



TESIS DOCTORAL

E-learning y gamificación como apoyo al aprendizaje de programación

JÉFFERSON BELTRÁN MORALES

Programa de Doctorado en Tecnologías Informáticas y Comunicaciones
(TINC)

Conformidad de los directores:

Fdo.: Héctor Sánchez Santamaría

Fdo.: María Mercedes Rico

2017

DECLARACIÓN DE AUTORÍA

Yo, Jéfferson Beltrán, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; que he consultado las referencias bibliográficas que se incluyen en este documento y que no he plagiado dicha información.

Jéfferson Beltrán

DEDICATORIA

A Dios, por darme la vida, ser mi guía y darme la fuerza necesaria para alcanzar este sueño.

A mis padres, Tarcicio y Angelita, por su apoyo incondicional toda mi vida.

A mi esposa Aracely, por su amor y comprensión.

A mi princesita Kamy, que con solo una sonrisa me da infinita felicidad y motivación cada día.

A mis verdaderos amigos, quienes confiaron siempre en mí.

Con mucho amor, este trabajo es para ustedes.

Jéfferson

AGRADECIMIENTOS

Mi agradecimiento imperecedero a todas las personas que permitieron la realización de este trabajo de investigación, en especial a mi familia, padres, esposa y mi hija Kamila por su amor incondicional, a mis amigos por su apoyo total.

Gracias a las autoridades de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador por su apoyo durante el doctorado, y a mis queridos estudiantes de Programación I quienes fueron parte fundamental en este proyecto.

Gratitud infinita a mis queridos y respetados directores de tesis, Dr. Héctor Sánchez Santamaría y Dra. María Mercedes Rico de la Universidad de Extremadura, quienes guiaron cada paso en este difícil, pero a la vez emocionante y enriquecedor camino al doctorado.

Gracias siempre

COMPROMISO DE NO PLAGIO

A través del presente documento, dejo constancia de que el desarrollo de esta investigación es fruto de la conjugación de mi esfuerzo, perseverancia y mucha dedicación, la cual no contiene material previamente publicado o escrito por otra persona, ni material que de manera sustancial haya sido aceptado sin que se haya hecho el reconocimiento debido del texto.

Todo el proceso de investigación se ha ceñido estrictamente a los reglamentos y normas vigentes en la Universidad Central de Ecuador, UCE, y en la Universidad de Extremadura, UEX; en consecuencia, este trabajo investigativo no es producto de plagio.

Quito, mayo de 2017

Jéfferson Beltrán

ÍNDICE

ÍNDICE	I
ÍNDICE DE FIGURAS	III
ÍNDICE DE TABLAS	VII
CAPÍTULO I. INTRODUCCIÓN	1
1.1. .Planteamiento del problema	3
1.2. .Justificación	5
1.3. .Objetivos.....	7
1.4. .Hipótesis	7
1.5. Estructura y sinopsis de esta tesis.....	8
CAPÍTULO II. BASE TEÓRICA Y TECNOLÓGICA	9
2.1. Modelos de enseñanza de la programación	9
2.2. .Teoría de la gamificación	11
2.2.1. La gamificación aplicada al proceso de enseñanza-aprendizaje.....	18
2.2.2. Gamificación como apoyo al aprendizaje de programación.....	20
2.2.3. Framework D6 en el diseño de gamificación de la asignatura	22
2.3. .Las TIC y el aprendizaje de la programación	24
2.3.1. El e-learning.....	25
2.3.2. El uso de plataformas virtuales en la educación	31
2.4. .Plataforma Moodle	31
2.4.1. Gamificación en Moodle	36
2.5. .Antecedentes a la investigación.....	38
CAPÍTULO III. CONTEXTO EN QUE SE DESARROLLA EL ESTUDIO	43
3.1. .El desarrollo de las TIC en la enseñanza universitaria en Ecuador	43
3.2. .Universidad Central de Ecuador. Principales características del centro de estudios	49
3.3. .Enseñanza de programación en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la UCE.....	52
CAPÍTULO IV. METODOLOGÍA	57
4.1. .Operacionalización de la variable.....	59

4.2. .Metodología objetivo 1.....	61
4.3. .Metodología objetivo 2.....	65
4.4. .Metodología objetivo 3.....	66
CAPÍTULO V. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS.....	73
5.1. .Resultados objetivo 1.....	73
5.1.1. Características metodológicas	73
5.1.2. Análisis cuantitativo	75
5.1.3. Análisis cualitativo	87
5.2. .Resultados objetivo 2.....	110
5.2.1. Justificación de la propuesta en la asignatura Programación I.....	110
5.2.2. Diseño de la asignatura gamificada	112
5.2.3. Implementación de gamificación en la plataforma Moodle	135
5.3. .Resultados objetivo 3.....	169
5.3.1. Análisis cuantitativo	169
5.3.2. Análisis cualitativo	178
5.3.3. Comprobación de hipótesis.....	194
CONCLUSIONES	199
LÍNEAS DE INVESTIGACIÓN FUTURAS.....	217
BIBLIOGRAFÍA	219
ANEXOS.....	229

ÍNDICE DE FIGURAS

Figura 1. El cono de aprendizaje de Dale [31]: aprendemos haciendo, adaptado por Fernández [32]	19
Figura 2. Diferencias entre <i>e-learning</i> y <i>b-learning</i>	30
Figura 3. Estudiantes aprobados y suspensos de los matriculados	75
Figura 4. Alumnos aprobados vs. suspensos por procedencia	77
Figura 5. Alumnos aprobados vs. suspensos por género	78
Figura 6. Aprobados vs. suspensos por tipo de colegio	79
Figura 7. Porcentaje de estudiantes aprobados y suspensos por carrera	80
Figura 8. Porcentaje de estudiantes aprobados por número de matrícula	81
Figura 9. Porcentaje de estudiantes aprobados y suspensos desde 2009 hasta 2015	82
Figura 10. Cantidad de estudiantes por profesor en cada semestre.....	83
Figura 11. Porcentaje de estudiantes aprobados y suspensos en Programación I, por cada docente.....	84
Figura 12. Relación de alumnos que aprobaron Programación I y luego recibieron Programación II	86
Figura 13. Análisis de elementos que influyen en que un alumno apruebe o no Programación I.....	86
Figura 14. Herramientas que facilitan el aprendizaje de Programación I.....	88
Figura 15. Métodos de enseñanza que facilitan el aprendizaje de Programación I	90
Figura 16. Opinión sobre la metodología de acuerdo a las competencias adquiridas.....	92
Figura 17. Motivación como factor importante en el aprendizaje	93
Figura 18. Criterio sobre la gamificación usada en un ambiente <i>e-learning</i>	94
Figura 19. Áreas de conocimiento que les gustaría estudiar	96
Figura 20. Importancia de aprender a programar.....	97
Figura 21. Opinión de los alumnos sobre si aprender a programar es difícil.....	97
Figura 22. Opinión de los alumnos respecto a si aprender a programar es aburrido.	98
Figura 23. Uso de las herramientas tecnológicas en el aprendizaje de Programación I	99
Figura 24. Contribución de la metodología usada para el aprendizaje de Programación I.....	103

Figura 25. Metodología usada para el aprendizaje.....	104
Figura 26. Opinión de los estudiantes sobre la motivación como factor fundamental en el aprendizaje	105
Figura 27. Grado de motivación en el aprendizaje de Programación I.....	106
Figura 28. Mejora en el nivel de conocimientos de programación	107
Figura 29. Satisfacción general respecto al curso de Programación I.....	108
Figura 30. Niveles de Programación Mágica.....	114
Figura 31. Ejemplo de insignias.....	115
Figura 32. Tablero de puntos de experiencia ganados	115
Figura 33. Barra de progreso.....	116
Figura 34. Estrategias de gamificación aplicadas en el nivel Agua – Introducción a la POO	122
Figura 35. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras secuenciales	125
Figura 36. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras selectivas.....	127
Figura 37. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras repetitivas.....	129
Figura 38. Estrategias de gamificación aplicadas en el nivel Aire – Arreglos y matrices. Fuente: elaboración propia.....	131
Figura 39. Estrategias de gamificación aplicadas en el nivel Fuego – Métodos.....	134
Figura 40. Portada de la plataforma Moodle gamificada antes de autenticarse	137
Figura 41. Cursos implementados en la plataforma Moodle	138
Figura 42. Barra de progreso en Moodle gamificado	139
Figura 43. Barra de progreso en Moodle gamificado	139
Figura 44. Insignias recientes en Moodle gamificado	140
Figura 45. Insignias del usuario en Moodle gamificado	141
Figura 46. <i>Ranking</i> en Moodle gamificado.....	142
Figura 47. Listado de participantes del curso.....	143
Figura 48. Perfil personal de un aprendiz	144
Figura 49. Bloque Navegación en Moodle gamificado	145

Figura 50. Calificaciones de un estudiante en Moodle gamificado	146
Figura 51. Calificaciones de los alumnos en Moodle gamificado	147
Figura 52. Calificaciones de alumnos en Moodle gamificado.....	148
Figura 53. Tarea Actualizar avatar en Moodle gamificado.....	149
Figura 54. Tarea Hechizo de programación en Moodle gamificado.....	150
Figura 55. Interfaz para realizar hechizos de programación en Moodle gamificado	151
Figura 56. Evaluación de Hechizo de programación en Moodle gamificado	152
Figura 57. Tarea Reto de hechicería en Moodle gamificado	153
Figura 58. Premios virtuales en Moodle gamificado	154
Figura 59. Taberna virtual en Moodle gamificado.....	155
Figura 60. Información del curso gamificado	157
Figura 61. Nivel Agua – Introducción a la POO.....	158
Figura 62. Nivel Tierra – Fundamentos de Java – Estructuras secuenciales	160
Figura 63. Nivel Tierra – Fundamentos de Java – Estructuras condicionales	161
Figura 64. Nivel Tierra – Fundamentos de Java – Estructuras repetitivas.....	162
Figura 65. Nivel Aire – Arreglos y matrices.....	164
Figura 66. Nivel Fuego – Métodos	166
Figura 67. Hechizo final en Moodle gamificado	167
Figura 68. Comparación de resultados académicos en los tres semestres	177
Figura 69. Usabilidad de la plataforma Moodle gamificada.....	179
Figura 70. Moodle gamificado vs. plataforma virtual.....	180
Figura 71. Gamificación en ambiente <i>e-learning</i>	181
Figura 72. Criterio de los alumnos sobre la metodología tradicional y la gamificada.....	182
Figura 73. Aprender a programar usando la plataforma virtual y metodología gamificadas es divertido	183
Figura 74. Grado de motivación en el aprendizaje de Programación I.....	191
Figura 75. Grado de motivación en el aprendizaje del segundo hemisemestre comparado con el primero	192
Figura 76. Nivel de conocimiento de programación.....	193
Figura 77. Nivel de satisfacción general del curso gamificado	194

ÍNDICE DE TABLAS

Tabla 1.	Superficie de los laboratorios	53
Tabla 2.	Capacidad de los laboratorios	53
Tabla 3.	Metodologías usadas en cada semestre.....	58
Tabla 4.	Cuadro de operacionalización.....	60
Tabla 5.	Descripción de la muestra de estudiantes encuestados.....	95
Tabla 6.	Herramientas tecnológicas utilizadas en el aprendizaje de Programación I.....	99
Tabla 7.	Recursos y tecnologías utilizadas en el aprendizaje de Programación I	100
Tabla 8.	Aspectos metodológicos que favorecen más el aprendizaje de Programación I.....	101
Tabla 9.	Formas de evaluar que favorecen más el aprendizaje de Programación I.....	105
Tabla 10.	Capítulos, niveles, avatar y puntuación de la unidad Fundamentos de Java ...	119
Tabla 11.	Distribución de la asignatura gamificada	120
Tabla 12.	Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 1	171
Tabla 13.	Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 2	173
Tabla 14.	Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 3	175
Tabla 15.	Metodología y plataforma gamificadas utilizadas como apoyo al aprendizaje de Programación I	184
Tabla 16.	Dinámicas de juegos que favorecen más el aprendizaje.....	186
Tabla 17.	Mecánicas de juego que favorecen más el aprendizaje de programación	188
Tabla 18.	Test de normalidad de Kolmogorov-Smirnov	196
Tabla 19.	Test t-Student de las notas del semestre octubre 2015-marzo 2016.....	196
Tabla 20.	Test no paramétrico de las notas del semestre abril-septiembre 2016.....	197
Tabla 21.	Test no paramétrico de las notas de los semestres 1, 2 y 3.....	198

CAPÍTULO I. INTRODUCCIÓN

El desarrollo de la sociedad está condicionado por la evolución biológica que permitió al ser humano adaptarse a los entornos nuevos y/o cambiantes a lo largo del tiempo y por los cambios sociales y culturales; es así que el lenguaje entra a formar parte indisoluble de este proceso y con él las distintas maneras de regular el conocimiento. Los avances científicos son elementos de peso en esta evolución y la educación los toma como punto de partida y referencia.

La aparición y desarrollo de la electrónica constituyó el primer paso para que la era digital se convirtiera en un suceso social de amplia connotación, logrando una conexión estrecha y directa entre la electrónica, las telecomunicaciones y la informática.

Las tecnologías de la información y las comunicaciones (TIC) se convierten de este modo, en un elemento de peso para las relaciones humanas, el conocimiento y la evolución de la sociedad. A través de las TIC se puede producir, almacenar, comunicar, estructurar, registrar, procesar y documentar información desde todas partes y sin límites de tiempo, lo que nos lleva a pensar en ellas como contenedoras y productoras de conocimientos.

La educación se ha servido de las TIC para encaminar el proceso de conocimiento de los estudiantes, como uno de tantos métodos para establecer la relación entre los maestros, los estudiantes, la sociedad y el conocimiento.

Las TIC en el medio educativo posibilitan la colaboración, comunicación, análisis y creatividad como instrumentos del conocimiento, convirtiéndose en una vía para lograr mayor interés y motivación por parte de los alumnos, estimulando su creatividad y capacidad de discernimiento, lo que hace que el proceso de conocimiento sea atractivo, aprendan en menor tiempo, exista una mayor relación profesor-estudiante, así como para maximizar el desempeño académico.

Para lograr esto, el profesor necesita estar en una constante búsqueda de métodos y estrategias innovadoras, por eso su labor debe ir más allá del fortalecimiento de su preparación, creando incluso sus propios instrumentos de trabajo para impartir las clases.

La incesante búsqueda para encontrar nuevas formas de llevar los contenidos y recursos académicos a los educandos promueve estudios como este que analiza alternativas para contribuir al proceso de la enseñanza-aprendizaje, que se puede lograr a partir de la introducción de las TIC como método novedoso donde se pueden conjugar los conocimientos teóricos y prácticos.

Es el caso puntual de la gamificación, aplicación de elementos de los juegos [1], que permite integrar al proceso educacional algunas dinámicas que se basan en recompensas, logros, competencia y el estatus al que puede llegar el estudiante según su esfuerzo y motivación, y las mecánicas que potencien el interés de los estudiantes, tales como acumular puntos, escalar niveles, premios, regalos y otros.

Al respecto, se asumen los criterios de Lee, Ceyhan, Jordan-Cooley y Sung [1], quienes exponen que a partir de la gamificación pueden desarrollarse herramientas que ofrezcan al alumno la oportunidad de aprender a través de una solución motivadora. En tanto, como instrumento educativo, resulta interesante debido a que «la gamificación puede ser una estrategia de gran alcance que promueva la educación entre las personas y un cambio de comportamiento» [1, p. 15]. Por lo anteriormente dicho se sustenta que la gamificación en el espacio académico puede provocar una situación de dependencia sana.

Con esta investigación se busca cambiar la actitud de los estudiantes hacia la asignatura Programación I a través del uso de las dinámicas y mecánicas de juego y su implementación, utilizando la dimensión lúdica en entornos virtuales como alternativa a la coerción que muchas veces se usa en la enseñanza para sugestionar a los alumnos en el proceso de aprendizaje.

1.1. Planteamiento del problema

Algunos de los problemas que enfrentan los docentes giran alrededor de cómo conseguir de sus estudiantes una motivación adecuada y el compromiso en el proceso de aprendizaje. En los últimos años, de acuerdo con Oviedo y Ortiz [2], los profesores de programación observan que, por lo general, los alumnos de los cursos de programación, básicamente los de primer y segundo semestres, muestran algunos rasgos que dificultan el aprendizaje, tales como:

- Falta de implicación estudiantil.
- Apatía por la programación [2].

Por su parte, García, Lías, Mahílló, Pinero [3], exponen que el abandono en Ingeniería Informática es mayor que en otras ingenierías y mencionan que esto ocurre con mayor frecuencia en el primer curso y que está relacionado, en cierta medida, con dificultades de adaptación a la universidad, ya sea por causas académicas tales como: deficiente preparación previa y bajo rendimiento; como por posibles causas sociales relacionadas con algunos problemas de los estudiantes para vincularse al medio académico.

Como resultado de este estudio, se proponen medidas para que el futuro estudiante conozca la oferta académica en detalle, con programas de información, procesos de orientación y su posterior acogida. Su enfoque no está orientado a analizar los problemas de aprendizaje sino a hacerle conocer al alumnado lo que significa la carrera.

Para Oviedo y Ortiz [2], los estudiantes sufren de problemas derivados del sistema educativo básico actual y que se evidencian en su desempeño inicial en la carrera. En tal caso, los estudiantes del primer semestre, en su cambio del nivel medio al superior, pasan de manera abrupta de un ambiente escolar controlado a otro de casi libertad total, lo que les provoca descontrol y desubicación temporal. Por otro lado, el estudiante entra en un ambiente social que le cautiva, crea amistades con otros compañeros que manifiestan intereses comunes y con los que se identifica. Su nuevo estatus le atrae más que los asuntos académicos relacionados con su carrera y no presta la atención necesaria a los mismos.

En tal caso, Jenkins [4] asegura que los estudiantes deben estar motivados para que puedan participar de manera apropiada. Con una adecuada motivación sus actitudes cambian a medida que avanzan a través del curso. Algunos estudiantes pueden descubrir un interés genuino en la informática de manera general y después por la programación, y desarrollar una motivación intrínseca que es lo que realmente se debe buscar. Otros, en cambio, pueden llegar a valorar el aprendizaje de programación por razones básicamente extrínsecas, por ejemplo, tener un buen futuro financiero una vez que empiecen a trabajar.

Steinmann, Bosh y Aiassa [5] plantean que la motivación es un condicionante decisivo del aprendizaje y de allí se deriva la importancia de profundizar en su estudio. Desde este punto de vista se hace imprescindible conocer las expectativas de los estudiantes, cuáles son sus motivaciones, teniendo en cuenta que la interacción entre motivaciones intrínsecas y extrínsecas, así como la desmotivación, se encuentran relacionadas con diversos factores que inciden en su rendimiento. Estas motivaciones se relacionan, al menos en parte, con las características del medio socioeconómico y cultural, así como con la experiencia escolar. Las condiciones sociopsicológicas de las instituciones escolares impactan en los logros académicos y emocionales de los estudiantes [5].

La motivación, como elemento psicológico relacionado con el proceso humano, está determinada por la interacción de las personas con el contexto, de ahí que no funcione igual para todos, ni en todo momento, por tanto, debe manejarse como un proceso individual en medio de un entorno colectivo.

Es importante señalar que, a nivel individual, para gamificar una actividad es imprescindible hallar la manera correcta de motivar a la persona específica en el momento adecuado. Para lo anterior es importante referirse a los distintos tipos de motivaciones:

- **Intrínsecas:** inherentes a la persona. Lo realiza por su propio bien o interés (por ejemplo: estatus, poder, acceso a ciertas aptitudes) o para contribuir a un bien común [6].
- **Extrínsecas:** exteriores a la persona. Lo realiza por la recompensa o retroalimentación. También es importante el componente social, o lo que es lo

mismo, el contar con otras personas con las que pueda competir, colaborar y comparar logros. En el juego social los objetivos pueden ser competitivos o colaborativos [6].

En tal sentido, en los juegos de equipo deben diferenciarse las mecánicas que influyen en el equipo (proyectos, puntuaciones de grupo, etc.) de las mecánicas que solo se aplican al individuo (motivación, refuerzo positivo, entre otras) [7].

Sería importante destacar que, aunque la motivación intrínseca y la extrínseca pueden estar estrechamente relacionadas entre sí, este trabajo se centrará más en el estudio de la motivación intrínseca, por el interés establecido en generar en el estudiante el placer por la asignatura en cuestión, ya que esta permite una mejora del conocimiento, potencia las actividades de voluntariado, un mayor reconocimiento personal, y la obtención de mayores responsabilidades.

Si se tiene en cuenta que para que los alumnos logren aprobar la asignatura Programación I existen varios aspectos como son el profesor, el nivel educativo con el que el estudiante se enfrenta al estudio, entre otros, no nos cabe la menor duda de que la motivación del estudiante puede ser crucial. Para ello se tomará en consideración cómo establecer vínculos directos con el comportamiento dinámico de la conducta, deseos, metas, desempeño, así como utilizar métodos que permitan determinar hacia dónde se puede dirigir la conducta del alumno. En consonancia con lo expuesto anteriormente, se plantea como problemática la necesidad de incidir sobre el aprendizaje de estos alumnos a través del desarrollo de su motivación por los estudios de programación.

1.2. Justificación

Una vez establecida la problemática, esta investigación asume la tesis de que los avances de las TIC aportan instrumentos para el desarrollo del proceso educativo de una manera más integrada, y ayudan a promover la motivación entre los estudiantes, ya que, como en nuestro caso, puede servirse de los entornos virtuales de aprendizaje (EVA), también conocidos por las siglas en inglés LMS (*Learning Management System*) y de propuestas educativas basadas en la gamificación.

Por otro lado, Muntean [8] investiga cómo puede aumentar la participación de los alumnos con una aplicación *e-learning* y sus tareas específicas mediante la implementación de dinámicas y mecánicas de juego.

Investigadores del Departamento de Informática de la Universidad Ramon Llull [9] presentaron un catálogo de herramientas de ayuda para que los estudiantes se sintieran motivados y así conseguir el máximo rendimiento en la asignatura y afianzar gradualmente los conocimientos sin depender de la proximidad del examen. Normalmente los estudiantes que no tienen la suficiente motivación dedican más tiempo a estudiar antes de la celebración del examen, debido quizá a que la principal motivación no es el aprendizaje sino la calificación necesaria para aprobar la asignatura. Plantean además la problemática de la adaptación de la nueva dinámica de estudio que se tiene al ingresar a la universidad [9].

Estos investigadores proponen tres modalidades para evitar los problemas de motivación. En primer lugar, un entorno vía web que comunique a los alumnos y a los profesores. En segundo lugar, un plan de trabajo continuo que, a su vez, permita conocer la evolución de los alumnos. Y finalmente, emplear una aplicación de entorno educativo para introducir los conocimientos [9].

De acuerdo con estos análisis, el diseño de una propuesta metodológica que asuma la teoría de la gamificación y los LMS para contribuir a la motivación de los estudiantes debe ser de interés para la universidad ecuatoriana, puesto que el resultado de la investigación y de las lecciones aprendidas mediante su aplicación práctica en la enseñanza de Programación I se extrapolan al resto de asignaturas dictadas, con el objetivo de inspirar a los estudiantes a que se comprometan, que aumenten el nivel de actividad, la motivación, la interactividad y la participación en el proceso de aprendizaje presencial con el apoyo de la gamificación y el *e-learning*. Además del interés de la tesis para la institución en la que se encuadra, es importante justificar el interés como aportación científica en general en el área de la gamificación en entornos *e-learning*.

Para la consecución de la propuesta de este estudio se necesitó la conformación de objetivos que dirijan y guíen la investigación.

1.3. Objetivos

Para la ejecución del estudio se partió de los siguientes objetivos planteados:

Objetivo general

Diseñar una propuesta metodológica para la realización de tareas autónomas basadas en el desarrollo de las TIC, que influya en la motivación de los estudiantes y mejore su desempeño académico, en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

Para la consecución de este estudio, deben lograrse los siguientes objetivos específicos:

Objetivos específicos

- Analizar la metodología de enseñanza existente en la modalidad presencial de la asignatura Programación I, para identificar los factores que influyen en el proceso de aprendizaje de los estudiantes.
- Diseñar e implementar estrategias como apoyo al proceso de aprendizaje presencial, desarrollando herramientas de *e-learning* y gamificación.
- Evaluar la actividad pedagógica de las estrategias de gamificación en un entorno *e-learning*, diseñadas para la modalidad presencial en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

1.4. Hipótesis

De acuerdo con lo anterior, se expone para el desarrollo del estudio, la siguiente hipótesis:

La implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje para desarrollar tareas autónomas influye sobre la motivación de los estudiantes y contribuye a mejorar su rendimiento académico [5], en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

1.5. Estructura y sinopsis de esta tesis

La presente investigación se desarrolla en capítulos estructurados de la siguiente forma:

En el capítulo 1 se establece la problemática de la investigación y a partir de esta los objetivos que estructuraron el proceso de indagación. Se parte de la prioridad de incidir sobre el aprendizaje de los estudiantes a través del desarrollo de su motivación por los estudios de programación y al respecto se asume la tesis de Lee, Ceyhan, Jordan-Cooley y Sung [1], quienes muestran que a partir de la gamificación puede crearse una herramienta práctica que permita al educando la oportunidad de aprender de forma continua a través de una solución integradora y llamativa.

Después en el capítulo 2 se exponen las teorías sobre los modelos de enseñanza de programación y el apoyo que puede significar la inserción de mecánicas y dinámicas de juego en el proceso de aprendizaje, unido a la implementación en los entornos virtuales. Se expone el *e-learning* y los entornos virtuales para facilitar la educación y capacitación mediante herramientas tecnológicas.

Luego en el capítulo 3 se establece el contexto en el que se desarrolla el estudio, tomando en cuenta la utilización de los entornos virtuales de aprendizaje en algunas universidades ecuatorianas, además de establecer características de la Universidad Central de Ecuador y de la metodología con la que se imparte la asignatura de Programación I en esta institución de enseñanza superior.

Posteriormente en el capítulo 4 se plantea la metodología de investigación científica, donde se establecen los procedimientos a seguir por el investigador para lograr los objetivos de este estudio. Por cada objetivo se plantean métodos y técnicas para su desarrollo.

Al final, en el capítulo 5 se analizan e identifican los factores que influyen en el proceso de enseñanza-aprendizaje a partir de la metodología establecida, y en base a esto se diseñan e implementan las estrategias a seguir para la realización de tareas autónomas en un entorno virtual de aprendizaje, lo que lleva a una evaluación pedagógica de las estrategias de gamificación aplicadas y a la posterior validación de la hipótesis de investigación.

CAPÍTULO II. BASE TEÓRICA Y TECNOLÓGICA

2.1. Modelos de enseñanza de la programación

Las diversas teorías sobre los modelos de enseñanza, de forma general, van desde los tradicionales a los constructivistas [10]. Sin embargo, es importante puntualizar que en la enseñanza universitaria no se aplica un modelo único o ideal, ya que se hace necesario contar con la realidad que acontece en el aula, el contexto y la interacción entre profesor y estudiantes. Es decir, en el proceso de enseñanza-aprendizaje se evidencian características disímiles que condicionan la utilización de un modelo sobre otro con la utilización de estrategias didácticas específicas [11].

Es así como, a través de la historia, en la enseñanza de la programación han cohabitado diferentes enfoques y tendencias [12]. En la actualidad se puede mencionar que no hay una estrategia única en la utilización de métodos de enseñanza de la programación, dada la diversidad de enfoques y la orientación de los métodos didácticos aplicados [11].

Teniendo esto en cuenta, Kaasbøll [13] propone tres modelos didácticos que han sido ampliamente utilizados para enseñar programación: la escalera semiótica, que se enfoca en la utilización de lenguajes de programación y en la búsqueda de instrucciones, que evidencian evolución de secuencias sintácticas, luego semánticas y finalmente, la pragmática del lenguaje como herramienta fundamental; el de objetivos de taxonomía cognitiva, basado en el uso de secuencias a través de instrucciones; y el de resolución de problemas; establecido más como un modelo de aprendizaje que como una estrategia de enseñanza [11].

En las ciencias de la programación, los resultados de las investigaciones de Volet y Lund [14] puntualizan que el aprendizaje de los estudiantes no debe restringirse al nivel sintáctico del conocimiento, sino pasar al nivel de conocimiento conceptual, unido al desarrollo de habilidades cognitivas básicas y al uso de habilidades metacognitivas dirigidas a la resolución de nuevos problemas de programación. También enfatizan la importancia de desarrollar en los estudiantes sus propios modelos mentales, una

descomposición de la estructura jerárquica y la relación que se establece entre los objetos y una valoración del avance de cada estudiante.

El modelo didáctico tiene diversos puntos de coincidencia con el modelo constructivista, en el que se basa específicamente el modelo de aprendizaje basado en la resolución de problemas. Los docentes centran su actividad con los alumnos en la solución de problemas de la vida cotidiana a los que se enfrentará el futuro ingeniero en la asignatura de Programación I; en estas estrategias proponen ejercicios que se han creado y seleccionado previamente por el profesor.

En los procesos metodológicos empleados por los profesores para profundizar en los conceptos objeto de enseñanza, mediante la resolución de problemas, se evidencia que los pasos didácticos giran en torno a:

- Explicación por parte del docente mediante la solución de problemas demostrativos.
- Ejercicios dirigidos a que los alumnos resuelvan problemas.
- Evaluación y retroalimentación.

La retroalimentación es una actividad que los profesores realizan constantemente y está estrechamente relacionada a la evaluación, que no solo debe verse como el proceso donde se obtiene un resultado cuantitativo para obtener notas parciales o finales, sino que es un proceso continuo, de cada clase. Es, además, un aspecto inherente a la evaluación, ya que permite conocer a los alumnos su desempeño, sus procesos de aprendizaje y la forma como se pueden potencializar o favorecer, lo que la convierte en una herramienta de mucha importancia para el docente.

Entre los problemas más comunes e importantes que se encuentran los profesores en el aula está el conseguir en sus estudiantes una motivación adecuada y que esta se mantenga a lo largo de la impartición de la asignatura. Es por eso que el papel del profesor es crucial para promover el interés de sus educandos como condicionante del aprendizaje.

Por eso es necesario tener claro cuáles son las expectativas, las motivaciones, ya sean intrínsecas o extrínsecas, así como las desmotivaciones a la hora de evaluar su rendimiento.

Todo esto está potencialmente vinculado con las peculiaridades del medio que los rodea, desde el punto de vista social, económico y cultural.

Lo que sí salta a la vista es que la tasa de estudiantes aprobados que cursaron la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador fue del 53,0 % y la tasa de suspensos del 47,0 % [15], por lo que los métodos educativos tienen que estar en consecuencia con estos resultados a fin de resolver esta dificultad. Es aquí cuando se empieza a hablar de la gamificación aplicada a la educación como un método para mejorar el rendimiento y la motivación en los estudiantes si se tiene en cuenta que los jóvenes de hoy, de manera general, dominan los medios digitales.

De ese modo hay que tener en cuenta que las estrategias a seguir deben estar basadas en el diseño de actividades dirigidas a los estudiantes y centradas en el uso de las técnicas de gamificación, así como en su implementación en un entorno virtual de aprendizaje.

2.2. Teoría de la gamificación

El actual desarrollo de las TIC y los diferentes métodos de enseñanza han provocado que maestros y alumnos tengan en su poder distintas herramientas que permiten el apoyo y la motivación del proceso educativo. La gamificación, o como también se le suele llamar ludificación, es «el empleo de mecánicas de juego en entornos y aplicaciones no lúdicos, con el fin de potenciar la motivación, la concentración, el esfuerzo, la fidelización y otros valores positivos comunes a todos los juegos» [6, 16, 17].

Si tenemos en cuenta que el proceso de aprendizaje en el aula se puede volver muy monótono y aburrido desde el punto de vista tradicional, la gamificación puede convertirse en la solución práctica a estas dificultades, pues a través de ella se puede estimular la participación activa de los estudiantes, lo que genera un proceso de retroalimentación entre los compañeros de estudio, interviene en la construcción de una identidad propia bien definida, y afianza la confianza entre los alumnos, convirtiendo actividades difíciles en procesos más sencillos.

Al respecto, las técnicas usadas en la gamificación se transforman en estrategias útiles para aplicarlas en la enseñanza, contribuyendo a la motivación de los alumnos hacia el aprendizaje. Zichermann y Cunningham [18] la definen como «un proceso relacionado con el pensamiento del jugador y las técnicas de juego para atraer a los usuarios y resolver problemas» [18, p. 11]. Kapp [19] señala que la gamificación es «la utilización de mecanismos, la estética y el uso del pensamiento, para atraer a las personas, incitar a la acción, promover el aprendizaje y resolver problemas» [19, p. 9].

Los tres autores defienden que la finalidad de la gamificación es influir en la conducta psicológica y social del jugador. Los autores indican que a través del uso de ciertos elementos presentes en los juegos (como insignias, puntos, niveles, barras de progreso, avatares, etc.), los jugadores incrementan su tiempo en el juego, así como su predisposición psicológica a seguir en él.

Antes de profundizar sobre los aspectos positivos de la gamificación, es oportuno hacer una distinción entre esta y los videojuegos convencionales. Los investigadores Hamari y Koivisto [20], en su estudio sobre este tema establecían las siguientes diferencias:

1. Los videojuegos se definen como juegos electrónicos realizados con el fin de visualizarse a través de una pantalla y la gamificación está trabajada a partir del uso de mecánicas y métodos de juego que se utilizan en contextos que no son de esparcimiento o juego propiamente dichos.
2. Uno de los principales objetivos de la gamificación es influir en la conducta de las personas, además de algunos objetivos secundarios como el placer que se pueda sentir durante el tiempo que se realice esta actividad, y los videojuegos tienen la finalidad exclusiva del placer por el juego.
3. La gamificación genera experiencias, sensaciones de dominación e independencia en las personas logrando así un notable cambio en su comportamiento, mientras que los videojuegos solamente conciben destrezas hedonistas por los audiovisuales.
4. Los juegos educativos en el aula y la gamificación tienen una diferencia marcada ya que los juegos educativos están pensados para incorporar elementos del aprendizaje, los cuales pueden ser atractivos, pero también pueden ser repetitivos y están en

función de establecer experiencias básicamente en el aula, mientras que la gamificación basa su enfoque en otra perspectiva, sus metas, puntos, premios u otros elementos pueden desprenderse del contexto educacional y llevarse a un terreno que no sea exactamente de juego, siempre con la intención de motivar y enseñar [19].

Entre los aspectos positivos de la gamificación podemos señalar el motivacional, que estimula al estudiante; el cognitivo, que interviene en mejorar la atención y el tiempo de reacción; el social, que mueve las destrezas aprendidas fuera del entorno lúdico, y el emocional, que regula las emociones y conductas positivas.

En la gamificación se utilizan elementos de juego, como los incentivos y puntos, para lograr el comportamiento que se quiere de parte del jugador. Estos elementos tienen la misión de hacer sentir al jugador o usuario que el juego tiene determinadas metas a cumplir y estas se determinan por los premios que se logran a partir de acciones, que a su vez pueden ser visualizados por otros jugadores, lo que hace que la motivación en el usuario sea mayor, con el fin de obtener mayor reconocimiento; estos logros son objeto de muchos estudios en el área de la economía y de la teoría de los juegos [21].

Además de estos elementos, existen otros como el reto, que tiene una importante carga psicológica y cuyo principal fin es influir en el comportamiento del usuario. El reto está considerado como un elemento crucial en la gamificación. En este caso la psicología expresa la necesidad del jugador de conseguir superar sus expectativas o las expectativas que el juego le impone: conseguir el reto es un ejemplo de superación para el usuario [22].

En la obra de Tejeiro, Pelegrina del Río y Gómez [23] se describe una aproximación a las implicaciones de la psicología en la gamificación. Los autores indican que los usuarios son conscientes de la exigencia del juego y que son ellos los primeros en advertir la existencia de elementos como el reto, una de las razones fundamentales por las que usan los videojuegos. A principios de los años noventa, la doctora Shotton [24] sugería que, además de incrementar la destreza manual y la coordinación visomotora, los juegos aceleraban las vías neuronales, permitiendo que el conocimiento viaje a mayor velocidad, acelerando los juicios y las decisiones.

En este trabajo nos centramos especialmente en la gamificación porque esta tiene una serie de elementos de uso frecuente que proyectan a su vez una gran carga psicológica haciendo que el usuario esté más tiempo en el juego, que colabore más y sobre todo que aprenda gracias al contenido de la información que recibe durante su estancia en el mismo. De esta forma se consigue que el jugador adquiera conocimientos de los que carecía, además de conocer este tipo de juegos con el fin de que los consuma, en este caso, para fines educativos.

Para una mejor comprensión de la gamificación, es necesario conocer una serie de elementos que suelen estar presentes en ella. Kapp [19] menciona algunas de las particularidades de la gamificación, muchas ellas compartidas por Zichermann y Cunningham [18], y por Werbach y Hunter [25].

- **La base del juego:** es donde aparece la oportunidad de jugar, de aprender, y transmitir la información y la existencia de un reto que estimule al juego. Se necesita establecer algunas normativas en el juego, la interactividad y la retroalimentación.
- **Dinámicas:** se pueden definir como aquellos patrones, pautas y sistemas presentes en los juegos pero que no forman parte de los mismos. Mediante estas, se adaptan las mecánicas a los diferentes tipos de jugadores. Las dinámicas se basan en los deseos básicos de las personas: reconocimiento, consecución de objetivos, capacidad de expresión, etc. Entre las principales se identifican las siguientes:
 - **Recompensas:** es algo de valor que se obtiene tras la realización de una acción o la consecución de un objetivo, y se traducen en puntos, medallas, acceso a niveles del juego superiores, bienes virtuales, entre otras.
 - **Estatus:** se entiende como un sentimiento de reconocimiento por parte de terceros y un sentimiento de popularidad, de prestigio, de respeto o de captación de la atención. Es un factor considerado de la máxima importancia, por lo que deben contener elementos que pongan de manifiesto este estatus, como los niveles, las medallas, las clasificaciones, etc.
 - **Logros:** es conseguir algo que tiene un cierto nivel de dificultad, alto pero

factible tras un esfuerzo razonable; puede ser un buen elemento de motivación para muchas personas. Una buena recompensa es el reconocimiento de que se ha conseguido un logro.

- **Autoexpresión:** las personas buscan diferenciarse del resto mediante el aspecto, las acciones que llevan a cabo, etc. Es una manera de conseguir autonomía, mostrarse original ante terceros y, en cierto modo, forjarse una determinada identidad particular.
- **Competición:** una de las maneras de obtener satisfacción en el desempeño de una acción es comparando nuestros resultados con los de los demás. Es un factor de motivación.
- **Altruismo:** se basa en la entrega de bienes o servicios a los demás sin una compensación directa o prevista.
- **Retroalimentación o *feedback*:** permite a los jugadores/usuarios conocer cuál es su grado de cumplimiento o progreso en el juego o sistema en el cual participan. Es un elemento básico en cualquier juego y, sobre todo, en cualquier sistema gamificado.
- **Diversión (*fun*):** contiene elementos de diversión como el sentimiento de victoria, la resolución de problemas, los deseos de exploración, el trabajo en equipo, la sorpresa, la imaginación, el compartir, la asunción de nuevos roles, el relajamiento [26].
- **Mecánicas:** son los sistemas y elementos que hacen que el progreso en el juego o sistema sea visible. Entre las mecánicas destacan por su popularidad y presencia masiva en juegos y en sistemas gamificados las identificadas como PBL, del inglés *Points* (puntos), *Badges* (insignias o medallas) y *Leaderboards* (clasificaciones) [26]. Entre las mecánicas y componentes de juego se tienen los siguientes:
 - **Puntos:** se ganan por las acciones de los participantes.
 - **Insignias o medallas:** son representaciones gráficas de los logros obtenidos.
 - **Tabla de clasificaciones:** ordena de manera visual a los participantes según los puntos ganados.
 - **Barra de progreso:** muestra visualmente el cumplimiento y progreso de las

diferentes tareas.

- **Niveles:** indicador del grado de progreso en el juego; cuando se supera un nivel, se desbloquea el contenido del siguiente nivel.
 - **Avatares:** representaciones personalizadas y únicas de los jugadores por medio de una imagen.
 - **Retroalimentación automática:** forma en que los jugadores saben si están avanzando correctamente en el juego. En programación una forma es la compilación, ejecución y evaluación automática mediante pruebas unitarias de programas.
 - **Colaboración:** mediante los foros, para motivar a los jugadores a interactuar y ayudarse entre ellos.
 - **Misiones y retos:** serie de acciones a llevar a cabo para conseguir un objetivo, por ejemplo, pruebas y ejercicios.
 - **Bienes virtuales, regalos o premios:** bienes tangibles o intangibles que los participantes ganan en el juego [26].
-
- **Estética:** la utilización de imágenes agradables a la vista del jugador.
 - **Idea del juego:** a través de estas dinámicas y mecánicas de juego el jugador va recibiendo información, a veces evidente solamente por su subconsciente, con la que se logra que asimile actividades que son reales en el entorno virtual y con ello obtenga habilidades que antes no poseía.
 - **Conexión juego-jugador:** se pretende encontrar un acuerdo entre el jugador y el juego. Padilla, Halley y Chantler [27] indican que el jugador necesita encontrar con relativa facilidad lo que está buscando, los botones que vaya a usar o las instrucciones para jugar. Si no los encuentra fácilmente podrá crearse un sentimiento de frustración hacia el juego, y la relación jugador-juego será negativa.
 - **Jugadores:** existen diferentes perfiles de jugadores, jóvenes o no, estudiantes o no. Debido a esa diversidad, Kapp [19] realiza una división entre los jugadores que quieran ser parte del proceso creativo y que pudieran estar motivados a actuar en el juego, y los que no.
 - **Motivación:** la predisposición psicológica de la persona a participar en el juego es

decididamente un desencadenante. Respecto a la motivación en la gamificación se considera que no debe ser sin muchos desafíos, pues puede volverse aburrida, ni con demasiados pues podría derivar en ansiedad y frustración. Y como el aprendizaje se basa en la repetición y en el tiempo, los desafíos deben aumentar progresivamente para mantenerse a la altura de sus habilidades crecientes [28, p. 9], hay que buscar una media para que el usuario no se sienta incapaz de lograr el objetivo, y por tanto abandone el juego o, todo lo contrario, que el juego se presente tan fácil de resolver que no tenga atractivo para el jugador.

- **Promover el aprendizaje:** la gamificación incorpora técnicas de la psicología para fomentar el aprendizaje a través del juego. Técnicas tales como la asignación de puntos y el *feedback* correctivo.
- **Resolución de problemas:** el objetivo final del jugador es llegar a la meta, resolver el problema, eliminar a su enemigo en combate, superar los obstáculos, etc.

De forma general se manifiesta la necesidad de motivar a los estudiantes utilizando las dinámicas y mecánicas de juego que además permiten incrementar la responsabilidad con el aprendizaje y el vínculo del estudiante con el contenido y con las tareas en sí.

Al respecto, Werbach y Hunter [25] plantean tres tipos de gamificación:

- **Interna:** para mejorar la motivación dentro de una clase.
- **Externa:** cuando se busca involucrar a los estudiantes mejorando las relaciones con los profesores.
- **Cambio de comportamiento:** busca generar nuevos hábitos en los alumnos, hasta conseguir que escojan opciones más sanas al rediseñar la clase para conseguir que se aprenda más mientras se disfruta.

2.2.1. La gamificación aplicada al proceso de enseñanza-aprendizaje

El concepto de gamificación aplicado a la enseñanza trata de sistematizar un procedimiento, a la hora de impartir las clases, o en la realización de actividades de aprendizaje guiando a los estudiantes en la acumulación de conocimientos a partir de novedosas técnicas que les generen un aprendizaje más relevante. En definitiva, establecer situaciones de aprendizaje a través de las que obtengan ciertas competencias y conocimientos. Se ha usado con dos propósitos principales: fomentar el aprendizaje de conductas deseadas, y promover la participación de los estudiantes en actividades de aprendizaje [29]. Entre las muchas definiciones de este concepto, puede asumirse que «la gamificación en educación es el uso de los elementos de juego (dinámicas, mecánicas y componentes) en un ambiente de aprendizaje, generalmente apoyado en las TIC» [30], con el objetivo de que los estudiantes logren apropiarse de forma adecuada de ciertos resultados, como la satisfacción, la persistencia, y los logros académicos [29].

Parte de la esencia que sostiene el uso de estas nuevas técnicas es la forma en la que aprenden los seres humanos. Se sabe que con estímulos verbales (clase tradicional/lección magistral) se obtiene un 10 % de tasa de recuerdo a las 72 horas. Es decir, de todo lo que «se dice» en las clases, aproximadamente un año después ya no se recuerda casi nada. Pero si a esos mismos alumnos se les muestra un total de 2500 imágenes, con una frecuencia de 10 segundos por imagen, la tasa de recuerdo de dichas imágenes a las 72 horas sería del 90 %. Y al cabo de un año podrían recordar el 63 % de las imágenes visualizadas [31].

Si se analiza el cono de aprendizaje de Dale [31], se aprecia que la mayor tasa de aprendizaje se puede lograr con un usuario activo y que se contrapone a simulaciones o situaciones reales.



Figura 1. El cono de aprendizaje de Dale [31]: aprendemos haciendo, adaptado por Fernández [32]

Esto se consigue mediante la dimensión lúdica como un compendio de conductas en las que se preparan y adaptan las habilidades específicas para algunas de las actividades vitales que los estudiantes realizarán a lo largo de su vida [32].

A los alumnos el aspecto lúdico les favorece en la asimilación de la información y en su selección. Eligen elementos relevantes que después utilizarán a lo largo del juego. También se aprenden reglas, personajes, además se memoriza sin manual de instrucciones, lo que se precisa para poder desarrollar el juego. Se genera la capacidad de aplicar esas reglas a las diferentes etapas del juego, y los jugadores no desconocen las reglas, sino que indagan, interaccionan con los compañeros, socializan, buscan soluciones que puedan ser aplicadas a su juego, las verbalizan, cooperan entre ellos, se adaptan e interactúan. Los alumnos quieren participar, no quieren ser meros consumidores. En definitiva, se desarrollan todas las etapas del aprendizaje consignadas por García Hoz que generan el aprendizaje significativo [32, pp. 43-48].

Detrás de cada juego hay una serie de aprendizajes tanto de contenidos como de valores, tolerancia a la frustración, memorización de reglas, estrategias para ganar, anticipación ante las posibles acciones del otro. Cualquier juego desarrolla competencias esenciales como la observación, probabilidad, rapidez, empatía, intuición, toma de riesgos y de decisiones. A veces gano y a veces pierdo, pero siempre las decisiones tienen consecuencias, mis acciones generan consecuencias sobre el entorno.

Por tanto, de acuerdo con Fernández [32], un espacio lúdico es un lugar donde se pueden hacer dos cosas esenciales: se puede experimentar, aunque sea de modo virtual, y en segundo lugar y aún más importante, el estudiante puede equivocarse para aprender. Una de tantas dificultades en el desarrollo de la docencia es la incapacidad de dejar a los alumnos equivocarse y volver a empezar.

En el área particular de las ciencias exactas, estudios como el de Dávila [33] muestran que el uso de la gamificación unida a la implementación de un LMS en el aula de clase, no solo ayuda a mejorar el índice de aprobados, sino también a acrecentar la motivación hacia el estudio de estas disciplinas en específico.

Si tenemos en cuenta que la gamificación está dirigida a la inducción de la motivación en el acto del aprendizaje, entonces podemos relacionarla con los buenos resultados de los estudiantes, tales como: mejoras académicas, satisfacción y persistencia.

Por lo que el uso de la gamificación en un ambiente *e-learning* y utilizando Moodle además de otras herramientas puede ayudar en el incremento de la motivación por parte de los estudiantes, así como a la interacción profesor-estudiante.

2.2.2. Gamificación como apoyo al aprendizaje de programación

Como se ha señalado anteriormente, y a pesar de estar muy interrelacionados, la definición de gamificación no se basa en la creación o uso de videojuegos, sino en el aprovechamiento del componente motivador de estos, que, utilizados en entornos diferentes, procura atraer al usuario y lograr que realice ciertas acciones de forma satisfactoria.

Puede plantearse que para gamificar una actividad se parte desde la óptica del diseño de juegos. Según Cook [34], cualquier acción que cumpla las siguientes condiciones puede ser gamificada:

- La actividad puede ser aprendida.
- Las acciones del usuario pueden ser medidas.
- Las retroalimentaciones pueden ser entregadas de forma oportuna al usuario.

Por tanto, es muy factible que las actividades formativas en la asignatura de Programación I puedan ser gamificadas [6].

A lo largo de los años, numerosas técnicas se han utilizado para hacer la enseñanza de habilidades de programación más eficaz, mediante la adopción de un sistema de ambientes simplificados de aprendizaje. Los métodos generales incluyen la simplificación del lenguaje. Otro enfoque para hacer que el proceso de aprendizaje sea más efectivo para enseñar programación es el que promueve el uso de los juegos de ordenador. La capacidad de los juegos para enseñar habilidades de resolución de problemas a través de un ciclo repetido de experiencia [11] los hace ideales en la enseñanza de programación, ya que se requiere que los estudiantes desarrollen una jerarquía de habilidades a través de varias técnicas e incrementen sus conocimientos en la resolución de problemas prácticos [35].

Watson, Li y Lau [36] plantean que un método eficaz de enseñar conceptos de programación puede ser obtenido mediante la fusión de las capacidades de razonamiento con un juego altamente motivador. Investigaciones anteriores han sugerido que los ambientes de esta naturaleza pueden ser muy eficaces [27]. Sin embargo, los ejemplos de tales sistemas para la educación de programación son escasos. Para apoyar a los estudiantes en el desarrollo de una comprensión, el enfoque incluye diferentes formas de retroalimentación correctiva: implícita, explícita y a nivel de la lógica del programa, con el objetivo de transmitir conceptos de codificación de una manera más clara.

Estos mismos autores refieren que para enseñar los conceptos de programación a los estudiantes de una manera más visual e interactiva que de la manera en que se hace, puede utilizarse como soporte al sistema convencional de *e-learning*, una interfaz de aprendizaje basada en juegos [37]. La propuesta de Watson, Li y Lau [36] se diferencia de las aplicaciones estándar de los juegos de ordenador dentro de los cursos de programación, pues este enfoque no se limita a incluir juegos como un método para facilitar el proceso de aprendizaje, es decir, no solo combinan juegos con tareas de programación tradicionales para hacerlas más entretenidas y atractivas para los alumnos, sino que basan la singularidad de su enfoque en transformar la mecánica de un juego de computación en las tareas de

programación, de tal manera que el estudiante puede aprender a programar completando las tareas.

Por otra parte, el uso de esta estrategia ofrece una fuerte herramienta de motivación, donde los educandos pueden ver resultados inmediatos de la ejecución de código sin tener que dominar una gran cantidad de conocimientos. Por su parte, Anderson y McLoughlin [38] plantean que el escenario de juego también se puede personalizar, lo que permite mayor variedad de ejemplos y el aumento adicional de la motivación.

Watson, Li y Lau [36] proponen un método para la enseñanza de conceptos de programación mediante la generación de diferentes formas de retroalimentación correctiva con una interfaz de motivación basada en el juego para la visualización de los conceptos de codificación y la ejecución de código. A diferencia de los materiales de clase tradicionales, este concepto apoyado en las mecánicas de juego que ofrece la gamificación permite a los estudiantes desarrollar rápidamente una comprensión de los conceptos de codificación a través de un conjunto de ejemplos claros y visuales.

2.2.3. Framework D6 en el diseño de gamificación de la asignatura

El Framework D6 del profesor Werbach [39] propone seis pasos a seguir para el diseño de estrategias de gamificación; en este caso, tiene como objetivo guiar a los docentes en el proceso de diseño de una asignatura gamificada:

1. **Definir los objetivos del curso:** se especifican los resultados positivos que se esperan en el curso, se enfatiza en el objetivo final o los objetivos del diseño gamificado en lugar de en los medios a través de los cuales se va a lograr el objetivo.
2. **Diseñar las conductas objetivo:** se describen los comportamientos que queremos de los estudiantes y cómo estos ayudan a alcanzar los objetivos esperados, además de métricas para que los jugadores puedan saber cuándo están participando con éxito en los comportamientos deseados.

3. **Describir a los jugadores, en este caso a los estudiantes:** se debe tener claro qué tipo de elementos de juego y otras estructuras es probable que sean efectivas para los jugadores.
4. **Elaborar los bucles de actividad:** se explora en mayor detalle cómo motivar a los jugadores con el compromiso y la progresión de los bucles de actividades; y se describen los tipos de retroalimentación que tendrán para fomentar aún más la acción, y cómo esta trabajará para motivarlos. Para esto se crean tareas que mantengan una secuencia de realización de eventos y procesos que mantengan el interés del estudiante por seguir progresando. Se diseña el sistema de puntuación, niveles, clasificaciones, tableros, premios, insignias, narrativa, retroalimentación, etc.
5. **No se olvide de la diversión:** es algo muy importante al diseñar el curso gamificado, se busca que el estudiante se motive intrínsecamente. Se identifica qué aspectos del juego podrían seguir motivando a los estudiantes a participar, incluso sin recompensas.
6. **Implementar las herramientas adecuadas:** se describe por ejemplo si el sistema gamificado se usará principalmente en ordenadores personales, dispositivos móviles, o alguna otra plataforma; qué opciones específicas se necesitan para desplegar el sistema, etc. Por último, se hace un análisis de si todas las decisiones tomadas están atadas a los otros cinco pasos, especialmente a los objetivos del curso.

También es importante tener en cuenta el modelo para la gamificación de actividades educativas propuesto por González y Mora [6]:

- **Experimentación repetida:** se permitirá que el estudiante-jugador pueda realizar repeticiones de la actividad para alcanzar una meta.
- **Inclusión de ciclos de retroalimentación rápida:** es necesario proporcionar información inmediata que ayude a los estudiantes a mejorar su habilidad y conseguir un mejor momento para triunfar la próxima vez que lo intente.
- **Adaptación de las tareas a los niveles de habilidad:** los buenos juegos ayudan a los jugadores a valorar realmente sus probabilidades de triunfar. Los diferentes

niveles con objetivos adaptados a las habilidades de los estudiantes permiten mejorar su motivación

- **Intensificación progresiva de la dificultad de las tareas:** ayuda a los estudiantes a mejorar sus habilidades y suponen nuevos retos.
- **División de tareas complejas en subtarear más cortas y simples:** esto ayuda a los estudiantes a hacer frente a la complejidad de la tarea.
- **Diseño de diferentes rutas hacia el éxito:** la planificación de diferentes formas de alcanzar los objetivos es una forma de personalización de las actividades.
- **Incorporación de recompensas y actividades de reconocimiento social (por ejemplo, profesores y compañeros):** ser recompensado y valorado promueve el estatus social de los estudiantes [6].

2.3. Las TIC y el aprendizaje de la programación

Las TIC son reconocidas como el soporte que interviene en el procesamiento de la información, entre ellas se encuentran la informática, las telecomunicaciones y los distintos tipos de redes que logran integrar al hombre con los dispositivos que proporcionan el intercambio de información.

Las TIC tienen como principal función aumentar y transformar la información y desarrollar las comunicaciones para obtener mayores conocimientos.

Cuando es posible el buen uso de las TIC, estamos en presencia de ventajas tales como: la mejora de las comunicaciones, se acortan y hasta pueden eliminarse las barreras de tiempo y espacio, se beneficia el trabajo cooperativo, la calidad de vida aumenta debido a que se tienen nuevos productos y servicios para beneficio de la humanidad.

Por consiguiente, las TIC se caracterizan por ser básicamente interactivas, intangibles, poseen una enorme capacidad de intercomunicación, son instantáneas y diversas pues pueden manifestarse en formas tecnológicas y con diferentes aplicaciones.

Las TIC pasaron a desempeñar un rol importante en la educación en general, gracias a Internet, estudiantes y profesores pueden acceder a gran variedad de información; su

incidencia positiva en la sociedad y en particular en la educación ha replanteado el papel de la educación y del profesorado, debido a medios que se deben utilizar para su mejor funcionamiento.

La educación con tecnología hace más viable el proceso de enseñanza-aprendizaje en la asignatura Programación I, pues las herramientas de *software* facilitan la investigación y el conocimiento en general.

Estas herramientas están clasificadas en:

- Herramientas de calificación automática.
- Aplicaciones multimedia.
- Sistemas inteligentes de tutoría.
- Aprendizaje virtual [40].

Por tanto, la repercusión de las TIC se ha generalizado y es considerada una puerta abierta al conocimiento y a las transformaciones sociales.

2.3.1. El *e-learning*

La educación que se lleva a cabo a través de Internet se ha dimensionado y extendido a niveles mundiales y se materializa a través de la enseñanza virtual, o como se la conoce mejor, del *e-learning*, modo de formar a los estudiantes a partir de las posibilidades que ofrece Internet y con la interacción y predominio de la comunicación profesor-alumno.

Este modo de formación semipresencial ha logrado que esta educación llegue a un número mayor de personas, haciendo que desaparezcan las barreras espacio-tiempo, pero siempre el alumno es el centro, la formación será sencilla, y el profesor deja de ser un mero comunicador de contenidos para ser un guía que orienta y ayuda en los procesos formativos; la comunicación siempre será constante y los contenidos actualizados.

El *e-learning* también apoya el proceso de aprendizaje a través de las plataformas LMS donde el profesor y los estudiantes se acompañan de sitios en los que pueden acceder a

todos los recursos académicos. Son herramientas que apoyan el proceso de enseñanza presencial y permiten desarrollar la enseñanza semipresencial y no presencial [41].

Cuenta con un método de trabajo sincrónico y asincrónico donde los profesores y alumnos pueden interactuar instantánea e intermitentemente, sin tener que estar conectados ambos en el mismo momento, lo que aporta ventajas, como son: la motivación del estudiante, fomento del proceso individual de aprendizaje, la retroalimentación inmediata y el acceso no lineal a la información [42]. El uso de plataformas para entornos *e-learning* tiene como principal objetivo permitir la creación y gestión de los espacios de aprendizaje por Internet, donde los profesores y los alumnos se puedan comunicar e interaccionar con la información durante el proceso de formación.

Dicho espacio es el que permite al alumnado establecer competencias y motivarse para el curso, aunque se debe esclarecer que el profesor es la persona más importante en el curso para lograr que los estudiantes se motiven y venzan satisfactoriamente el mismo con notas relevantes [43]. Durante varios años se ha ido perfeccionando el término *e-learning* por lo que han surgido algunas definiciones dependiendo del espacio de aprendizaje, que puede ser el aula de un centro educativo, en la enseñanza presencial con el apoyo de las TIC, o los entornos virtuales, en la enseñanza semipresencial o no presencial con el apoyo de plataformas virtuales desarrolladas para el proceso docente a distancia.

Además del *e-learning* existen otros espacios de aprendizaje que se basan en la tecnología, entre ellos se conocen:

b-learning (blended learning)

Es la forma de aprender mediante la combinación de la enseñanza presencial con la tecnología no presencial: «which combines face-to-face and virtual teaching» [44, p. 5]. El término *blended learning* en el artículo publicado por Antonio Bartolomé [44, p. 5] expresa que «sigue una tendencia con una marcada raíz procedente del campo de la psicología escolar en la que destaca el término “aprendizaje” como contrapuesto al de “enseñanza”».

Esta modalidad de aprendizaje tiene como ventajas: aumento significativo en la correlación entre enseñar-aprender convirtiendo en más activo el proceso y centrado en el estudiante [44].

Esta modalidad de aprendizaje no consiste en poner a disposición materiales en Internet sino en aprovechar los materiales existentes en la red. La intención es no cambiar el medio sin necesidad y aprovechar lo existente: «Las líneas básicas del proyecto no son reproducir electrónicamente material didáctico cuyo soporte ideal es el impreso, sino aprovechar la enorme cantidad de información disponible en Internet» [45].

El *blended learning* puede ser alcanzado mediante el uso de recursos virtuales, físicos, o la mezcla de ambos, como es el uso de materiales basados en la tecnología y conferencias presenciales, favoreciendo el proceso de enseñanza eficaz.

Los materiales a utilizar son los que ya se encuentran disponibles en Internet, permitiéndoles a los estudiantes desarrollar diferentes destrezas como:

- Realizar búsquedas en Internet y encontrar información relevante.
- Desarrollar criterios para valorar la información y tener indicadores de calidad.
- Aplicar información a la elaboración de nueva información y a situaciones reales.
- Trabajar en equipo compartiendo y elaborando información.
- Tomar decisiones en grupo [44].

m-learning (mobile learning)

En los últimos años, el desarrollo de la tecnología ha avanzado mucho, existiendo, por ejemplo, teléfonos inteligentes con propiedades muy parecidas a las de un ordenador, todo un avance teniendo en cuenta que el concepto de *e-learning* ha evolucionado dando entrada, parcialmente o en su totalidad, a la participación de alumnos y profesores en acciones formativas a través de sus dispositivos móviles (*m-learning*).

El *m-learning* es considerado como el siguiente paso en el desarrollo del aprendizaje virtual, ya que maximiza la idea de aprendizaje desde cualquier lugar y en cualquier momento [46].

t-learning (transformative learning)

Es un subconjunto del *e-learning* basado en el aprendizaje transformativo, donde el uso de las tecnologías es un elemento esencial en el proceso. Es un proceso de enseñanza-aprendizaje basado en la correlación de tecnologías televisivas, telecomunicaciones y aplicaciones en correspondencia con el sector educativo. Ofrece la personalización de contenidos y el uso multimedia, además de integrar los modelos *e-learning* y *m-learning*. Según Bates [47], otras definiciones del *t-learning* son: aprendizaje interactivo a través de la televisión, o el acceso interactivo a contenidos educativos.

u-learning (ubiquitous Learning)

El *u-learning* o aprendizaje universal permite «trasladar el aprendizaje fuera del aula hacia distintos ambientes de la vida cotidiana apoyado en una tecnología flexible, invisible y omnipresente que nos provee la información que necesitamos en todo momento» [48, 49].

p-learning (pervasive learning)

Otro de los modelos que se puede considerar como una evolución del *m-learning* se denomina *p-learning* (*pervasive learning*, en español aprendizaje generalizado). En este modelo, se utilizan los «ordenadores que obtienen información sobre el contexto de aprendizaje a través de pequeños dispositivos inteligentes integrados como sensores o etiquetas» [48, 49].

Este modelo de aprendizaje generalizado tiene como objetivos:

- Proceder de acuerdo a los planes y objetivos de los usuarios.
- Ser sensible, capaz de adaptarse y responder a las habilidades, hábitos y necesidades del usuario.

- Ser accesible de una manera dinámica, omnipresente y segura a través de una interacción natural entre la información, los dispositivos y el usuario [48, 49].

c-learning (cloud learning)

El *c-learning* es un nuevo universo educativo basado en espacios de aprendizaje en la nube. Es decir, cualquier tipo de aprendizaje obtenido haciendo uso de medios sociales con espacios abiertos para la comunicación y colaboración, donde la formación se extiende a través de redes sociales y herramientas de realidad virtual como Second Life y OpenSim, redes sociales (Facebook, Tuenti), microblogs (Twitter) y blogs [50].

El autor de la presente investigación considera que la propuesta metodológica para el aprendizaje de Programación I en los estudiantes de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, con el apoyo de las TIC está enfocada principalmente en la modalidad *b-learning*, ya que la asignatura de Programación I es presencial pero se encuentra apoyada por una herramienta tecnológica como es la plataforma Moodle para motivar a los estudiantes, con componentes y recursos propios de la plataforma y *plugins* de terceros.

En la Figura 2 que aparece a continuación se representan las diferencias existentes entre el *e-learning* y el *b-learning* planteadas por la autora Calderón [51, p. 98].

E-LEARNING	B-LEARNING
El tutor dedica ciertas horas a sus estudiantes según lo preestablecido.	El docente dedica mayor cantidad de horas a sus estudiantes pues desempeña el papel de profesor en las clases presenciales y tutor en las actividades virtuales.
Educación por medio de herramientas tecnológicas.	Educación por tutorización con apoyo tecnológico.
La práctica se realiza ocupando fundamentalmente Internet.	La práctica se realiza bajo un diseño instrutivo que mezcla clases y/o actividades pedagógicas presenciales con clases y/o actividades 100 % en línea.
Cubre los objetivos del aprendizaje siendo sus puntos fuertes la interacción, comunicación, personalización.	A más de los puntos fuertes del <i>e-learning</i> se suman los de la educación presencial: trabajo directo de actitudes y habilidades.
Modalidad en línea que aprovecha las potencialidades de la web que son interacción, rapidez, flexibilidad, economía, acceso.	Modalidad que incluye formación presencial en el aula y las potencialidades de la web.
El estudiante aprende con la adquisición del conocimiento por medio del estudio o de la experiencia.	Es estudiante aprende a aprender que implica la capacidad de reflexionar en la forma en que se aprende y actúa.
Facilita la interacción entre estudiantes-estudiantes.	Facilita la interacción tanto entre profesores-estudiantes como entre estudiantes-estudiantes.
La retroalimentación entregada por el tutor es descriptiva.	La retroalimentación o entrega de información al estudiante acerca de su desempeño con el propósito de mejorarlo es descriptiva, específica, oportuna.
El tutor se convierte en orientador, facilitador y dinamizador.	El profesor tiene varios roles ya que debe ser tutor y profesor a la vez, siendo algunas de sus funciones: académica/pedagógica, social, organizativa, orientadora, de gestión, técnica.
Se necesitan herramientas educativas basadas en Internet para la publicación de contenidos.	Usa herramientas para las clases presenciales y herramientas educativas basadas en web para la parte en línea.

Figura 2. Diferencias entre *e-learning* y *b-learning*.

Fuente: Calderón [51, p. 98]

2.3.2. El uso de plataformas virtuales en la educación

Una LMS es una aplicación informática a la cual se puede acceder mediante la red de redes (Internet/Intranet), que permite a los profesores confeccionar materiales docentes y publicarlos a disposición de los alumnos en Internet. Facilita la interrelación pedagógica entre los implicados de un curso, que puede ser de modo virtual y semipresencial [52, p. 51], además sirve como apoyo en un curso presencial.

En la actualidad existen diversas plataformas virtuales desarrolladas para el ámbito educativo y que son utilizadas por instituciones como colegios, universidades, centros de aprendizaje o de impartición de cursos, empresas en general para la formación de su personal, etc. Todas las plataformas virtuales incorporan herramientas similares y se basan para su desarrollo en plataformas anteriores.

Como consecuencia de la aplicación de estas tecnologías, en el ámbito de la educación, el *e-learning* es definido como un conjunto de tecnologías, aplicaciones y servicios orientados a facilitar la enseñanza y el aprendizaje a través de Internet/Intranet, proporcionando el acceso a la información y a la comunicación con otros participantes. Cuando se combinan con clases presenciales, se denomina *b-learning*. Para la implementación de este tipo de estrategias formativas resultan de especial interés las plataformas educativas LMS.

2.4. Plataforma Moodle

Moodle es una plataforma de aprendizaje diseñada para proporcionarles a educadores, administradores y estudiantes un sistema integrado único, robusto y seguro para crear ambientes de aprendizaje personalizados [53].

Esta plataforma virtual de aprendizaje fue creada en 1999 por Martin Dougiamas como su tesis de doctorado, es un sistema de gestión de contenido de código abierto y gratuito que apoya el aprendizaje basado en el «construccionismo social». A continuación, se muestran algunas características de Moodle que lo hacen flexible y configurable [54]:

- Los profesores con rol de editor del curso pueden añadir, quitar y modificar actividades, secciones y/o bloques, según su necesidad.

- Dependiendo del contexto se definen los diferentes roles de la plataforma.
- Permite realizar calificaciones y actividades según la configuración que realice el profesor.
- Es flexible en cuanto a modificar las preferencias de apariencia a nivel de sitio, curso y actividad.
- Es compatible con otros sistemas externos y cuenta con varios módulos y bloques que facilitan el desarrollo de cursos.

Moodle fue diseñado con el objetivo de facilitar la creación de cursos en línea. Es una plataforma educativa basada en las ideas del constructivismo social, permitiendo que se convirtiera en una de las plataformas de aprendizaje a distancia más extendidas y usadas a nivel mundial, con una amplia comunidad de usuarios.

Moodle contiene varias herramientas de comunicación que permiten que los cursos sean dinámicos y entre los usuarios de un curso puedan intercambiar información, mediante los foros (debates en la web), el chat (conversación en tiempo real), el glosario (vocabulario creado en común), los calendarios, las *wikis*, encuestas, cuestionarios, tareas o recursos, el taller donde cada alumno es evaluado.

Esta plataforma cuenta con una amplia comunidad de usuarios, lo que facilita el desarrollo y soporte de soluciones para las distintas versiones disponibles en sus foros oficiales, *wiki* y sistema de registro de incidentes. Todos esos archivos se pueden subir y manejarse en el servidor, o pueden ser creados sobre la marcha usando formularios web (de texto o HTML) [55].

Estos entornos virtuales le permiten al estudiante establecer su ritmo de trabajo, la entrega de actividades, las cuales son monitoreadas y evaluadas por el profesor. También ayuda a los educadores a crear comunidades de aprendizaje en línea, a configurar herramientas externas para motivar a los estudiantes en el aprendizaje de programación ya que pueden implementar código fuente en diferentes lenguajes de programación y en la misma interfaz realizar correcciones, pudiendo saber si lo que están realizando es correcto, siendo este un

importante elemento a tener en cuenta por los profesores cuando van a realizar un examen evaluativo de programación.

Un profesor que opera desde este punto de vista crea un ambiente centrado en el estudiante que le ayuda a construir ese conocimiento en base a sus habilidades y conocimientos propios en lugar de simplemente publicar y transmitir la información que se considera que los estudiantes deben conocer [56].

Moodle cuenta con componentes que permiten crear cursos de fácil entendimiento para los estudiantes, contribuyendo al proceso de enseñanza-aprendizaje basado en el principio de comunicación entre estudiantes o estudiante-profesor. En la plataforma Moodle 2.x o 3.x estándar hay 14 diferentes tipos de actividades principales. Ejemplos de las actividades son:

- **Foros:** la suscripción a un foro puede ser personal o el profesor puede hacerla manual y de cualquier forma le llegará a cada participante una copia de los mensajes a su correo electrónico. Permite debatir cualquier tema, y siendo bien configurado se puede realizar evaluaciones de la participación en el mismo. Se puede adjuntar imágenes y configurar la forma en que se visualizan los mensajes en el foro [56].
- **Wikis:** potencian la colaboración entre los miembros de un grupo, para realizar conocimiento consensuado [56].
- **Chat:** es el espacio donde los usuarios participan en una discusión sincrónica en tiempo real de un tema específico [56].
- **Consulta:** el profesor puede hacer preguntas, con el objetivo de obtener criterios del grupo sobre algún tema [56].
- **Cuestionario:** otra forma de evaluar a los estudiantes, diseñado para facilitar el tipo de selección: opción múltiple, falso/verdadero y respuestas cortas, entre otros. Además, los cuestionarios pueden tener fechas u horarios determinados para realizarse y dependiendo de eso se da la opción de realizar uno o varios intentos del mismo cuestionario. La calificación es de cada intento que se realice y de forma automática puede o no mostrar la calificación y/o las respuestas correctas. Todo depende de cómo el profesor configura el cuestionario [56].

- **Encuesta:** provee a los profesores de varios instrumentos para analizar y estimular el aprendizaje en ambientes en línea, conociendo lo que piensan los estudiantes sobre un tema y poder reflexionar así sobre su práctica educativa [56].
- **Etiqueta:** es la anotación que usan los profesores para aclarar algún aspecto en el curso [56].
- **Glosario:** este recurso permite la creación de un glosario de términos referentes al curso, el cual puede ser generado por los profesores o los estudiantes [56].
- **Asignatura:** se puede acceder a la información que el profesor desea hacerles llegar a los estudiantes, ya sea en formato Word, Power Point, Excel, Acrobat, o cualquier otro que la plataforma reconozca [56].
- **Taller:** facilita el trabajo en grupo y la evaluación de las actividades que envía el profesor. También coordina la recopilación y distribución de esas evaluaciones [56].
- **Tarea:** le permite al profesor asignar trabajos, calificarlos y hacer comentarios sobre los archivos subidos, ya sea en línea o fuera de línea [56].
- **Lección:** proporciona contenidos de forma flexible para repasar conceptos o contenidos que el profesor crea importantes [56].
- **Herramienta externa:** los profesores pueden utilizar estas herramientas para dar mayores facilidades a los estudiantes y aumentar la motivación en el curso ya que existen varias de este tipo como VPL, javaunittest, Ranking Block, Progress Bar, etc. Los usuarios interactúan con otros recursos y actividades de enseñanza-aprendizaje compatibles con la plataforma [56].
- **Sharable Content Object Reference Model (SCORM):** permite incluir paquetes SCORM como contenido del curso [56].

Ventajas de la plataforma Moodle:

- **Alta disponibilidad:** es lo suficientemente robusto como para satisfacer las diversas necesidades de miles de estudiantes, administradores, creadores de contenidos y profesores simultáneamente [55].
- **Escalabilidad:** es posible ampliar su infraestructura o escalar para resolver el futuro crecimiento, tanto en términos de volumen de contenidos educativos como del número de estudiantes [55].
- **Facilidad de uso:** apoya un conjunto de servicios automatizados y personalizados, tales como aprender a ritmo individual y perspectivas específicas de aprendizaje; el acceso, la entrega y la presentación de materiales son fáciles de utilizar y muy intuitivos como navegar por la web [55].

La plataforma Moodle permite la importación/exportación de objetos reutilizables de aprendizaje empaquetados de acuerdo con los estándares IMS Content Packaging y SCORM. Las preguntas de los cuestionarios pueden ser exportadas en el formato estándar internacional IMS Question & Test Inoperability 2 (IMS QTI 2). Los canales de noticias RSS pueden ser integrados a otros sistemas o sitios web con funcionalidad RSS y acceder a los foros como noticias RSS [56].

Bases pedagógicas de Moodle

Moodle está cimentado en un modelo pedagógico de constructivismo social, basado en cuatro conceptos principales y subyacentes:

- **Constructivismo:** este punto de vista mantiene que el usuario renueva constantemente los conocimientos a partir de su interacción con lo que le rodea. Todo lo que ocurre a su alrededor entra en contradicción con sus experiencias anteriores y si logra integrarlo en el mundo interior de su mente, podrá formar el nuevo conocimiento del que se apropiará para siempre. Este conocimiento será mayor si logra integrarlo con éxito en el entorno más cercano [57].
- **Construccionismo:** describe que el aprendizaje es muy efectivo cuando se fabrica algo que se transmite a otros. Esto puede ir desde una frase propiamente dicha o

mandar un mensaje en Internet, a otros artilugios más complejos como un cuadro, un departamento o un paquete de *software* [57].

- **Constructivismo social:** extiende las ideas anteriores a la construcción de elementos de un grupo social hacia otro, para crear una cultura colaborativa de artefactos con significados compartidos. Cuando las personas forman parte de una cultura de este tipo, aprenden todo el tiempo a formar parte de esa cultura en todos los aspectos [57].
- **Conectados y separados:** esta idea explora más profundamente las motivaciones de los individuos en una discusión. Un comportamiento separado es cuando alguien intenta permanecer «objetivo», se remite a situaciones específicas y tiende a cuidar sus propias ideas a partir de la lógica y tratando de encontrar brechas en los razonamientos de sus contrarios. La conducta constructiva se da cuando alguien puede apropiarse de estas aproximaciones y tiene la capacidad de seleccionar una de ellas como la conveniente para cada caso particular [57].

2.4.1. Gamificación en Moodle

La gamificación en Moodle se ha convertido en una de las soluciones más efectivas para llevar la enseñanza a otro nivel y proporcionar una forma de aprendizaje más motivadora y actual; muchas de las instituciones educativas han tomado partido a favor de Moodle por las múltiples ventajas que ofrece este entorno virtual de enseñanza-aprendizaje y por las posibilidades para gamificar el aula virtual.

Moodle tiene varias características y funcionalidades propias de la plataforma o con *plugins* de terceros destinadas a facilitar el proceso de aprendizaje usando gamificación, entre ellas: perfiles, avatares, puntos y recompensas, visibilidad del progreso de los estudiantes, retroalimentación, visualización del resultado de cuestionarios o lecciones, foros, niveles, insignias, notificaciones, tabla de posiciones, actividades condicionales para restringir el acceso a contenidos, etc. [58].

Desde el punto de vista de la programación, Moodle posee el *plugin* VPL mediante el cual se puede gestionar tareas de programación en varios lenguajes, y en particular Java y

programar en el mismo entorno de Moodle. Además, los estudiantes pueden compilar el código fuente de los programas para detectar si tiene algún error, ejecutar de forma interactiva los programas en el navegador, y detectar el antiplagio ya que busca similitud entre los archivos subidos por los estudiantes, establece restricciones de edición y evita pegar texto externo [59, 6]. Estos elementos mencionados anteriormente hacen que los estudiantes sean motivados, ya que saben si el código que escribieron se ejecuta correctamente, obtienen retroalimentación en línea y ven sus notas automáticamente cuando envían la tarea; sin embargo, cuando no se utilizaba este medio tenían que esperar a que el profesor pudiese revisarla para saber su nota.

Algunos *plugins* que permiten el uso de la gamificación en Moodle son:

Módulos de actividades y recursos

- **Asistencia (*mod_attendance*):** sirve para registrar la asistencia de los estudiantes a clases presenciales.
- **Certificado simple (*mod_simplecertificate*):** habilita entregar a los estudiantes un certificado o credencial que los acredita como aprobados en el curso.
- **Laboratorio virtual de programación (*mod_vpl*):** permite codificar, compilar, ejecutar y evaluar automáticamente programas escritos en los principales lenguajes de programación desde Moodle.

Bloques

- **Barra de progreso (*block_progress*):** muestra las actividades o recursos programados en el curso, su estado actual, avance, progreso, fecha esperada de cumplimiento de manera amigable.
- **Bloque de *ranking* (*block_ranking*):** facilita el despliegue de un tablero y la clasificación con la puntuación otorgada según las actividades que vaya completando el alumno.
- **Estadísticas de uso (*block_use_stats*):** acumula el tiempo que el alumno pasa en cada una de las opciones de un curso en la plataforma.

Tipos de preguntas

- **javaunittest** (*qtype_javaunittest*): evalúa automáticamente la respuesta de un método escrito en Java.

Temas

- **Esencial** (*theme_essential*): es uno de los módulos que se emplean para personalizar la imagen de Moodle.

2.5. Antecedentes a la investigación

Para el desarrollo de este estudio se hizo necesario el análisis de varias investigaciones que contribuyeron a la consolidación y estructuración del mismo. Los antecedentes a esta investigación se detallan a continuación:

Steinmann, Bosh y Aiassa [5] indagaron sobre la motivación, las expectativas, las actitudes frente al aprendizaje y sobre la metodología de enseñanza de ciencias en estudiantes de educación secundaria y universitaria de la ciudad de Río Cuarto, Argentina. La investigación tuvo carácter exploratorio-descriptivo, y se aplicó una encuesta a estudiantes de estos niveles [5].

Proponen que habría que trabajar para lograr una mejor enseñanza, que valore las motivaciones y los pensamientos de los demás, que sea cuidadosamente pensada y que proporcione experiencias y logros significativos, tanto desde el punto de vista emocional como profesional y cognitivo.

En la Universidad de los Andes en Colombia, se realizó el proyecto CUIP2 [60], un estudio que mejora la motivación en los estudiantes de programación, y en el que se propone una solución integral al problema de enseñar y aprender a programar. Menciona que en la enseñanza de la programación existe en general un bajo nivel de motivación entre los estudiantes y en consecuencia altas tasas de abandono. A esto se une la falta de un estudio a fondo de las habilidades que los estudiantes deben adquirir y, por consiguiente, muchas

veces se reduce el programa de los cursos a un recorrido de estructuras sintácticas de un lenguaje de programación específico [60].

El proyecto CUIPI2 establece una estrategia para enfrentar estos problemas, que además pudiera alinearse con la visión del perfil de los profesionales en ingeniería. El objetivo de los cursos de programación no es únicamente que el estudiante aprenda a escribir programas, sino que deben generar una variedad de competencias asociadas con cualquier profesional en ingeniería (aprender a entender un problema, a plantear soluciones efectivas, a manejar lenguajes para expresar una solución, a utilizar herramientas que entiendan esos lenguajes, a probar que la solución sea válida, a justificar las decisiones tomadas, etc.) [60].

Jenkins [4], de la Universidad de Kent, investiga sobre las motivaciones de los estudiantes para cursar una carrera de informática, y en particular para el estudio de programación. Concluye que la programación es una habilidad difícil de adquirir, que se aprende mejor con la práctica y que será más efectiva si la práctica es autodirigida. El papel del profesor, por lo tanto, es clave para persuadir a los estudiantes a realizarla y así motivarlos [4].

Cacho y Rodrigo [58] proponen un diseño especificando un conjunto de elementos y componentes de juego para gamificar un LMS de código abierto, en particular Moodle LMS, y crearon un prototipo sobre la base de este diseño. Proponen realizar la medición de la eficacia de su prototipo en un experimento con maestros y alumnos [58].

En particular en la educación universitaria de programación, Larsen [61] crea una plataforma gamificada, Game of Exams, para aprendizaje de programación en su trabajo de fin de máster, explorando las facetas de la motivación relacionada con la autonomía y la competencia; en las conclusiones indica que se deben tener herramientas que permitan la creación de ejercicios creativos de programación y poder evaluarlos automáticamente, para que los estudiantes adquieran la habilidad de resolver ejercicios en forma autónoma, y anima a una mayor investigación de la metodología en un proyecto similar al de su tesis [61].

Martínez y Duarte [62] aplicaron la gamificación para aprender ciencias de la computación en carreras no relacionadas con la informática. Mencionan que, al transformar las

actividades de clase en desafíos, fomentando el liderazgo de los participantes, recompensando la eficiencia y el esfuerzo en el desarrollo de actividades, los estudiantes consiguen las siguientes competencias: pensamiento lógico, capacidad para seguir procedimientos complejos, reconocimiento de patrones, uso de tecnología y comunicación, trabajo colaborativo y resolución de problemas. Para esto evaluaron los sentimientos de los estudiantes con una encuesta, y presentaron el análisis de los datos derivados de la experiencia comparando los resultados con un grupo que tomó la misma clase con un enfoque normal [62].

Arenas [63] describe un modelo de aprendizaje de programación como un complemento que promueva la motivación intrínseca usando técnicas de gamificación. Se basa en el Gamification Framework D6 propuesto por Werbach [39]. Realiza un prototipo de una herramienta computacional que incluye algunos de los elementos del modelo usando WordPress CMS y el *plugin* BadgeOS para demostrar la factibilidad técnica de su implementación. Propone que el prototipo puede funcionar como base para un desarrollo posterior que incluya todos los elementos de juego y pueda ser utilizado en un salón de clases como herramienta de apoyo al aprendizaje [63].

Behnke [64] investiga el impacto de la gamificación en la motivación (intrínseca y extrínseca) y el aprendizaje del estudiante en varias actividades educativas introductorias en ciencias de la computación. El uso de técnicas de diseño de juegos en la educación ofrece el potencial de hacer que el aprendizaje sea más motivador y agradable para los estudiantes. Esta investigación demuestra lo complejo que puede ser el proceso de aprendizaje: lo que funciona para un estudiante en un contexto educativo, no necesariamente funciona para otros en un contexto diferente [64], de ahí la importancia de diseñar actividades centradas en el estudiante tomando en cuenta la realidad y cultura donde se apliquen.

Por consiguiente, este trabajo de investigación tiene la finalidad de cambiar la visión sobre el proceso educativo como un proceso tradicional, a partir de la introducción de la gamificación en el desarrollo del aprendizaje, de combinar el placer con el aprender, lo que se traduce en una renovación de las técnicas de motivación, en utilizar lo agradable de los juegos en la realización de tareas autónomas de programación.

Se partió de conceptos establecidos para cambiar una realidad que se hace difícil, y es el hecho de que los estudiantes de programación no tienen suficiente motivación, pues las clases pueden llegar a ser monótonas o muy densas, por lo que el cansancio puede vencerles, por el no gusto de realizar tareas fuera del aula, así que la gamificación, con sus aportes y la combinación de lo divertido y eficaz, es la respuesta más acertada que se ha encontrado.

CAPÍTULO III. CONTEXTO EN QUE SE DESARROLLA EL ESTUDIO

3.1. El desarrollo de las TIC en la enseñanza universitaria en Ecuador

En la actualidad, las discusiones teórico-filosóficas que se desarrollan en el campo de la educación se centran en la necesidad de optimizar la calidad de la enseñanza que se brinda a las nuevas generaciones. En el caso de América Latina, y Ecuador en específico, la mayor preocupación está generada por los pobres resultados académicos que obtienen los estudiantes en asignaturas básicas como matemáticas y lengua materna. De acuerdo con Briones, Rivas, Viteri y Estrada [65]:

El Ecuador, desde siempre ha cargado con un nivel de educación deficiente. Esto resulta preocupante para una sociedad que se encuentra en vías de desarrollo, ya que la educación es uno de los pilares fundamentales para forjar una nación exitosa y libre. Entre los principales problemas que ha tenido la educación pública se encuentran la falta de infraestructura, poca preparación de los docentes, sistema muy centralizado, salarios bajos de los profesores, entre otros [65, p. 1].

Esto promueve la necesidad de proyectar investigaciones sobre estrategias para el aprendizaje e innovaciones pedagógicas que incentiven y aumenten la eficacia de la educación. En tal sentido se inscriben los estudios donde se integran las TIC al proceso de enseñanza-aprendizaje, así lo reflejan Fernández, Carballos y Delavaut [66]:

Las TIC ofrecen un nuevo reto al sistema educativo: pasar de un modelo unidireccional de formación, donde por lo general los saberes recaen en el profesor o en su libro de texto, a modelos más abiertos y flexibles, donde la información situada en grandes bases de datos tiende a ser compartida entre diversos estudiantes. Por otra parte, se rompe la exigencia de que el profesor esté presente en el aula y tenga bajo su responsabilidad a un único grupo de estudiantes. El estudiante puede interactuar con otros compañeros y profesores que no tienen porqué estar situados en su misma sala [66, p. 137].

La inclusión de las TIC en la universidad es un hecho estratégico para el desarrollo de la enseñanza en este nivel de estudios. Estas ofrecen nuevas rutas de aprendizaje, transforman el papel del profesor y ponen en cuestionamiento los métodos pedagógicos actuales. La eventualidad de acceder a un flujo mayor de información hace que el docente ceda su acción transmisora de conocimientos y centre su trabajo en el aprendizaje. En tal caso, la instrucción universitaria se encamina a liberar los métodos de aprendizaje con el propósito de guiar al alumno hacia la elaboración de su propia sapiencia a través del grupo de elementos y de información aprovechables:

Las TIC no pueden suponer por sí solas una garantía de cambio positivo en la universidad, estos retos que es necesario afrontar son, entre otros: nuevos programas docentes, el control de la calidad de los materiales y servicios virtuales, buenas prácticas docentes en el uso de las TIC, formación del profesorado. Las potencialidades de las TIC en la universidad dependerán no solo de su modalidad de uso, que puede ir desde la educación a distancia hasta la educación presencial, sino del modelo educativo en el que se apoyen, de la manera de concebir la relación profesor-alumno, de la integración de estas técnicas como un medio más para repetir patrones educativos anteriores o para modificarlos [67, p. 214].

Al analizar las experiencias en Latinoamérica acerca de la introducción de las TIC en la universidad, la mayoría de las investigaciones están encaminadas a documentar el fenómeno de la educación virtual y, en forma menor, a la modalidad híbrida o mixta. No deja de ser curioso este fenómeno, dada la cantidad de universidades interesadas en introducir mejoras y modernizaciones en su docencia presencial, mediante la introducción de las TIC [68].

En América, como manifiesta Bustos [68], la innovación se ha basado en el enriquecimiento de la docencia presencial con la ayuda de la tecnología, favoreciendo a los docentes en la creación de tácticas encaminadas a provocar en los estudiantes una optimización del aprendizaje curricular, así como el progreso en habilidades y competencias que no se adquieren con facilidad en la modalidad presencial.

Ecuador también se introduce en este ámbito de aprendizaje, así ocurre en la Universidad de las Américas que dicta maestrías, que desarrolla un periodo interactivo vía Internet con el envío y recepción de material pedagógico y trabajo, chats en videoconferencias entre los catedráticos y los estudiantes [69].

El entorno virtual de aprendizaje de la Pontificia Universidad Católica de Ecuador (PUCE) es el llamado PUCE Moodle, plataforma que se utiliza esencialmente como complemento de la enseñanza presencial. La universidad hace uso de las TIC como herramientas para comunicación síncrona y asíncrona, las videoconferencias como forma de establecer un diálogo didáctico mediado por el computador. Los educadores tienen a su alcance instrumentos de la web 2.0 que favorecen compartir conocimiento, brindar repositorios de objetos de aprendizaje, mensajería instantánea, bibliotecas digitales y redes sociales [70].

La propuesta de la Universidad Salesiana de Ecuador (UPS) creó un modelo que se define por la unicidad de la propuesta, en la filosofía, en la misión y en la metodología guía del proceso de formación. La UPS busca además la internacionalización. El modelo intenta que la infraestructura tecnológica de base sea desarrollada en el curso de las fases de implementación, para tener herramientas dinámicas de apoyo a las actividades de producción. Se trata de un modelo abierto que une las tecnologías, los contenidos y los modelos formativos en un reporte dialéctico que advierte sus propias potencialidades, orienta el desarrollo de los modelos formativos y favorece la revisión de los contenidos [71].

Por otro lado, en la Universidad Técnica de Ambato se incentivan estudios para la utilización de la plataforma virtual Moodle en la carrera de Docencia en Informática de la Facultad de Ciencias Humanas y de la Educación [72], debido a que por sus características de herramienta de código abierto u *open source* es considerada un instrumento adecuado para la implementación de sistemas de gestión de aprendizaje en la educación, en la optimización de la enseñanza-aprendizaje perfectible que desea ofrecer esa universidad.

Por su parte, la Universidad Estatal del Sur de Manabí (UNESUM) desarrolló un diagnóstico sobre la poca utilización de entornos virtuales y la necesidad de implementarlos para que las universidades ecuatorianas entren en una nueva era. En cuanto a usos de la

educación virtual en esta institución, la única experiencia reconocida es el proyecto de implementación de una plataforma Moodle en posgrado, trabajo que se llevó a cabo con la premisa de mejorar la eficacia del proceso educativo en las maestrías con modalidad semipresencial [73].

La Escuela Superior Politécnica de Chimborazo (ESPOCH) desarrolló en 2015 un estudio para determinar el uso que dan los maestros de la Facultad de Informática y Electrónica (FIE) a los LMS; en su análisis establecieron la existencia de 386 entornos virtuales activos creados por los profesores. Los cursos virtuales en esa universidad se distribuyen en cuatro carreras de ingeniería que conforman la FIE: Sistemas Informáticos (119), Diseño Gráfico (54), Ingeniería Electrónica en Control (113) e Ingeniería Electrónica en Telecomunicaciones (100). En los resultados de esa indagación se plantea que los profesores de ese centro de estudios utilizan los entornos virtuales como instrumento para apoyar la labor docente presencial, además, se especifica su utilización para compartir información, recibir tareas, enviar comunicados, y como espacio combinado con el aula física (*blended learning*) [74].

De forma general, en los intentos de introducir las tecnologías en las universidades ecuatorianas existe la tendencia a colocar en el extremo izquierdo de una línea horizontal imaginaria al modelo de educación presencial tradicional y en el otro extremo al modelo de educación totalmente virtual [70]. Cada centro de estudios define su propio modelo trasladándose entre los dos extremos. La mezcla de educación presencial y virtual se establece en atención al tipo de educación que se ofrece, al contexto donde se coloca la institución y a las líneas estratégicas que busquen las diferentes universidades.

En la educación superior ecuatoriana, la virtualización se identifica en la identificación de actores, información, conocimientos, procesos y objetos asociados a actividades de enseñanza-aprendizaje, investigación, extensión y gestión, así como instrumentos cuya manipulación permite al usuario realizar operaciones a través de Internet, tales como aprender mediante la interacción con cursos virtuales, suscribirse a foros, consultar documentos en una biblioteca virtual, comunicarse multidireccionalmente entre profesor y alumno y entre estudiantes [70].

Como afirma Negrete [70], en las instituciones de enseñanza de prestigio que son normalmente presenciales, se hace imprescindible la necesidad de asumir las TIC dentro de su patrón educativo, se entrevé la importancia de contar con modelos pedagógicos actualizados, concentrados en el alumno, donde estos asuman el protagonismo de un aprendizaje significativo y no como actores pasivos frente al reto que simboliza una educación innovadora de esta época.

En este punto es importante exponer que para una mejor reestructuración de la integración de las TIC en el ámbito educativo, es relevante que este sea un proceso amparado por normativas legales en las que el Estado ofrezca respaldo económico y legal a las universidades, las instituciones públicas y privadas o cualquier centro educativo: la Constitución de la República en el Título Séptimo: Régimen del Buen Vivir, Capítulo Primero: Inclusión y Equidad, Sección Uno: Educación, el art. 347 numeral 8 dice:

Art. 347.- Será responsabilidad del Estado: 8. Incorporar las tecnologías de la información y comunicación en el proceso educativo y propiciar el enlace de la enseñanza con las actividades productivas o sociales [75, p. 71].

Un artículo más concreto en lo respectivo al respaldo del Estado a la inserción de las nuevas tecnologías en la educación ecuatoriana es el 350:

Art. 350.- El sistema de educación superior tiene como finalidad la formación académica y profesional con visión científica y humanista; la investigación científica y tecnológica; la innovación, promoción, desarrollo y difusión de los saberes y las culturas; la construcción de soluciones para los problemas del país, en relación con los objetivos del régimen de desarrollo [75, p. 162].

En otra instancia, la Ley Orgánica de Educación Superior plantea los recursos que las entidades educativas necesitan para su desarrollo académico y en el aprovechamiento de las nuevas tecnologías como un instrumento pedagógico. En su Título Segundo: Autonomía Responsable de las Universidades y Escuelas Politécnicas, Capítulo Segundo: Patrimonio y Financiamiento de las Instituciones de Educación Superior, en su artículo 35 la ley especifica:

Art. 35.- Asignación de recursos para investigación, ciencia, tecnología e innovación.- Las instituciones del Sistema de Educación Superior podrán acceder adicional y preferentemente a los recursos públicos concursables de la preasignación para investigación, ciencia, tecnología e innovación establecida en la ley correspondiente [76, p. 10].

Otros artículos que refieren lo primordial de la introducción de las TIC en las universidades de Ecuador son los artículos 143 y 144 de la Ley Orgánica de Educación Superior. En tal sentido, en el Título Séptimo: Integralidad, Capítulo Primero: Del principio de integralidad, Sección Tercera: Del Funcionamiento de las Instituciones de Educación Superior se expone:

Art. 143.- Bibliotecas.- Las instituciones de educación superior públicas y particulares desarrollarán e integrarán sistemas interconectados de bibliotecas a fin de promover el acceso igualitario a los acervos existentes, y facilitar préstamos e intercambios bibliográficos. Participarán en bibliotecas digitales y sistemas de archivo en línea de publicaciones académicas a nivel mundial [76, p. 23].

Art. 144.- Tesis Digitalizadas.- Todas las instituciones de educación superior estarán obligadas a entregar las tesis que se elaboren para la obtención de títulos académicos de grado y posgrado en formato digital para ser integradas al Sistema Nacional de Información de la Educación Superior de Ecuador para su difusión pública respetando los derechos de autor [76, p. 23].

Lo más importante de estos artículos es que el Estado ecuatoriano haya contemplado la creación de leyes específicas para respaldar el proceso de enseñanza universitaria; le brinda tanto a este como a las universidades y escuelas politécnicas obligaciones referentes a la actualización del sistema educacional, emprendiendo en lo referente a la implementación de equipos tecnológicos de acuerdo a las particularidades de las diferentes instituciones educativas, el sustento al profesor con adiestramientos en el uso de las TIC para que este integre dichos conocimientos en sus asignaturas, y sobre la pertinencia del fortalecimiento de la enseñanza en el país de la educación superior [69].

3.2. Universidad Central de Ecuador. Principales características del centro de estudios

La Universidad Central de Ecuador (UCE) se sitúa en una parte de la Ciudadela Universitaria, establecida en el centro-occidente de la ciudad de Quito, en la Av. América y Av. Universitaria. Su perímetro geográfico es de 203 507,61 m², para un total de 38 340 alumnos matriculados hasta 2016.

La UCE tiene 19 facultades, la mayoría de ellas dispuestas dentro del campus universitario, las cuales se mencionan a continuación:

Artes, Arquitectura y Urbanismo, Ciencias Administrativas, Ciencias Agrícolas, Ciencias Biológicas, Ciencias de la Discapacidad, Ciencias Económicas, Ciencias Médicas, Ciencias Psicológicas, Ciencias Químicas, Comunicación Social, Cultura Física; Ingeniería, Ciencias Físicas y Matemática; Filosofía, Letras y Ciencias de la Educación; Ingeniería en Geología, Minas, Petróleo y Ambiental; Ingeniería Química; Jurisprudencia, Ciencias Políticas y Sociales; Medicina Veterinaria y Zootecnia; Odontología.

De acuerdo con Moreno [77], el modelo pedagógico de la Universidad Central de Ecuador se basa en:

Para la Universidad Central el modelo pedagógico tiene una estructura didáctica compuesta por: el sujeto cognoscente, a través de un proceso de enseñanza-aprendizaje y comunicación; el objeto de conocimiento, compuesto por conocimientos, saberes, prácticas, expresiones y manifestaciones; y la acción que los vincula y los constituye [77, p. 9].

En atención a estos preceptos, se concuerda con la concepción constructivista que esboza la importancia de brindar al estudiante «las herramientas que le permitan crear sus propios procedimientos para resolver una situación problemática. Aquí el rol del docente es de un modelador, coordinador, facilitador, mediador y también un participante activo» [77, p. 9].

En atención a lo anterior, en este centro académico se buscó introducir los métodos del *e-learning*, con el uso de instrumentos electrónicos y multimedia para la enseñanza

universitaria, dispuestos en conjunto con las estrategias pedagógicas clásicas. La introducción de las TIC se caracterizó, en un primer paso, por la preparación de los docentes a partir de la capacitación para el funcionamiento de aulas virtuales. Luego, en los últimos años se ha buscado la modernización y conexión general dentro de la UCE. Es así que se llevó a cabo el diseño total de conectividad de la universidad, teniendo varios componentes:

- Integración de las facultades y la administración central a través de un campus conectado con fibra óptica a 10 GB.
- Implementación de una red inalámbrica para todo el campus con alrededor de 900 *access points*, para garantizar el acceso a Internet y a la Intranet [78].

El objetivo de esto fue crear una red inalámbrica para la UCE que sea centralizada, permitiendo así la conexión de puntos de acceso, puntos de distribución y núcleos para la comunicación de sus facultades y sedes. El diagrama lógico que se implementa se compone de enlaces de fibra óptica para el *back bone* del campus, *access points* de cuatro tipos diferentes, controladoras de los dispositivos inalámbricos, *switching* de distribución, de borde, de acceso y un sistema de monitoreo general para implementar acuerdos de nivel de servicio SLA (*Service Level Agreement*, por sus siglas en inglés), trabajando en una estandarización de procedimientos bajo las buenas prácticas de *IT Infrastructure Library* (ITIL) [78].

Para el diseño de la red se implementan:

- *Wireless* externo TIPO1, con antenas externas para cubrir áreas extensas abiertas y ubicadas sobre los edificios altos del campus.
- *Wireless* externo TIPO2, con antenas internas para cubrir áreas cerradas como parques pequeños o entre edificios.
- *Wireless* interno TIPO1, con antenas internas por seguridad ya que estarían ubicadas dentro de aulas de clases, pasillos, salas de reuniones, laboratorios, etc.
- *Wireless* interno TIPO2, ideales para áreas interiores con alta densidad de usuarios como auditorios y bibliotecas [78].

El proyecto de conexión incluye las siguientes fases:

- *Data center*: sistema convergente de servidores, almacenamiento, contra incendio, accesos biométricos, arreglos de obra civil, sistema eléctrico, aire acondicionado.
- Seguridades: implementación de *firewall* de próxima generación.
- *Software* de virtualización y sistemas operativos [78].

Una de las arterias importantes que soporta la colaboración en este nuevo modelo es la transformación de la universidad hacia una *smart university* acoplada a la innovación; entre los principales logros se tiene la primera *hackathon* de innovación para aplicaciones que permitan convertirla en una universidad inteligente. Las aplicaciones que se presentaron fueron:

- Medición de radiaciones UV.
- Sistema de control de inventarios con *Radio Frequency Identification* (RFID) para el Laboratorio de Innovación UCE LAB.
- Sistema de registro de asistencia con sensores de presión implementados en sillas.
- Sistemas de control de ingreso de alumnos con RFID y huella.
- Sistema de apertura de puertas con RFDI, aplicación móvil.
- Sistema de basureros inteligentes con sensores de llenado y sensores de acercamiento.
- Bastón con sonido para ayuda de personas no videntes.
- Sistema de riego automático con implementación de sensores de humedad.
- Sistema de parqueaderos con sensores de ultrasonido.
- Registro de asistencia con analítica de video implementada en cámara IP.
- Proyecto de medición de rayos UV, aplicación móvil y creación de *data sets* en tiempo real.
- Proyecto de control de inventarios con RFID.
- Proyecto de basureros inteligentes.
- Metodología para la creación de proyectos, mapa de datos y reutilización de los mismos [78].

Por otro lado, entre los planes futuros de la UCE se encuentra poner en funcionamiento el portal de datos abiertos, implementar el ecosistema de innovación y socializar la plataforma de seguimiento de proyectos de innovación y poner en práctica el sistema integral de información.

3.3. Enseñanza de programación en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la UCE

La Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, a partir de los programas de pregrado y posgrado, brinda la oportunidad de acceder al conocimiento en las siguientes carreras: Ingeniería Civil, Ingeniería Informática, Ingeniería en Computación Gráfica, Ingeniería en Diseño Industrial, e Ingeniería Matemática.

Entre las disciplinas que se imparten en la facultad se estudia la asignatura de Programación I; en el sílabo de esta se explica que a través de ella, se capacita al alumno en las concepciones básicas y compendios de la programación a través de diferentes técnicas algorítmicas, lenguajes de programación y herramientas de desarrollo de *software* para escribir programas de computación, y de esta forma hacer de manera más simple el mundo de la programación [79].

Dicha facultad cuenta con laboratorios de informática conformados por un total de 8 salas y un laboratorio de redes, los cuales constan de áreas (en m²) descritas a continuación:

Tabla 1. Superficie de los laboratorios

SUPERFICIE DE LOS LABORATORIOS	
INSTALACIONES	ÁREA (m²)
LAB. DE REDES	56,24
SALA 1	52,52
SALA 2	45,77
SALA 3	55,73
SALA 4	66,71
SALA 5	55,96
SALA 6	39,52
SALA 7	31,99
SALA 8	22,04
TOTAL	426,48

Fuente: elaboración propia

La capacidad de cada laboratorio para servir a sus estudiantes se encuentra descrita a continuación:

Tabla 2. Capacidad de los laboratorios

CAPACIDAD DE LOS LABORATORIOS		
INSTALACIONES	NÚMERO DE ESTUDIANTES	NÚMERO DE COMPUTADORAS
LAB. DE REDES	28	9
SALA 1	52	20
SALA 2	37	16
SALA 3	91	20
SALA 4	48	20
SALA 5	46	20
SALA 6	35	16
SALA 7	17	9
SALA 8	13	10
TOTAL	289	140

Fuente: elaboración propia

Entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015 se matricularon en Programación I, 2858 estudiantes. En el periodo abril-septiembre 2015 estuvieron matriculados 234 alumnos; en el semestre octubre 2015-marzo 2016, 131 estudiantes; y en el periodo 2016-2016, 163 alumnos.

De acuerdo con la información publicada en la página web de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central, se conoce que el número de estudiantes matriculados en dicha facultad en el semestre abril-septiembre 2015 fue de 2406, entre octubre 2015-marzo 2016 fue de 2174 y en abril-septiembre 2016 de 2068.

La asignatura Programación I tiene como objetivo principal que el educando asimile los elementos de un lenguaje de programación (Java) con el que orientará su práctica de los conocimientos teóricos, entre otros elementos. Lo asimilado será aprovechado a través del desarrollo de programas básicos y simples que se presentan frecuentemente en el proceso de aprendizaje de la programación, se enfoca en programas aplicados a la carrera que estudia el alumno [79].

De acuerdo con el sílabo de Programación I, estos son los logros de aprendizaje establecidos, en atención al perfil de la asignatura [79]:

- Conoce los fundamentos y principales características de la programación y de los algoritmos.
- Analiza y diseña algoritmos básicos y sencillos usando los fundamentos y técnicas algorítmicas de la programación estructurada.
- Construye programas básicos y sencillos en lenguaje Java usando las diferentes técnicas algorítmicas de la programación estructurada y de la programación orientada a objetos (atributos y métodos) [79].

Los alumnos luego de graduados serán capaces de aplicar sus conocimientos de programación para resolver problemas específicos en entidades públicas o privadas. Esta es la primera disciplina en donde el alumno obtiene los elementos de la programación y se forma una mentalidad de programador. De igual modo contribuirá con conocimientos

básicos para diseñar algoritmos que serán implementados con programas de computador utilizando un lenguaje de programación como es Java [79].

Según Beltrán, Sánchez y Rico [15], Programación I se imparte en el primer semestre, donde el alumno obtiene los fundamentos de la programación y los conocimientos iniciales para diseñar algoritmos, que son establecidos en programas simples de computador utilizando NetBeans o Eclipse como instrumentos para desarrollar *software*, con el apoyo de las distintas técnicas algorítmicas de la programación estructurada y de la programación orientada a objetos (atributos y métodos). Programación II es la continuidad de Programación I, donde el estudiante asimila la programación orientada a objetos con el uso de Java, así como, además, se implementa UML para ejecutar el análisis y diseño de *software* [15].

Hasta el semestre abril-septiembre 2016, las asignaturas se valoraron con exámenes (50 %) y acciones suplementarias (50 %), como trabajos autónomos, individuales, grupales, trabajos integradores y pruebas parciales. El semestre se fragmentaba en dos hemisemestres¹, y cada uno se evaluó sobre 20 puntos, el educando que alcanzó 27,5 aprobó la asignatura, en tanto que, con resultados inferiores, tenía derecho a enfrentarse a un examen de recuperación, exceptuando a los que cursaron el ciclo con tercera matrícula. Para alcanzar esta prueba se sumaban las notas de los dos hemisemestres y se dividían por dos, y a este cociente se le añadía la nota del examen de recuperación. Si el alumno suspendiera la asignatura, o se marchara con notas parciales, se suponía suspendido o reprobado [15]. En el periodo abril 2016-septiembre 2017 se implementaron algunos cambios en la evaluación, para esta fecha la etapa académica continuaba dividida en semestres y estos a su vez en dos hemisemestres, con al menos cuatro evaluaciones, y para que un estudiante se promocionara debía superar 28,0 puntos entre los dos hemisemestres.

En cuanto al uso de las plataformas virtuales, hasta septiembre de 2016 se utilizó la plataforma virtual Sakai, un LMS de código abierto que posibilita estar en constante ampliación y mejora. Pero fue necesario investigar y analizar qué otra plataforma podría

¹ Término utilizado por la Universidad Central de Ecuador para referirse a la división de un semestre evaluativo en dos partes.

apoyar de mejor manera el aprendizaje de la asignatura Programación I, usando conceptos de gamificación para el proceso de enseñanza de los estudiantes de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central, por lo que se decidió utilizar la plataforma virtual Moodle. A partir de octubre de 2016, Moodle se comienza a utilizar en toda la universidad en su versión 2.9. Sin embargo, según la apreciación del análisis efectuado por Beltrán, Sánchez y Rico [15], los recursos e instrumentos tecnológicos usados por los docentes no cumplen con los requerimientos de aprendizaje, además no son utilizados de manera generalizada y estandarizada.

CAPÍTULO IV. METODOLOGÍA

El desarrollo de la investigación parte del objetivo general y de los objetivos específicos planteados en el capítulo I, donde su resultado es la propuesta de un conjunto de estrategias que influyan en la motivación de los estudiantes por la realización de tareas autónomas y, por lo tanto, en el aprendizaje de la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, con el apoyo de la gamificación y el *e-learning*. La metodología a seguir en los objetivos específicos se describe en el presente capítulo.

Para identificar los factores que inciden en el proceso de aprendizaje de la asignatura (antes de intervenir en el mismo), los autores Beltrán, Sánchez y Rico [15] realizaron un análisis estadístico de los datos entregados por la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, referente a los estudiantes inscritos y matriculados entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015, así como a aquellos que aprobaron y suspendieron la asignatura Programación I. Además, en el semestre abril 2014-septiembre 2015 mediante encuestas electrónicas anónimas a docentes y estudiantes de Programación I, se analizó cualitativamente la estrategia de aprendizaje aplicada.

De todos los factores identificados, se decide trabajar en proponer una solución al problema que enfrentan los docentes para lograr que sus estudiantes se encuentren motivados y comprometidos por el proceso de aprendizaje de Programación I. Es por eso que el autor de este trabajo diseñó y aplicó estrategias de gamificación en la ejecución de tareas autónomas en la unidad Fundamentos de Java que se estudia en el segundo semestre de la asignatura Programación I, sobre la plataforma de gestión de aprendizaje Moodle, permitiendo que los estudiantes desarrollen las actividades de forma autónoma y autodidacta para que adquieran las competencias de la asignatura satisfactoriamente.

Después, se analizó cuantitativamente (Tabla 3) los diagnósticos aplicados al culminar el semestre abril-septiembre 2015, en la carrera de Ingeniería en Computación Gráfica, y al final de los periodos octubre 2015-marzo 2016 y abril-septiembre 2016, en la carrera de Ingeniería Matemática. En el primer semestre de cada periodo se usó la metodología

tradicional, mientras que en el segundo semestre del semestre abril-septiembre 2015 se aplicó la estrategia de aprendizaje gamificada para todos los estudiantes. En el segundo semestre del periodo octubre 2015-marzo 2016 se usaron las metodologías de aprendizaje tradicional y gamificada; donde se crearon dos grupos, uno de control y otro experimental, en forma aleatoria para aplicar las metodologías por separado. Finalmente, en el segundo semestre del semestre abril-septiembre 2016, se usó la metodología tradicional en los estudiantes de Programación I de Ingeniería en Diseño Industrial como grupo de control, y como grupo experimental a los alumnos de Programación I de Ingeniería Matemática aplicando la metodología gamificada tomando en cuenta las experiencias adquiridas anteriormente. En cada experiencia vivida por el autor se aplicaron mejoras metodológicas y tecnológicas a partir de la evaluación de la experiencia anterior.

Tabla 3. Metodologías usadas en cada semestre

Semestre	Carrera	Metodología tradicional (T) - gamificada (G)							
		Hemisemestre		Hemisemestre			Hemisemestre		
		I	II	I	II	I	II	I	II
Abril-septiembre 2015	Ing. Computación Gráfica	T	G						
Octubre 2015-marzo 2016	Ing. Matemática			T	T	G			
Abril-septiembre 2016	Ing. Matemática						T		G
	Ing. Diseño Industrial						T	T	

Fuente: elaboración propia

Además, con un enfoque cualitativo se aplicaron encuestas a los estudiantes que recibieron la asignatura con la metodología gamificada en esos periodos de tiempo, con el objetivo de valorar cuál de las estrategias de gamificación aplicadas motivan, incentivan y comprometen a los estudiantes en el aprendizaje de Programación I y permiten obtener mejores calificaciones. Estos tres aspectos (motivan, incentivan y comprometen) son de vital importancia y están estrechamente relacionados, por lo que si uno deja de funcionar los demás se verán afectados.

A continuación, se muestran los objetivos y su descripción en la metodología definida a seguir.

4.1. Operacionalización de la variable

Hipótesis

La implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje para desarrollar tareas autónomas influye sobre la motivación de los estudiantes y contribuye en mejorar su rendimiento académico (Steinmann, Bosch, & Aiassa, 2013), en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

Variable independiente: estrategias de gamificación.

Variable dependiente: motivación y rendimiento.

Tabla 4. Cuadro de operacionalización

Variabes	Concepto	Dimensiones	Indicadores	Instrumento de medición
Var. independiente Estrategias de gamificación	La aplicación de estrategias de gamificación en los entornos virtuales tiene el objetivo de promover el aprendizaje e influir en los intereses de los estudiantes por el estudio. Para esto se aplican dinámicas y mecánicas de juego en un ambiente <i>e-learning</i> en la realización de tareas autónomas, incentivando a los estudiantes para la programación	Tareas autónomas de programación	Tareas autónomas de programación	Análisis de las encuestas realizadas a los estudiantes después de implementar las estrategias de gamificación en un ambiente <i>e-learning</i>
Var. dependiente Rendimiento	Rendimiento académico de los estudiantes (resultados de las notas finales)	Exámenes Tareas autónomas Pruebas parciales	Nota final	Exámenes Tareas autónomas Pruebas parciales
Var. dependiente Motivación por la realización de tareas	Motivación de los estudiantes por la realización de las tareas autónomas en la unidad Fundamentos de Java de la asignatura Programación I	Tareas autónomas de programación	Puntos de experiencia de tareas autónomas Insignias obtenidas por los estudiantes	Puntos de experiencia e insignias ganadas por los estudiantes en la plataforma gamificada

Fuente: elaboración propia

4.2. Metodología objetivo 1

Analizar la metodología de enseñanza existente en la modalidad presencial de la asignatura Programación I, para identificar los factores que influyen en el proceso de aprendizaje de los estudiantes.

1) Enfoque de investigación

En el cumplimiento del objetivo 1, se realiza una investigación con enfoque mixto (cuantitativo y cualitativo); inicialmente se identifican los factores que influyen en el proceso de aprendizaje de los estudiantes que cursan la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador bajo la metodología de enseñanza existente, a través de un estudio cuantitativo del porcentaje de estudiantes aprobados y suspensos entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015. En un segundo momento se realiza un estudio cualitativo, a través de la aplicación de encuestas semiestructuradas a docentes y estudiantes en el semestre octubre 2014-marzo 2015.

2) Tipo de investigación

Existen varias tipologías para los estudios, en el desarrollo del objetivo 1 se aplica la modalidad de investigación documental y de campo, pues inicialmente se analiza la metodología de enseñanza existente a partir de fuentes de carácter documental, desarrolladas en capítulos anteriores y de un estudio cuantitativo del porcentaje de alumnos aprobados y suspensos en Programación I. A partir de este análisis se realiza un estudio de campo, donde se aplican encuestas a docentes y estudiantes, para identificar finalmente los principales factores que inciden en el proceso de aprendizaje, y proponer métodos y herramientas que conduzcan a mejorar la motivación y el compromiso de los estudiantes por la asignatura Programación I [15].

3) Nivel de investigación

En el objetivo 1 se realiza un estudio con alcance explicativo, pues inicialmente se exploran y describen un conjunto de variables con posibles influencias en el proceso de aprendizaje, luego se evalúan las posibles asociaciones y las relaciones identificadas son confirmadas a través de métodos estadísticos confirmatorios multivariados.

4) Diseño de investigación

Se realiza un diseño de investigación no experimental de corte transversal, pues no hay control de variables, llevando a cabo el estudio de los factores que influyen en el proceso de aprendizaje y de la metodología de enseñanza existente en un solo momento de tiempo, es decir, entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015.

5) Población y muestra

La población y muestra de estudio para el cumplimiento del objetivo 1 se presentan a continuación:

- **Población:** en la presente investigación, para el análisis cuantitativo se tomó como grupo poblacional a todos los estudiantes matriculados en la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015, representando 2858 estudiantes. Para el análisis cualitativo la población es de 505 alumnos, que fueron los matriculados en la asignatura Programación I en el semestre octubre 2014-marzo 2015 y que tenían registrado su correo electrónico en el sistema académico de la universidad, y los nueve docentes que impartieron la asignatura en el mismo periodo (uno de los profesores se retiró de la universidad).
- **Muestra:** la muestra se seleccionó de forma diferente para los estudios cuantitativo y cualitativo, como se observa a continuación:
 - Para el análisis cuantitativo se tomó el porcentaje de los alumnos aprobados y suspendidos entre los periodos octubre 2009-marzo 2010 y octubre 2014-marzo 2015, siendo seleccionados para la muestra todos los alumnos matriculados en ese

periodo, representando un total de 2858 estudiantes, que coincide con el total poblacional.

- En el análisis cualitativo se tomaron como muestra los 85 estudiantes y seis docentes (de la población definida), que respondieron la encuesta electrónica anónima. La muestra de alumnos es significativa a un nivel de confianza del 95,0 %, con un error máximo de estimación de un 10,0 %.

6) Método

El método aplicado fue el analítico-sintético.

Se analizan los factores que influyen en el bajo rendimiento de los estudiantes en la asignatura Programación I, para lograr optimizar la problemática planteada. En tal sentido fue pertinente una adecuada segmentación de los referentes teóricos en el aprendizaje de la asignatura y las herramientas tecnológicas que influyen en la motivación del estudiante.

7) Técnicas e instrumentos

a) Recolección de datos

Las técnicas e instrumentos para la recolección de datos aplicados en el objetivo 1 de la investigación con enfoque cuantitativo son:

- **Observación:** es la técnica utilizada para obtener información sobre las causas por las cuales los estudiantes aprueban o suspenden la asignatura entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015. El investigador realizó una observación intencionada y selectiva para poder interpretar el fenómeno e identificar las formas de mitigarlo. Como instrumento se utilizaron registros de observación.
- **Documental:** esta técnica permitió la revisión de registros documentales de la facultad y del profesor para obtener información de variables relacionadas con la muestra de estudio, como la cantidad de estudiantes retirados y matriculados, así como aquellos aprobados y suspensos; género, provincia de procedencia, tipo de colegio de donde provienen, carrera que cursan, semestre, profesorado, etc.

La técnica e instrumento aplicados en el objetivo 1 de la investigación con enfoque cualitativo es:

- **Encuesta:** esta técnica se utiliza para obtener información sobre los elementos positivos y negativos del proceso de aprendizaje de la asignatura Programación I, usando como instrumento de recopilación cuestionarios. Se aplicó un cuestionario a docentes y otro a estudiantes (Anexos 2 y 3).

b) Procesamiento y análisis estadístico

La información obtenida tanto en el estudio cuantitativo como en el cualitativo fue procesada a través de una base de datos utilizando el *software* de procesamiento estadístico Pasw Statistics (SPSS versión 23.0).

En el estudio cuantitativo, los resultados se organizan en tablas de frecuencia, contingencia y gráficos, utilizando estadígrafos descriptivos como frecuencia absoluta y porcentajes. Para evaluar la posible asociación entre variables cualitativas, se usó el estadígrafo Chi cuadrado. Además, se utiliza un análisis de componentes principales y algoritmos de Bayes, como métodos multivariados confirmatorios.

En todos los casos, se fijó un intervalo de confianza del 95 %; la significación estadística se interpretó según el siguiente criterio:

- Si $p > 0,05$ no existen diferencias significativas.
- Si $p \leq 0,05$ existen diferencias significativas.

En el caso de los datos provenientes de las encuestas, se organiza la información en tablas de frecuencia y contingencia, utilizando para la descripción de las mismas, estadísticos descriptivos tales como: frecuencias absolutas y porcentajes.

Cada vez que se entiende conveniente para facilitar la visualización de los resultados, las tablas se acompañan de gráficos representados según el tipo de información.

4.3. Metodología objetivo 2

Diseñar e implementar estrategias como apoyo al proceso de aprendizaje presencial, usando herramientas de *e-learning* y gamificación.

1) Tipo de investigación

Para dar cumplimiento al objetivo 2, se empleó la modalidad de estudio documental, pues se recolectó en el capítulo II (*Base teórica y tecnológica*) toda la información referente a la gamificación y el *e-learning*, y a su aplicación en la educación y en particular en el aprendizaje de programación, para luego proceder al diseño e implementación de la propuesta.

2) Nivel de investigación

En el objetivo 2, se asume un nivel de investigación descriptivo, donde se describe de forma exacta la información, limitándose a señalar específicamente las características o información de base analizada. Este tipo de estudio se aplica con el propósito de diseñar estrategias y técnicas de gamificación en un entorno virtual para el aprendizaje de la unidad Fundamentos de Java de la asignatura Programación I². También, le permite al investigador sintetizar las experiencias adquiridas después de implementar las estrategias de la gamificación en un ambiente *e-learning* en la asignatura Programación I durante tres semestres (abril-septiembre 2015, octubre 2015-marzo 2016 y abril-septiembre 2016) que impartió.

En tanto, se proponen métodos y herramientas que influyan en la motivación y el compromiso de los estudiantes en el proceso de aprendizaje de la asignatura para la realización de las tareas autónomas, posibilitando orientar la investigación de forma sintetizada y alcanzar un planteamiento adecuado del problema de investigación, así como la formulación de la hipótesis.

² Fundamentos de Java en la asignatura Programación I: en este objetivo es llamada Programación Mágica I.

3) Métodos

El método aplicado fue el inductivo-deductivo. Se realiza el análisis de la información recolectada en el capítulo II estableciendo los elementos necesarios para la elaboración de la propuesta solución de la investigación en marcha. Permite desde lo inductivo establecer una verdadera comprensión del fenómeno, y desde lo deductivo beneficia la concreción, de lo más significativo para establecer una propuesta de estrategias de gamificación en un ambiente *e-learning* para lograr influir en la motivación de los estudiantes por la asignatura Programación I, en la realización de tareas autónomas.

Con los datos obtenidos en el capítulo II y los resultados alcanzados a partir del primer objetivo, el investigador se encuentra en condiciones de elaborar la propuesta metodológica de las estrategias de gamificación implementadas en la plataforma Moodle, para la realización de las tareas autónomas, solventando de esta forma las carencias detectadas en el objetivo 1.

Como marco de trabajo para el diseño de la propuesta y su posterior implementación en una plataforma virtual se tomaron en consideración las siguientes referencias: el Framework D6 propuesto por Kevin Werbach [39] y el modelo para la gamificación de actividades educativas propuesto por González y Mora [6].

4.4. Metodología objetivo 3

Evaluar la actividad pedagógica de las estrategias de gamificación en un entorno *e-learning* diseñadas para la modalidad presencial en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

1) Enfoque de investigación

En el objetivo 3 se realiza un estudio analítico con enfoque mixto, se evalúan mediante un estudio cuantitativo los resultados obtenidos por los estudiantes en los semestres donde se implementaron las estrategias y técnicas de gamificación en un entorno virtual. Además, se presenta el comportamiento de los resultados obtenidos en la encuesta aplicada a la muestra de estudiantes utilizada (estudio cuantitativo).

2) Tipo de investigación

En el desarrollo del objetivo 3 se aplica una investigación documental y de campo. Se evalúa la actividad pedagógica de las estrategias de gamificación en un entorno *e-learning* a partir de los resultados del diseño e implementación, mediante un análisis cuantitativo. Seguidamente se realiza una investigación de campo, donde se aplican cuestionarios a estudiantes y se procesan los resultados de los mismos.

3) Nivel de investigación

En el objetivo 3 se realiza un estudio con alcance explicativo, pues se exploran, describen y relacionan variables a través del análisis cuantitativo y cualitativo, para finalmente probar la hipótesis de investigación.

4) Diseño de investigación

En el objetivo 3, se asume un diseño cuasiexperimental con grupos experimental y de control, en el que se emplean grupos intactos de comparación ya existentes, no equivalentes (los grupos constituyen entidades formadas naturalmente), aunque similares. Este tipo de diseño de forma general controla los principales efectos de la historia, la maduración, la administración de test y la instrumentación donde la diferencia para el grupo experimental no puede explicarse por efectos principales de esas variables, como los que afectarían tanto al grupo experimental como al de control.

En el primer hemisemestre de cada periodo (abril-septiembre 2015, octubre 2015-marzo 2016, abril-septiembre 2016) se usó la metodología tradicional, mientras que en el segundo hemisemestre del semestre abril-septiembre 2015 se aplicó la estrategia de aprendizaje gamificada para todos los estudiantes. En el segundo hemisemestre del periodo octubre 2015-marzo 2016 se usaron las metodologías de aprendizaje tradicional y gamificada, donde se crearon dos grupos (control y experimental) seleccionados de forma aleatoria para aplicar las metodologías por separado. Finalmente, en el segundo hemisemestre del semestre abril-septiembre 2016, se aplicó la metodología gamificada a todos los alumnos tomando en cuenta

las experiencias adquiridas en los semestres anteriores, siendo esta la propuesta de solución de la presente investigación.

5) Población y muestra

La población y muestra de estudio para el cumplimiento del objetivo 3 se presentan a continuación:

- **Población**

La población para este objetivo se divide para el estudio cuantitativo (notas finales) y el cualitativo (encuestas):

- **Análisis cuantitativo:** se toma como grupo poblacional a los 17 estudiantes de la carrera de Ingeniería en Computación Gráfica del semestre abril-septiembre 2015; los 25 estudiantes del semestre octubre 2015-marzo 2016 y los 33 del semestre abril-septiembre 2016, de la carrera de Ingeniería Matemática; y los 31 alumnos de la carrera de Ingeniería en Diseño Industrial del periodo abril-septiembre 2016. Todos son alumnos de la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.
- **Análisis cualitativo:** el grupo poblacional son todos los estudiantes de Programación I donde se aplicó las estrategias y plataforma gamificada, es decir, a los 17 estudiantes de la carrera de Ingeniería en Computación Gráfica del semestre abril-septiembre 2015; los 13 estudiantes del semestre octubre 2015-marzo 2016 y los 33 del semestre abril-septiembre 2016, de la carrera de Ingeniería Matemática.

- **Muestra**

Al igual que la población, la muestra seleccionada en este objetivo se divide para el estudio cuantitativo (notas finales) y el cualitativo (encuestas):

- **Análisis cuantitativo:** para el estudio cuantitativo, se usaron las notas finales de los estudiantes que recibieron la asignatura gamificada, representando los 17 estudiantes del semestre abril-septiembre 2015 de la carrera de Ingeniería en

Computación Gráfica. En el semestre octubre 2015-marzo 2016 de la carrera de Ingeniería Matemática, el curso se dividió en dos grupos de forma aleatoria, 12 alumnos utilizaron la metodología tradicional, constituyendo el grupo experimental, y 13 la metodología gamificada en un ambiente *e-learning*, tomado como grupo de control. Para el periodo abril-septiembre 2016 el grupo de control estuvo conformado por los 31 estudiantes de la carrera de Ingeniería en Diseño Industrial y el grupo experimental, por los 33 alumnos de la carrera de Ingeniería Matemática.

- **Análisis cualitativo:** para el estudio cualitativo se realizó un muestreo por criterios, quienes contestaron la encuesta fueron aquellos estudiantes que terminaron el curso gamificado en los diferentes semestres: 9 estudiantes en el semestre abril-septiembre 2015 en la carrera de Ingeniería en Computación Gráfica, 11 alumnos del semestre octubre 2015-marzo 2016, y 30 estudiantes del semestre abril-septiembre 2016 de la carrera de Ingeniería Matemática.

6) Método

El método aplicado fue el de análisis-síntesis.

Con este método se procedió al análisis de la información pertinente, en esta ocasión, las notas y encuestas de los alumnos en los tres semestres en los cuales se aplicaron las propuestas descritas en el objetivo 2. Luego se realizó una síntesis concreta en la cual se pudiera identificar los resultados de la implementación del producto metodológico y tecnológico de la investigación.

7) Técnicas e instrumentos

a) Recolección de datos

La técnica aplicada en el objetivo 3 de la investigación con enfoque cuantitativo es:

- **Observación no estructurada:** permitió detectar durante los tres semestres donde se pusieron en práctica las estrategias diseñadas, elementos que deberían ser mejorados,

incluidos y eliminados, para de esta forma llegar a la estrategia de gamificación en un entorno *e-learning* adecuado para la asignatura Programación I, como se describe en el objetivo 2. Se utilizó como instrumento registros de observación.

- **Documental:** la técnica documental permitió al investigador documentar el análisis realizado sobre el cumplimiento de las tareas autónomas del curso gamificado y las notas obtenidas por los estudiantes en los semestres abril-septiembre 2015, octubre 2015-marzo 2016 y abril-septiembre 2016, además del porcentaje de estudiantes que aprueban o no la asignatura. Finalmente, se hace una evaluación de los resultados donde se muestra la correlación que existe entre los resultados académicos de los alumnos, y las estrategias y técnicas de gamificación usadas en la asignatura.

La técnica aplicada en el objetivo 3 de la investigación con enfoque cualitativo es:

- **Encuesta:** es la técnica más importante de obtención de información con enfoque cualitativo en la investigación. El cuestionario elaborado como instrumento fue aplicado al finalizar los semestres impartidos por el investigador con la metodología y plataforma gamificada con las mismas interrogantes (Anexo 4). Tras la aplicación de esta técnica los estudiantes opinaban lo positivo o negativo de la estrategia implementada, permitiendo que se fuera perfeccionando progresivamente, hasta lograr las estrategias que se proponen en la presente investigación.
- **Entrevista no estructurada:** se trabajó con preguntas abiertas para recoger información adicional a partir de cuestionarios y la observación en las clases. Es decir, se realizaron preguntas en función de las experiencias que surgieron durante las clases.

b) Procesamiento y análisis estadístico

La información obtenida por diferentes medios (notas y encuestas) fue procesada a través de una base de datos utilizando el *software* de procesamiento estadístico Pasw Statistics (SPSS versión 23.0).

Para dar cumplimiento al objetivo 3 se realizó el análisis de los datos. En el caso de los datos provenientes de las encuestas, se organiza la información en tablas de frecuencia y

contingencia, utilizando para la descripción de las mismas, estadísticos descriptivos tales como: frecuencias absolutas y porcentajes.

En el estudio cuantitativo, los resultados se organizan en tablas de frecuencia, contingencia y figuras; utilizando estadígrafos descriptivos como frecuencia absoluta y porcentajes.

Las tablas se acompañan de figuras representadas según el tipo de información cada vez que se entiende conveniente para facilitar la visualización de los resultados.

El análisis estadístico para constatar la hipótesis se hace con test referidos a la variable «rendimiento» como se muestra a continuación:

Rendimiento: inicialmente se usó el test de Kolmogorov-Smirnov, para verificar la normalidad de las notas finales en cada semestre. En aquellos semestres donde la distribución de notas fue normal, se utilizó el test paramétrico t-Student para evaluar las diferencias entre las medias en cada grupo (normal o control y gamificado o experimental). En el caso de que la distribución no fuera normal, se utiliza el test no paramétrico para muestras independientes U-Mann-Whitney y la prueba de la mediana para k muestras. Además, con el objetivo de evaluar las posibles diferencias significativas en los grupos donde se utiliza la metodología gamificada para cada semestre, se aplicó la prueba anova de un factor de Kruskal-Wallis y el test de las medianas para k muestras.

En todos los casos, se fijó un intervalo de confianza del 95,0 %; la significación estadística se interpretó según el siguiente criterio:

- Si $p > 0,05$ no existen diferencias significativas.
- Si $p \leq 0,05$ existen diferencias significativas.

CAPÍTULO V. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS

5.1. Resultados objetivo 1

Objetivo 1. Analizar la metodología de enseñanza existente en la modalidad presencial de la asignatura Programación I, para identificar los factores que influyen en el proceso de aprendizaje de los estudiantes.

Para identificar los factores que influyen en el proceso de aprendizaje de los estudiantes que cursan la modalidad presencial de la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la UCE, con la metodología de enseñanza existente, se recolectaron las evaluaciones finales de los estudiantes entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015. Se realizó un análisis cuantitativo de las variables: «género», «tipo de provincia», «tipo de colegio», «carreras que cursan», «cantidad de matrículas», «semestre» y «profesorado», permitiendo identificar aquellas que influyen en que el estudiante apruebe o no dicha asignatura. Además, se llevó a cabo un estudio cualitativo, basado en la aplicación de encuestas no estructuradas a docentes y estudiantes en el semestre octubre 2014-marzo 2015, con el fin de obtener información referente a las causas que ocasionan el alto porcentaje de estudiantes suspensos, y el criterio de los docentes y alumnos sobre la metodología y herramientas utilizadas por los docentes al impartir la asignatura.

5.1.1. Características metodológicas

La asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador se imparte en las carreras de Ingenierías: Informática, Matemática, Computación Gráfica y Diseño Industrial. Todas estas carreras cuentan con un mismo sílabo o temario por el cual los profesores deben guiarse para impartir la asignatura, pero los ejercicios a desarrollar son relacionados con cada una de ellas para que sea más fácil de aprender y los alumnos vean su importancia en la carrera que cursan.

La asignatura se imparte en un semestre de 240 horas, las cuales son divididas en 96 horas de clases y 144 horas de tutoría (prácticas de laboratorio, tutorías presenciales, aprendizaje en aula virtual y trabajo autónomo); un semestre es dividido en periodos de dos hemisemestres.

En el primer periodo del semestre, el estudiante adquiere los fundamentos de la programación y los conocimientos iniciales para diseñar algoritmos en pseudocódigo y diagramas de flujo usando la herramienta PSeInt. En el segundo periodo o hemisemestre, se aprenden los fundamentos de programación en Java, mediante la implementación de programas básicos y sencillos, empleando las herramientas NetBeans o Eclipse. También se usan diferentes técnicas algorítmicas de programación estructurada y lo básico de la programación orientada a objetos (atributos y métodos). La continuación de la asignatura Programación I en el plan de estudios de las carreras (Anexo 1) es Programación II, en la cual el alumno aprende la programación orientada a objetos más avanzada, donde se realiza además el análisis y diseño de *software* a través de UML [15].

La evaluación de la asignatura es la suma de los dos hemisemestres o periodos y se evalúa con un examen (50 %) y actividades complementarias orientadas en el sílabo (50 %), como: evaluación escrita o práctica, trabajo autónomo y/o virtual, trabajos individuales, trabajos grupales, trabajos integradores y las pruebas parciales. De estas actividades cada profesor decide qué actividades utilizará al impartir la asignatura.

Cada hemisemestre o periodo se evalúa sobre 20 puntos, para un total de 40 puntos en el semestre. Así, el estudiante que reúna 27,5 superará la asignatura, mientras que, con un valor inferior, puede rendir un examen de recuperación, excepto si cursa el ciclo con tercera matrícula ya que esta sería la tercera vez que un estudiante cursa la asignatura por haberla suspendido en las dos matrículas anteriores. Para poder presentarse al examen de recuperación, se suman las notas de los dos hemisemestres y se divide por dos, cociente al que se agrega la nota del examen de recuperación, si suma 27,5 o más, el alumno aprueba la asignatura. Si el estudiante no aprueba la asignatura, o se retira teniendo alguna nota parcial, se considera suspenso o reprobado.

Es una asignatura difícil de aprender y que necesita de mucha dedicación, pues independientemente de saberse las palabras reservadas de los lenguajes de programación, así como la forma de implementar el código fuente, se necesita interpretar y analizar la solución a obtener. Muchas veces cuando la implementación de un programa no funciona correctamente o da error, puede llevar mucho tiempo corregirla y eso desmotiva al desarrollador. También,

necesita de mucho esfuerzo y compromiso propio para poder aprender a programar. Además, se imparte en el primer semestre de la carrera, etapa en la que aún los estudiantes no se han adaptado a la universidad, ni al nuevo modelo de estudio.

Teniendo en cuenta las características de la asignatura y los resultados académicos obtenidos, el investigador observó que existían problemas de motivación y compromiso en el proceso de aprendizaje por parte de los alumnos.

Como se había mencionado anteriormente, el análisis cuantitativo nos permite identificar aquellas variables que influyen en el proceso de aprendizaje de Programación I, en la modalidad presencial bajo la metodología de enseñanza tradicional, a partir de la evaluación final de las mismas entre los semestres octubre 2009-marzo 2010 y octubre 2014-marzo 2015.

5.1.2. Análisis cuantitativo

Inicialmente se realizó una caracterización de la población, determinando la cantidad de estudiantes retirados y matriculados, así como aquellos aprobados y suspensos. La figura siguiente presenta las estadísticas encontradas:

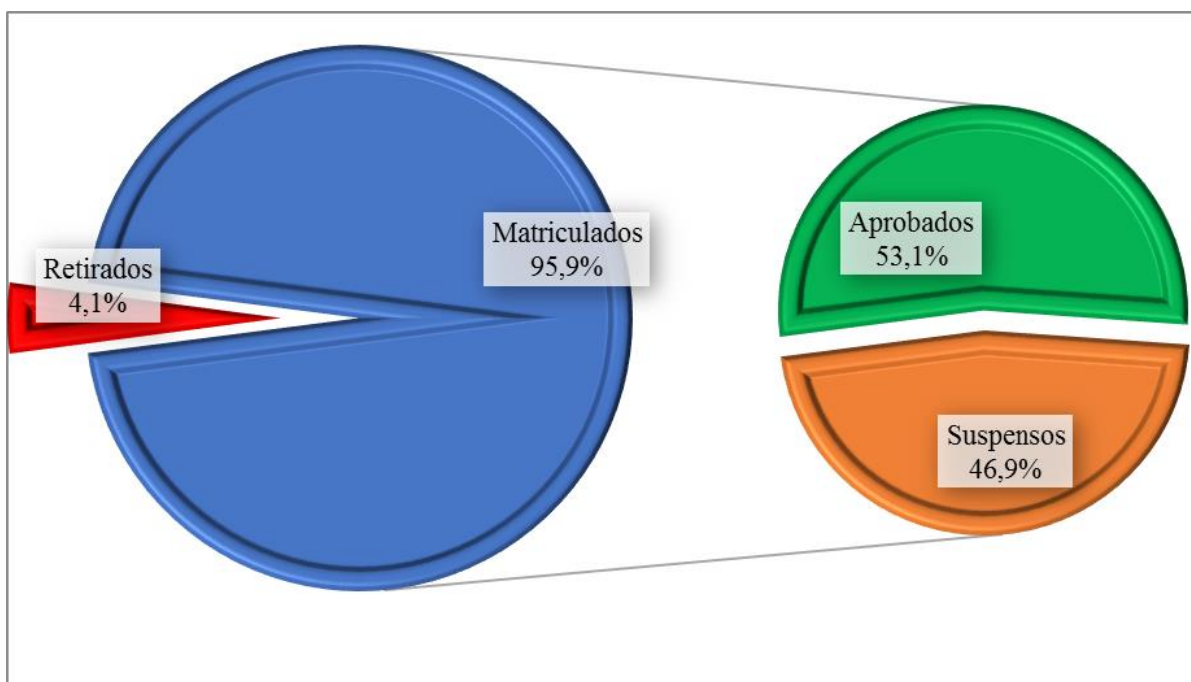


Figura 3. Estudiantes aprobados y suspensos de los matriculados

Fuente: elaboración propia

El número de estudiantes inscritos en el periodo de octubre 2009-marzo 2010 a octubre 2014-marzo 2015 fue de 2980, de los cuales 2858 (95,9 %) se matricularon, y 122 se retiraron sin cursar la asignatura, representando un 4,1 %. De los estudiantes que se matricularon, 1519 (53,1 %) aprobaron, y 1339 (46,9 %) suspendieron o reprobaron, demostrando que existen problemas en el proceso de enseñanza-aprendizaje de la asignatura, por el alto porcentaje de estudiantes que no lograron superar la asignatura.

Una vez determinado que un alto porcentaje de los estudiantes matriculados suspendieron, debemos identificar aquellas variables que pudieron influir en estos resultados; para ello analizaremos: «provincia de procedencia», «género», «colegio de secundaria de donde provienen», «cantidad de matrículas», «semestre» y «profesorado que imparte la asignatura».

Provincia de procedencia: este aspecto fue analizado ya que se pensaba que los estudiantes provenientes de colegios de la provincia de Pichincha, en particular de la ciudad de Quito, poseen mayores conocimientos que los del resto de provincias de Ecuador. Como resultado se tiene que, el 81,0 % de los estudiantes son de la provincia de Pichincha (92,0 % de ellos de Quito), con 53,3 % de aprobados y 46,7 % de suspensos; y el 19,0 % son del resto del país, de los cuales el 52,6 % aprobaron y el 47,4 % suspendieron. Al aplicar el test de independencia Chi cuadrado no se encontró asociación ($p = 0,769 < 0,05$) entre el lugar de procedencia y los resultados de la evaluación.

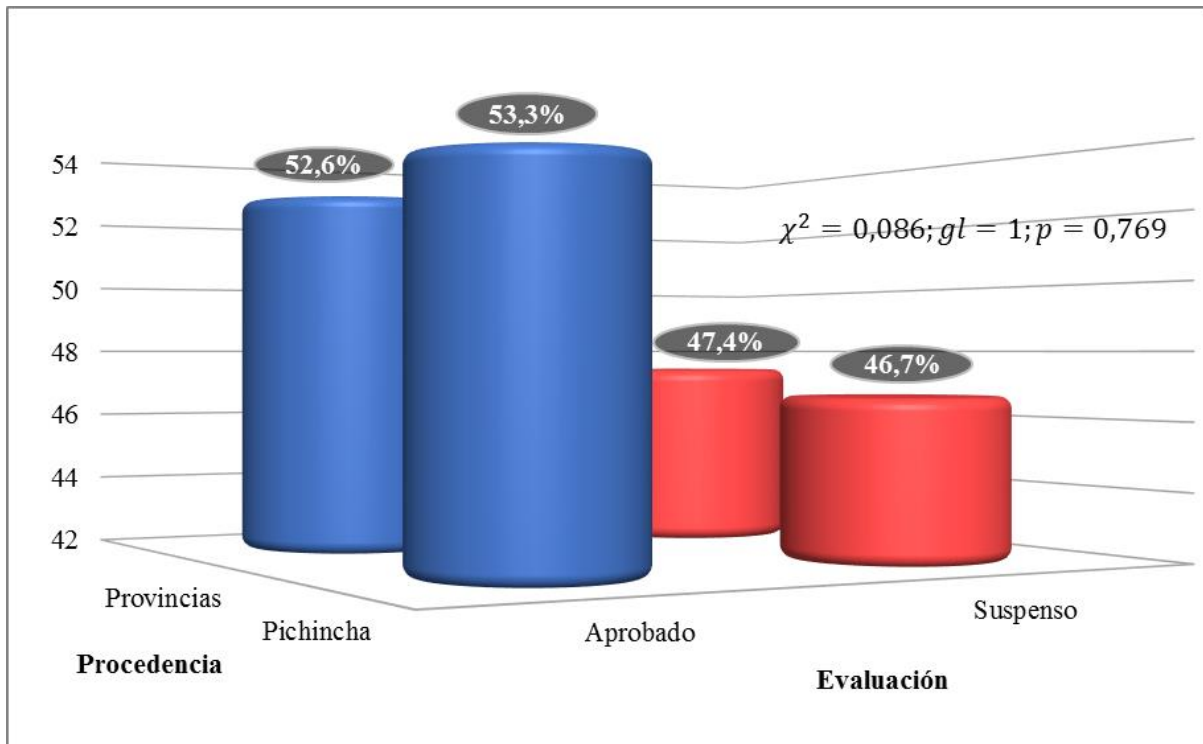


Figura 4. Alumnos aprobados vs. suspensos por procedencia
Fuente: elaboración propia

Género: esta variable fue analizada debido a que la mayoría de los estudiantes que estudian carreras de ingeniería son hombres, por lo que pudiéramos pensar que el género sea un factor que influya en los resultados de la evaluación. La Figura 5 demuestra lo descrito, donde del total de matriculados el 70,3 % son masculinos (2009) y el 29,7 % son del sexo femenino (849 estudiantes). Al relacionar el género con los resultados de la evaluación, encontramos que existe una distribución muy parecida por cada grupo, pues en el caso de los hombres el 53,8 % aprobó y el 46,2 % suspendió; y dentro del grupo de las féminas, el 51,7 % aprobó y un 48,3 % suspendió. Estos resultados se corroboraron a partir del test de independencia Chi cuadrado, el cual determinó que no existe una asociación entre la variable género y los resultados de la evaluación.

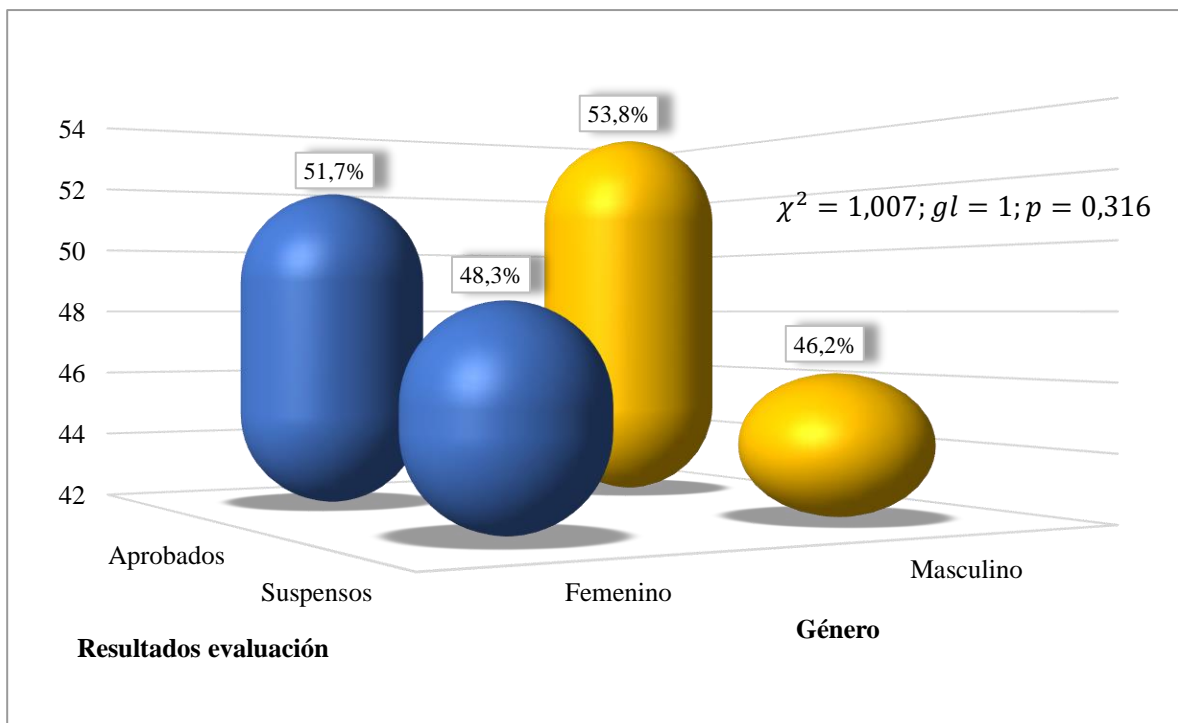


Figura 5. Alumnos aprobados vs. suspensos por género
Fuente: elaboración propia

Colegios de secundaria: en cuanto a la variable «tipo de colegio de donde provienen», tenemos que el 76,1 % de los estudiantes de Programación I proviene de colegios fiscales o estatales³, el 18,5 % de colegios particulares o privados y el 5,4 % restante se distribuye entre colegios municipales⁴, fiscomisionales⁵ y otros. Los alumnos que venían de colegios fiscales que aprobaron representan el 43,4 % del total y los suspensos el 46,6 %; en el caso de los estudiantes de colegios privados, los aprobados y suspensos representan el 55,3 % y 44,7 % respectivamente; el 49,2 % aprobaron, y el 50,8 % suspendieron de colegios municipales; de colegios fiscomisionales aprobaron el 57,1 % y suspendieron el 42,9 % (Figura 6). De manera general, la distribución por tipo de colegio con respecto a los resultados de las notas se distribuye de forma homogénea, contrastándose que estas dos variables son independientes, es decir, no existe una asociación estadísticamente significativa entre ellas.

³ Colegios fiscales o estatales: son los colegios públicos que son financiados por el Estado.

⁴ Colegios municipales: son los colegios financiados por el municipio de la ciudad.

⁵ Colegios fiscomisionales: son colegios que reciben apoyo del Estado, pero también se autosolventan con el cobro de pensiones.

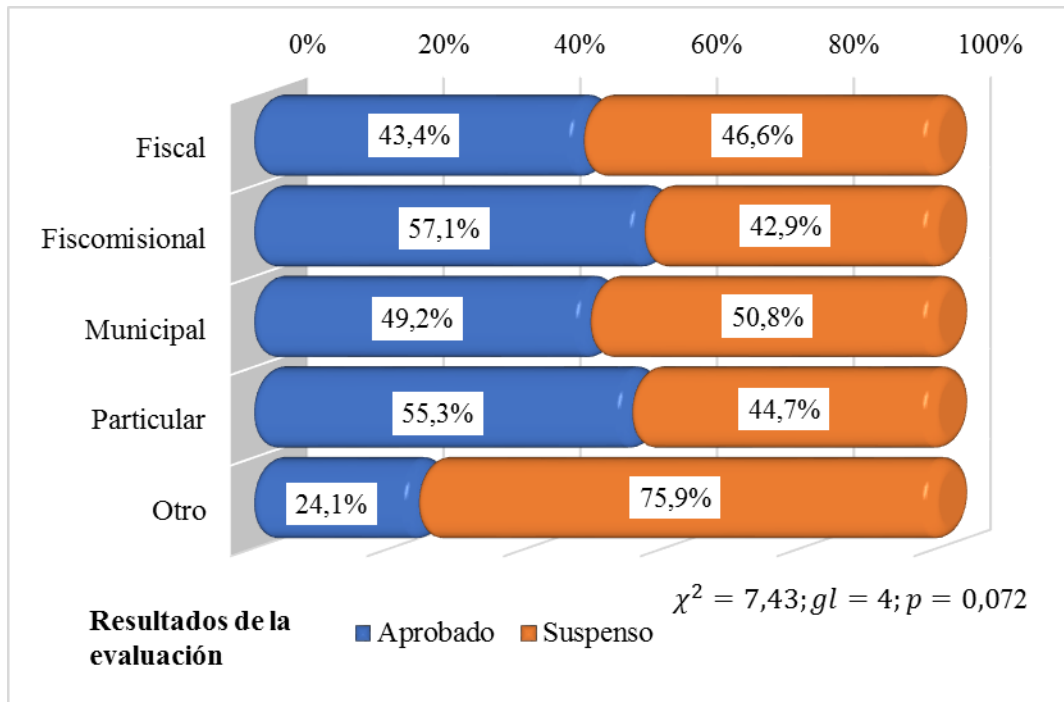


Figura 6. Aprobados vs. suspensos por tipo de colegio
Fuente: elaboración propia

Carrera que cursan: al ser la muestra heterogénea en cuanto a la carrera que cursan los estudiantes de la investigación, pudiera ser que el resultado de la evaluación se viera influenciado por esta variable, por lo que se evaluó si existe una asociación entre la carrera que cursan y los resultados de la evaluación.

La Figura 7 muestra el porcentaje de estudiantes matriculados en primera matrícula, siendo del 43,0 % en la carrera de Ingeniería Informática, de un 19,5 % en Ingeniería en Computación Gráfica y del 22,3 % y 15,2 % en Ingeniería en Diseño Industrial y en Ingeniería Matemática, respectivamente.

En Ingeniería Informática, el 59,0 % de los estudiantes aprobó, y el 41,0 % reprobó o suspendió. En Ingeniería en Computación Gráfica aprobaron el 52,0 %, y el 48,0 % suspendieron; en Ingeniería en Diseño Industrial encontramos un 50,0 % de aprobados, y 50,0 % de suspendidos; mientras que en Ingeniería Matemática el 49,0 % aprobó, y el 51,0 % suspendió (Figura 7). Los resultados evidencian que aprobaron más estudiantes Programación I en Informática y Computación Gráfica, carreras donde la asignatura es de especialidad.

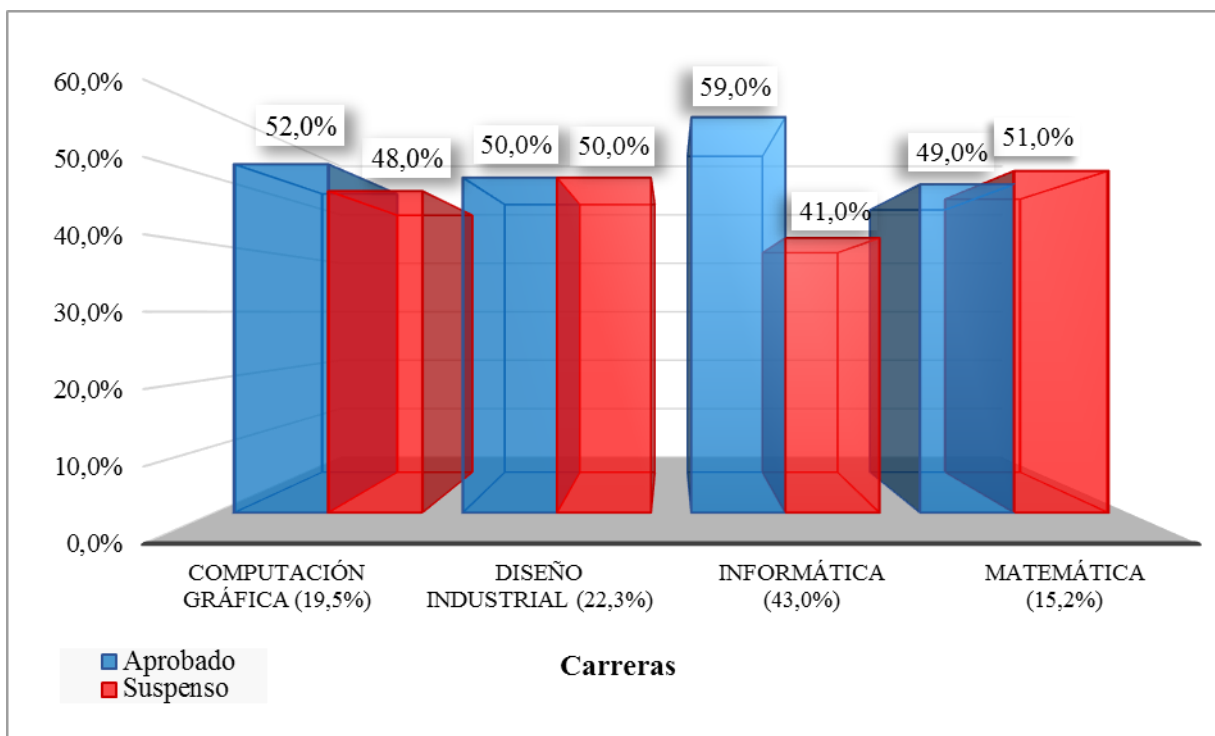


Figura 7. Porcentaje de estudiantes aprobados y suspensos por carrera
Fuente: elaboración propia

En los semestres analizados antes de intervenir en el proceso de aprendizaje de Programación I con la presente investigación, los cursos de Programación I tuvieron estudiantes de distintas carreras; los alumnos consideraban que los ejercicios y su complejidad, y los contenidos al ser genéricos no tenían mayor aplicabilidad en la carrera que estudiaban, por lo tanto, se sentían desmotivados. Si bien el sílabo de Programación I fue el mismo, se necesitó adaptar los contenidos y ejercicios a la necesidad de cada carrera, y de este modo tener mayor motivación de los alumnos. Luego del semestre abril-septiembre 2015, los cursos tienen solamente estudiantes de una misma carrera.

Un alto porcentaje de estudiantes que pierden la asignatura no la vuelven a tomar. El 29,0 % de los estudiantes, aunque aprobaron Programación I, no tomaron la asignatura Programación II. Las posibles razones son: el cambio de carrera o su abandono, postergar la matrícula para aprobar el resto de asignaturas del semestre y después retomar Programación II [15].

Número de matrículas: antes de intervenir en el proceso de aprendizaje de los estudiantes en la asignatura Programación I, en primera matrícula estuvieron el 75,1 % de los estudiantes, en

segunda el 22,5 % y en tercera el 2,4 %. De los estudiantes que tomaron primera matrícula, el 50,0 % aprobaron y el 50,0 % suspendieron; de quienes tomaron segunda matrícula, aprobaron el 59,5 % y el 40,5 % suspendieron, y de quienes se matricularon en tercera matrícula, aprobaron el 77,6 % y suspendieron el 22,4 %, como se muestra en la Figura 8.

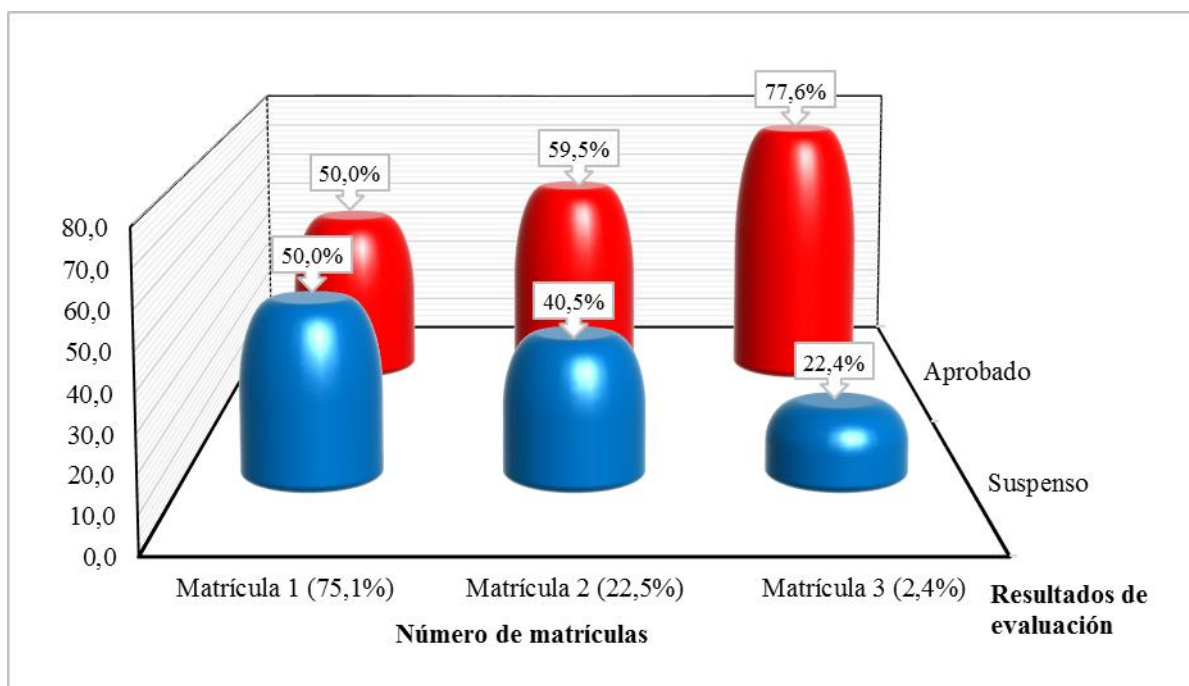


Figura 8. Porcentaje de estudiantes aprobados por número de matrícula
Fuente: elaboración propia

Teniendo en cuenta que los estudiantes de primera matrícula no tienen conocimiento previo de programación y que también influye el profesor con el que recibieron la asignatura, muchos suspenden u obtienen malas calificaciones; ya en la segunda matrícula sí poseen conocimientos de programación por lo que las calificaciones deben ser mucho mejores, así mismo sucede con los estudiantes de tercera matrícula, pues ya por tercera vez reciben la misma asignatura.

Semestre: otra de las variables a tener en cuenta son los «semestres», con el objetivo de determinar si dicha variable influye o no en los resultados de las evaluaciones; de ahí que se estudie la evolución del número de aprobados y suspensos por semestres.

Entre los semestres octubre 2009-marzo 2010 hasta abril-septiembre 2012, exceptuando el semestre octubre 2011-marzo 2012, el porcentaje de estudiantes suspensos o reprobados fue mayor que los aprobados. A partir del semestre octubre 2012-marzo 2013 la relación fue totalmente contraria como se muestra en la Figura 9, ya que a partir de este semestre varios docentes se retiraron de la asignatura, e ingresaron nuevos profesores.

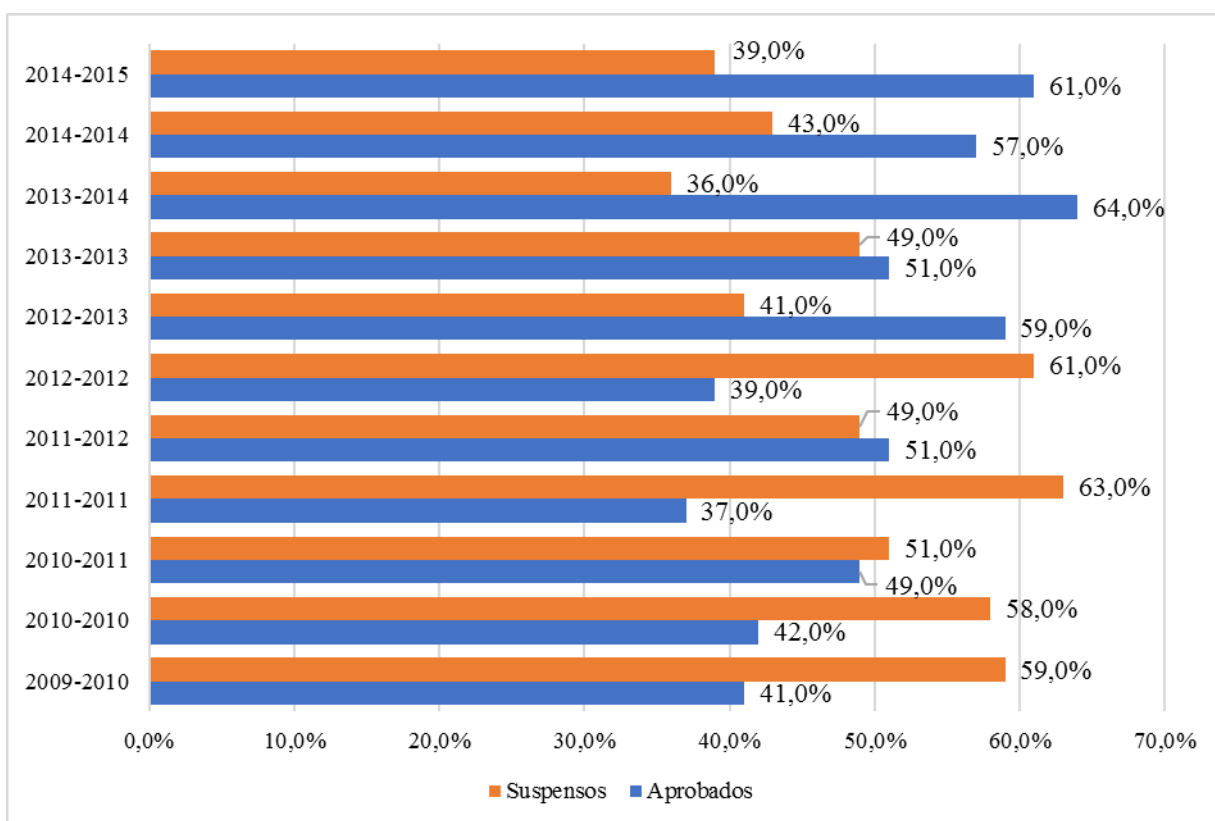


Figura 9. Porcentaje de estudiantes aprobados y suspensos desde 2009 hasta 2015
Fuente: [15]

Profesorado: el análisis de la variable «profesor» tiene como objetivo evaluar la influencia de los mismos en los resultados académicos, ya que como vimos anteriormente, a partir del semestre octubre 2012-marzo 2013, hubo una mejora, etapa que coincidió con el retiro de aquellos docentes con mayor experiencia, más exigencia y complejidad en los exámenes de la asignatura.

A partir del semestre octubre 2012-marzo 2013 comienzan a impartir la asignatura varios nuevos docentes, y el promedio de aprobados subió significativamente a un 73,0 % y el

promedio de suspensos bajó al 27,0 %, comparados con los semestres anteriores donde el porcentaje de aprobados fue del 43,2 % y el 56,8 % de suspensos.

En el semestre octubre 2012-marzo 2013 ingresa el Profesor 2, en el semestre abril-septiembre 2013 se integran los Profesores 5 y 14, en el semestre octubre 2013-marzo 2014 se integra el Profesor 22 y durante el semestre abril-septiembre 2014 ingresan los Profesores 1, 11 y 6. Durante el periodo de estudio y hasta la actualidad, el único profesor que se mantiene impartiendo ininterrumpidamente la asignatura Programación I es el Profesor 4 (Figura 10). Estos resultados muestran que el porcentaje de estudiantes aprobados y suspensos depende significativamente del nivel de exigencia, complejidad de los exámenes y dedicación de los docentes que imparten la asignatura.

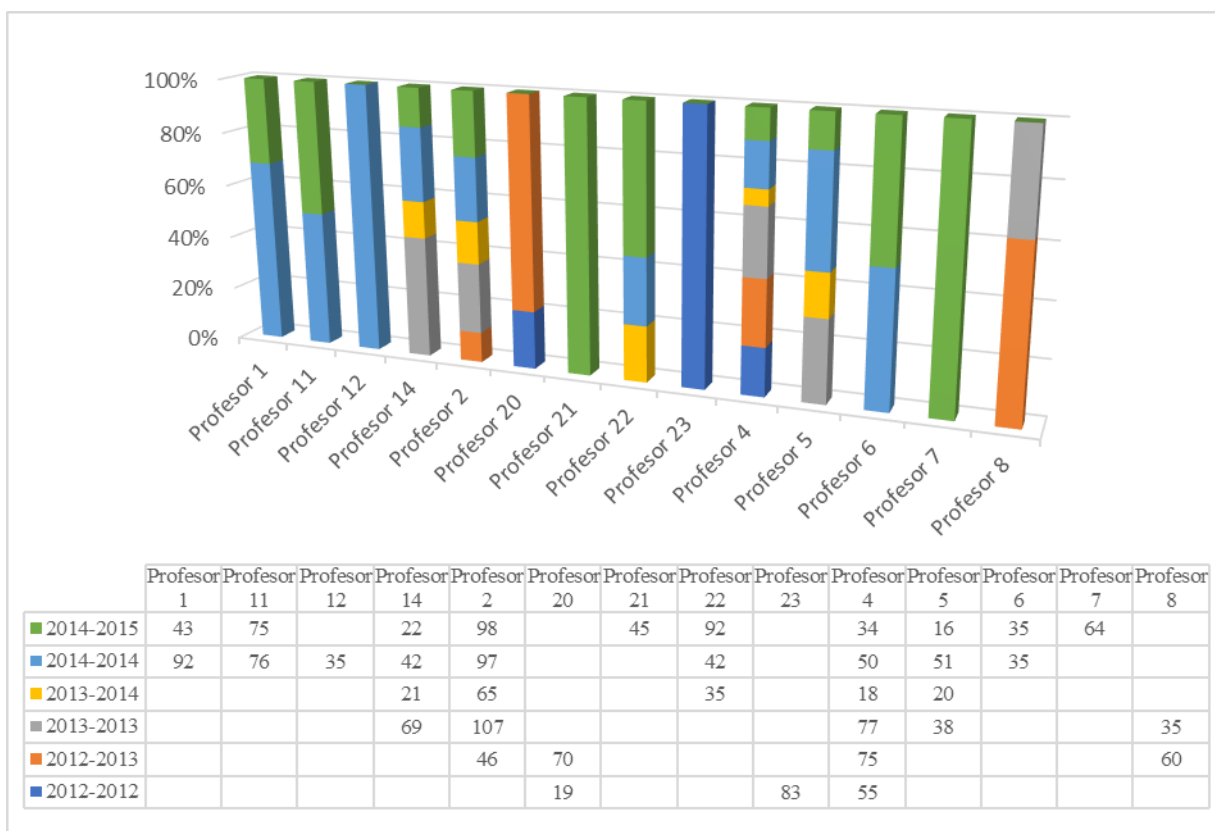


Figura 10. Cantidad de estudiantes por profesor en cada semestre
Fuente: elaboración propia

Teniendo en cuenta que diferentes profesores impartieron la asignatura en distintos momentos, se pudo hacer el análisis siguiente y que se representa en la Figura 11, donde se lista el porcentaje de estudiantes que aprobaron y suspendieron Programación I por cada docente. Con el Profesor 5, el porcentaje de aprobados fue del 87,0 %, con el Profesor 2 del 71,0 %, mientras que con el Profesor 4 el índice de aprobados se situó en un 24,0 %, y con el Profesor 14 en un 26,0 %. Aunque el sílabo y el programa docente es el mismo, se aprecia que el porcentaje de alumnos aprobados y suspensos entre profesores no fue equitativo, y en algunos casos difiere mucho del porcentaje general, 53,1 % aprobados y 46,9 % suspensos [15].

Con los profesores 4 y 14 aprobaron menos estudiantes. Ellos tuvieron un elevado nivel de exigencia, además que los exámenes aplicados tenían más complejidad en comparación con los de los demás profesores.

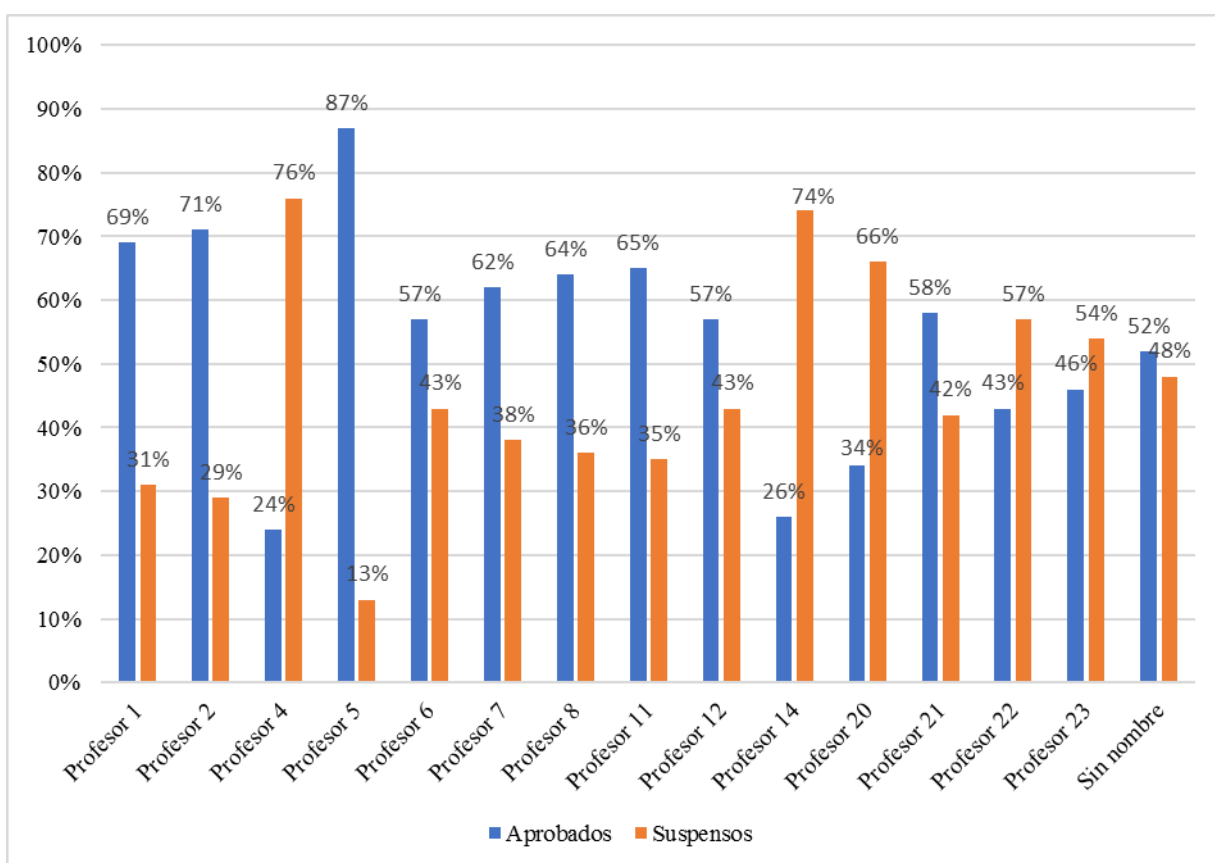


Figura 11. Porcentaje de estudiantes aprobados y suspensos en Programación I, por cada docente
Fuente: [15]

Los autores Beltrán, Sánchez y Rico [15], durante su investigación detectaron que, entre los semestres octubre 2009-marzo 2010 y abril-septiembre 2014 en Programación I aprobaron 1188 estudiantes, pero solo el 71,0 % de ellos se matricularon en la asignatura Programación II entre los semestres abril-septiembre 2010 y octubre 2014-marzo 2015. De los matriculados en Programación II, el 72,0 % aprobaron y el 28,0 % suspendieron el curso.

En Programación II, con los profesores 15 y 9 aprobaron el 52,0 %, y reprobaron el 48,0 %; en cambio, con el Profesor 19, aprobaron el 94,0 %, y reprobaron el 6,0 %.

Los estudiantes que más aprobaron Programación II con los profesores 9 y 15 vienen de los cursos de Programación I con los profesores 4 y 14 (con quienes menos aprobaron Programación I). En cambio, los alumnos que más suspendieron Programación II con los profesores 9 y 15, tomaron Programación I con los profesores 2 y 5 (con quienes más aprobaron Programación I). Con el Profesor 19 fue con quien mayor cantidad de estudiantes aprobaron, representando el 94,0 %, sin importar con qué docente recibieron Programación I. Existe entonces una relación entre el porcentaje de estudiantes que aprobaron y suspendieron Programación II con el profesor que dictó Programación I (Figura 12). En las columnas se encuentran los docentes de Programación II que mayor cantidad de alumnos tuvieron, y en las filas el docente con el que aprobaron los estudiantes Programación I.

Se puede afirmar que el porcentaje de estudiantes que aprueban o no Programación I y II, en general depende del profesor que imparte cada asignatura; específicamente la metodología y herramientas usadas, así como la exigencia de los docentes y la dificultad de los exámenes.

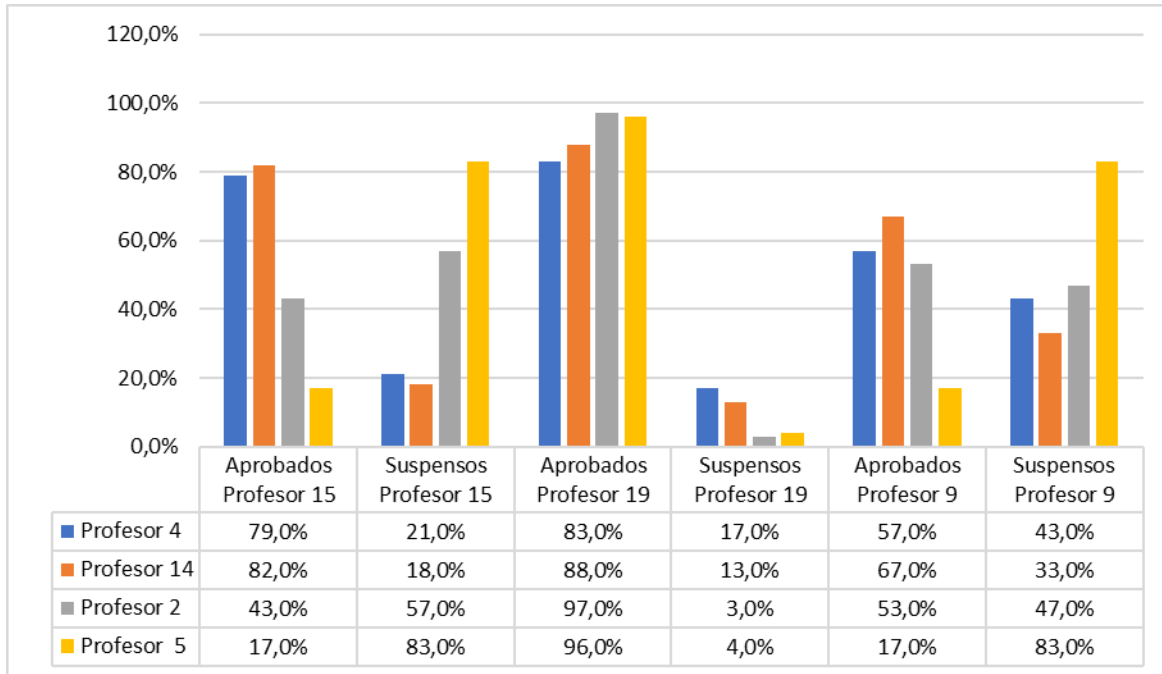


Figura 12. Relación de alumnos que aprobaron Programación I y luego recibieron Programación II
Fuente: [15]

Para confirmar los resultados anteriormente descritos, se realizó un análisis de componentes principales (Anexo 1), y de minería de datos, realizado en Microsoft SQL Server, el cual aplica el algoritmo de Bayes. De esta forma se confirmó que la variable de mayor impacto sobre los resultados es el profesor con el que se cursó la asignatura [15].

Análisis de influenciadores clave - Algoritmo de Bayes			
Variable	Valor	Favorece aprobado	Favorece suspenso
Profesor	Profesor 4		100
Profesor	Profesor 5	55,6	
Profesor	Profesor 2	51,8	
Profesor	Profesor 14		38,8
Número matrícula	1		15,9
Número matrícula	3	11,2	
Profesor	Profesor 1	8,1	
Profesor	Profesor 20		8,1
Número matrícula		6	
Profesor	Profesor 11	3,2	
Profesor	Profesor 22		2,3

Figura 13. Análisis de elementos que influyen en que un alumno apruebe o no Programación I
Fuente: [15]

A modo de resumen del análisis cuantitativo después del análisis realizado por el investigador se encontró que las variables «género», «tipo de provincia» y «tipo de colegio» no influyeron en que el estudiante aprobara o no Programación I. Con respecto al «semestre» se muestra que hasta el periodo octubre 2012-marzo 2013, con excepción del semestre octubre 2011-marzo 2012, el porcentaje de estudiantes suspensos fue mayor que el de aprobados. Desde el semestre octubre 2012-marzo 2013 en adelante, la relación fue la contraria. Además, se evidenció que el profesorado tiene gran influencia en la cantidad de estudiantes aprobados y suspensos en Programación I y II.

5.1.3. Análisis cualitativo

Una vez identificado que el «profesorado» es la variable que más influye en los resultados de la evaluación, se llevó a cabo un estudio cualitativo, basado en la aplicación de encuestas anónimas a docentes y estudiantes en el semestre octubre 2014-marzo 2015, con el fin de obtener información referente a las causas que ocasionan el alto porcentaje de estudiantes suspensos, y el criterio de los docentes y alumnos sobre la metodología y herramientas utilizadas por los docentes al impartir la asignatura.

En el análisis cualitativo realizado por los autores Beltrán, Sánchez y Rico [15] se obtuvieron los siguientes resultados:

5.1.3.1. Encuestas a docentes

La encuesta fue aplicada a seis de los nueve docentes de la universidad (uno de los profesores se retiró) que impartieron la asignatura Programación I en el semestre octubre 2014-marzo 2015 (Anexo 2).

P1. Las siguientes herramientas tecnológicas que usted utiliza, ¿facilitan el aprendizaje de Programación I?

A continuación, se muestran las opiniones principales de los docentes sobre las herramientas tecnológicas utilizadas que facilitan el aprendizaje de Programación I.

Los resultados evidencian que el 100,0 % de los docentes opinan que el *software* especializado (PSeInt, NetBeans o Eclipse) facilita el aprendizaje. Otro resultado relevante está asociado a la herramienta de plataforma virtual, donde el 50,0 % de los docentes cree que facilita el aprendizaje y el resto que lo facilita parcialmente. De manera similar, el 50,0 % de los docentes opina que Internet facilita el aprendizaje, el 33,3 % que lo facilita parcialmente y el 16,7 % que lo facilita mínimamente. En cuanto al correo electrónico, redes sociales y audio, los profesores afirmaron que facilitan, facilitan parcialmente y mínimamente el aprendizaje en el 33,3 %.

Exceptuando los *softwares* especializados para el aprendizaje de programación, los demás que se mencionan en la Figura 14 son herramientas que permiten investigar y aprender la teoría de la asignatura estudiada para luego practicar lo aprendido.

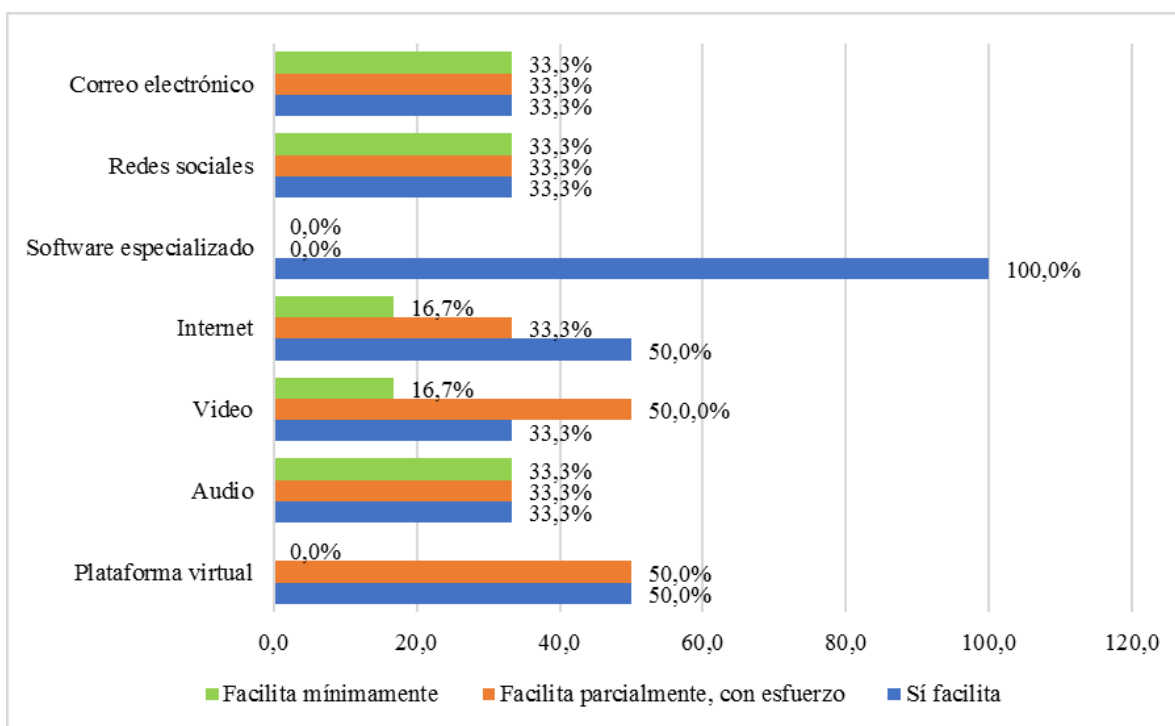


Figura 14. Herramientas que facilitan el aprendizaje de Programación I
Fuente: [15]

Los IDE (Entorno de Desarrollo Integrado, *Integrated Development Environment* por sus siglas en inglés) de desarrollo de *software* NetBeans y Eclipse son muy buenos para comenzar a aprender a programar en Java [80].

P2. Los siguientes métodos de enseñanza que usted utiliza, ¿facilitan el aprendizaje de Programación I?

Los aspectos metodológicos principales aplicados en la enseñanza de Programación I por los docentes se resumen en la Figura 15. Es en estos aspectos donde el profesor tiene el control del proceso de enseñanza-aprendizaje durante las clases que imparte en el aula, además de ser los que facilitan mayormente el aprendizaje de programación, como prácticas de laboratorio, exposición oral y prácticas de campo; y no siendo así, cuando el estudiante realiza trabajo autónomo fuera del aula.

En los criterios de los docentes se evidencia que las tareas que deben desarrollar los estudiantes fuera del aula facilitan el aprendizaje en un 33,3 %, el 50,0 % respondieron que lo facilitan parcialmente y el 16,7 % mínimamente. Este es un elemento que debe ser resuelto en la asignatura Programación I por ser una asignatura que requiere de mucha dedicación y estudio para ser entendida. En tanto, el autor Cantor [81, p. 18] expresa que «el trabajo autónomo es un instrumento donde el estudiante obtiene un resultado sin ayuda del profesor y aunque sean pocos y de bajo nivel los ejercicios a realizar, se debe lograr enfatizar y ayudar al estudiante para su aprendizaje». Además, mediante los trabajos y/o las tareas escolares, los docentes pueden identificar a los estudiantes con deficiencias o dificultades en su aprendizaje, para poder hacer trabajo personalizado y que mejoren.

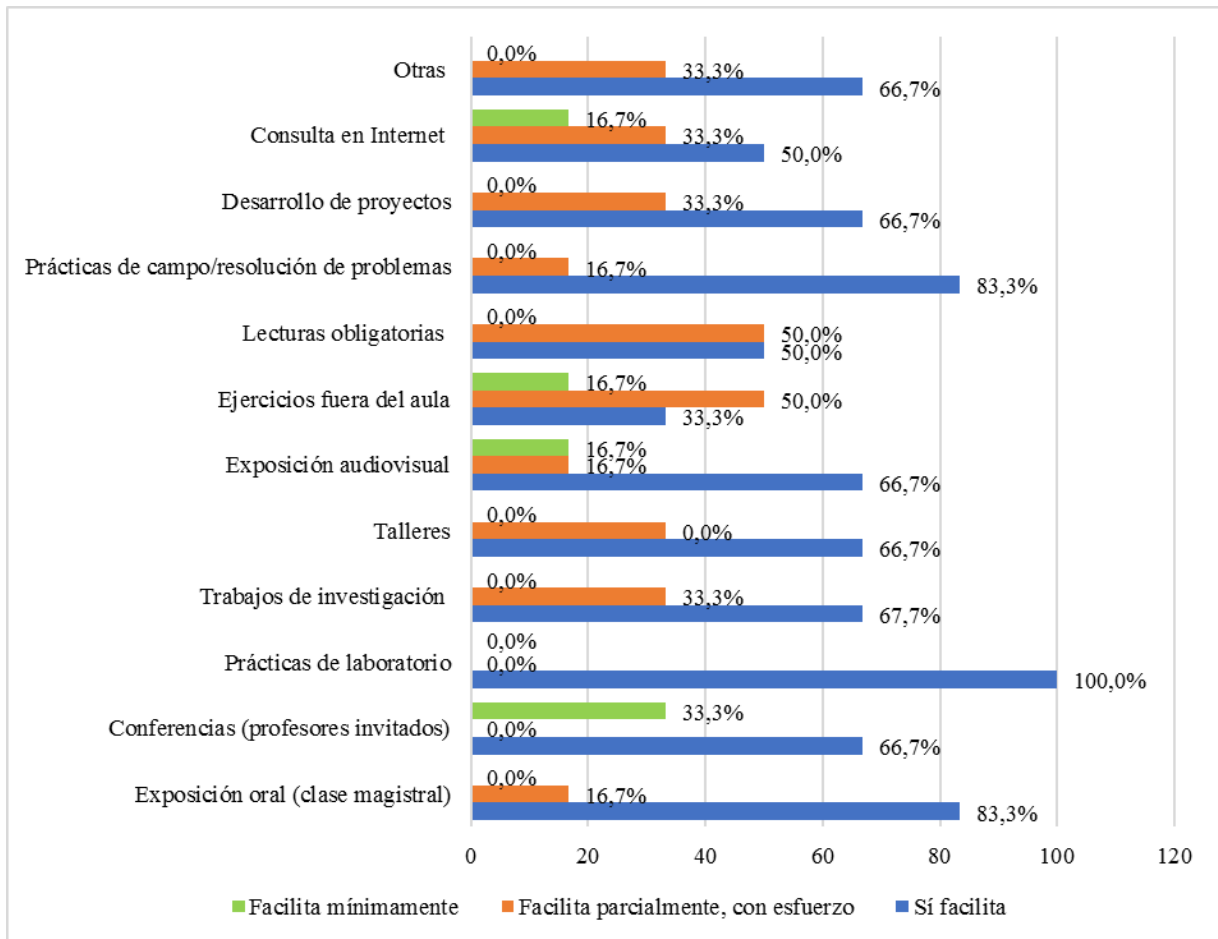


Figura 15. Métodos de enseñanza que facilitan el aprendizaje de Programación I
Fuente: [15]

De acuerdo con Cantor, el trabajo autónomo fuera del aula beneficia a los alumnos en los siguientes aspectos [81, p. 18]:

- Practican y refuerzan los contenidos y habilidades académicas adquiridas en el aula.
- Propicia la formación del alumno y se fortalecen buenos hábitos de estudio.
- Es el medio que les permite desarrollar su creatividad, iniciativa e ingenio.
- Les crea hábitos de estudio.
- Aprenden a planificar su tiempo libre en actividades importantes para ellos y los demás.
- Promueve su responsabilidad, su autonomía y su autoestima.

P3. ¿Cuáles considera que son los principales problemas que enfrenta el estudiante en el aprendizaje de Programación I?

Ante esta pregunta abierta, los docentes expresaron que los principales problemas que enfrentan los estudiantes en el aprendizaje de la asignatura Programación I son los siguientes [15]:

- Falta de motivación para el autoaprendizaje.
- Falta de conocimientos matemáticos básicos previos.
- Deficiencia de equipos en los laboratorios.
- Deficiencia en el acceso a Internet.
- Muchos alumnos por curso.

P4. ¿Qué aspectos considera usted que faltan dentro de la metodología aplicada por la facultad para un adecuado aprendizaje de Programación I?

Los docentes expresan que deben considerarse algunos elementos metodológicos que afectan al aprendizaje de programación, y la mejora de los mismos puede lograr satisfactorios resultados académicos. Los elementos son los siguientes [15]:

- El examen unificado y la exigencia análoga, por parte de todos los profesores de una misma carrera.
- La estandarización del contenido y el alcance de la asignatura deben ser los mismos para todos los profesores de una misma carrera, y deben regirse al sílabo de la asignatura.
- Reforzar los contenidos, guías, prácticas de laboratorio y recursos para el proceso de enseñanza-aprendizaje.
- La mejora del acceso a Internet, laboratorios y el uso de la plataforma virtual.

Por último, los docentes en una misma carrera se guiaron por el sílabo de la asignatura, pero aplicaron la metodología de aprendizaje de diferente forma, así como los recursos y herramientas tecnológicas usados por los profesores no fueron los mismos, y tampoco la

exigencia en el proceso evaluativo, acarreando distintos resultados académicos de los grupos de estudiantes por docente.

P5. ¿Cómo calificaría usted la metodología usada en el proceso de enseñanza-aprendizaje, de acuerdo a las competencias de Programación I?

La opinión de los docentes acerca de la metodología usada en el proceso de enseñanza-aprendizaje de acuerdo a las competencias de Programación I se muestra en la Figura 16.

Con respecto al análisis y desarrollo de algoritmos sencillos utilizando fundamentos y técnicas de programación estructurada, el 16,7 % los califica de muy buenos, el 66,6 % de buenos y el resto de regulares. Un resultado similar se obtuvo en el indicador relacionado con la construcción de programas sencillos en lenguaje Java, donde el 16,7 % de los docentes lo calificó como muy bueno y el 83,3 % restante de bueno.

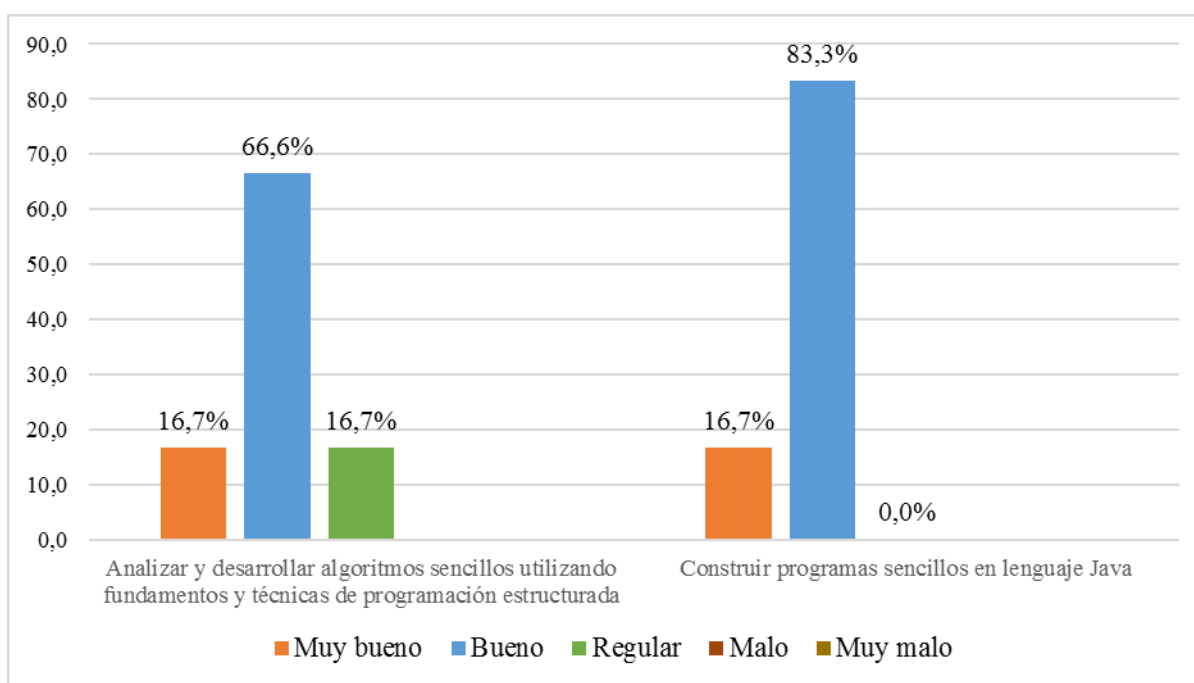


Figura 16. Opinión sobre la metodología de acuerdo a las competencias adquiridas
Fuente: elaboración propia

P6. ¿Considera que la motivación es un factor importante en el aprendizaje de programación?

Ante la pregunta sobre si la motivación es un factor importante en el aprendizaje de programación, todos los docentes respondieron estar totalmente de acuerdo.

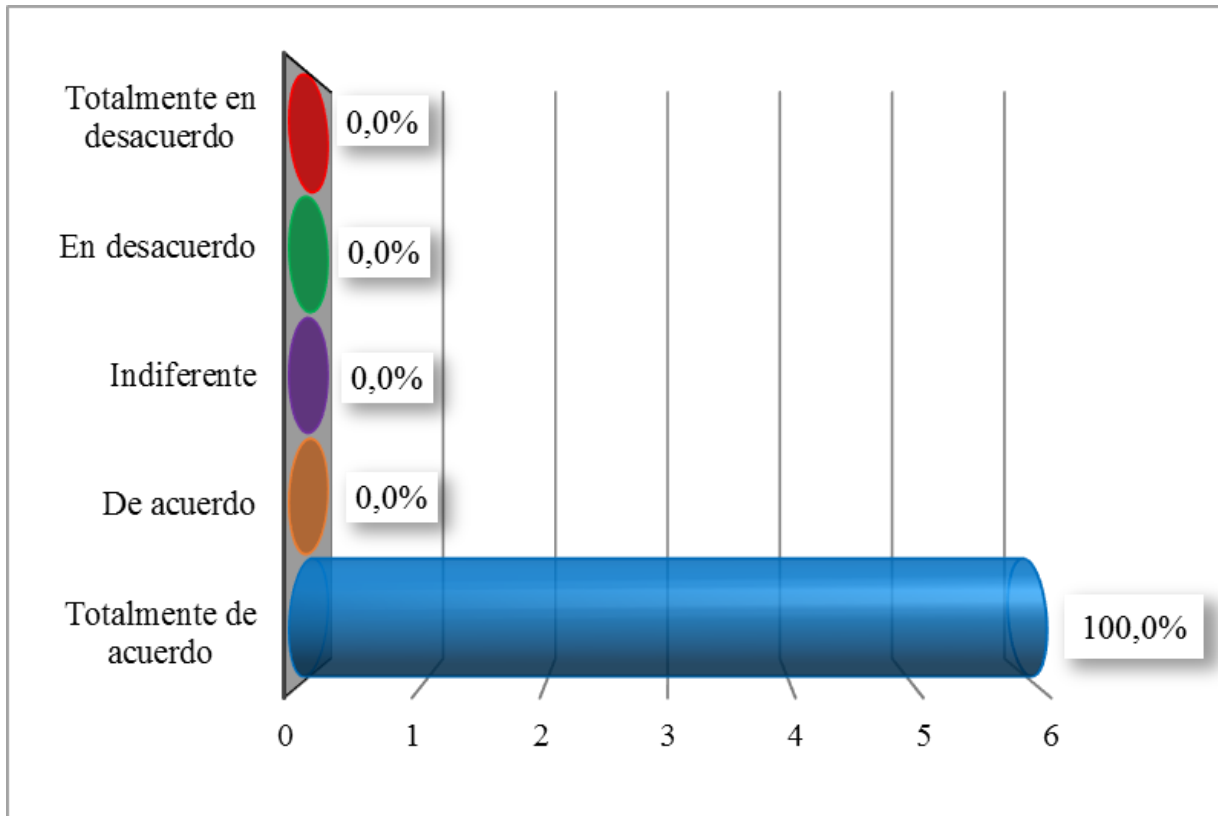


Figura 17. Motivación como factor importante en el aprendizaje
Fuente: elaboración propia

P7. ¿Considera que la gamificación usada en un ambiente *e-learning* puede ayudar en el proceso de enseñanza-aprendizaje de Programación I?

La Figura 18 expone el criterio de los docentes sobre la gamificación usada en un ambiente *e-learning*; donde el 66,6 % está totalmente de acuerdo con que la misma puede ayudar en el proceso de aprendizaje, el 16,7 % está de acuerdo y el resto, 16,7 %, sostiene una opinión indiferente. Ninguno de los profesores manifestó estar en desacuerdo en alguna medida.

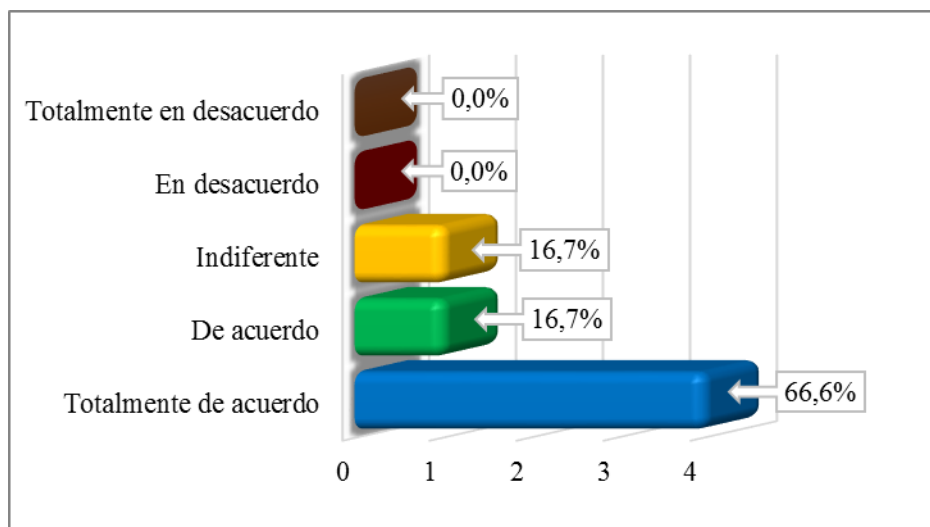


Figura 18. Criterio sobre la gamificación usada en un ambiente *e-learning*
Fuente: elaboración propia

5.1.3.2. Encuestas a estudiantes

Como se mencionó en el capítulo IV, *Metodología*, la encuesta fue aplicada a 85 estudiantes de los 505 matriculados en Programación I en el semestre octubre 2014-marzo 2015, a quienes se envió un correo electrónico solicitando rellenasen la encuesta en forma anónima.

Caracterización de la muestra: de los estudiantes encuestados el 36,5 % son mujeres y el 63,5 % hombres, su edad oscila entre 17 y 28 años. En cuanto al colegio en el que se graduaron, el 69,4 % de los estudiantes estudiaron en colegios públicos, el 25,9 % en colegios privados y el 4,7 % en colegios municipales (Tabla 5).

Los perfiles a los cuales pertenecían los estudiantes son los siguientes: de Ingeniería Informática un 32,9 %, de Ingeniería en Diseño Industrial el 29,4 %, de Ingeniería en Computación Gráfica el 21,2 %, y de Ingeniería Matemática el 16,5 %. De ellos el 54,1 % son de primera matrícula y el 45,9 % de segunda. Pasaron el curso solo el 70,6 % y el 29,4 % no lo aprobaron (Tabla 5).

Tabla 5. Descripción de la muestra de estudiantes encuestados

Variables		Frecuencia	Porcentaje %
Género	Masculino	54	63,5
	Femenino	31	36,5
Colegio de graduado	Fiscal/público	59	69,4
	Particular/privado	22	25,9
	Municipal	4	4,7
Carrera	Informática	28	32,9
	Matemática	14	16,5
	Computación Gráfica	18	21,2
	Diseño Industrial	25	29,4
Universidad Central como primera opción	Sí	62	72,9
	No	23	27,1
Carrera que estudia es primera opción	Sí	37	43,5
	No	48	56,5
Número de matrículas	Primera	46	54,1
	Segunda	39	45,9
	Tercera	0	0,0
Pasó el curso	Sí	60	70,6
	No	25	29,4
Alguna asignatura previa de programación	Sí	26	30,6
	No	59	69,4

Fuente: elaboración propia

Al consultar a los estudiantes si fue la carrera que estudian su primera opción, en la Figura 19 se exponen las áreas de conocimiento que les gustaría estudiar a aquellos alumnos que no la eligieron como primera opción. En más de la mitad de los casos (55,3 %) les gustaría estudiar carreras relacionadas con ingeniería, industria y construcción.

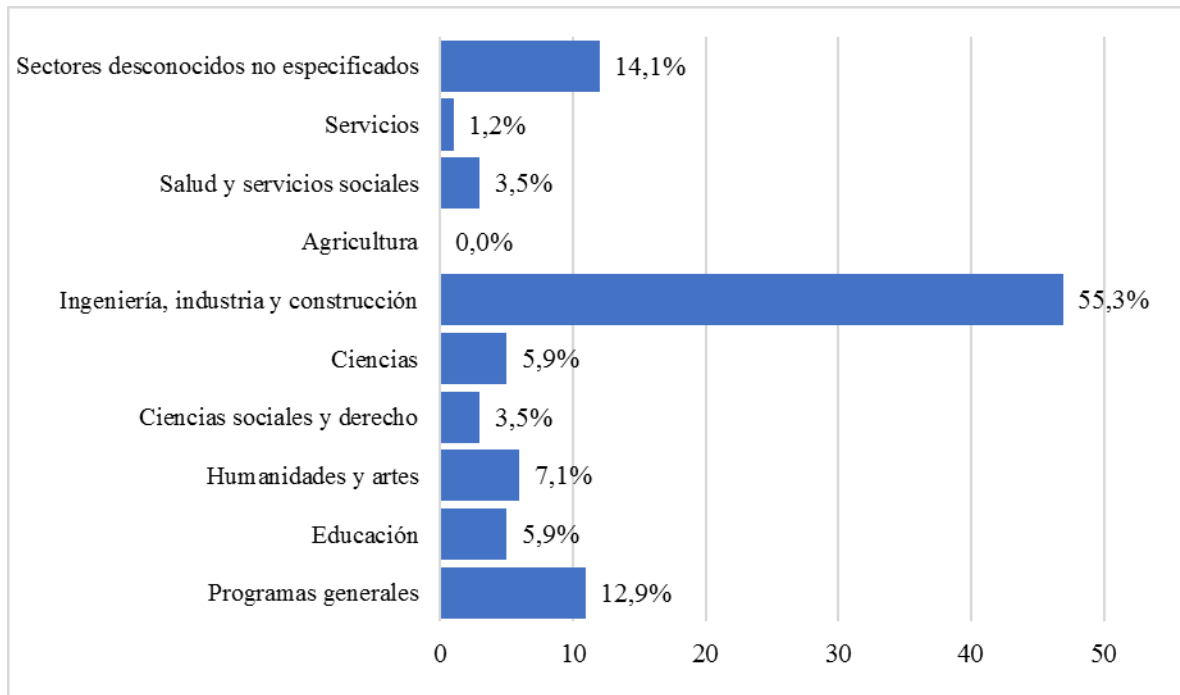


Figura 19. Áreas de conocimiento que les gustaría estudiar
Fuente: elaboración propia

Las preguntas realizadas a los estudiantes fueron:

P1. ¿Considera que aprender a programar es importante para su futuro profesional?

En la Figura 20 se muestra que el 67,1 % de los alumnos expresó estar totalmente de acuerdo y el 27,1 % de acuerdo, que para su futuro como profesional es importante aprender a programar sin importar la carrera que estudien; el 3,50 % fue indiferente a la pregunta y el 2,40 % estuvo en desacuerdo y totalmente en desacuerdo.

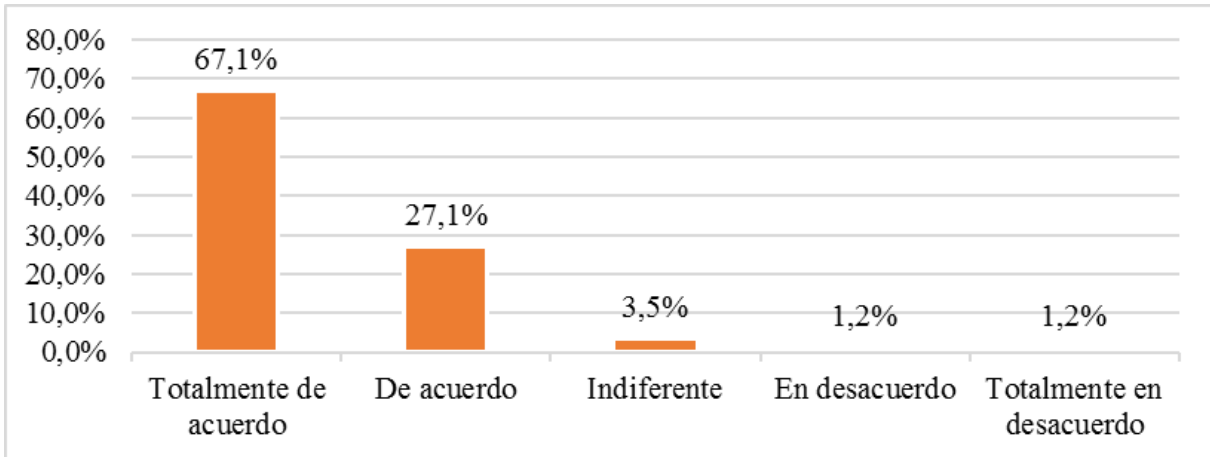


Figura 20. Importancia de aprender a programar.
Fuente: elaboración propia

P2. ¿Considera que aprender a programar es difícil?

En tanto, los resultados de esta pregunta muestran que aprender a programar es difícil como se ve en la Figura 21, donde el 55,2 % está totalmente de acuerdo y de acuerdo, un 27,1 % fue indiferente a la pregunta y el 17,70 % se encuentra en desacuerdo y total desacuerdo. Por tanto, la mayoría de la muestra plantea que es difícil aprender a programar, siendo necesario que los alumnos dediquen gran parte del tiempo al estudio de la asignatura y para lograrlo deben estar motivados e incentivados por la programación.

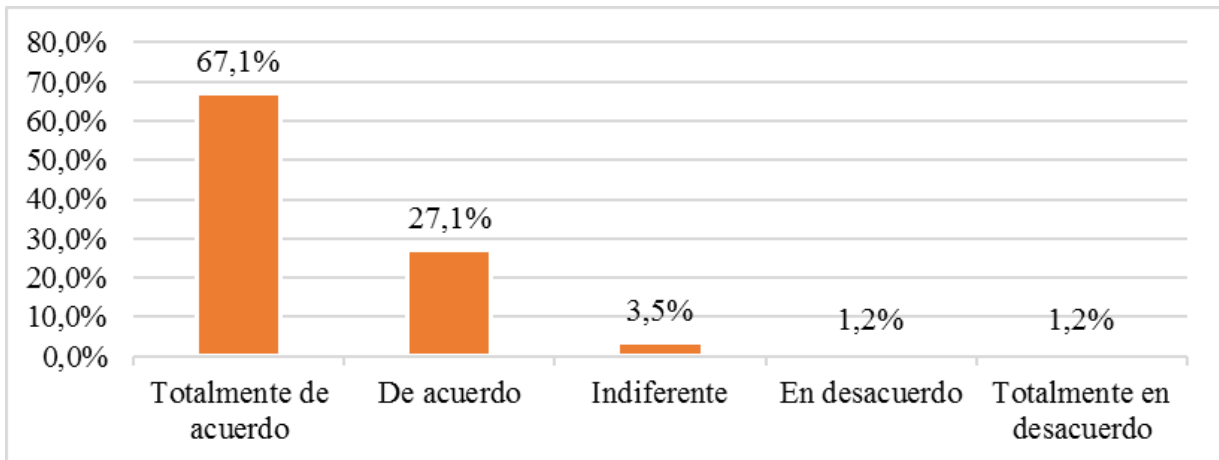


Figura 21. Opinión de los alumnos sobre si aprender a programar es difícil.
Fuente: elaboración propia

P3. ¿Considera que aprender a programar es aburrido?

El criterio de los alumnos respecto a si aprender a programar es aburrido acarrió que la mayoría se encuentra en total desacuerdo y en desacuerdo como se representa en la Figura 22, representando un 25,9 % y 48,2 % respectivamente, resultado que se explica pues consideran divertido hacer programas y ponerlos en funcionamiento.

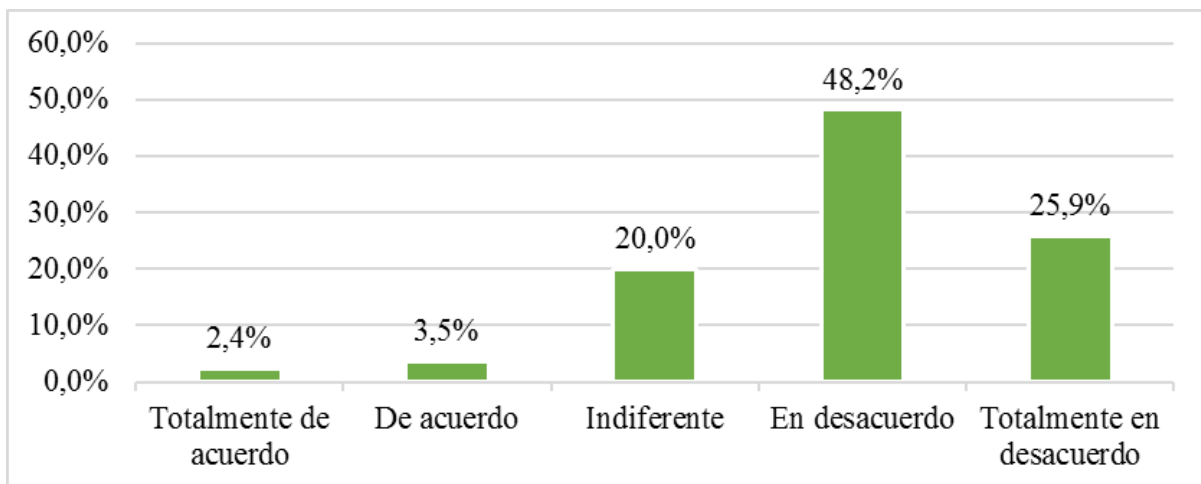


Figura 22. Opinión de los alumnos respecto a si aprender a programar es aburrido.
Fuente: elaboración propia

P4. ¿Qué herramientas tecnológicas utiliza en el aprendizaje de Programación I?

Según se muestra en la Tabla 6, la herramienta más utilizada como apoyo en el aprendizaje de la asignatura Programación I es Internet, usada siempre o frecuentemente por un 67,1 % de los estudiantes. Los *softwares* especializados (PSeInt, NetBeans o Eclipse) se usaron siempre o frecuentemente en un 62,3 %, la plataforma virtual Sakai en el 32,9 %, siendo menos usadas las redes sociales con un 20,0 %.

Tabla 6. Herramientas tecnológicas utilizadas en el aprendizaje de Programación I

Herramienta	Siempre		Frecuentemente		Algunas veces		Rara vez		Nunca	
	Frec.	%	Frec.	%	Frec.	%	Frec.	%	Frec.	%
Plataforma virtual	11	12,9	17	20,0	32	37,6	15	17,6	10	11,8
Audio	8	9,4	17	20,0	23	27,2	18	21,2	19	22,4
Video	17	20,0	21	24,7	22	25,9	9	10,6	16	18,8
Internet	36	42,4	21	24,7	14	9,4	8	9,4	6	7,1
Software especializado	29	34,1	24	28,2	18	27,1	8	9,4	6	7,1
Redes sociales	6	7,1	11	12,9	23	27,1	26	20,6	19	22,4
Correo electrónico	10	11,8	16	18,8	25	29,4	17	20,0	17	20,0

Fuente: elaboración propia

P5. Considera que el uso de las herramientas tecnológicas utilizadas en el aprendizaje de Programación I fue:

En cuanto al uso de las herramientas tecnológicas en el aprendizaje de programación I (Figura 23), el 44,7 % de los estudiantes considera que fue muy bueno y bueno, el 44,7 % afirmó que fueron regulares las herramientas usadas, y el 10,6 % afirmó que son malas o muy malas.

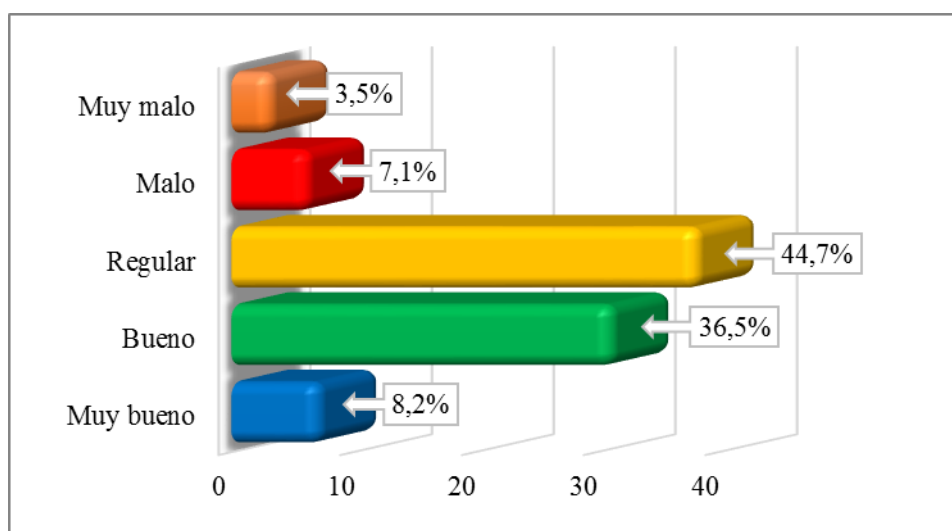


Figura 23. Uso de las herramientas tecnológicas en el aprendizaje de Programación I

Fuente: elaboración propia

P6. Considera usted que los recursos y tecnologías utilizadas en el aprendizaje de Programación permiten:

Según los recursos y tecnologías utilizadas en el aprendizaje de Programación I (Tabla 7), los estudiantes expresaron estar totalmente de acuerdo y de acuerdo en un 84,71 % en que el uso de los recursos tecnológicos favorece a la práctica, en un 74,1 % que los motivan, el 71,8 % mencionó que hace más fácil el aprendizaje, el 70,6 % que permiten mantener el interés en las clases y un 76,4 % que optimizan el aprendizaje. Sin embargo, un 30,6 % de estudiantes pensaron en forma indiferente, estando en desacuerdo o totalmente en desacuerdo con la efectividad del uso de las herramientas usadas.

Esto evidencia nuevamente que el uso de las herramientas tecnológicas no se encuentra estandarizado, ni generalizado entre los docentes en su estrategia de enseñanza, pudiendo ser optimizado para el uso de los estudiantes y así tener mayor efectividad en el aprendizaje de la asignatura Programación I [15].

Tabla 7. Recursos y tecnologías utilizadas en el aprendizaje de Programación I

Recursos y tecnologías	Totalmente de acuerdo		De acuerdo		Indiferente		En desacuerdo		Totalmente en desacuerdo	
	Frec.	%	Frec.	%	Frec.	%	Frec.	%	Frec.	%
Motivan al estudiante	20	23,5	43	50,6	19	22,4	2	2,4	1	1,2
Hacen más fácil el aprendizaje	21	24,7	40	47,1	15	17,6	8	9,4	1	1,2
Permiten mantener el interés en las clases	20	23,5	40	47,1	16	18,8	6	7,1	3	3,5
Favorecen la práctica	34	40,0	38	44,7	11	12,9	2	2,4	0	0,0
Optimizan el aprendizaje	28	32,9	37	43,5	12	14,1	8	9,4	0	0,0
No son los más adecuados para el aprendizaje	13	15,3	13	15,3	23	27,1	30	35,3	6	7,1

Fuente: elaboración propia

P7. ¿Cuáles son los aspectos metodológicos que usted considera favorecen más en el aprendizaje de Programación I?

Los resultados de este aspecto analizado en la encuesta arrojaron que los estudiantes se sintieron más a gusto con las metodologías aplicadas en el aula, tales como: prácticas de laboratorio y el desarrollo de ejercicios en clase. Sin embargo, los trabajos autónomos como: ejercicios fuera del aula, trabajos investigativos, lecturas obligatorias y la exposición oral o clase magistral, no fueron muy acogidos por los alumnos. Además, se puede observar que a los alumnos les gusta más la práctica que la teoría (Tabla 8).

Tabla 8. Aspectos metodológicos que favorecen más el aprendizaje de Programación I

Aspectos metodológicos	Totalmente de acuerdo		De acuerdo		Indiferente		En desacuerdo		Totalmente en desacuerdo	
	Frec.	%	Frec.	%	Frec.	%	Frec.	%	Frec.	%
Exposición oral	17	20,0	30	35,3	26	30,6	8	9,4	4	4,7
Ejercicios dentro de clase	50	58,8	32	37,6	3	3,5	0	0,0	0	0,0
Conferencias (profesores invitados)	24	28,2	34	40,0	20	23,5	5	5,9	2	2,4
Prácticas de laboratorio	65	76,5	19	22,4	1	1,2	0	0,0	0	0,0
Trabajos de investigación	25	29,4	34	40,0	17	20,0	7	8,2	2	2,4
Talleres	38	44,7	37	43,5	9	10,6	1	1,2	0	0,0
Exposición audiovisual	33	38,8	36	42,4	12	14,1	3	3,5	1	1,2
Ejercicios fuera del aula	29	34,1	32	37,6	17	20,0	4	4,7	3	3,5
Lecturas obligatorias	16	18,8	25	29,4	26	30,6	9	10,6	9	10,6
Prácticas de campo/resolución de problemas	45	52,9	30	35,3	8	9,4	1	1,2	1	1,2
Desarrollo de proyectos	37	43,5	20	23,8	15	17,6	4	4,7	0	0,0
Consulta en Internet	34	40,0	35	41,2	12	14,1	4	4,7	0	0,0
Otros	18	21,2	25	29,4	36	42,4	2	2,4	4	4,7

Fuente: elaboración propia

En el criterio planteado por los estudiantes respecto a la metodología usada por los docentes al impartir la asignatura, se evidencia la importancia de aplicar una estrategia metodológica con una nueva perspectiva que incentive a los estudiantes al autoaprendizaje mediante la

realización de ejercicios fuera del aula, trabajos investigativos, lectura, reforzar lo aprendido en clase, pues todo estudiante debe ser capaz de aprender nuevas habilidades o desarrollar y optimizar las que ya tienen sin la ayuda de otras personas. Según la autora Gallegos [82], los métodos de autoaprendizaje hacen que un aprendiz cubra los espacios vacíos que existen en su memoria. Los alumnos no son receptores pasivos del conocimiento, sino todo lo contrario, son participantes activos en la interpretación de los modelos que ellos o el docente proponen para que intenten aprender aquello que aún no saben.

P8. Califique la contribución de la metodología usada para el aprendizaje de Programación I en las destrezas listadas.

En la Figura 24 se recogen las calificaciones de los estudiantes respecto a la contribución de la metodología usada para el aprendizaje de Programación I. En cuanto a la destreza de análisis y diseño de algoritmos sencillos utilizando fundamentos de programación estructurada, el 75,3 % refiere que es muy buena y buena, el 18,8 % la evalúa de regular y solo el 5,9 % calificó este indicador de mala y muy mala.

La destreza relacionada con la construcción de programas sencillos en lenguaje Java fue calificada de muy buena y buena por el 67,1 % de los estudiantes, de regular por el 16,5 %; sin embargo, en comparación a la anterior destreza, aumentaron las calificaciones negativas, ya que el 16,5 % de los estudiantes dieron calificaciones de mala y muy mala.

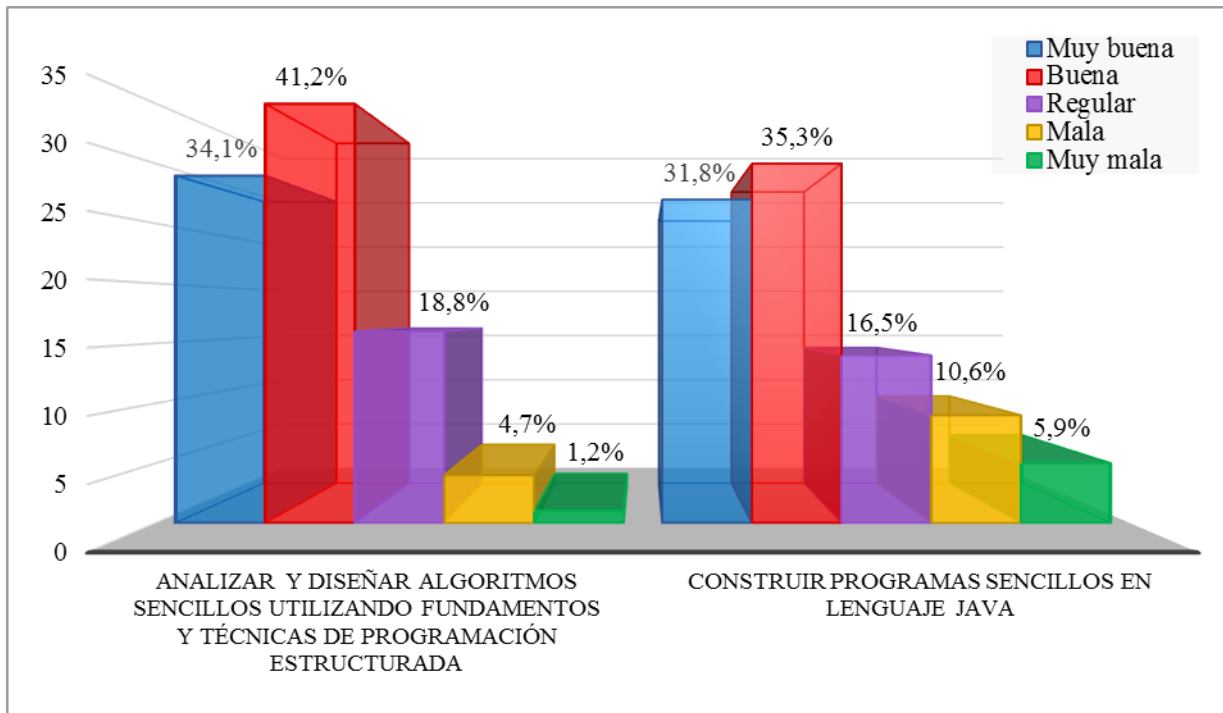


Figura 24. Contribución de la metodología usada para el aprendizaje de Programación I
Fuente: elaboración propia

P9. Considera en general que la metodología utilizada para el aprendizaje de Programación I es...

La Figura 25 indica la evaluación de los estudiantes acerca de la metodología para el aprendizaje de Programación I. De los 85 estudiantes encuestados, cuatro evaluaron la metodología de muy buena representando un 4,7 %, el 85,9 % consideró que era buena y regular, el 7,1 % mala y solo el 2,4 % la evaluaron de muy mala.

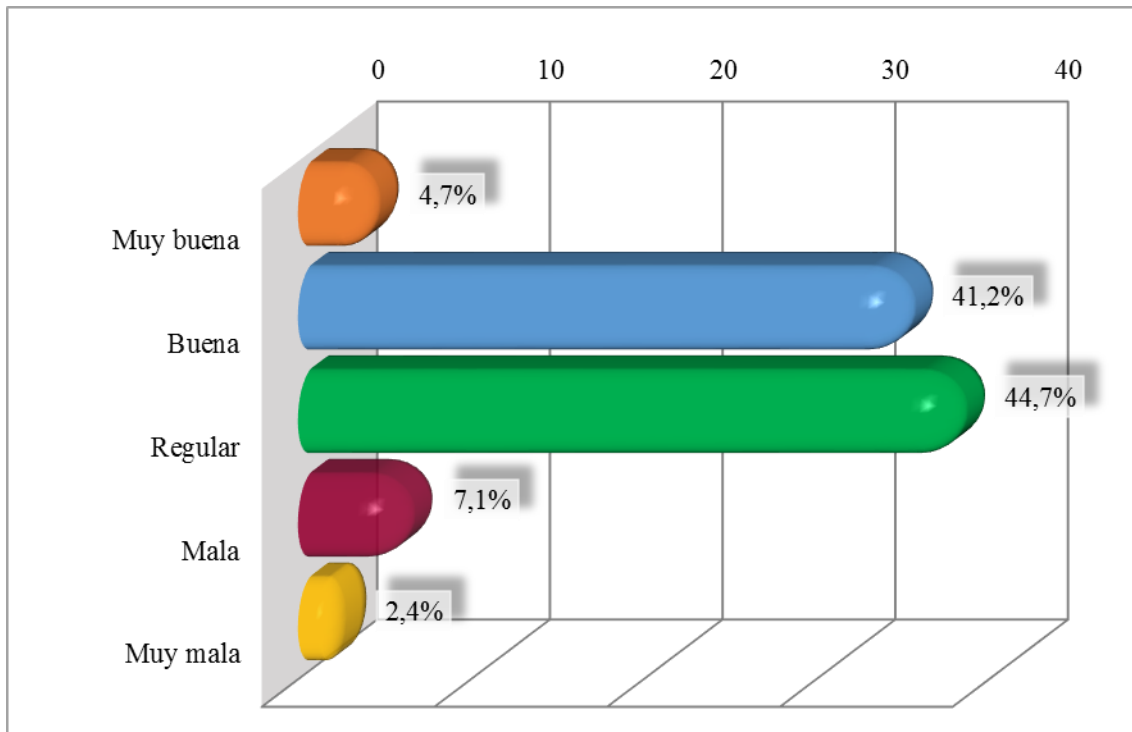


Figura 25. Metodología usada para el aprendizaje
Fuente: elaboración propia

P10. ¿Cuáles son las formas de evaluar que usted considera favorecen más el aprendizaje de Programación I?

La Tabla 9 recoge el criterio de los estudiantes acerca de las formas de evaluación que favorecen más el aprendizaje. Del total de estudiantes, el 87,0 % está totalmente de acuerdo y de acuerdo con que el examen final es la forma de evaluación que más favorece el aprendizaje, seguido de los trabajos fuera del aula en un 84,7 %, resultado que justifica la propuesta realizada. En menor medida, encontramos la participación en las clases (83,6 % de los estudiantes), las pruebas parciales (82,4 %), el trabajo en equipo (82,3 %) y la asistencia a prácticas (81,2 %).

Tabla 9. Formas de evaluar que favorecen más el aprendizaje de Programación I

Formas de evaluar	Totalmente de acuerdo		De acuerdo		Indiferente		En desacuerdo		Totalmente en desacuerdo	
	Frec.	%	Frec.	%	Frec.	%	Frec.	%	Frec.	%
Pruebas parciales	26	30,6	44	51,8	12	14,1	3	3,5	0	0,0
Trabajos y tareas fuera del aula	30	35,3	42	49,4	8	9,4	5	5,9	0	0,0
Participación en las clases	35	41,2	36	42,4	11	12,9	2	2,4	1	1,2
Informes de laboratorio	20	23,5	23	27,1	25	29,4	13	15,3	4	4,7
Examen final	32	37,6	42	49,4	9	10,6	2	2,4	0	0,0
Asistencia a prácticas	36	42,4	33	38,8	13	15,3	2	2,4	1	1,2
Trabajo en equipo	37	43,5	33	38,8	8	9,4	4	4,7	3	3,5
Otras	21	24,7	27	31,8	30	35,3	3	3,5	4	4,7

Fuente: elaboración propia

P11. ¿Considera que la motivación es un factor importante en el aprendizaje de programación?

De acuerdo con el autor Santrock [83], la motivación es «el conjunto de razones por las que las personas se comportan de las formas en que lo hacen. El comportamiento motivado es vigoroso, dirigido y sostenido»; factor que se decidió evaluar en la encuesta para determinar si influye en el aprendizaje de Programación I, arrojando que el 98,8 % de los alumnos expresaron que la motivación es un factor fundamental en su proceso de aprendizaje, y el 1,20 % fue indiferente a esta pregunta (Figura 26).

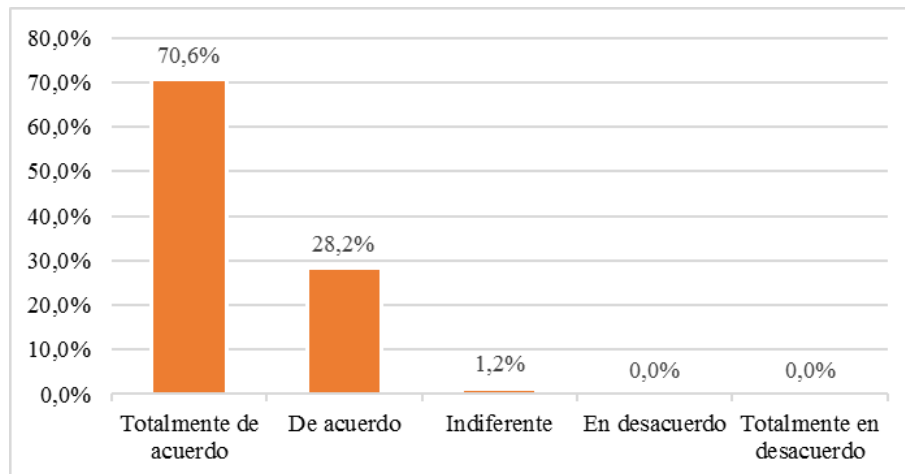


Figura 26. Opinión de los estudiantes sobre la motivación como factor fundamental en el aprendizaje

Fuente: elaboración propia

P12. Califique su grado de motivación en el semestre en el aprendizaje de Programación I.

Como se expone en la Figura 27, el 72,9 % manifestó un grado de motivación bajo y muy bajo, mientras que el 27,1 % restante tenía una motivación muy alta, alta y regular.

De aquí que el grado de motivación es bajo de forma general, hecho que ocurre a medida que la complejidad de los ejercicios aumenta, ya que no pueden resolverlos fácilmente. Por tal motivo es necesario aplicar una metodología con la que, indistintamente de la complejidad de los ejercicios, los estudiantes se encuentren motivados para hacerlos y presten interés al aprendizaje de programación.

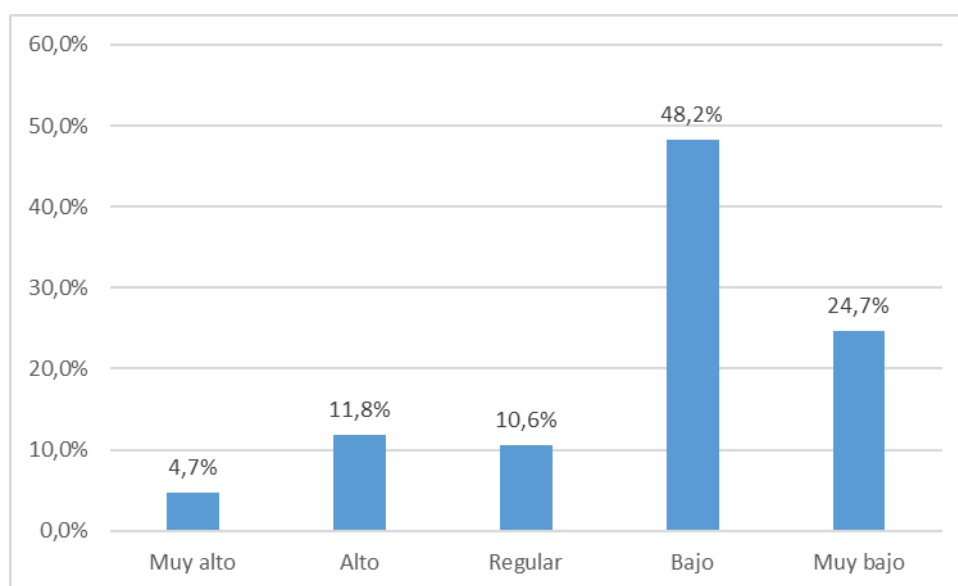


Figura 27. Grado de motivación en el aprendizaje de Programación I
Fuente: elaboración propia

P13. ¿Considera que ha mejorado su nivel de conocimientos de programación en este semestre?

Respecto a la pregunta sobre si ha mejorado su nivel de conocimientos de programación en el semestre, el 81,2 % consideró una mejoría de mucho y medianamente, mientras que el 17,6 % expresó que la mejoría fue poca y un 1,2 % que su nivel de conocimientos no mejoró (Figura 28).

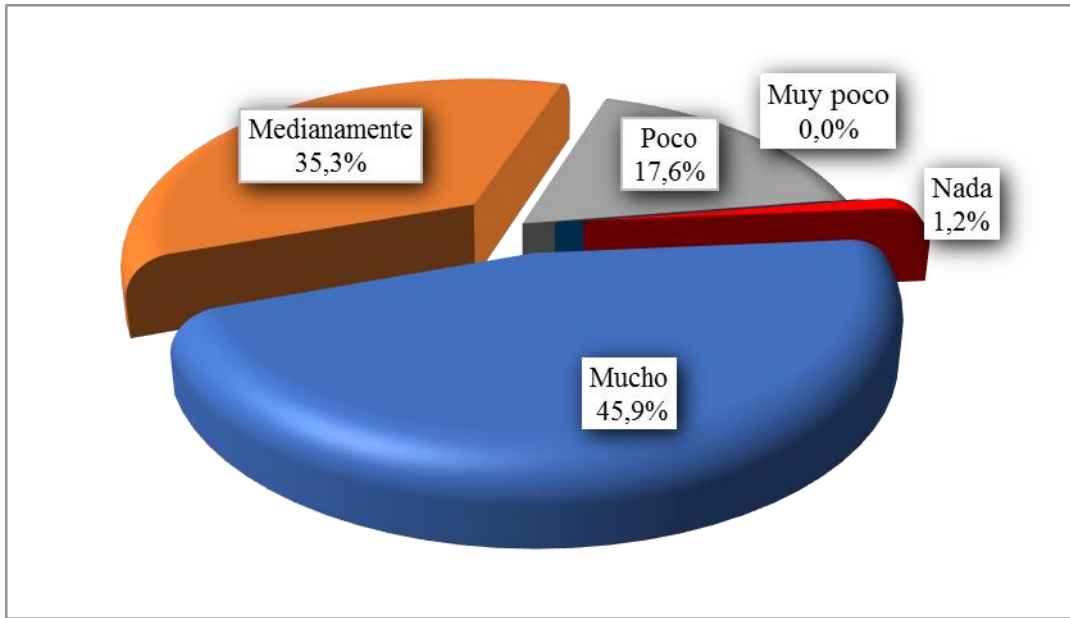


Figura 28. Mejora en el nivel de conocimientos de programación
 Fuente: elaboración propia

P14. Califique su satisfacción general respecto al curso de Programación I.

La Figura 29 expone la satisfacción general respecto al curso de Programación I, donde se evidencia que el 68,2 % se encuentra muy satisfecho o satisfecho, y un 31,8 % indiferente, insatisfecho o muy insatisfecho.

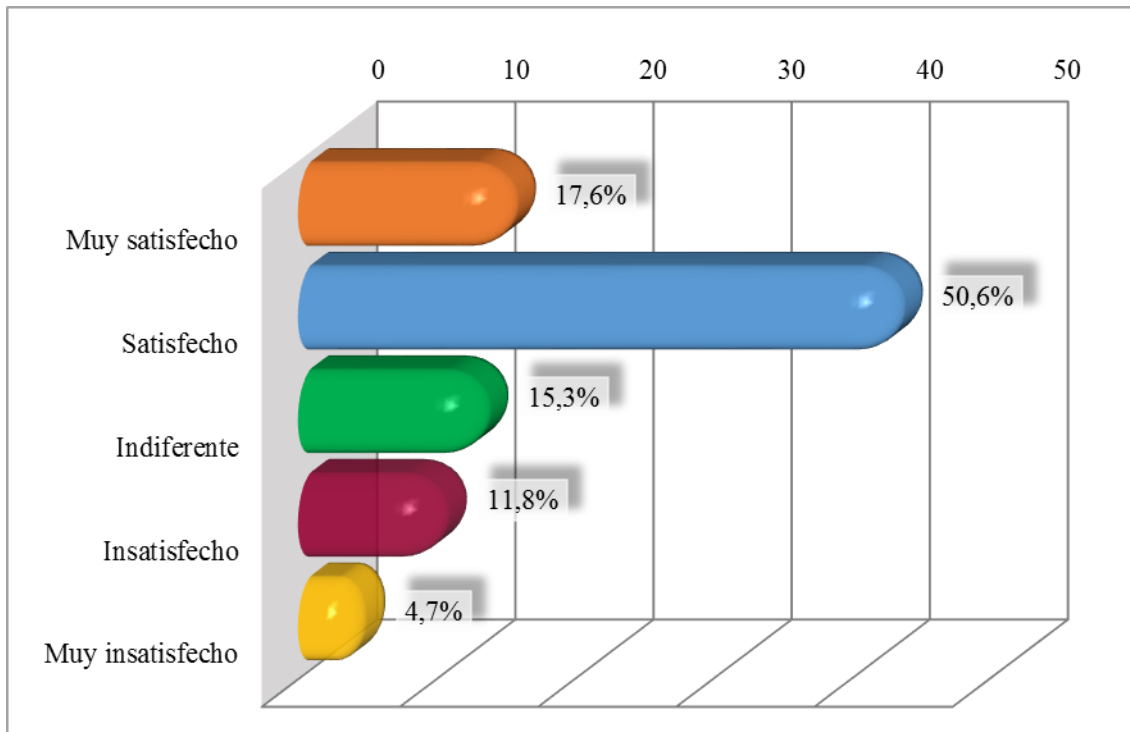


Figura 29. Satisfacción general respecto al curso de Programación I
Fuente: elaboración propia

A modo de resumen de los problemas identificados en el objetivo 1, el autor del trabajo expone que a partir del estudio cuantitativo:

- Existen problemas en el proceso de enseñanza-aprendizaje de Programación I, debido al alto porcentaje de estudiantes que suspendieron y muchos de ellos no vuelven a tomar la asignatura. De los alumnos que aprobaron, el 29,0 % no tomaron Programación II.
- Las variables geográficas y demográficas «género», «provincia de procedencia» y «tipo de colegio» no influyeron en que el estudiante apruebe o no Programación I.
- La variable con mayor influencia en el aprendizaje de Programación I, que decide si los estudiantes aprueban o no la asignatura Programación I, es el docente que les imparte la asignatura, ya que estos en ocasiones no utilizan las herramientas adecuadas y no aplican la metodología correctamente. Además, no es equitativo el nivel de complejidad de los exámenes elaborados por los docentes, ni el nivel de exigencia.

A partir de las encuestas se puede concluir que:

- En los resultados del análisis realizado a la encuesta se obtuvo que una minoría de alumnos (17,6 %) están muy satisfechos con la metodología tradicional aplicada en los cursos, viéndose la necesidad de proponer estrategias que faciliten el aprendizaje de Programación I. Además, los profesores deben cumplir con el sílabo de la asignatura y el mismo grado de exigencia con todos los estudiantes.
- La motivación en los estudiantes se encuentra muy baja y este es un factor importante que se encuentra estrechamente vinculado con el compromiso del aprendizaje de Programación I y las calificaciones que obtienen los alumnos. Este factor debe ser elevado en todos los estudiantes que cursan la asignatura Programación I para evitar que se retiren o suspendan.
- Aprender a programar es sustancial para todo profesional que desee ejercer las carreras de Ingenierías: Informática, Computación Gráfica, Matemática y Diseño Industrial.

En este sentido el autor de la presente investigación, sin pretender solventar todos los problemas detectados, propone el uso de la gamificación en la asignatura Programación I, como apoyo al proceso de aprendizaje de los estudiantes en la modalidad de estudio presencial, en un ambiente *e-learning*. La propuesta incluye la utilización de la plataforma Moodle y herramientas complementarias que permitan incrementar la motivación, mejorar su compromiso con el desarrollo de las tareas autónomas y las calificaciones de los alumnos.

Moodle actualmente cuenta con varios módulos que permiten asignar puntos, insignias, tableros, la retroalimentación en línea para el desarrollo de ejercicios y que los estudiantes vean su calificación una vez subida la solución de los mismos, desbloqueo de contenido, etc. Los docentes también podrán controlar el proceso de aprendizaje de los estudiantes, por lo que el 83,4 % de los docentes está totalmente de acuerdo con que la gamificación en un ambiente *e-learning* puede ayudar a mejorar el proceso de aprendizaje de la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

5.2. Resultados objetivo 2

Objetivo 2. Diseñar e implementar estrategias como apoyo al proceso de aprendizaje presencial, usando herramientas de gamificación y de *e-learning*.

En el presente objetivo se describen las estrategias para fortalecer el proceso de aprendizaje en la asignatura Programación I. Además, se presenta la distribución detallada de la asignatura en el segundo semestre donde se estudian los fundamentos de la programación orientada a objetos (atributos y métodos) en Java, con la aplicación de dinámicas y mecánicas de juego.

Se detalla además el proceso de implementación de la propuesta de estrategias de gamificación en un entorno *e-learning* para la asignatura Programación I en la realización de tareas autónomas, con el uso de las TIC a través de la plataforma de aprendizaje Moodle.

Para conseguir las estrategias de aprendizaje, usando gamificación en Moodle, más adecuadas y la propuesta metodológica óptima, que fue implementada en el semestre abril-septiembre 2016 en la carrera de Ingeniería Matemática, y que es la que se describe en el presente objetivo, se realizaron dos aproximaciones a dicha propuesta metodológica que fueron implantadas y evaluadas en los semestres abril-septiembre 2015 en la carrera de Ingeniería en Computación Gráfica [84], y octubre 2015-marzo 2016 en Ingeniería Matemática. Fruto de esa evaluación y de la experiencia adquirida por el investigador al impartir esos cursos, se propusieron mejoras para la siguiente experiencia que se listan al final de este apartado.

Por efecto de facilidad de lectura, la evaluación de los resultados de cada semestre donde se usaron las estrategias de gamificación en Moodle se presenta en la sección 5.3.1 del presente capítulo.

5.2.1. Justificación de la propuesta en la asignatura Programación I

El aprendizaje de programación es parte esencial en las carreras donde se estudia esta ciencia. Sin embargo, actualmente la enseñanza de la programación se ha transformado en una problemática para los docentes [84], pues el desarrollo de actividades básicas de programación como son el diseño de algoritmos en pseudocódigo y su posterior implementación en un lenguaje de programación, son difíciles de entender para el estudiante.

De acuerdo a lo expuesto por diversos autores como González [85] y Arenas [63], existen factores que inciden en el aprendizaje de Programación I, tales como: los estilos de aprendizaje, la motivación, el modelo de enseñanza-aprendizaje utilizado por los docentes, la falta de conciencia estudiantil, en general el desinterés por sus estudios.

En tanto, los autores Beltrán, Sánchez y Rico [15] mencionan que a los estudiantes les gusta más la práctica que la teoría, pero no les gusta realizar trabajos autónomos, identificando que existe una falta de motivación por el autoaprendizaje. Proponen que se deben plantear métodos y herramientas para mantener e incrementar la motivación por los trabajos y tareas fuera del aula [15], así como hacer del aprendizaje de programación una actividad interesante y agradable [86].

Durante algunos años los docentes han propuesto numerosos enfoques y herramientas para resolver esta problemática [85, 63], sin embargo, hasta este momento no parece existir un enfoque o una solución completamente satisfactoria.

Como aporte innovador al proceso de enseñanza-aprendizaje de programación, el autor de la presente investigación diseña estrategias de aprendizaje para la asignatura Programación I a través de la gamificación en un ambiente *e-learning* en la realización de tareas autónomas, donde su finalidad es influenciar en la conducta psicológica de los estudiantes, con el objetivo de que vean la asignatura de Programación I como un aprendizaje divertido, para que aumenten su predisposición de continuar desarrollando las tareas autónomas y que exista una sana competencia entre ellos [8, 87]. El diseño final cuenta con las experiencias de tres cursos impartidos aplicando estrategias de gamificación en Moodle, que se basan en:

- La programación es una habilidad difícil de adquirir, que se aprende mejor con la práctica y que será más efectiva si la práctica es autodirigida [4].
- El desarrollo de buenas habilidades de programación, por lo general requiere de mucha práctica, que no se puede sostener a menos que los alumnos estén adecuadamente motivados [86].

- El Framework D6 propuesto por Werbach [39] para el diseño de estrategias de gamificación, posibilitando a las instituciones educativas desarrollar el proceso docente de forma motivadora y flexible, y que los estudiantes aprendan de mejor manera.
- El modelo para la gamificación de actividades educativas planteado por González y Mora [6].

5.2.2. Diseño de la asignatura gamificada

El autor de la presente investigación, después de identificar la problemática existente en el aprendizaje de la asignatura Programación I, decide proponer una solución al problema de motivación y compromiso de los estudiantes en la realización de las tareas autónomas mediante el uso de gamificación en Moodle.

El diseño de la experiencia gamificada se basó en los seis pasos planteados en el Framework D6 [39] que se muestran a continuación:

1. Definir los objetivos del curso

Mediante la implementación del aprendizaje de Programación I usando gamificación en un ambiente *e-learning*, en particular gamificando las tareas autónomas, se busca aumentar los conocimientos y capacidades de los estudiantes de Programación I, y por lo tanto mejorar su rendimiento académico.

2. Diseñar las conductas objetivo

Se desea que los estudiantes incrementen la motivación y el compromiso en la realización de las diferentes tareas de programación fuera del aula (programas en lenguaje Java, lecciones, lecturas, etc.), fomentando la competitividad y colaboración entre los alumnos, desarrollando de esta manera el autoaprendizaje. Se busca además que los estudiantes incrementen su participación dentro del aula.

3. Describir a los jugadores-estudiantes

Los jugadores serán los estudiantes de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador. Sus edades están comprendidas entre 18 y 21 años. En general no tienen experiencia previa en programación y tampoco tienen clara su importancia para su futuro profesional, y no les gusta la realización de tareas fuera del aula, lo que hace que su motivación inicial no sea la mejor.

4. Elaborar los bucles de actividad

La experiencia gamificada se ambienta en la Edad Media, periodo repleto de historias, donde lo fantástico es a menudo tenido por real y cuyas leyendas sobreviven aún, además de en el imaginario popular, literatura, películas o tradiciones, en la ambientación de juegos por ordenador de carácter lúdico. Tomando esta metáfora, el docente es el sabio del pueblo, los alumnos de Programación I serán los aprendices de hechiceras o magos de la magia de la programación, los libros y materiales de programación son los libros y recetas de magia, las tareas como ejercicios y pruebas son las actividades de hechicería, los programas Java son hechizos o conjuros, y los foros y chats las tabernas virtuales donde los aprendices podrán interactuar y ayudarse mutuamente.

La misión y reto consiste en viajar por los diferentes niveles de la Programación Mágica (Agua, Tierra, Aire, Fuego y Vida), el estudiante iniciará como aprendiz de mago o hechicera (nivel Agua), para luego ser conjurador (nivel Tierra), invocador (nivel Aire), mago (nivel Fuego) y finalmente archimago (nivel Vida) (Figura 30). Cada nivel tiene mayor complejidad que el anterior, para que de esta manera el reto siga creciendo y motivando al aprendiz a superarlo. Los niveles pueden subdividirse en subniveles, y en cada nivel o subnivel deben realizar distintas actividades como: tareas, ejercicios, lecturas, pruebas, foros, etc., para adquirir progresivamente las competencias de los fundamentos de la programación, hasta llegar al nivel más alto que es ser un archimago. Para pasar de un nivel al siguiente, se necesita de la realización correcta de un número mínimo de tareas que se establece en cada nivel, o en caso contrario una vez transcurrido el tiempo límite para su realización, se pasa de nivel sin acumular los puntos de las tareas.



Figura 30. Niveles de Programación Mágica
Fuente: elaboración propia

El estudiante podrá identificarse con un avatar que escoja en cada uno de los niveles. Se tendrá retroalimentación inmediata en la realización de cada tarea, el estudiante podrá intentar las veces que sean necesarias para cumplir con ellas, mientras el periodo de tiempo esté disponible; se busca con ello que adquiera las competencias de programación necesarias mientras realiza cada intento, además de que pierda el miedo al fracaso y aprenda de los errores. Para poder colaborar entre aprendices y hechiceras, se tendrá una taberna virtual (foros).

Por el cumplimiento de cada tarea autónoma el estudiante obtendrá puntos (llamados puntos de experiencia). Cuando finalice un nivel o subnivel con la realización de las tareas determinadas, ganará insignias que muestran su avance dentro de la plataforma, además de ser una recompensa por su esfuerzo (Figura 31). Podrá también ganar insignias otorgadas por el sabio del pueblo por la realización de tareas excepcionales o por la ayuda que brinde a otros aprendices. Además, podrá conseguir premios virtuales por la consecución de algunas competencias o cuando el sabio lo decida, como libros de magia de la programación o ejemplos de hechizos y conjuros.



Figura 31. Ejemplo de insignias
Fuente: elaboración propia

Se tendrá un tablero donde los alumnos podrán ver los puntos de experiencia ganados frente al resto de aprendices para motivar la competencia (Figura 32), además de que podrán ver gráficamente el avance que tienen y lo que les falta por cumplir para terminar un nivel o subnivel (Figura 33).

Posición	Nombre completo	Puntos
1	 Pablo Eduardo	629.0
2	 Stalin Fabricio	621.0
3	 Jose David	619.3
4	 Luis Fernando	615.8
5	 Dennis Roberto	615.0
6	 Melani Fernanda	613.5
7	 BRENDA IRENE	612.4
8	 Danny Alexander	611.8
9	 Daniel Alexander	608.5
10	 César Alexander	606.6

Figura 32. Tablero de puntos de experiencia ganados
Fuente: elaboración propia



Figura 33. Barra de progreso
Fuente: elaboración propia

Los puntos de experiencia que ganen se transformarán en puntos de tareas autónomas para su nota final (50 % de la nota), el 50 % restante corresponde al examen final, donde pondrán en práctica todo lo aprendido. Pasar el curso será consecuencia de haber aprendido a programar.

De acuerdo con el diseño, se identifican las siguientes dinámicas (deseos y necesidades humanas) que motivan el aprendizaje y que se alcanzan al realizar las mecánicas de juego:

- **Las dinámicas:**
 - **Recompensas:** es algo de valor que se obtiene tras la realización de una tarea o la consecución de un objetivo.
 - **Estatus:** sentimiento de reconocimiento por parte de terceros y un sentimiento de popularidad, prestigio, respeto o captación de atención.
 - **Competencia:** una de las maneras para obtener satisfacción en el desempeño de una tarea o acción, se compara el resultado con el de los demás.
 - **Logros:** una buena recompensa es el reconocimiento por haber conseguido un logro.
 - **Expresión e identidad propia:** manera de conseguir autonomía, identidad y originalidad ante terceros.

- **Altruismo y ayuda:** para fomentar las relaciones e interacciones con los demás estudiantes.
- **Retroalimentación:** permite conocer el progreso o el grado de cumplimiento dentro del juego.
- **Mecánicas y componentes:**
 - **Puntos:** se ganan por la realización de las tareas en la plataforma.
 - **Insignias o medallas:** representaciones gráficas de los logros obtenidos en los niveles o subniveles.
 - **Tabla de clasificaciones:** ordena de manera visual a los participantes según los puntos ganados.
 - **Barra de progreso:** muestra visualmente el cumplimiento y progreso de las diferentes tareas.
 - **Niveles:** indicador del grado de progreso en el juego, cuando se supera un nivel o subnivel, se desbloquea el contenido del siguiente nivel o subnivel.
 - **Avatares:** representaciones personalizadas y únicas de los jugadores por medio de una imagen.
 - **Retroalimentación automática:** codificación, compilación, retroalimentación y evaluación automática de programas Java.
 - **Colaboración:** mediante foros.
 - **Misiones y retos:** serie de tareas a llevar a cabo para conseguir un objetivo, por ejemplo, pruebas y ejercicios.
 - **Bienes virtuales, regalos o premios:** bienes tangibles o intangibles que los participantes ganan en el juego.

5. No se olvide de la diversión

Se busca motivación intrínseca cuando el aprendiz termine cada tarea, especialmente los ejercicios de programación (hechizos y conjuros), y vea que va adquiriendo todas las competencias progresivamente y que puede hacer cada vez hechizos más complicados, todo

esto en un ambiente divertido, social y fácil de usar para jóvenes que están al día con la tecnología.

6. Implementar las herramientas apropiadas

La plataforma gamificada será implementada en un ambiente *e-learning*, de código abierto, que sea fácilmente configurable sin tener que incluir código de programación adicional (Moodle), que pueda ser accesible a través de los principales navegadores en computadores personales y dispositivos móviles. Debe poderse usar para el aprendizaje de los principales lenguajes de programación, en particular, de Java.

Se busca que puedan extenderse la metodología y plataforma al aprendizaje de otras materias en los diferentes niveles educativos.

5.2.2.1. Objetivos y resultados de aprendizaje

El estudiante construye programas básicos y sencillos que se presentan comúnmente en el proceso de aprendizaje de programación en lenguaje Java, utilizando las diferentes técnicas algorítmicas de la programación estructurada y de la programación orientada a objetos (atributos, métodos) [79].

5.2.2.2. Competencias

Competencia específica:

El estudiante, al aprobar la asignatura, estará en capacidad de analizar y diseñar algoritmos básicos y sencillos utilizando los fundamentos y técnicas algorítmicas de la programación estructurada, y construir pequeños programas usando un lenguaje de programación, trabajando en forma responsable y colaborativa [79].

Competencias transversales:

- Capacidad de abstracción, análisis y síntesis sobre problemas específicos.
- Capacidad de aplicar los conocimientos en la práctica.

- Capacidad de investigación.
- Capacidad de aprender y actualizarse permanentemente.
- Habilidad para trabajar en forma autónoma.
- Se expresa con propiedad dentro del ámbito de la asignatura.
- Describe contenidos de la asignatura.

5.2.2.3. Contenido y distribución del curso

Luego de exponer las estrategias de gamificación diseñadas para el aprendizaje de Programación I, en la Tabla 10 se representa la distribución general de la asignatura gamificada para el segundo semestre, donde se estudia la unidad Fundamentos de Java [79]. En la tabla se muestra cada capítulo de Fundamentos de Java de la asignatura Programación I, y su correspondiente nivel y avatar en la Programación Mágica I, así como los puntos de experiencia que se pueden ganar en cada nivel.

Tabla 10. Capítulos, niveles, avatar y puntuación de la unidad Fundamentos de Java

Capítulos de la unidad Fundamentos de Java	Nivel	Avatar	Puntos
Introducción a la Programación Orientada a Objetos (POO)	Agua	Aprendiz	120
Fundamentos de Java (estructuras secuenciales, estructuras selectivas: <i>if-else</i> , <i>switch</i> , estructuras repetitivas: <i>for</i> , <i>while</i> , <i>do-while</i>)	Tierra	Conjurador	420
Arreglos y matrices	Aire	Invocador	220
Métodos	Fuego	Mago	320
	Vida	Archimago	-

Fuente: elaboración propia

En la Tabla 11 se detallan los subcapítulos de Fundamentos de Java y su respectivo subnivel en la unidad gamificada, además de la distribución del tiempo en semanas. El semestre tiene una duración aproximada de 8 semanas, más los exámenes. En cada semana se imparten tres clases de 2 horas cada una.

Los aprendices de la magia de la programación a medida que desarrollen las actividades acumulan puntos, ganan insignias y regalos virtuales, y se les activa el próximo nivel o subnivel de acuerdo a las estrategias de gamificación establecidas. Esto sucede consecutivamente con cada uno de los niveles o subniveles hasta vencerlos todos. Cada nivel

tiene mayor complejidad que el anterior. Una vez terminado el tiempo definido para cada nivel o subnivel se activará el siguiente (Tabla 11); con esto los aprendices que no vencieron todas las actividades definidas, pueden acceder al nuevo contenido, para avanzar con el curso presencial. Es importante esclarecer que una vez activo un contenido o material este no se desactiva, pero las tareas sí, tales como: ejercicios y exámenes, cuando vence el tiempo definido para ello; es decir, dejarán de ganar esos puntos los aprendices.

En la primera semana de cada subnivel se imparte teoría y práctica, luego solo se realizan actividades prácticas. Se tienen dos o tres semanas para aplicar los exámenes finales de la asignatura y entregar las notas en la semana nueve o diez; luego en la semana once se realiza el examen de recuperación a los estudiantes que no superan los 27,5 puntos en los dos hemisemestres del semestre.

Tabla 11. Distribución de la asignatura gamificada

Capítulo	Subcapítulo	Nivel	Subnivel	Semana								Examen		Examen recuperación			
				1	2	3	4	5	6	7	8	9	10	11			
Introducción a la POO		Agua		1													
Fundamentos de Java	Estructuras secuenciales	Tierra	Secuenciales	1	2												
	Estructura selectiva: <i>if-else</i>		<i>if-else</i>			3	4										
	Estructura selectiva: <i>switch</i>		<i>switch</i>			3	4										
	Estructura repetitiva: <i>for</i>		<i>for</i>				4	5	6								
	Estructura repetitiva: <i>while</i>		<i>while</i>				4	5	6								
	Estructura repetitiva: <i>do-while</i>		<i>do-while</i>				4	5	6								
Arreglos y matrices	Arreglos	Aire	Arreglos						6	7	8						
	Matrices		Matrices						6	7	8						
Métodos	Métodos	Fuego	Métodos-hechizos integradores							7	8	9					
		Vida											9	10	11		

Fuente: elaboración propia

5.2.2.4. Estrategias para gamificar las tareas autónomas en cada capítulo

En este epígrafe se muestran los niveles, subniveles y sus estrategias de gamificación para la realización de tareas autónomas de la unidad Fundamentos de Java de la asignatura Programación I, donde se describirá cada una de ellas teniendo en cuenta los capítulos que conforman la unidad. Es importante tomar en cuenta que la complejidad en cada nivel aumenta, por lo que se tendrán menos tareas autónomas de hechizos de programación (ejercicios en Java) pero con mayor complejidad y puntuación mientras se avanza en los niveles.

Recordemos que los libros y materiales de programación son los libros y recetas de magia, los retos de hechicería son pruebas teóricas o prácticas, los hechizos son ejercicios de programación, y las tabernas virtuales son foros.

Capítulo 1: Introducción a la Programación Orientada a Objetos (POO) – Nivel Agua

A continuación, se muestra en la Figura 34 el primer capítulo de la unidad Fundamentos de Java de la asignatura Programación I llamado *Introducción a la POO* que pertenece al nivel Agua y que para superarlo los estudiantes deben vencer las cuatro tareas autónomas para adquirir los 120 puntos de experiencia. La principal tarea es el Reto de hechicería – Examen de Fundamentos de la POO, donde se pone a prueba lo aprendido (80 puntos). La actividad de Taberna virtual – Foro de Fundamentos de la POO (foro para comentar sobre clases y objetos) tiene un valor de 20 puntos, que para ganarlos el aprendiz debe añadir un nuevo tema de discusión y comentar otros cinco. Existe otra actividad de 20 puntos que es actualizar el avatar del nivel. Cada tarea se habilita de acuerdo a las actividades que se han realizado previamente y que están definidas para el nivel. Es en este nivel donde se adquiere la insignia *Aprendiz de mago* que es ganada al inscribirse en el curso. Si el alumno realiza todas las actividades o tareas ganará un premio virtual adicional para aumentar su motivación.


Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir tareas	Insignia hechizos	Gana insignia hechizos al cumplir tareas	XP de nivel	XP de tarea	Premio virtual
Introducción a la POO		Agua		1	Crea tu avatar de Aprendiz de mago	Al inscribirse	Aprendiz de mago		Al inscribirse			120	20	-
				2	Lectura de fundamentos de la POO	Al inscribirse							0	
				3	Taberna virtual – Foro Fundamentos de la POO	2							20	-
				4	Reto de hechicería – Examen Fundamentos de la POO	3							80	Sí

Figura 34. Estrategias de gamificación aplicadas en el nivel Agua – Introducción a la POO
Fuente: elaboración propia

Capítulo 2: Fundamentos de Java – Nivel Tierra

En las Figuras 35, 36 y 37 se muestra el segundo capítulo de la unidad Fundamentos de Java de la asignatura Programación I llamado *Fundamentos de Java* que corresponde al nivel Tierra. Es el nivel más extenso, por eso cuenta con varios subniveles o subcapítulos: Estructuras secuenciales o condicionales (Figura 35), Estructuras selectivas (*if-else* y *switch*) (Figura 36) y Estructuras repetitivas (*for*, *while* y *do-while*) (Figura 37). Tiene 43 tareas autónomas entre lecturas, hechizos de programación, y actualización del avatar para que el aprendiz se identifique en el nivel; el cumplimiento de todas es de 420 puntos de experiencia. Se tiene la mayor cantidad de hechizos de programación (40) de todos los niveles, por ser el primero donde se realizan ejercicios prácticos de Java, en cada uno de ellos se ganan 10 puntos de experiencia. También se obtienen insignias de hechizos al cumplir todas las tareas o actividades determinadas en cada subnivel, en este caso serían seis insignias, la insignia del nivel (*Conjurador*) se gana después de cumplir con las tareas 2, 3 y 4 (Figura 35); es decir, cuando se vence el nivel Agua.

Es importante poder leer las matrices donde se describen las estrategias de gamificación aplicadas en cada nivel, puesto que todos tienen la misma estructura. Como ejemplo, en la Figura 35 se puede observar que se habilitan las tareas 5, 6 y 7 al iniciar el capítulo en las clases presenciales o cuando se cumple la tarea 4 del nivel Agua. El estudiante ganará 20 puntos de experiencia al realizar la actividad 5 (Actualiza tu avatar de conjurador), que le identifica en este nivel. Se hará acreedor de la insignia de nivel Conjurador al terminar correctamente las tareas 2, 3 y 4. Las tareas de hechizos de programación 9 a 18 que corresponden al subnivel Secuenciales, se habilitan al terminar la actividad 8, y esta al cumplir la 7, y tienen un puntaje de 10 cada una. Para ganar la insignia de hechizos que corresponde a este subnivel, el aprendiz deberá cumplir con todos los hechizos del mismo, en este caso, las actividades: 9, 10, 11, 12, 13, 14, 15, 16, 17 y 18.

Para habilitar el siguiente subnivel *if-else* se necesita terminar 8 de las 10 actividades del subnivel anterior, o sea, las tareas 9, 10, 11, 12, 13, 14, 15, 16 (Figura 36), o que las clases del capítulo Estructura selectiva *if-else* inicien.

El aprendiz de mago de la programación ganará además premios virtuales como libros de hechicería o ejemplos de conjuros (código Java) luego de realizar los laboratorios virtuales (hechizos) 20 a 27, es decir, al vencer el subnivel *if-else* (Figura 36). Por último, si el docente o sabio del pueblo identifica un comportamiento excepcional en alguno de los alumnos, puede entregar insignias manuales, como justo premio a su rendimiento y para fomentar aún más la motivación (Figura 39).

- Estructuras secuenciales





Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual	
Fundamentos Java	Estructuras secuenciales	Tierra				Inicia el capítulo en el curso o se cumple 4	Conjurador		2, 3, 4			420		-	
				5	Actualiza tu avatar de Conjurador	Inicia el capítulo en el curso o se cumple 4							20	-	
				6	Lectura de recursos iniciales antes de hacer los primeros hechizos	Inicia el capítulo en el curso o se cumple 4							0	-	
				7	Lectura de los Fundamentos de Java	Inicia el capítulo en el curso o se cumple 4							0	-	
				8	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	7								-	
				9	Secuenciales: Hechizo 1	8								10	-
				10	Secuenciales: Hechizo 2	8								10	-
	11	Secuenciales: Hechizo 3	8		9, 10, 11, 12, 13, 14, 15, 16, 17, 18	10	-								
	12	Secuenciales: Hechizo 4	8			10	-								
	13	Secuenciales: Hechizo 5	8			10	-								
	14	Secuenciales: Hechizo 6	8			10	-								
	15	Secuenciales: Hechizo 7	8			10	-								
	16	Secuenciales: Hechizo 8	8			10	-								
	17	Secuenciales: Hechizo 9	8			10	-								
	18	Secuenciales: Hechizo 10	8			10	-								

Figura 35. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras secuenciales

Fuente: elaboración propia




- Estructuras selectivas o condicionales: *if-else* y *switch*

Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual
Fundamentos Java	Estructura selectiva: if-else	Tierra	if-else	19	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 9, 10, 11, 12, 13, 14, 15, 16					20, 21, 22, 23, 24, 25, 26, 27, 28, 29			
				20	Condicionales if-else: Hechizo 1	19			10					
				21	Condicionales if-else: Hechizo 2	19			10					
				22	Condicionales if-else: Hechizo 3	19			10					
				23	Condicionales if-else: Hechizo 4	19			10			-		
				24	Condicionales if-else: Hechizo 5	19			10			-		
				25	Condicionales if-else: Hechizo 6	19			10			-		
				26	Condicionales if-else: Hechizo 7	19			10			-		
				27	Condicionales if-else: Hechizo 8	19			10			-		
				28	Condicionales if-else: Hechizo 9	19			10			-		
	29	Condicionales if-else: Hechizo 10	19			10	-							
	Estructura selectiva: switch	Tierra	switch	30	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 20, 21, 22, 23, 24, 25, 26, 27					31, 32, 33, 34, 35			Sí
				31	Condicionales switch: Hechizo 1	30			10			-		
32				Condicionales switch: Hechizo 2	30			10	-					

Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual
				33	Condicionales switch: Hechizo 3	30							10	-
				34	Condicionales switch: Hechizo 4	30							10	-
				35	Condicionales switch: Hechizo 5	30							10	-

Figura 36. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras selectivas
Fuente: elaboración propia

- Estructuras repetitivas: *for*, *while* y *do-while*

Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual		
Fundamentos Java	Estructura repetitiva: for	Tierra	for	36	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 31, 32, 33, 34					37, 38, 39, 40, 41					
				37	Repetitivas for: Hechizo 1	36			10							
				38	Repetitivas for: Hechizo 2	36			10							
				39	Repetitivas for: Hechizo 3	36			10							
				40	Repetitivas for: Hechizo 4	36			10			-				
	41		Repetitivas for: Hechizo 5	36			10	-								
	Estructura repetitiva: while		while	42	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 37, 38, 39, 40					43, 44, 45, 46, 47					-
				43	Repetitivas while: Hechizo 1	42			10			-				
				44	Repetitivas while: Hechizo 2	42			10			-				
				45	Repetitivas while: Hechizo 3	42			10			-				
				46	Repetitivas while: Hechizo 4	42			10			-				
	47		Repetitivas while: Hechizo 5	42			10	-								
	Estructura repetitiva: do-while		do-while	48	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 43, 44, 45, 46					49, 50, 51, 52, 53					Sí
				49	Repetitivas do-while: Hechizo 1	48			10			-				
				50	Repetitivas do-while: Hechizo 2	48			10			-				
				51	Repetitivas do-while: Hechizo 3	48			10			-				

Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual
				52	Repetitivas do-while: Hechizo 4	48							10	-
				53	Repetitivas do-while: Hechizo 5	48							10	-

Figura 37. Estrategias de gamificación aplicadas en el nivel Tierra – Estructuras repetitivas

Fuente: elaboración propia

Capítulo 3: Arreglos y matrices – Nivel Aire

En la Figura 38, se encuentra el tercer capítulo de la unidad Fundamentos de Java de la asignatura Programación I llamado *Arreglos y matrices* que hace correspondencia al nivel Aire. Se divide en dos subniveles o subcapítulos: Arreglos y Matrices. Consta de 13 tareas autónomas como actualizar el avatar, lecturas y hechizos de la magia de la programación; todas suman 220 puntos de experiencia. Es un nivel más complejo que los anteriores, por esto, se tienen solo cinco hechizos para Arreglos y cinco para Matrices, cada uno valorado en 20 puntos de experiencia. El presente nivel cuenta con una insignia de nivel (*Invocador*), que se obtiene al vencer el nivel anterior (tareas 49, 50, 51 y 52), y dos insignias de hechizos, una para el subnivel Arreglos y que se gana al terminar las tareas 58, 59, 60, 61 y 62; y otra para el subnivel Matrices al terminar todas las actividades de hechicería (64, 65, 66, 67 y 68). Se puede obtener también premios virtuales al realizar la tarea 56 y al terminar el subnivel Arreglos (actividades 58, 59, 60 y 61).






Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual
Arreglos y matrices		Aire			Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 49, 50, 51, 52	Invocador		49, 50, 51, 52			220		
				54	Actualiza tu avatar de Invocador	Inicia el capítulo en el curso o se cumple 49, 50, 51, 52							20	
				55	Lectura Fundamentos de Java	Inicia el capítulo en el curso o se cumple 49, 50, 51, 52							0	
				56	Lectura Arreglos y matrices	Inicia el capítulo en el curso o se cumple 49, 50, 51, 52							0	
			Arreglos	57	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	56							0	Sí
				58	Arreglos: Hechizo 1	57							20	-
				59	Arreglos: Hechizo 2	57							20	-
				60	Arreglos: Hechizo 3	57							20	-
				61	Arreglos: Hechizo 4	57							20	-
				62	Arreglos: Hechizo 5	57							20	-
			Matrices	63	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación	Inicia el capítulo en el curso o se cumple 58, 59, 60, 61								Sí
				64	Matrices: Hechizo 1	63							20	-
				65	Matrices: Hechizo 2	63							20	-
				66	Matrices: Hechizo 3	63							20	-
				67	Matrices: Hechizo 4	63							20	-
				68	Matrices: Hechizo 5	63							20	-

Figura 38. Estrategias de gamificación aplicadas en el nivel Aire – Arreglos y matrices. Fuente: elaboración propia

Capítulo 4: Métodos – Nivel Fuego

En la Figura 39 se representa el cuarto capítulo de la unidad Fundamentos de Java de la asignatura Programación I llamado *Métodos*, perteneciente al nivel Fuego donde se adquiere la insignia de nivel *Mago* al terminar el nivel Aire (tareas 64, 65, 66 y 67), y una sola insignia de hechizos para el subnivel Métodos-Hechizos integradores al cumplir las actividades 73, 74, 75 y 76. Se adquieren 320 puntos de experiencia con el desarrollo de las ocho tareas autónomas configuradas como actualizar el avatar (20 puntos), lecturas, hechizos de programación (150 puntos) y una prueba final-reto de recetas de hechizos (100 puntos). Es el nivel más complejo donde se hacen hechizos integradores de los niveles anteriores y se pone a prueba todo lo aprendido en el curso gamificado. También se ganan 50 puntos por la asistencia al curso. El último premio virtual se adquiere al cumplir con la actividad 71. Como es el último capítulo de la asignatura, al terminar las tareas 73, 74 y 76 se gana la mayor insignia de la magia de la programación, *Archimago*, que confirma haber llegado al máximo nivel, Vida.

Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual						
Métodos		Fuego				Inicia el capítulo en el curso o se cumple 63, 64, 65, 66	Mago		64, 65, 66, 67			320								
				69	Actualiza tu avatar de Mago	Inicia el capítulo en el curso o se cumple 63, 64, 65, 66							20							
				70	Lectura Fundamentos de Java	Inicia el capítulo en el curso o se cumple 63, 64, 65, 66							0							
				71	Lectura Métodos	Inicia el capítulo en el curso o se cumple 63, 64, 65, 66							0							
				Métodos-Hechizos integradores	72	Marcar con un <i>check</i> para iniciar los Laboratorios virtuales de programación							71						Sí	
			73		Métodos: Hechizo 1	72												50	-	
			74		Métodos: Hechizo 2	72												50	-	
			75		Métodos: Hechizo 3	72												73, 74, 75, 76	50	-
			76		Retos de recetas de hechizos – Fundamento de Java	(73, 74) o 75													100	-
			77		Asistencia a la escuela de magia	Inscribirse en el curso													50	-
		Vida	Matrices	63		73, 74, 76	Archimago		73, 74, 76		73, 74, 76			-						
							Muy bien		Otorgada por el sabio al trabajo excepcional		Otorgada por el sabio al trabajo excepcional			-						


Capítulo	Subcapítulo	Nivel	Subnivel	Núm. tarea	Tareas autónomas	Se habilita al cumplir las tareas	Avatar nivel	Insignia nivel	Gana insignia de nivel al cumplir las tareas	Insignia hechizos	Gana insignia hechizos al cumplir las tareas	XP nivel	XP tarea	Premio virtual
							Buen trabajo, hechizo complejo		Otorgada por el sabio al trabajo excepcional		Otorgada por el sabio al trabajo excepcional			-
											Puntos de experiencia total	1080		
											Nota final tareas	Puntos de experiencia total /100 (con un máximo de 10 puntos)		

Figura 39. Estrategias de gamificación aplicadas en el nivel Fuego – Métodos
Fuente: elaboración propia

5.2.2.5. Criterios e instrumentos de evaluación

En la Figura 39, se tiene que el total de puntos de experiencia que un aprendiz puede ganar es de 1080. Para la nota final de las tareas autónomas, los puntos ganados al actualizar el avatar en cada nivel (20 puntos por cada nivel, con un total de 80) son considerados como adicionales, es decir, sirven para que los estudiantes tengan la opción de perder hasta 80 puntos. Por lo tanto, el puntaje máximo de tareas autónomas es de 1000 puntos.

Para transformar los puntos de experiencia en la nota de las tareas autónomas (con un máximo de 10 puntos) del segundo hemisemestre, se usa el siguiente algoritmo:

```
notaTareasAutonomas = puntosExperiencia / 100
```

```
Si puntosExperiencia >= 10,8 entonces
```

```
    notaTareasAutonomas = 10
```

```
FinSi
```

La nota final del hemisemestre está compuesta por dos elementos: nota de las tareas autónomas (10 puntos), y el examen final (10 puntos) que es una evaluación de ejercicios prácticos en Java que evidencien lo aprendido en todo el hemisemestre.

5.2.3. Implementación de gamificación en la plataforma Moodle

Las estrategias de gamificación diseñadas para la realización de tareas autónomas de la unidad Fundamentos de Java de la asignatura Programación I, y que están descritas en el epígrafe 5.2.2 del presente capítulo, se implementaron en el entorno virtual Moodle. Los requisitos de *hardware* y *software* están detallados en el Anexo 5. De esta manera se creó la plataforma Moodle gamificada que se encuentra publicada en el sitio:

<www.aprendizajegamificado.com> [88].

Moodle gamificado permite a los estudiantes aprender los contenidos de la unidad Fundamentos de Java de la asignatura Programación I y la realización de tareas autónomas de forma divertida y motivadora. Programación I en Moodle gamificado [84] se llama

Programación Mágica I, que transforma a los alumnos de programación en aprendices de hechiceras o magos de la época medieval, y su misión será adquirir el nivel más alto de la magia de la programación, que es llegar a ser archimago. Al realizar todas las tareas el estudiante irá adquiriendo los poderes mágicos de la programación e irá superando los niveles de la magia desde aprendiz, para luego realizar conjuros básicos, después aprenderá a invocar conjuros avanzados, para más tarde ser promovido a mago, y si supera este último nivel llegará a ser archimago [88].

A continuación, se describe cómo se encuentra implementado Moodle gamificado, sus partes o características principales, las tareas autónomas configuradas, y cada uno de los niveles o capítulos de Fundamentos de Programación Java. La evaluación de los resultados académicos al implementar las estrategias de aprendizaje usando gamificación en Moodle gamificado en los estudiantes de Programación I, se realizará en el objetivo 3 del presente capítulo.

5.2.3.1. Estructura de Moodle gamificado

La Figura 40 muestra la portada principal de la plataforma Moodle gamificada antes de autenticarse; en esta pantalla se describe una síntesis de Programación Mágica I en general y de los cursos disponibles.

Programación Mágica

Serás una Hechicera o un Mago de la época medieval y tu misión será adquirir el nivel más alto de la Magia de la Programación, que es llegar a ser Archimago.

Realizando todas las tareas irás adquiriendo los poderes mágicos de la Programación, e irás subiendo de niveles, desde Aprendiz, para luego realizar Conjuros básicos, después aprenderás a Invocar Conjuros avanzados, para luego ser promovido a Mago, si superas este nivel serás un Archimago de la Programación.

Te divertirás aprendiendo los Fundamentos de Programación Java.



★ PROGRAMACIÓN MÁGICA ING. MATEMÁTICA 2016-2016



En el curso de la Magia de la Programación, serás una Hechicera o un Mago de la época medieval y tu misión será llegar a ser Archimago. Aprenderás en forma divertida los Fundamentos de Programación Java.

Programación Mágica

★

★

Figura 40. Portada de la plataforma Moodle gamificada antes de autenticarse

Fuente: Moodle gamificado [88]

La Figura 41 muestra la interfaz para un estudiante autenticado en Moodle gamificado. Se divide en tres columnas, en la primera están los bloques de materiales y actividades del curso, y en las restantes columnas se encuentran los bloques que permiten la gamificación y el resto de bloques generales. Se describen a continuación los principales módulos usados en la plataforma.



Figura 41. Cursos implementados en la plataforma Moodle
Fuente: elaboración propia

Bloques de gamificación

- **Barra de progreso:** muestra de manera fácil de interpretar las actividades o recursos programados en cada nivel o subnivel, su estado actual, avance, progreso y fecha esperada de cumplimiento (Figura 42). Esto motiva al estudiante a seguir avanzando en el curso. Se usa el *plugin* Progress Bar adicional a Moodle [89].



Figura 42. Barra de progreso en Moodle gamificado
Fuente: Moodle gamificado [88]

Los docentes pueden también acceder a ver el progreso de las tareas de todos los aprendices para cada nivel o subnivel, optimizando el tiempo de revisión de las actividades (Figura 43).

Vista general de alumnos

Grupos visibles: Todos los participantes | Rol: Estudiante

Página: 1 2 (Siguiente)

	Nombre / Apellido(s)	Última en línea	Barra de Progreso	Progreso
	Fernanda Morales	miércoles, 11 de enero de 2017, 23:24		100%
	Alex Dario Anilema Molina	lunes, 15 de agosto de 2016, 17:52		100%
	Alexander Vinicio Yupangui Segovia	martes, 20 de septiembre de 2016, 21:47		100%
	ALISSON YADIRA QUIMBITA TAXI	sábado, 26 de noviembre de 2016, 09:36		100%
	Andy Mauricio Molina Coronel	martes, 22 de noviembre de 2016, 22:15		100%

Figura 43. Barra de progreso en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Insignias recientes:** las insignias (*badges*) son una buena forma para celebrar un logro y mostrar el progreso. Las insignias pueden otorgarse basándose en una variedad de criterios elegidos por el docente. Este bloque despliega las insignias que ha ganado el aprendiz (Figura 44).



Figura 44. Insignias recientes en Moodle gamificado
Fuente: Moodle gamificado [88]

También se puede acceder a las insignias a través del menú de usuario como se ve en la Figura 45.



Figura 45. Insignias del usuario en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Ranking o tablero de clasificación:** clasifica a los aprendices de mayor a menor de acuerdo a su puntuación según las actividades que han completado (Figura 46). Fomenta la sana competencia entre aprendices. Se usa el módulo Ranking Block [90] adicional a Moodle.

Posición	Nombre completo	Puntos
1	Pablo Eduardo	629.0
2	Stalin Fabricio	621.0
3	Jose David	619.3
4	Luis Fernando	615.8
5	Dennis Roberto	615.0
6	Melani Fernanda	613.5
7	BRENDA IRENE	612.4
8	Danny (Ντόνη) Alexander	611.8
9	Daniel Alexander	608.5
10	César Alexander	606.6

Figura 46. Ranking en Moodle gamificado
Fuente: Moodle gamificado [88]

Bloques generales

- **Últimas noticias:** se muestran las publicaciones más recientes hechas en el Foro de novedades, junto con un enlace a las noticias anteriores archivadas.
- **Gente:** este bloque contiene el enlace a un listado de todos los participantes del curso (Figura 47) donde se puede enviar mensajes y correos a todos los participantes del curso.

Participantes

Mis cursos

Programación Mágica I 2016-2016. Ing. Matemática ▾

Grupos

separados:
Thomas Edison

Mostrar usuarios que han estado inactivos durante más de

Lista de usuarios

Resumen ▾

Seleccionar período ▾

Rol actual

Todos los participantes ▾

Todos los participantes:9

Nombre : **Todos** A B C D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z

Apellido(s) : **Todos** A B C D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z

Imagen del usuario	Nombre / Apellido(s)	Ciudad	País	Último acceso al curso ▾
	Melani Fernanda Nuñez Nuñez	Quito	Ecuador	38 días 7 horas
	Fernanda Morales	Quito	Ecuador	71 días 22 horas
	César Alexander Guayasamín Morales	Quito	Ecuador	122 días 23 horas
	Anthony David Chávez Cárdenas	Quito	Ecuador	130 días 9 horas
	Angie Monserrath Chisaguano	Quito	Ecuador	221 días 3 horas

Figura 47. Listado de participantes del curso

Fuente: Moodle gamificado [88]

A través de este listado se accede al perfil de un usuario en particular como se ve en la Figura 48; entre la información principal que se tiene está el avatar, las insignias ganadas, detalles del usuario y del curso.

Programación Mágica I 2016-2016. Ing. Matemática



César Alexander Guayasamín Morales

Detalles de usuario

▪ [Editar perfil](#)

País

Ecuador

Ciudad

Quito

Insignias

Insignias de Aprendizaje Gamificado:



Sabio. Ing
Matemática 2016-
2016



Muy Bien
Matemática 2016
2016



Buen Trabajo
Matemática 2016-
2016



Archimago
Matemática 2016-
2016

Detalles del curso

Perfiles de curso

▪ [Programación Mágica I 2016-2016. Ing. Matemática](#)

Roles

[Estudiante](#)

Grupo

[Thomas Edison](#)

Misceláneos

- [Perfil completo](#)
- [Entradas del blog](#)
- [Mensajes en foros](#)
- [Discusiones del foro](#)

Reportes

- [My feedback](#)

Actividad de ingresos

Último acceso al curso

[lunes, 21 de noviembre de 2016, 21:39 \(122 días 23 horas\)](#)

Figura 48. Perfil personal de un aprendiz

Fuente: Moodle gamificado [88]

- **Usuarios en línea:** muestra una lista de los usuarios que han ingresado al curso actual. La lista es actualizada regularmente cada cinco minutos por defecto.
- **Buscar foros:** permite buscar dentro de los foros del curso por palabras o frases.
- **Eventos próximos:** despliega los eventos futuros en una lista resumida. Los eventos son generados directamente desde el calendario y/o las fechas finales de actividades,

proporcionando un enlace hacia los detalles completos o directamente hacia la actividad.

- **Actividades:** este bloque enlista y permite la navegación entre las diferentes actividades disponibles en el curso como Asistencias, Exámenes, Foros, Laboratorios de programación virtuales, Recursos y Tareas.
- **Navegación:** contiene un menú de árbol expandible que incluye el Tablero, Páginas del sitio y Cursos. Permite navegar rápidamente entre los diferentes niveles del curso gamificado (Figura 49).

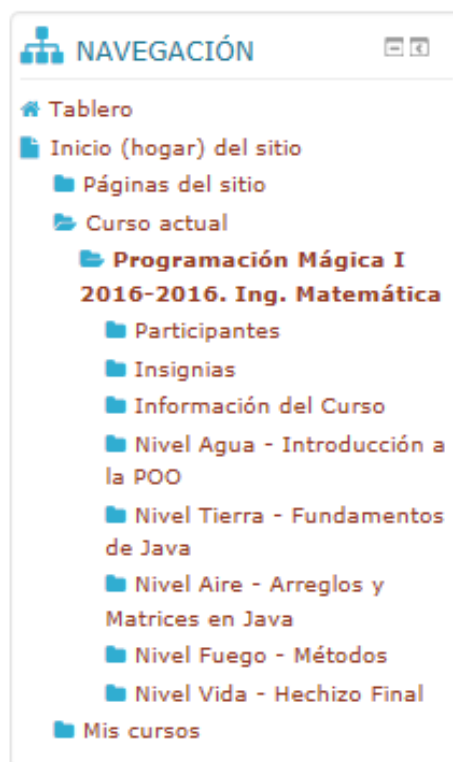


Figura 49. Bloque Navegación en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Administración:** permite al aprendiz visualizar las calificaciones de todas las tareas que ha realizado (Figura 50).

Reporte de usuario - César Alexander Guayasamín

Ítem de calificación	Ponderación calculada	Calificación	Rango	Porcentaje
Programación Mágica I 2016-2016. Ing. Matemática				
Asistencia	-	50.00	0-50	100.00 %
Crea tú Avatar de Aprendiz de Mago de la Programación	-	20.00	0-20	100.00 %
Actualiza tú Avatar de Conjurador	-	20.00	0-20	100.00 %
Actualiza tú Avatar de Invocador	-	20.00	0-20	100.00 %
Actualiza tú Avatar de Mago	-	20.00	0-20	100.00 %
Taberna Virtual - Fundamentos de la POO	-	18.67	0-20	93.33 %
Reto de Hechicería - Fundamentos de la POO	-	80.00	0-80	100.00 %
Reto de Recetas de Hechizos - Fundamentos de Java	-	100.00	0-100	100.00 %
Taberna Virtual de Ayuda -	-	-	No me gusta	-

Figura 50. Calificaciones de un estudiante en Moodle gamificado
Fuente: Moodle gamificado [88]

Los docentes pueden ver y descargar las calificaciones de todos los aprendices (Figura 51). La descarga puede ser en varios formatos, el más utilizado es en hojas de cálculo Excel por la facilidad de realizar cálculos.

Todos los participantes:33/33

Nombre :

Todos A B C D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z

Apellido(s) :

Todos A B C D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z

Apellido(s) ^ Nombre		Dirección Email	Programación Mágica I ...		
			Asistencia	Crea tú Avatar de Aprendiz ...	Actualiza tú Avatar de ...
	DEBORA BELEN AGUIRRE MOLINA	deboraguirre@hotmail.es	43.33	20.00	20.00
	Edison Javier Almachi Llano	edi1996_@hotmail.com	46.67	20.00	20.00
	Kevin David Anchapaxi Moreta	kerukevin@hotmail.com	26.67	20.00	20.00
	Alex Dario Anilema Molina	alexdariolcpunk@gmail.com	43.33	20.00	20.00
	Carlos Andres Asqui Caarte	andrescarl_10@hotmail.com	43.33	20.00	20.00

Figura 51. Calificaciones de los alumnos en Moodle gamificado

Fuente: Moodle gamificado [88]

Principales módulos de actividades y recursos en Moodle gamificado

Cada uno de los niveles o subniveles del curso tiene la siguiente distribución general que se ve en la Figura 52, donde están listadas las tareas autónomas definidas que el aprendiz debe vencer. La realización correcta de una tarea se muestra con un *check* o visto marcado a la derecha.

Las tareas se agrupan en: actualizar el avatar del nivel, hechizos y conjuros, actividades de hechicería, bienes virtuales y tabernas virtuales.



Figura 52. Calificaciones de alumnos en Moodle gamificado.

Fuente: elaboración propia

- **Actualizar el avatar del nivel:** esta tarea permite actualizar el avatar con el cual el aprendiz quiere ser identificado en el nivel que superó dentro de la plataforma (Figura 53). Tiene un valor de 20 puntos de experiencia por cada nivel.

Actualiza tú Avatar de Mago

[Return to: Nivel Fuego - M...](#)

Estimad@ Aprendiz de Hechicera o Mago, ya eres un Mago, actualiza por favor tú Avatar que te identificará durante el nivel Fuego del Curso de Programación Mágica.
Debes actualizar en tú perfil y enviar la imagen el esta tarea.

Puedes usar como ejemplo los siguientes Avatares:

Nivel Tipo de Mago Avatar Mago

Avatar Hechicera

Fuego Mago



Estatus de la entrega

Número de intento	Este es el intento 1 (1 intentos permitidos)
Estatus de la entrega	Enviado para calificar
Estatus de calificación	Calificado
Fecha de entrega	domingo, 14 de agosto de 2016, 23:55
Tiempo restante	La tarea fue enviada 49 días 5 horas antes
Última modificación	domingo, 26 de junio de 2016, 18:40
Envíos de archivo	 9229836.jpg
Comentarios al envío	» Comentarios (0)

Retroalimentación

Calificación	20.00 / 20.00
Calificado en	martes, 19 de julio de 2016, 23:14
Calificado por	 Jefferson Beltrán

Figura 53. Tarea Actualizar avatar en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Hechizos y conjuros:** en esta sección se tiene acceso a recursos y materiales como libros y recetas de hechizos y conjuros de programación que el aprendiz debe leer antes de realizar los ejercicios de programación.

- **Actividades de hechicería - Hechizos:** son ejercicios de programación en Java que el aprendiz debe realizar en cada nivel o subnivel. Para cada hechizo se entrega al estudiante la definición del mismo y un código básico que le guía en cómo debe resolverlo (Figura 54).

Repetitivas do - while: Hechizo 3

Fecha realización: domingo, 24 de julio de 2016, 23:55

Archivos solicitados: SumaDivisibles.java ([Descarga](#))

Tipo de trabajo: Trabajo individual

Calcule la suma de los n primeros número naturales divisibles por un número ingresado.

Archivos solicitados

SumaDivisibles.java

```
1 import java.util.Scanner;
2 public class SumaDivisibles {
3
4     public static void main(String[] args) {
5         //su código aquí
6
7         System.out.println("Ingrese el número de repeticiones: ");
8         //su código aquí
9
10        System.out.println("Divisible: ");
11        //su código aquí
12
13        suma=...
14        System.out.println("Suma: " + suma);
15    }
16
17    public static int sumaDivisibles(int num, int divisible){
18        //su código aquí
19
20    }
21 }
22 }
23 }
```

Figura 54. Tarea Hechizo de programación en Moodle gamificado
Fuente: Moodle gamificado [88]

Se tiene una interfaz para resolver el ejercicio planteado que se ve en la Figura 55. Una vez que el estudiante edita el código y escribe la solución del problema, lo guarda y lo ejecuta para ver su resultado. Si piensa que está correcto lo evalúa y el sistema compara el resultado del código del aprendiz con las pruebas que están configuradas en la plataforma, teniendo una retroalimentación automática en línea de la ejecución de su código. El estudiante puede evaluarlo las veces que sean necesarias hasta que se tenga la ejecución correcta del código. Este módulo es uno de los más importantes en Moodle gamificado pues tener retroalimentación inmediata hace que la motivación aumente y se vuelva a intentar una vez más hasta que se venza el reto. El puntaje de

estas tareas depende del nivel en el que se esté, mientras más se avanza en los niveles, también avanza la dificultad y por lo tanto el puntaje.

El módulo externo de Moodle que se usa para la edición, compilación, ejecución y evaluación de código Java es Virtual Programming Lab (VPL) [59, 91].

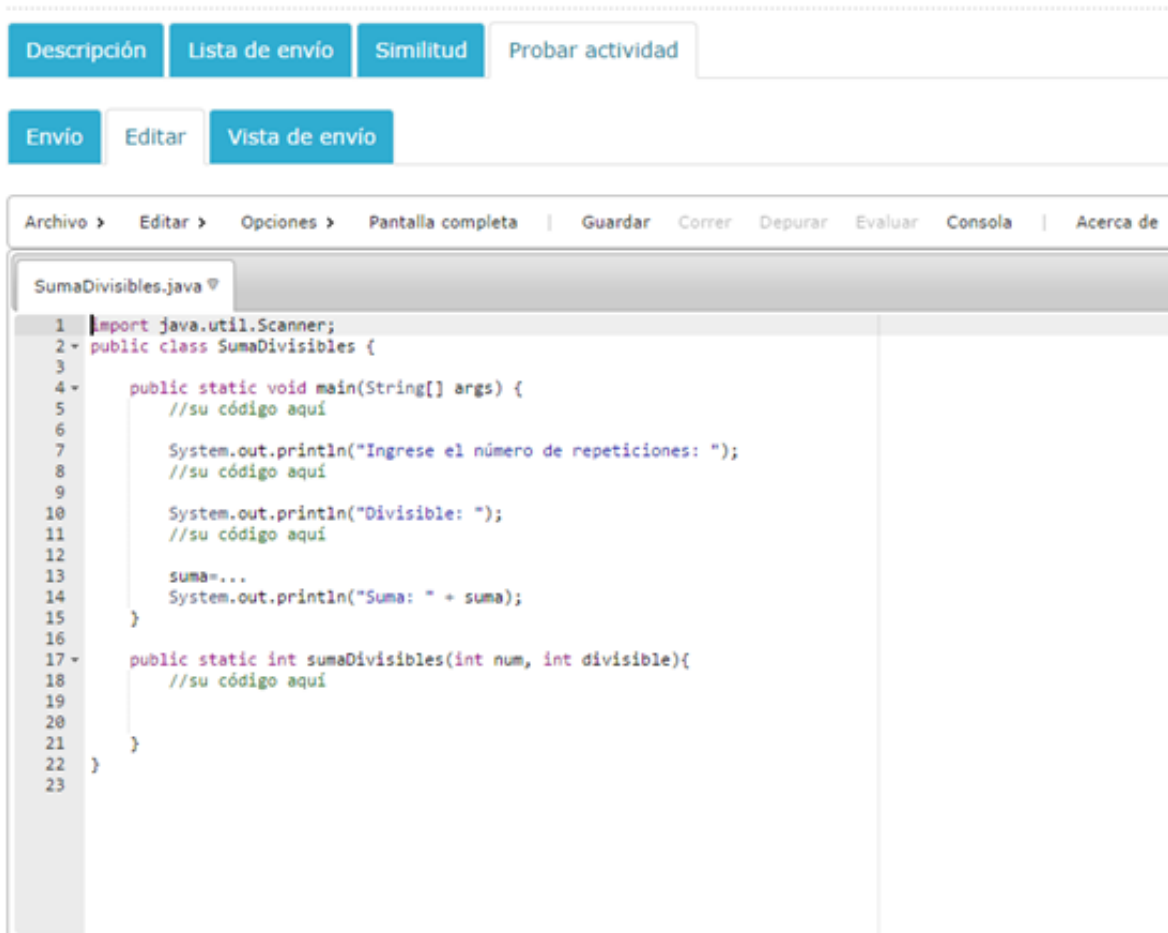


Figura 55. Interfaz para realizar hechizos de programación en Moodle gamificado
Fuente: Moodle gamificado [88]

En la Figura 56 se aprecia el resumen de la ejecución de código de un hechizo de programación, el número de pruebas unitarias pasadas y la calificación.

Calificación

Revisado en domingo, 19 de junio de 2016, 18:26 by Calificación automática

Calificación 10 / 10

Reporte de la evaluación

[-] Summary of tests

```
+-----+
| 4 tests run/ 4 tests passed |
+-----+
```

Enviado en domingo, 19 de junio de 2016, 18:25 (Descarga)

SumaDivisibles.java

```
1 import java.util.Scanner;
2 public class SumaDivisibles {
3
4     public static void main(String[] args)
5     {
6         int numero=0 ,divisb=0, suma=0;
7         Scanner leer = new Scanner(System.in);
8
9         System.out.println("Ingrese el número de repeticiones: ");
10        numero=leer.nextInt();
11        System.out.println("Divisible: ");
12        divisb=leer.nextInt();
13
14        suma=sumaDivisibles(numero, divisb);
15        System.out.println("Suma: " + suma);
16    }
17
18    public static int sumaDivisibles(int num, int divisible)
19    {
20        int i=1 ,suma=0;
21        do
22        {
23            if(i%divisible==0)
24            {
25                suma=suma+i;
26            }
27            i++;
28        }while(i<=num);
29
30        return suma;
31    }
32 }
33
```

Figura 56. Evaluación de Hechizo de programación en Moodle gamificado

Fuente: Moodle gamificado [88]

- **Actividades de hechicería - Reto de hechicería:** son lecciones o pruebas que generalmente están constituidas por una serie de preguntas y varias respuestas posibles. Se puede configurar distintos tipos de respuestas como verdadero/falso, opción múltiple, selección múltiple, emparejamiento, etc. En la Figura 57 se presenta el ejemplo de una pregunta de un reto de hechicería y su evaluación. El estudiante puede realizarla hasta cinco veces, con diferentes preguntas aleatorias; lo importante es el aprendizaje más que la nota final.

Programación Mágica I 2016-2016. Ing. Matemática

Comenzado el	martes, 19 de julio de 2016, 21:39	≡ NAVEGACIÓN DENTRO DEL EXAMEN
Estado	Terminado	
Finalizado en	martes, 19 de julio de 2016, 21:43	1 2 3 4 5 6 7 8 9
Tiempo empleado	4 minutos 53 segundos	
Calificación	100.00 de un total de 100.00	10
Comentario de retroalimentación	¡Excelente, buen trabajo! Sigue así.	Mostrar una página cada vez Finalizar revisión

Pregunta 1

Correcta

Puntúa 10.00 sobre 10.00

🚩 Marcar con bandera la pregunta

Considere el siguiente fragmento de código:

```
1. public class Question{
2. public static void main(String[] args) {
3. byte x = 21;
4. byte y = 13;
5. int z = x*y;
6. System.out.println(z);
7. }
8. }
```

¿Cuál es el resultado de este fragmento de código?

Seleccione una:

- a. Un error de compilación ocurre en la línea 5.
- b. 13
- c. 21
- d. 29
- e. 24 ✓

Su respuesta es correcta.

La respuesta correcta es: 24

Figura 57. Tarea Reto de hechicería en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Premios virtuales:** son premios que el docente entrega en la plataforma a sus aprendices cuando han vencido ciertos subniveles; se busca tener motivación a través de sorpresas que el alumno no esperaba (Figura 58).

Bien, bien, bien!!!, te has consolidado como Mago, eres un Archimago.

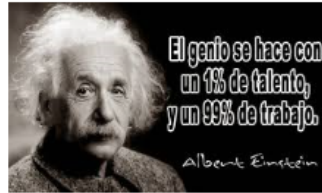
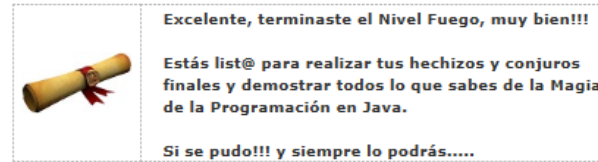


Figura 58. Premios virtuales en Moodle gamificado
Fuente: Moodle gamificado [88]

- **Taberna virtual:** son foros donde los aprendices pueden socializar, interactuar, colaborar y compartir conocimientos. Existen tabernas exclusivas para socializar y otras para el aprendizaje (Figura 59).

Taberna Virtual - Fundamentos de la POO

[Return to: Nivel Agua - In... ↩](#)

Estimado Aprendiz de Mago o Hechicera de la Programación Orientada a Objetos, participa en la **Taberna Virtual (Foro)** indicando ejemplos de qué es una Clase y qué es un Objeto y la diferencia entre ellos.

Cada Aprendiz debe enviar una discusión y responder a 5 para que la tarea esté terminada.

Éxito.


Discusión	Comenzado por	Rélicas	Último mensaje
Clase -Objeto :D	 Katherine Valeria Sisalima Coyago	0	Katherine Valeria Sisalima Coyago vie, 24 de jun de 2016, 10:47
clase - objeto	 Jose David Guzman Muoz	7	BRYAN FABRICIO PINEDA CH. vie, 17 de jun de 2016, 22:55
clase - objeto	 Angel David Santacruz Heredia	5	Pablo Eduardo Santillan Garcia vie, 17 de jun de 2016, 22:53
CLASE-OBJETO	 BRENDA IRENE USINA	7	BRYAN FABRICIO PINEDA CH. vie, 17 de jun de 2016, 22:50
Clase - Objeto	 BRYAN FABRICIO PINEDA CH.	1	Pablo Eduardo Santillan Garcia vie, 17 de jun de 2016, 22:49
objeto - clase	 Pablo Eduardo Santillan Garcia	1	BRYAN FABRICIO PINEDA CH. vie, 17 de jun de 2016, 22:46
Clase-Objeto	 Edison Salomon Cadena Serrano	10	Pablo Eduardo Santillan Garcia vie, 17 de jun de 2016, 22:45

Figura 59. Taberna virtual en Moodle gamificado

Fuente: Moodle gamificado [88]

Otra herramienta que se usa para motivar la colaboración entre los estudiantes es Scrapyfy [92], que permite crear equipos y trabajar simultáneamente en el código de programación. El código que escribe cada usuario tiene color distinto para identificarlo. El equipo puede chatear o activar el micrófono para conversar vía voz.

5.2.3.2. Niveles de Programación Mágica

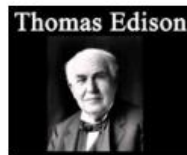
Una vez presentada la estructura general de Moodle gamificado, de sus bloques principales y en particular del bloque de recursos y actividades, se detalla a continuación la implementación de cada nivel o capítulo de Fundamentos de Programación Java de acuerdo al diseño de las estrategias para gamificar las tareas autónomas detalladas en el epígrafe 5.2.2.4.

Información del curso

En esta sección se tiene información general e introductoria al curso Fundamentos de Java gamificado. Se encuentra el sílabo de la asignatura, el reporte de asistencia a las clases y el puntaje obtenido por ello, así como el acceso a los foros generales y sociales donde los estudiantes y el docente pueden expresarse libremente, como Novedades y Cartelera en línea, y para colaboración, intercambiar información, realizar debates sobre temas de programación como las Tabernas virtuales (Figura 60).

En el curso se agrupó a los estudiantes en equipos denominados con el nombre de un científico (sabio) como: Albert Einstein, Isaac Newton, Pitágoras, Thomas Edison, fomentando la colaboración dentro del equipo y la competencia entre equipos. Para realizar la distribución de los estudiantes en los diferentes equipos se realizó una actividad de disfraces donde se sortearon papeles con los nombres de los grupos.

Información del Curso



Felicidades, estás en la casa de magia del Sabio Thomas Edison









-  Novedades
-  La Magia de la Programación
-  Sílabo Programación Mágica I
-  Asistencia
-  Cartelera en Línea
-  Taberna Virtual
-  Taberna Virtual Equipo Edison
-  Herramientas de Apoyo

Figura 60. Información del curso gamificado
Fuente: Moodle gamificado [88]


Nivel Agua – Introducción a la Programación Orientada a Objetos (POO)

En la Figura 61 se ve el primer nivel del curso (Agua), el cual corresponde al capítulo 1 de la unidad Fundamentos de Java de la asignatura Programación I. En este primer nivel, el aprendiz debe actualizar su avatar que lo identifique como aprendiz de hechicera o mago, para luego realizar actividades de hechizos y conjuros como la lectura de los Fundamentos de la Programación Orientada a Objetos, así como enviar al foro una discusión sobre clases y objetos y responder a cinco discusiones de otros aprendices; al cumplir este reto recibirá un premio virtual. Por último, realizará un reto de hechicería que es una lección o prueba sobre lo

aprendido en este nivel. En todo momento puede solicitar ayuda a sus compañeros o al sabio en la Taberna virtual del nivel.


Nivel Agua - Introducción a la POO

Inicia el impresionante mundo de la Magia de la Programación.




Estás en el Nivel Agua, donde eres un Aprendiz de Hechicera o Mago, aprenderás las bases de la Programación Orientada a Objetos y estarás listo para los primeros conjuros con la fuerza del agua.


Éxito.

 Crea tú Avatar de Aprendiz de Mago de la Programación

Hechizos y Conjuros

 Fundamentos de la Programación Orientada a Objetos

 Taberna Virtual - Fundamentos de la POO

 Felicitaciones, te has hecho acreedor de este fabuloso premio

Actividades de Hechicería

 Reto de Hechicería - Fundamentos de la POO

 Taberna Virtual de Ayuda - Nivel Agua

Figura 61. Nivel Agua – Introducción a la POO

Fuente: Moodle gamificado [88]

Nivel Tierra – Fundamentos de Java

En las Figuras 62, 63 y 64 que se ven a continuación, está el nivel Tierra y los subniveles de estructuras secuenciales, condicionales y repetitivas respectivamente, los cuales pertenecen al capítulo 2 de la unidad Fundamentos de Java. El aprendiz, una vez en este nivel, debe actualizar su avatar de conjurador, y leer el material de estudio que le permitirá realizar los hechizos de programación.

El nivel cuenta con diez hechizos para el subnivel secuenciales, 15 para condicionales (diez *if-else* y cinco *switch*) y 15 para repetitivas (cinco *for*, cinco *while* y cinco *do-while*) que deben ser vencidos por los estudiantes en el tiempo establecido para que obtenga todos los puntos del nivel. A medida que se venzan los hechizos se habilita el próximo grupo de hechizos o subnivel, de acuerdo a la planificación del mismo, es decir, si los estudiantes terminan los hechizos antes del tiempo establecido automáticamente se les activa el próximo grupo de hechizos, pero en caso de no terminarlos en tiempo solo se activa el próximo grupo o subnivel cuando concluye el tiempo, dejando de ganar los puntos de los hechizos que no hicieron. Podrán ganar también dos bienes virtuales en este nivel.

- Estructuras secuenciales

Nivel Tierra - Fundamentos de Java



Felicitaciones, has superado el Nivel Agua y ascendiste al Nivel Tierra.


Eres un Conjurador y aprenderás los **Fundamentos de Java** donde realizarás tus primeros Hechizos y Conjuros en lenguaje Java usando las estructuras Secuenciales, Condicionales y Repetitivas.


Éxito.

 Actualiza tú Avatar de Conjurador



Hechizos y Conjuros

 Recursos iniciales antes de hacer los primeros hechizos

 Fundamentos de Java





Actividades de Hechicería





Reto de Hechicería - Estructuras Secuenciales en Java


 Secuenciales: Hechizo 1


 Secuenciales: Hechizo 2


 Secuenciales: Hechizo 3


 Secuenciales: Hechizo 4

 Secuenciales: Hechizo 5

 Secuenciales: Hechizo 6

 Secuenciales: Hechizo 7

 Secuenciales: Hechizo 8

 Secuenciales: Hechizo 9


 Secuenciales: Hechizo 10

Figura 62. Nivel Tierra – Fundamentos de Java – Estructuras secuenciales
Fuente: Moodle gamificado [88]

- Estructuras condicionales o selectivas

 **Reto de Hechicería - Estructura Condicional if-else**

-  Condicionales If-else: Hechizo 1 
-  Condicionales If-else: Hechizo 2 
-  Condicionales If-else: Hechizo 3 
-  Condicionales If-else: Hechizo 4 
-  Condicionales If-else: Hechizo 5 
-  Condicionales If-else: Hechizo 6 
-  Condicionales If-else: Hechizo 7 
-  Condicionales If-else: Hechizo 8 
-  Condicionales If-else: Hechizo 9 
-  Condicionales If-else: Hechizo 10 



 **Reto de Hechicería - Estructura Condicional switch**

-  Condicionales switch: Hechizo 1 
-  Condicionales switch: Hechizo 2 
-  Condicionales switch: Hechizo 3 
-  Condicionales switch: Hechizo 4 
-  Condicionales switch: Hechizo 5 

 Felicidades!, tienes una sorpresa esperándote. 



Figura 63. Nivel Tierra – Fundamentos de Java – Estructuras condicionales
Fuente: Moodle gamificado [88]

- **Estructuras repetitivas**

 **Reto de Hechicería - Estructura Repetitiva for**

-  Repetitivas for: Hechizo 1 
-  Repetitivas for: Hechizo 2 
-  Repetitivas for: Hechizo 3 
-  Repetitivas for: Hechizo 4 
-  Repetitivas for: Hechizo 5 



 **Reto de Hechicería - Estructura Repetitiva while**

-  Repetitivas while: Hechizo 1 
-  Repetitivas while: Hechizo 2 
-  Repetitivas while: Hechizo 3 
-  Repetitivas while: Hechizo 4 
-  Repetitivas while: Hechizo 5 



 **Reto de Hechicería - Estructura Repetitiva do - while**

-  Repetitivas do - while: Hechizo 1 
-  Repetitivas do - while: Hechizo 2 
-  Repetitivas do - while: Hechizo 3 
-  Repetitivas do - while: Hechizo 4 
-  Repetitivas do - while: Hechizo 5 

 Felicitaciones!, tienes una conjuro de regalo. 

Figura 64. Nivel Tierra – Fundamentos de Java – Estructuras repetitivas
Fuente: Moodle gamificado [88]

Nivel Aire – Arreglos y matrices

En la Figura 65 que sigue a continuación, está el nivel Aire y los subniveles Arreglos y Matrices, los cuales corresponden al capítulo 3 de la unidad Fundamentos de Java de la asignatura Programación I. En este nivel debe el estudiante actualizar su avatar de invocador de hechizos, y leer la documentación sobre arreglos y matrices que le habilitará a realizar los hechizos de programación.

La complejidad aumenta en este nivel que cuenta con cinco hechizos para el subnivel Arreglos y cinco para Matrices. Al igual que en el nivel anterior, a medida que se venzan los hechizos se habilita el próximo grupo de hechizos o subnivel, en función de lo diseñado, es decir, si los aprendices terminan los hechizos antes del tiempo establecido automáticamente se les activa el próximo grupo de hechizos, pero en caso de no terminarlos en tiempo, solo se activará el próximo grupo o subnivel cuando concluya el tiempo, dejando de ganar los puntos de los hechizos que no hicieron. En este nivel se ganan dos premios virtuales, uno para cada subnivel.

Nivel Aire - Arreglos y Matrices en Java




Muchas felicidades, has terminado el Nivel Tierra, ascendiste al Nivel Aire.

Eres ahora un Invocador de Hechizos superiores, invocarás **Arreglos y Matrices de Java**.

Éxito.

 Actualiza tú Avatar de Invocador

Hechizos y Conjuros


 Fundamentos de Java

 Arreglos y Matrices


Actividades de Hechicería





Reto de Hechicería - Arreglos


 Arreglos: Hechizo 1

 Arreglos: Hechizo 2

 Arreglos: Hechizo 3


 Arreglos: Hechizo 4


 Arreglos: Hechizo 5

 Felicitaciones!, tienes un hechizo sorpresa esperándote.




Reto de Hechicería - Matrices


 Matrices: Hechizo 1

 Matrices: Hechizo 2

 Matrices: Hechizo 3

 Matrices: Hechizo 4

 Matrices: Hechizo 5

 Muy bien!, estás a punto de ser un gran mago de la programación.

 Taberna Virtual de Ayuda - Nivel Aire

Figura 65. Nivel Aire – Arreglos y matrices

Fuente: Moodle gamificado [88]

Nivel Fuego – Métodos

A continuación, en la Figura 66 se muestra el nivel Fuego, el cual pertenece al capítulo 4, Métodos, de la unidad Fundamentos de Java de la asignatura Programación I. El aprendiz se convierte en un mago por lo que debe actualizar su avatar correspondiente. Se cuenta con documentación sobre Fundamentos de Java y el uso de métodos. Es en este nivel donde realizará hechizos y conjuros usando lo aprendido en todos los niveles anteriores, por lo que la complejidad es la mayor de todo el curso.

Se tienen tres hechizos que deben realizar los estudiantes para vencer el nivel, si realizan los hechizos 1 y 2, o terminan el hechizo 3 (el de mayor reto en todo el curso), podrán realizar el reto de recetas de hechizo que es la prueba final de teoría del nivel. Al vencer el nivel recibirán un último regalo virtual que les acredita como archimago, el nivel más alto de la magia de la programación y actualizarán nuevamente su avatar (archimago).

Nivel Fuego - Métodos

Muy bien!, has terminado el Nivel Aire, estás ahora en el Nivel Fuego.



Felicidades, eres un Mago de la Programación en Java, invocarás conjuros y hechizos aprendidos en los niveles anteriores, y los juntarás para hacer hechizos más fuertes.

Al terminar este último nivel, serás un gran Archimago de la Programación.

Exito.

-  Actualiza tú Avatar de Mago
-  **Hechizos y Conjuros**
 -  Fundamentos de Java
 -  Métodos
-  **Actividades de Hechicería**
-  **Reto de Hechicería - Métodos**
Hechizos integradores
 -  Métodos: Hechizo 1
 -  Métodos: Hechizo 2
 -  Métodos: Hechizo 3
 -  Reto de Recetas de Hechizos - Fundamentos de Java
-  Bien, bien, bien!!!, te has consolidado como Mago, eres un Archimago.
-  Taberna Virtual de Ayuda - Nivel Fuego

Figura 66. Nivel Fuego – Métodos
Fuente: Moodle gamificado [88]

Luego de haber superado todos los niveles y retos, el alumno está listo para rendir el hechizo o examen final (Figura 67), si lo vence se convertirá en un nuevo sabio de la magia de la programación.

Por último, se deberá participar en una encuesta electrónica anónima que permite al autor de la investigación mejorar las estrategias de gamificación y la plataforma gamificada para el aprendizaje de programación.



The screenshot shows a Moodle course interface with a blue header containing 'Moodle Gamificado', 'Mis Cursos', and 'Este curso'. The main content area is titled 'Nivel Vida - Hechizo Final'. It features an illustration of a wizard and a congratulatory message: 'Felicidades Archimag@, ahora estás list@ para realizar tus últimos conjuros y hechizos donde demostrarás los conocimientos adquiridos sobre la Magia de la Programación. Muchos Éxitos.' Below this is a poll icon labeled 'Encuesta' with a deadline: 'Disponible hasta antes de 8 de agosto de 2016, 23:55'.

Figura 67. Hechizo final en Moodle gamificado
Fuente: Moodle gamificado [88]

Principales mejoras en la implementación de Moodle gamificado

Como se había mencionado la metodología gamificada y la plataforma virtual Moodle gamificada se usaron en tres semestres diferentes con los estudiantes de la asignatura Programación I. A modo de resumen, se listan a continuación las principales mejoras que se hicieron entre cada semestre de acuerdo a la experiencia adquirida por el investigador, y a las encuestas y entrevistas no estructuradas realizadas a los estudiantes al terminar cada semestre. Con esto el autor pudo tener la propuesta óptima que es la que se describió en el epígrafe 5.2.2 y su implementación en el epígrafe 5.2.3.

- Cambios en la escala de los puntos de experiencia

Los puntos de experiencia que los estudiantes podían ganar en las tareas autónomas eran hasta 100 puntos en el periodo abril-septiembre 2015, luego en los demás

semestres son hasta 1080 puntos; con esto los alumnos tuvieron mayor percepción de recompensa por las tareas que realizan en la plataforma gamificada.

- Cambio del módulo para la evaluación automática de programas Java

En el semestre abril-septiembre 2015 se usó el *plugin* javaunittest [93, 94] adicional a Moodle para el desarrollo de las actividades prácticas, ya que daba la posibilidad a los estudiantes de escribir métodos en Java en una interfaz dentro de la plataforma Moodle y la respuesta se evalúa automáticamente realizando pruebas unitarias automatizadas. Este *plugin* no permite ver la ejecución de los programas sino solo el resultado de la evaluación. Los estudiantes querían ver cómo se ejecutaba su programa. Esta es la razón por la que en los periodos siguientes se decidió usar el *plugin* VPL [91] que permite codificar, compilar, ejecutar y evaluar programas, y no solo en lenguaje Java sino también en otros como C, C++, C#, Pascal, PHP, Python, etc.

- Incremento de insignias y premios virtuales que los estudiantes pueden ganar

En la primera experiencia gamificada del semestre abril-septiembre 2015 para reconocer los logros alcanzados y motivar a los estudiantes, se le asignó a cada nivel (Agua, Tierra, Aire y Fuego) una insignia, la cual se alcanzaba cuando cumplían con todas las tareas autónomas diseñadas en la asignatura.

A partir del semestre octubre 2015-marzo 2016, para dar más reconocimiento por las tareas que realizaban los estudiantes sin esperar a que terminasen un nivel completo, se crearon las insignias de hechizos o subniveles (Fundamentos de Java, secuenciales, *if-else*, *switch*, *for*, *while*, *do-while*, Arreglos, Matrices y Métodos), además se aumentó el número de bienes virtuales que podían ganar.

- Cambio en la estrategia de cumplimiento de niveles y subniveles

La plataforma virtual Moodle gamificada en los semestres de abril-septiembre 2015 y marzo 2016-octubre 2016, contaba con una secuenciación en los niveles y subniveles al pasar de uno a otro, es decir, no había tiempo límite para la realización de las tareas, ocasionando que hubiera en algunos estudiantes un desfase entre las tareas que

realizaban en la plataforma y las clases presenciales. Es por ello que se decide poner tiempo máximo de cumplimiento a las tareas autónomas, lo que además permitió tener mayor reto y por lo tanto motivación para vencer cada actividad planificada en la plataforma virtual.

Una vez diseñadas las estrategias de gamificación e implementadas en Moodle, se evalúan los resultados de su uso en la práctica en el objetivo 3.

5.3. Resultados objetivo 3

Objetivo 3. Evaluar la actividad pedagógica de las estrategias de gamificación en un entorno *e-learning*, diseñadas para la modalidad presencial en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

En el objetivo 3 se expondrán los resultados obtenidos en las encuestas a los estudiantes de Programación I sobre las estrategias de gamificación y la plataforma Moodle gamificada; los puntos de experiencia y el número de insignias ganadas en el cumplimiento de las tareas autónomas, y las notas finales; en los semestres abril-septiembre 2015, octubre 2015-marzo 2016, y abril-septiembre 2016. Serán analizados cualitativa y cuantitativamente para determinar que las estrategias de gamificación en un entorno virtual de aprendizaje con la plataforma Moodle influyen en la motivación de los estudiantes y contribuyen a elevar el nivel de conocimientos obtenidos, y en consecuencia sus calificaciones son mejores.

Para confirmar estadísticamente la contribución de las estrategias gamificadas implementadas en una plataforma virtual, al final del presente objetivo se realizan las pruebas de las hipótesis correspondientes sobre la variable rendimiento académico tomando en cuenta las notas finales comparando los grupos gamificado (experimental) y tradicional (control).

5.3.1. Análisis cuantitativo

Se realizó un análisis cuantitativo al finalizar la asignatura con el propósito de medir los resultados de los estudiantes con el uso de las estrategias de gamificación y la plataforma Moodle gamificada. Para esto se usaron las notas obtenidas por los estudiantes en la realización de tareas autónomas en la unidad Fundamentos de Java que se estudia en el

segundo hemisemestre, su cumplimiento a través del número de insignias obtenidas, el nivel y avatar alcanzados, y la nota final de la asignatura, que es la suma de las notas del primero y segundo hemisemestres si el alumno aprobó en los dos hemisemestres; caso contrario es la suma del promedio de las calificaciones de los dos hemisemestres y el examen de recuperación.

5.3.1.1. Cumplimiento de las tareas autónomas en Moodle gamificado

Recordemos que la misión y reto es superar los niveles de la magia de la programación desde un aprendiz (nivel Agua), conjurador (nivel Tierra), invocador (nivel Aire), mago (nivel Fuego) y finalmente archimago (nivel Vida). Los niveles contienen subniveles y estas tareas y actividades autónomas como hechizos de programación (ejercicios), retos de hechicería (lecciones y pruebas), lecturas de hechizos y conjuros (lectura de material bibliográfico), tabernas virtuales (participación en foros) y asistencia a la escuela de magia (asistencia a clases presenciales).

Por cada tarea realizada, el aprendiz gana puntos de experiencia. Cuando finalice un nivel o subnivel con la realización de las tareas determinadas, ganará insignias (automáticas) que muestren su avance y cumplimiento dentro de la plataforma. Podrá también ganar insignias manuales entregadas por el sabio del pueblo (profesor de la asignatura).

Los niveles se vencen por la realización correcta de un número mínimo de tareas que se establece en cada nivel, o cuando el tiempo límite termine. En este caso el estudiante deja de ganar los puntos de las tareas.

De acuerdo con este diseño (detallado en el epígrafe 5.2.2), la motivación de los estudiantes por la realización de tareas autónomas puede medirse por los puntos de experiencia y la cantidad de insignias ganadas.

Este diseño se implementó en la plataforma Moodle (Moodle gamificado) y se usó en el segundo hemisemestre donde se estudia Fundamentos de Java en la asignatura Programación I en los semestres abril-septiembre 2015 en la carrera de Ingeniería en Computación Gráfica, octubre 2015-marzo 2016 y abril-septiembre 2016 en la carrera de Ingeniería Matemática. El

sílabo de la asignatura es el mismo en todas las carreras, además del docente que impartió la asignatura; la complejidad de los ejercicios y exámenes fueron similares, por lo que se consideran grupos equivalentes para realizar el estudio.

Semestre abril-septiembre 2015 (semestre 1)

En la Tabla 12, se muestra un listado de los estudiantes que terminaron la asignatura gamificada con los niveles y avatares (clase de mago) alcanzados por cada uno de ellos, los puntos de experiencia, notas obtenidas en la realización de las tareas autónomas, así como el número de insignias manuales y automáticas (insignias por vencer un nivel o subnivel).

En este semestre, se podía ganar hasta 100 puntos de experiencia, y máximo dos insignias manuales y cinco automáticas. No existía límite de tiempo para terminar las tareas. Para ganar una insignia automática se necesitaba cumplir con el 100 % de las tareas, para vencer un nivel y pasar al siguiente solo del 80 %.

Tabla 12. Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 1

Estudiante	Puntos de experiencia	Nota tareas autónomas	Nivel alcanzado	Avatar	Insignias manuales	Insignias automáticas
Estudiante 5	99,50	9,95	Vida	Archimago	2	5
Estudiante 2	94,00	9,40	Vida	Archimago	2	5
Estudiante 7	94,00	9,40	Vida	Archimago	0	4
Estudiante 3	92,40	9,24	Vida	Archimago	2	5
Estudiante 4	91,80	9,18	Vida	Archimago	2	4
Estudiante 8	91,50	9,15	Vida	Archimago	2	5
Estudiante 6	90,10	9,01	Vida	Archimago	0	4
Estudiante 10	90,10	9,01	Vida	Archimago	0	5
Estudiante 1	88,90	8,89	Vida	Archimago	2	3
Estudiante 9	73,60	7,36	Fuego	Mago	0	4

Fuente: elaboración propia

Se observa que el 90 % de los estudiantes lograron el nivel superior (Vida) y el avatar de archimago, lo que significa que al menos realizaron correctamente el 80 % de las tareas en cada nivel. Un solo estudiante obtuvo el nivel de Fuego y el avatar de mago. El 50 % tuvo cinco insignias, es decir, completó satisfactoriamente todas las tareas en cada nivel; el 40 % ganó cuatro insignias, lo que implica que en un nivel no terminaron todas las tareas; y solo un alumno obtuvo tres insignias. El 60 % tuvo dos insignias manuales.

En cuanto a los puntos de experiencia, el 90 % de los estudiantes alcanzó más de 88,90 que equivale a 8,89 en la nota de tareas autónomas.

De acuerdo con esto, se puede decir que el uso de las estrategias de gamificación y la plataforma Moodle gamificada para la realización de las tareas autónomas fue fundamental para el cumplimiento de las mismas [84].

Semestre octubre 2015-marzo 2016 (semestre 2)

De los 13 estudiantes que iniciaron el curso gamificado en el segundo semestre del periodo octubre 2015-marzo 2016, en la Tabla 13 se detalla el resultado de la ejecución de las tareas autónomas de los diez estudiantes que terminaron el curso.

La escala de puntos de experiencia en este semestre pasó a un máximo de 1080 para tener mayor percepción de reconocimiento por las tareas que realizan los estudiantes; además se incrementaron nueve insignias de subnivel a las cinco de nivel que ya se tenía, para premiar el cumplimiento de los hechizos de programación al cumplir cada subnivel y no solo al terminar un nivel. Al igual que el semestre anterior, no existía límite de tiempo para realizar todas las tareas.

También se cambió el *plugin* para evaluación automática de código Java, en la primera experiencia se usó *javaunittest* [94] que permite codificar y evaluar métodos escritos en Java, pero no permite ejecutar y ver el resultado en pantalla. Los alumnos sabían si el resultado de su código estaba correcto, pero tenían la incertidumbre de cómo funciona realmente al correrlo. Desde este semestre se usó Virtual Programming Lab (VPL) [59] que permite codificar, compilar, ejecutar y evaluar automáticamente programas Java y otros lenguajes. Al

ver como corría su programa en pantalla y evaluarlo para determinar si estaba correcto, se pudo observar que se sentían más motivados para hacer los hechizos de programación que en el semestre anterior.

Tabla 13. Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 2

Estudiante	Puntos de experiencia	Nota equivalente	Nota tareas autónomas	Nivel alcanzado	Avatar	Insignias manuales	Insignias automáticas
Estudiante 25	1068,40	10,68	10,00	Vida	Archimago	2	14
Estudiante 1	1051,25	10,51	10,00	Vida	Archimago	1	14
Estudiante 12	1047,75	10,48	10,00	Vida	Archimago	2	14
Estudiante 4	1047,50	10,48	10,00	Vida	Archimago	1	14
Estudiante 11	1045,00	10,45	10,00	Vida	Archimago	2	14
Estudiante 13	1042,14	10,42	10,00	Vida	Archimago	1	14
Estudiante 15	1040,00	10,40	10,00	Vida	Archimago	1	14
Estudiante 21	1039,00	10,39	10,00	Vida	Archimago	2	14
Estudiante 24	1035,50	10,36	10,00	Vida	Archimago	1	14
Estudiante 23	1017,50	10,18	10,00	Vida	Archimago	1	14

Fuente: elaboración propia

El 100 % de los estudiantes alcanzaron el nivel superior (Vida), con el avatar de archimago, y ganaron el total de insignias, es decir, realizaron correctamente todos los retos de hechicería en cada uno de los niveles. Además, todos los alumnos tuvieron al menos una insignia entregada por el sabio (docente). Todos los estudiantes tuvieron más de 1000 puntos de experiencia que les da una nota final en tareas autónomas de 10,0.

De acuerdo con estos resultados en la aplicación de estrategias de gamificación para la realización de tareas autónomas, se puede mencionar que influyeron en la motivación para el cumplimiento de las mismas.

Semestre abril-septiembre 2016 (semestre 3)

En este periodo estuvieron matriculados 33 estudiantes, de ellos 32 terminaron el curso completo gamificado. En la Tabla 14 se presenta el resultado de estos estudiantes en el cumplimiento de las tareas autónomas.

En los dos semestres anteriores, al no tener tiempo límite para la realización de las tareas y que los aprendices avanzasen a su ritmo, ocasionaba que mientras se estaba estudiando un capítulo o subcapítulo en forma presencial, algunos estudiantes realizaban las tareas autónomas de otro nivel o subnivel. Al final, previo al examen, la mayoría se igualaba, en varios casos producto de copia, pues otros compañeros que ya tenían resueltos los ejercicios se los proporcionaban.

Razón por la que, con un criterio de evaluación continua, se puso tiempo límite para terminar o vencer los niveles y sus subniveles, que hiciera correspondencia con las clases presenciales; si un estudiante no terminó a tiempo las tareas definidas en el diseño, deja de ganar los puntos de experiencia y avatar del subnivel correspondiente.

Por una parte, se incrementó el reto al tener tiempo máximo para vencer los niveles y subniveles; además se influyó en la motivación, los alumnos intentaban una y otra vez los hechizos (ejercicios) y retos de hechicería (lecciones o pruebas) hasta vencerlos, para no dejar de ganar los puntos e insignias; y finalmente para ir al mismo ritmo de las clases que impartía el docente, autor de esta investigación.

Tabla 14. Puntos y notas de tareas autónomas; avatar, nivel, insignias obtenidas en el semestre 3

Estudiante	Puntos de experiencia	Nota equivalente	Nota tareas autónomas	Nivel alcanzado	Avatar	Insignias manuales	Insignias automáticas
Estudiante 17	1079,33	10,7933	10,00	Vida	Archimago	0	14
Estudiante 12	1078,67	10,7867	10,00	Vida	Archimago	2	14
Estudiante 6	1077,5	10,775	10,00	Vida	Archimago	0	14
Estudiante 7	1077,5	10,775	10,00	Vida	Archimago	2	14
Estudiante 15	1075,6	10,756	10,00	Vida	Archimago	0	14
Estudiante 2	1072,84	10,7284	10,00	Vida	Archimago	0	14
Estudiante 8	1064,33	10,6433	10,00	Vida	Archimago	1	14
Estudiante 26	1062,37	10,6237	10,00	Vida	Archimago	0	14
Estudiante 14	1061,6	10,616	10,00	Vida	Archimago	1	14
Estudiante 20	1060,33	10,6033	10,00	Vida	Archimago	2	14
Estudiante 28	1056,27	10,5627	10,00	Vida	Archimago	0	14
Estudiante 10	1054,7	10,547	10,00	Vida	Archimago	0	14
Estudiante 23	1052,67	10,5267	10,00	Vida	Archimago	0	14
Estudiante 1	1050,83	10,5083	10,00	Vida	Archimago	0	14
Estudiante 27	1050,5	10,505	10,00	Vida	Archimago	0	14
Estudiante 13	1049,93	10,4993	10,00	Vida	Archimago	0	14
Estudiante 29	1047,13	10,4713	10,00	Vida	Archimago	0	14
Estudiante 9	1045,83	10,4583	10,00	Vida	Archimago	2	14
Estudiante 31	1042,17	10,4217	10,00	Vida	Archimago	0	14
Estudiante 5	1041	10,41	10,00	Vida	Archimago	0	14
Estudiante 32	1040,87	10,4087	10,00	Vida	Archimago	0	14
Estudiante 11	1038,5	10,385	10,00	Vida	Archimago	1	14
Estudiante 4	1035,83	10,3583	10,00	Vida	Archimago	0	11
Estudiante 25	1030,66	10,3066	10,00	Vida	Archimago	0	14
Estudiante 18	1029,83	10,2983	10,00	Vida	Archimago	0	14
Estudiante 21	1026,36	10,2636	10,00	Vida	Archimago	0	14
Estudiante 19	1023,41	10,2341	10,00	Vida	Archimago	0	13
Estudiante 22	1023,21	10,2321	10,00	Vida	Archimago	0	13
Estudiante 16	1021,66	10,2166	10,00	Vida	Archimago	0	14
Estudiante 24	995,66	9,9566	9,96	Vida	Archimago	0	9
Estudiante 30	939,67	9,3967	9,40	Vida	Archimago	0	13
Estudiante 3	776,34	7,7634	7,76	Vida	Archimago	0	13

Fuente: elaboración propia

El 100 % de los estudiantes alcanzaron el nivel superior (Vida), con el avatar de archimago. El 81,3 % (26) ganaron todas las insignias automáticas, lo que implica que hicieron todas las

tareas autónomas en el tiempo establecido; el 12,5 % (4) ganaron 13 insignias, es decir, no terminaron todas las tareas a tiempo en un subnivel; un estudiante tuvo 11 insignias (3,1 %) y el resto nueve (3,1 %). Además, siete alumnos obtuvieron al menos una insignia manual por ser los mejores realizando tareas autónomas.

El 90,6 % tuvieron más de 1000 puntos de experiencia que les da una nota final en tareas autónomas de 10,0; dos estudiantes entre 900 y 1000 puntos (6,3 %), y solo uno menos de 900 puntos de experiencia (3,1 %).

Estos resultados, junto a los obtenidos en los dos semestres anteriores, muestran que se logró que los estudiantes manifestaran su interés genuino por la programación y que desarrollaran una motivación intrínseca que los incitara a ejercitar, estudiar y realizar las actividades y tareas planificadas, para de esta forma adquirir mayor nivel de conocimientos.

5.3.1.2. Notas finales de la asignatura Programación I

Luego de mostrar que las estrategias de gamificación implementadas en Moodle gamificado influyen en la motivación del cumplimiento de tareas autónomas, se necesita analizar si tienen impacto también en el rendimiento académico de los estudiantes.

Para ello se indaga sobre el porcentaje de estudiantes aprobados, no aprobados (terminaron el curso sin aprobarlo), retirados (no terminaron el curso) y suspensos (no aprobados + retirados), comparando los resultados entre los grupos de control usando la metodología tradicional y experimental con el uso de la metodología y plataforma gamificadas, además del porcentaje que aprobaron y suspendieron la asignatura.

En el semestre octubre 2015-marzo 2016 (semestre 2), se dividió el curso de Programación I de Ingeniería Matemática en grupo de control y experimental de forma aleatoria; en cambio, en el periodo abril-septiembre 2016 (semestre 3), el grupo tradicional o de control fueron los estudiantes de Programación I de la carrera de Ingeniería en Diseño Industrial que fue dictada por el Profesor 15, y el grupo gamificado o experimental fue impartido por el Profesor 4 en la carrera de Ingeniería Matemática. En los dos grupos, control y experimental, el sílabo y los exámenes de cada hemisemestre y de recuperación fueron comunes.

En la Figura 68 se evidencia que el porcentaje de estudiantes aprobados usando la metodología gamificada fue creciendo puesto que en el primer semestre el porcentaje de aprobados fue del 35,3 %, en el segundo del 38,5 %, y en el tercero del 97,0 %. Por lo tanto, el porcentaje de alumnos suspensos bajó entre cada semestre. Además, se identificó que se retiraban menos estudiantes, la gran mayoría finalizaba la asignatura con buenos resultados, ya que en el primer semestre se retiró el 41,2 %, en el segundo un 23,1 % y en el tercero solo un 3,0 %.

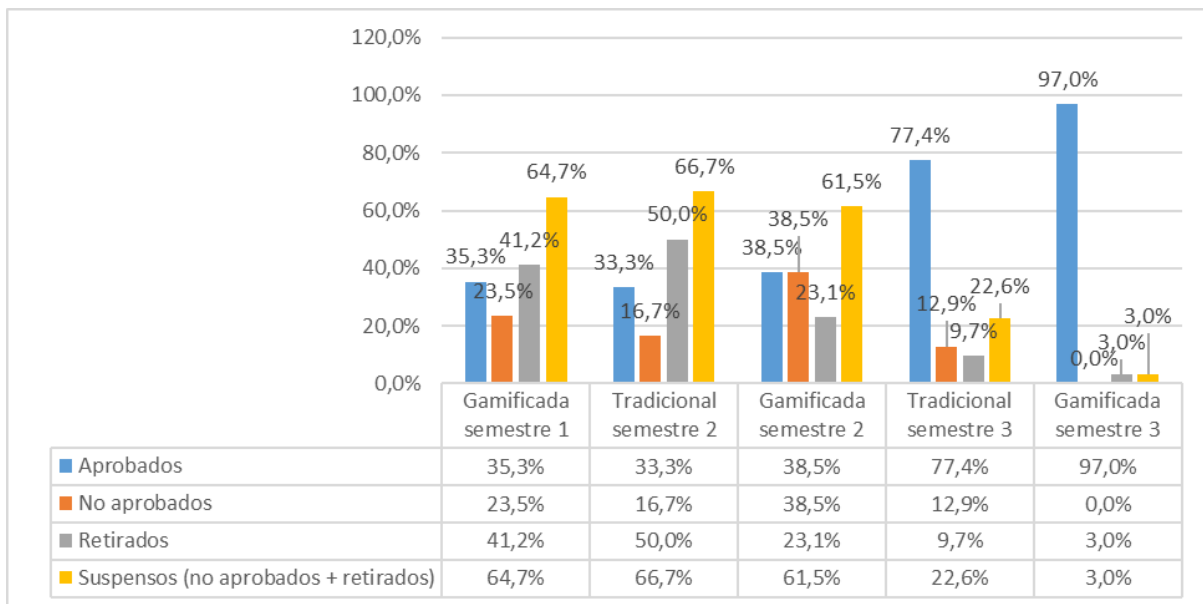


Figura 68. Comparación de resultados académicos en los tres semestres
Fuente: elaboración propia

Comparando la metodología tradicional y la gamificada usando Moodle gamificado, se observa que en el semestre 2, el porcentaje de alumnos aprobados fue ligeramente mayor en la metodología gamificada (38,5 %) que en la tradicional (33,3 %), y en el semestre 3 la diferencia fue mayor, 97,0 % usando la metodología gamificada frente a un 77,4 % en la tradicional.

La relación es contraria al analizar los estudiantes suspensos (quienes se retiraron y además los que no aprobaron terminando el curso), en el semestre 2 suspendió con la metodología gamificada el 61,5 % y con la tradicional el 66,7 %; en el semestre 3, con la metodología gamificada suspendió el 3,0 % y con la tradicional el 22,6 %.

Tomando en cuenta los estudiantes retirados, el porcentaje en los periodos semestre 2 y semestre 3 es mucho menor en el grupo experimental (gamificado) que en el grupo de control (normal); 3,0 % en el semestre 3 de la metodología gamificada y 9,7 % de la tradicional; el 23,1 % con la metodología gamificada y el 50,0 % con la tradicional en el semestre 2.

Los mejores resultados se tuvieron en el semestre 3 donde el investigador implementó las mejoras prácticas de las experiencias adquiridas en los periodos anteriores sobre el uso de las estrategias de gamificación en Moodle gamificado, encontrando de esta manera una alternativa óptima al problema planteado en la investigación sobre la baja motivación de los alumnos por la ejecución de las tareas que realizan fuera del aula; el 97,0 % aprobaron, solo el 3,0 % se retiró; de quienes terminaron la asignatura, el 100,0 % aprobó.

Al final del capítulo se realizarán las pruebas de las hipótesis correspondientes para contrastar la hipótesis de investigación.

5.3.2. Análisis cualitativo

Al finalizar la asignatura, se les realizó a los estudiantes una encuesta electrónica anónima, con el propósito de conocer su opinión sobre su experiencia al realizar las tareas autónomas en la plataforma Moodle gamificada, y sobre la metodología gamificada. Los resultados en los tres semestres (abril-septiembre 2015, octubre 2015-marzo 2016 y abril-septiembre 2016) fueron los siguientes:

P1. Usabilidad de la plataforma virtual Moodle gamificada.

La Figura 69 muestra las opiniones sobre el uso de la plataforma virtual Moodle gamificada, en los semestres estudiados. En los tres semestres, se encontró un alto porcentaje de evaluación de muy bueno, con un 77,8 %, 100,0 % y 93,3 % respectivamente. Sin duda, usar estas tecnologías en la asignatura es novedoso, implicando que el estudiante esté pendiente día a día de los ejercicios de cada uno de los niveles. Obtener los puntos de experiencia, recompensas, insignias y las calificaciones es cuestión de esfuerzo y voluntad.

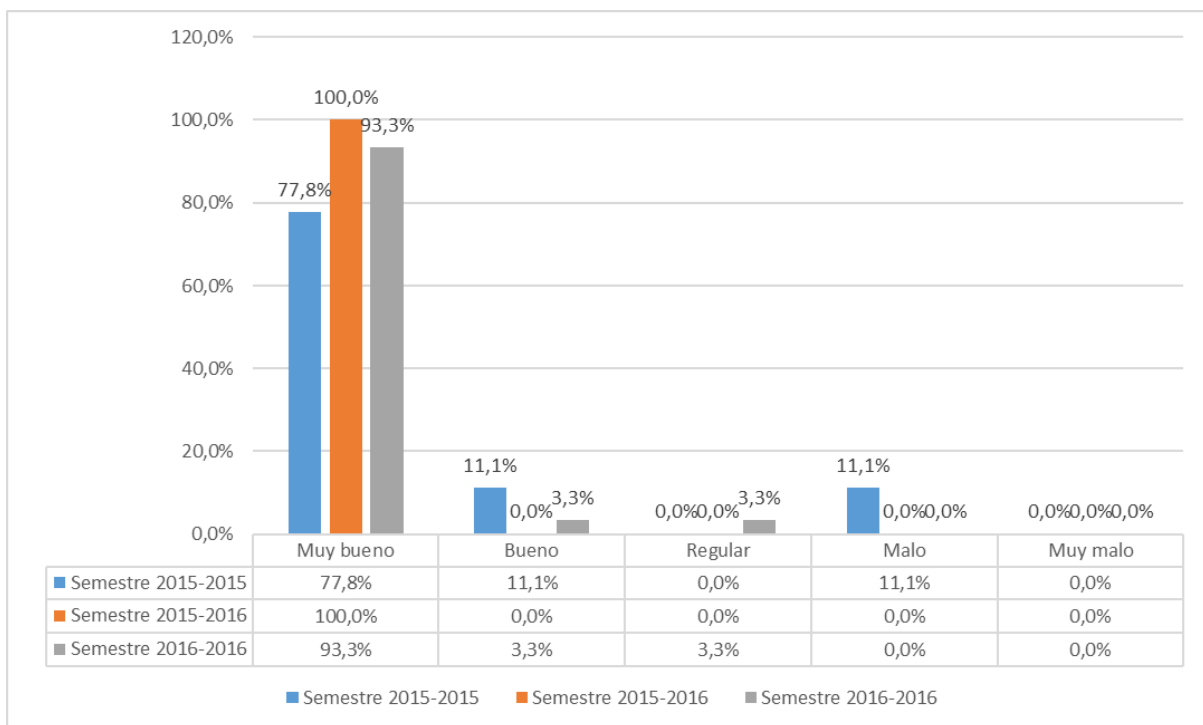


Figura 69. Usabilidad de la plataforma Moodle gamificada
Fuente: elaboración propia

P2. Moodle gamificado vs. plataforma virtual Sakai.

Los resultados obtenidos en esta pregunta por los tres semestres evidenciaron una convergencia en que la plataforma virtual Moodle gamificada comparada con la plataforma virtual Sakai, como apoyo en el proceso de aprendizaje de Programación I fue mucho mejor y mejor, con porcentajes por encima del 85,0 %. Solo en el primer y tercer semestres un 11,1 % y un 6,7 % respectivamente opinaron que era igual. Este criterio se debe a que Sakai solo era utilizada como un medio para subir tareas, realizar pruebas o exámenes teóricos y consultar información de Programación I, porque no posee *plugins* para diseñar un curso gamificado, como el caso de la plataforma virtual Moodle, que cuenta con varios *plugins* propios y externos que dinamizan el curso, como obtener en tiempo real la evaluación de las tareas autónomas de programación de los niveles, así como obtener insignias, identificarse con un avatar, el tablero de clasificación y ver cómo se va evolucionando en cada nivel.

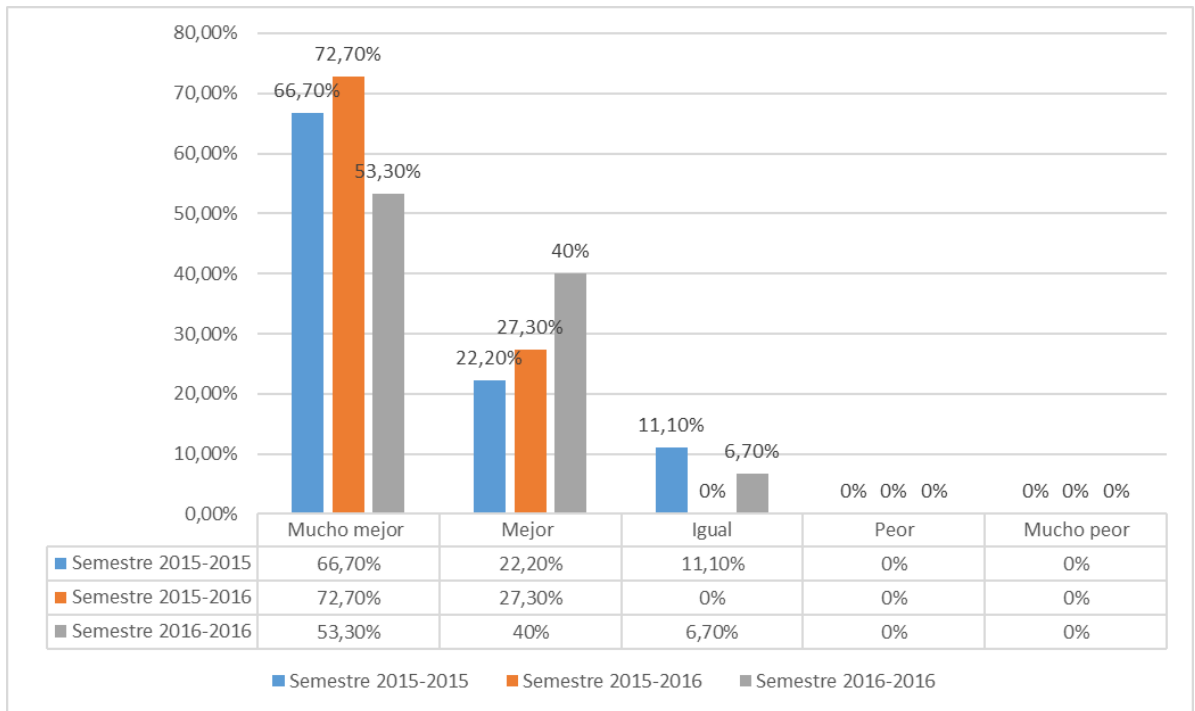


Figura 70. Moodle gamificado vs. plataforma virtual
Fuente: elaboración propia

La plataforma Moodle desde su versión 2.5 incluye componentes para diseñar cursos gamificados. Da la posibilidad de crear insignias o *badges* y hacer uso de las ya existentes en Moodlebadges. Las insignias pueden ser otorgadas automáticamente o de forma manual, además de poder exportarlas a otros sitios.

P3. Metodología de aprendizaje de gamificación en el ambiente *e-learning*.

La nueva metodología aplicada por el profesor investigador durante los tres semestres antes mencionados fue evaluada en sentido general como muy buena, con porcentajes por encima del 80,0 %. No se encontró ningún caso con calificación de regular, mala o muy mala de dicha metodología de aprendizaje. Los alumnos expresan que lograban desarrollar sus conocimientos de programación en Java, pudiendo realizar ejercicios que jamás pensaban poder hacer. Además, intentaban hacer los ejercicios varias veces hasta que lograban que funcionaran, no se daban por vencidos.

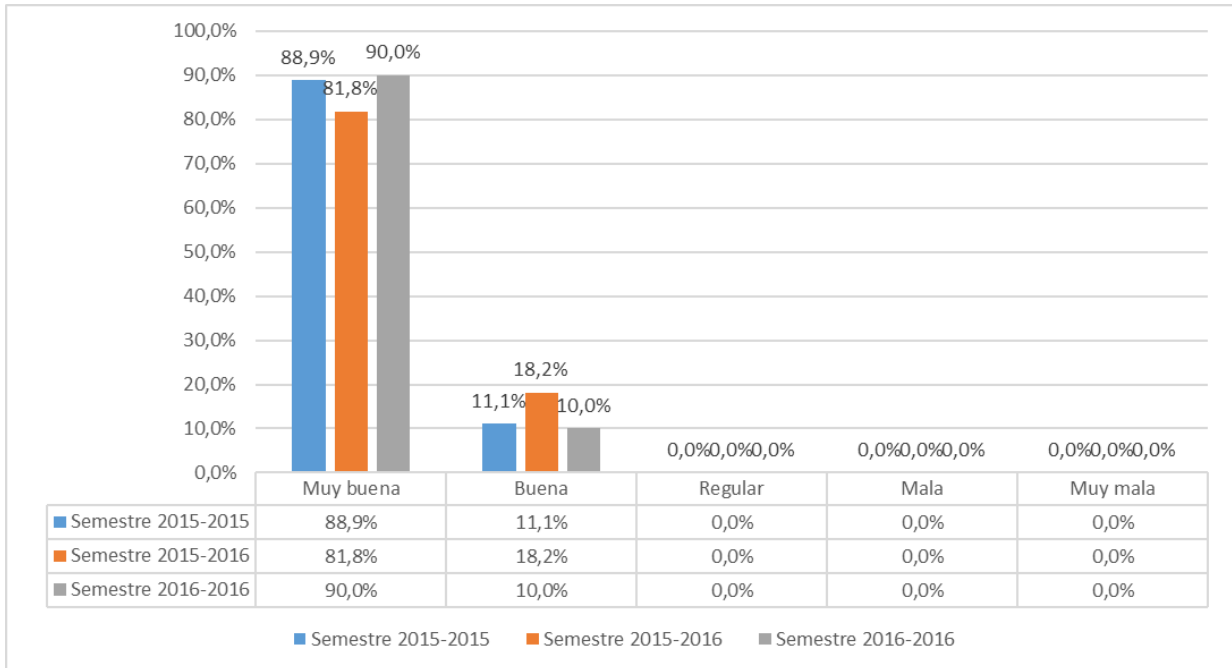


Figura 71. Gamificación en ambiente *e-learning*

Fuente: elaboración propia

P4. Metodología de *e-learning* y gamificación del segundo hemisemestre vs. metodología normal usada en el primer hemisemestre.

Comparando la metodología de *e-learning* y gamificación con la tradicional usada en el primer hemisemestre, se observa que excepto en el tercer semestre donde el 10,0 % expresaron que era igual, el 100,0 % de los estudiantes refirieron que fue mucho mejor y mejor (Figura 72).

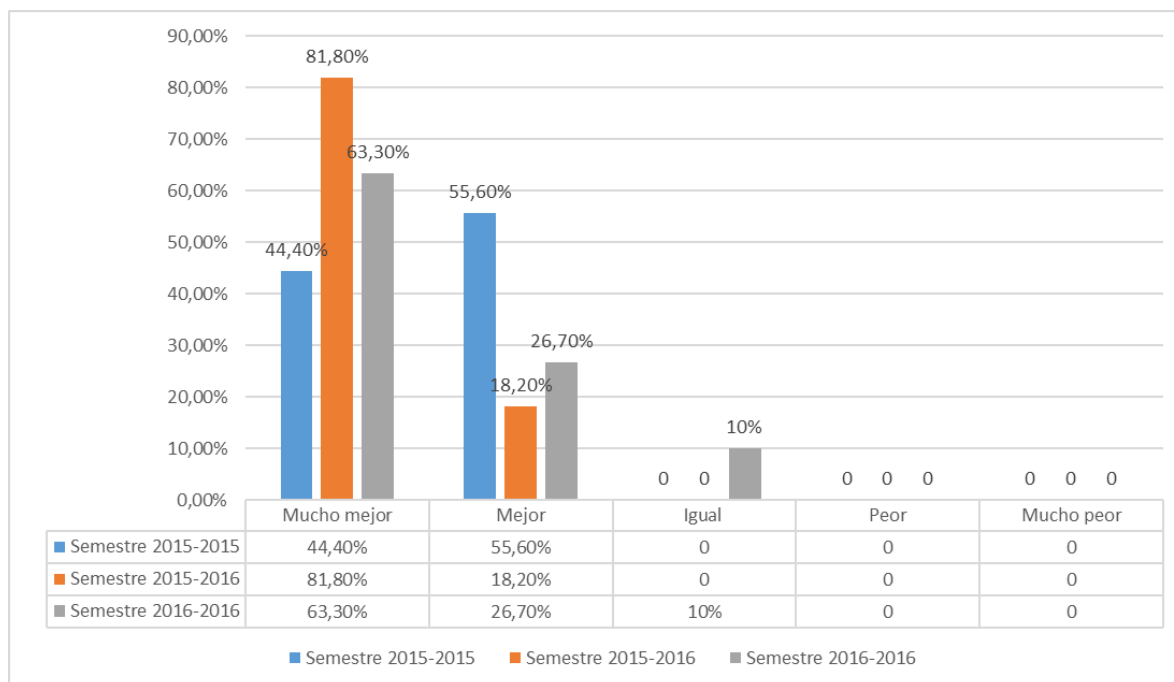


Figura 72. Criterio de los alumnos sobre la metodología tradicional y la gamificada
Fuente: elaboración propia

P5. ¿Considera que aprender a programar usando la plataforma virtual y metodología gamificadas fue divertido?

En la Figura 73 se expone el criterio de los estudiantes acerca de si aprender a programar usando la plataforma virtual y metodología gamificadas fue divertido. La mayoría de los estudiantes opinan estar totalmente de acuerdo y de acuerdo, porque en esta nueva experiencia existía competencia entre los estudiantes y a la vez colaboración entre ellos. Todos los estudiantes realizaban las tareas autónomas sin que existiera una orden del profesor, sino con el propósito de obtener mayor puntuación y premios para superar a sus compañeros. Solo se encontró en el tercer semestre un 10,0 % que refirió que le era indiferente y un caso (3,3 %) que estuvo en desacuerdo.

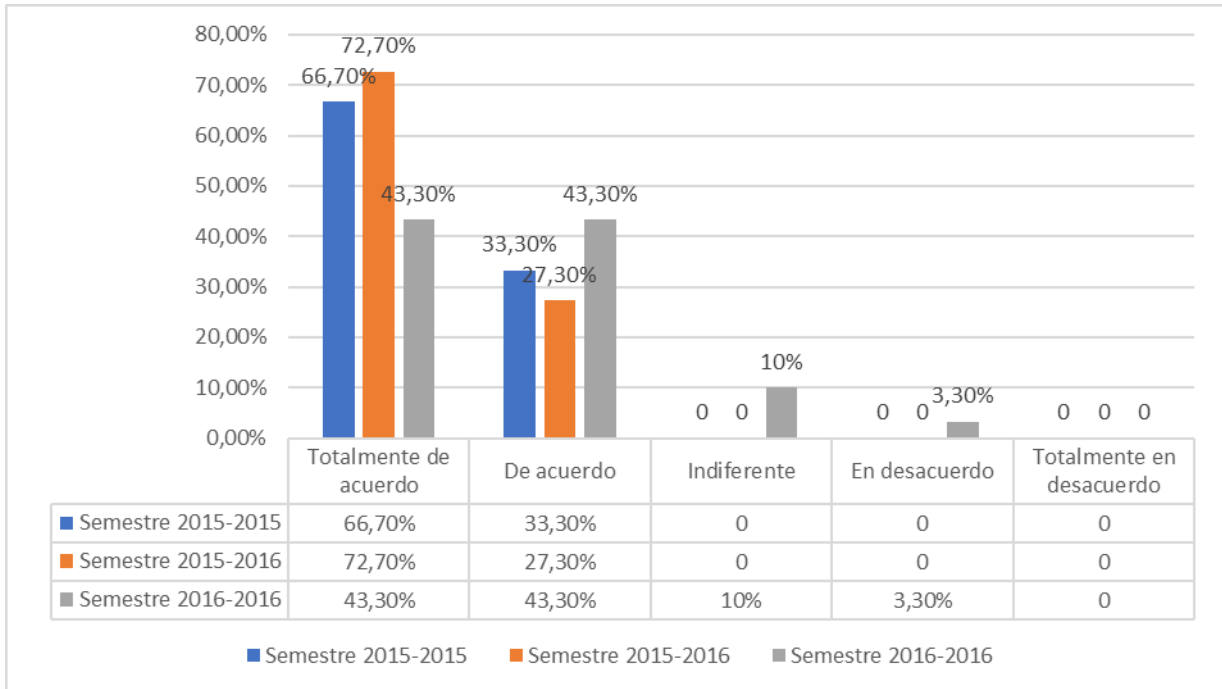


Figura 73. Aprender a programar usando la plataforma virtual y metodología gamificadas es divertido
 Fuente: elaboración propia

P6. Considera usted que la plataforma virtual y la metodología gamificada utilizada en el segundo semestre como apoyo en el aprendizaje de Programación I permiten (califique cada opción):

Tabla 15. Metodología y plataforma gamificadas utilizadas como apoyo al aprendizaje de Programación I

Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
Motivación al estudiante	6	9	21	3	2	8	0	0	1	0	0	0	0	0	0
Facilidad de aprendizaje	4	5	14	5	6	14	0	0	2	0	0	0	0	0	0
Interés	3	9	19	6	2	7	0	0	3	0	0	1	0	0	0
Favorecer a la práctica	5	10	16	2	0	10	2	1	3	0	0	0	0	0	1
Optimizar	7	7	15	1	4	14	1	0	1	0	0	0	0	0	0
Motivan el aprendizaje	6	9	19	2	1	10	1	1	0	0	0	0	0	0	1
Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %
Motivación al estudiante	66,7	81,8	70,0	33,3	18,2	26,7	0,0	0,0	3,3	0,0	0,0	0,0	0,0	0,0	0,0
Facilidad de aprendizaje	44,4	45,5	46,7	55,6	54,5	46,7	0,0	0,0	6,7	0,0	0,0	0,0	0,0	0,0	0,0
Interés	33,3	81,8	63,3	66,7	18,2	23,3	0,0	0,0	10,0	0,0	0,0	3,3	0,0	0,0	0,0
Favorecer a la práctica	55,6	90,9	53,3	22,2	0,0	33,3	22,2	9,1	10,0	0,0	0,0	0,0	0,0	0,0	3,3
Optimizar	77,8	63,6	50,0	11,1	36,4	46,7	11,1	0,0	3,3	0,0	0,0	0,0	0,0	0,0	0,0
Motivan el aprendizaje	66,7	81,8	63,3	22,2	9,1	33,3	11,1	9,1	0,0	0,0	0,0	0,0	0,0	0,0	3,3

Fuente: elaboración propia

S1: semestre abril-septiembre 2015
 S2: semestre octubre 2015-marzo 2016
 S3: semestre abril-septiembre 2016

En la Tabla 15 se muestran las respuestas de los estudiantes con respecto a la metodología y plataforma gamificadas utilizadas en la unidad Fundamentos de Java. En los tres semestres, vemos que, de forma general, los estudiantes opinan estar totalmente de acuerdo y de acuerdo con el uso de la plataforma virtual y la metodología gamificadas utilizadas como apoyo en el aprendizaje de Programación I. Los indicadores de mayor porcentaje de respuestas con respecto a las categorías totalmente de acuerdo y de acuerdo son: la motivación al estudiante (solo un estudiante en el tercer semestre dijo que le era indiferente), la facilidad de aprendizaje, la motivación del aprendizaje y la optimización del aprendizaje. En este sentido se puede decir que la mayoría de los alumnos se encuentran satisfechos con la nueva metodología utilizada para el aprendizaje, ya que les prestaban más interés a las clases para comprender las materias y luego resolver correctamente todas las tareas autónomas planificadas por el profesor en la plataforma. Esta metodología permite a los alumnos trabajar de manera didáctica e individualmente, y si tenían dudas podían ayudarse virtualmente a través de la misma plataforma.

P7. ¿Cuáles son las dinámicas de juego usadas en el segundo hemisemestre que usted considera favorecen más el aprendizaje?

Tabla 16. Dinámicas de juegos que favorecen más el aprendizaje

Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
Recompensas	5	8	19	3	3	10	1	0	1	0	0	0	0	0	0
Reconocimiento	6	6	16	2	3	11	1	2	2	0	0	1	0	0	0
Competencia	7	7	19	2	2	9	0	2	7	0	0	1	0	0	0
Logro	5	10	22	4	1	7	0	0	1	0	0	0	0	0	0
Expresión/identidad propia	3	8	17	3	2	10	3	1	2	0	0	1	0	0	0
Altruismo/ayudar	4	6	15	4	5	13	1	0	2	0	0	0	0	0	0
Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %
Recompensas	55,6	72,7	63,3	33,3	27,3	33,3	11,1	0,0	3,3	0,0	0,0	0,0	0,0	0,0	0,0
Reconocimiento	66,7	54,5	53,3	22,2	27,3	36,7	11,1	18,2	6,7	0,0	0,0	3,3	0,0	0,0	0,0
Competencia	77,8	63,6	52,8	22,2	18,2	25,0	0,0	18,2	19,4	0,0	0,0	2,8	0,0	0,0	0,0
Logro	55,6	90,9	73,3	44,4	9,1	23,3	0,0	0,0	3,3	0,0	0,0	0,0	0,0	0,0	0,0
Expresión/identidad propia	33,3	72,7	56,7	33,3	18,2	33,3	33,3	9,1	6,7	0,0	0,0	3,3	0,0	0,0	0,0
Altruismo/ayudar	44,4	54,5	50,0	44,4	45,5	43,3	11,1	0,0	6,7	0,0	0,0	0,0	0,0	0,0	0,0

Fuente: elaboración propia

S1: semestre abril-septiembre 2015
 S2: semestre octubre 2015-marzo 2016
 S3: semestre abril-septiembre 2016

En la Tabla 16 se representan las opiniones de los estudiantes respecto a las dinámicas de juego utilizadas en el curso, donde de forma general la mayoría refiere estar totalmente de acuerdo y de acuerdo en que todas las dinámicas favorecen la motivación del aprendizaje de Programación I. El indicador mejor evaluado fueron «los logros» donde un solo caso del tercer semestre manifestó una posición de indiferencia ante la pregunta. El uso de dinámicas de recompensas también tuvo una evaluación destacable con más de un 88,0 % de evaluación positiva por parte de los estudiantes. De aquí que se pueda afirmar que todas las dinámicas aplicadas en las estrategias de gamificación favorecen la motivación del aprendizaje de Programación I.

- **P8.** ¿Cuáles son las mecánicas de juego usadas en el segundo semestre que usted considera favorecen más el aprendizaje de programación?

Tabla 17. Mecánicas de juego que favorecen más el aprendizaje de programación

Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
Puntos	6	9	20	3	0	8	0	0	0	0	0	1	0	0	1
Insignias	5	6	18	4	4	8	0	1	3	0	0	1	0	0	0
Tabla de clasificación	5	7	18	3	4	8	1	0	2	0	0	1	0	0	1
Barra de progreso	1	8	19	3	2	8	5	1	1	0	0	2	0	0	0
Avatares	4	8	15	2	1	8	2	2	4	1	0	2	0	0	1
Niveles	5	9	23	3	2	5	1	0	1	0	0	1	0	0	0
Retroalimentación	6	9	18	3	2	9	0	0	1	0	0	2	0	0	0
Bonos o puntos adicionales	3	7	22	4	2	6	2	2	1	0	0	0	0	0	1
Colaboración	3	5	17	4	5	11	2	1	2	0	0	0	0	0	0
Desbloqueo de contenido	8	9	17	1	2	11	0	0	0	0	0	1	0	0	1
Misiones o retos	8	8	25	1	3	4	0	0	0	0	0	1	0	0	0
Bienes virtuales o retos	4	5	22	5	5	6	0	1	2	0	0	0	0	0	0
Regalos	4	6	20	4	4	8	1	1	1	0	0	0	0	0	1
Pregunta	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %	S1 %	S2 %	S3 %
Puntos	66,7	100,0	66,7	33,3	0,0	26,7	0,0	0,0	0,0	0,0	0,0	3,3	0,0	0,0	3,3
Insignias	55,6	54,5	60,0	44,4	36,4	26,7	0,0	9,1	10,0	0,0	0,0	3,3	0,0	0,0	0,0
Tabla de clasificación	55,6	63,6	60,0	33,3	36,4	26,7	11,1	0,0	6,7	0,0	0,0	3,3	0,0	0,0	3,3
Barra de progreso	11,1	72,7	63,3	33,3	18,2	26,7	55,6	9,1	3,3	0,0	0,0	6,7	0,0	0,0	0,0

Pregunta	Totalmente de acuerdo			De acuerdo			Indiferente			En desacuerdo			Totalmente en desacuerdo		
	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
Avatares	44,4	72,7	50,0	22,2	9,1	26,7	22,2	18,2	13,3	11,1	0,0	6,7	0,0	0,0	3,3
Niveles	55,6	81,8	76,7	33,3	18,2	16,7	11,1	0,0	3,3	0,0	0,0	3,3	0,0	0,0	0,0
Retroalimentación	66,7	81,8	60,0	33,3	18,2	30,0	0,0	0,0	3,3	0,0	0,0	6,7	0,0	0,0	0,0
Bonos o puntos adicionales	33,3	63,6	73,3	44,4	18,2	20,0	22,2	18,2	3,3	0,0	0,0	0,0	0,0	0,0	3,3
Colaboración	33,3	45,5	56,7	44,4	45,5	36,7	22,2	9,1	6,7	0,0	0,0	0,0	0,0	0,0	0,0
Desbloqueo de contenido	88,9	81,8	56,7	11,1	18,2	36,7	0,0	0,0	0,0	0,0	0,0	3,3	0,0	0,0	3,3
Misiones o retos	88,9	72,7	83,3	11,1	27,3	13,3	0,0	0,0	0,0	0,0	0,0	3,3	0,0	0,0	0,0
Bienes virtuales o retos	44,4	45,5	73,3	55,6	45,5	20,0	0,0	9,1	6,7	0,0	0,0	0,0	0,0	0,0	0,0
Regalos	44,4	54,5	66,7	44,4	36,4	26,7	11,1	9,1	3,3	0,0	0,0	0,0	0,0	0,0	3,3

Fuente: elaboración propia

S1: semestre abril-septiembre 2015
S2: semestre octubre 2015-marzo 2016
S3: semestre abril-septiembre 2016

De igual forma que la metodología utilizada y las dinámicas de juego, las mecánicas de juego fueron evaluadas de forma positiva en los tres semestres, ya que, en todos los indicadores, el promedio de los porcentajes de las evaluaciones totalmente de acuerdo y de acuerdo son superiores al 75,0 %. La mecánica de juego mejor evaluada fueron las «misiones o retos», donde en los dos primeros semestres el 100,0 % de los estudiantes estuvieron totalmente de acuerdo y de acuerdo, y en el tercer semestre solo un estudiante refirió una evaluación negativa al encontrarse en desacuerdo. Comportamiento muy similar fue encontrado en el uso de las mecánicas de «puntos», «desbloqueo de contenidos» y «retroalimentación». La mecánica peor evaluada fue la de «avatares», con porcentajes en las categorías totalmente de acuerdo y de acuerdo del 66,6 %, 81,8 % y 76,7 % en el primer, segundo y tercer semestre respectivamente.

P9. Califique su grado de motivación en el segundo semestre en el aprendizaje de Programación I.

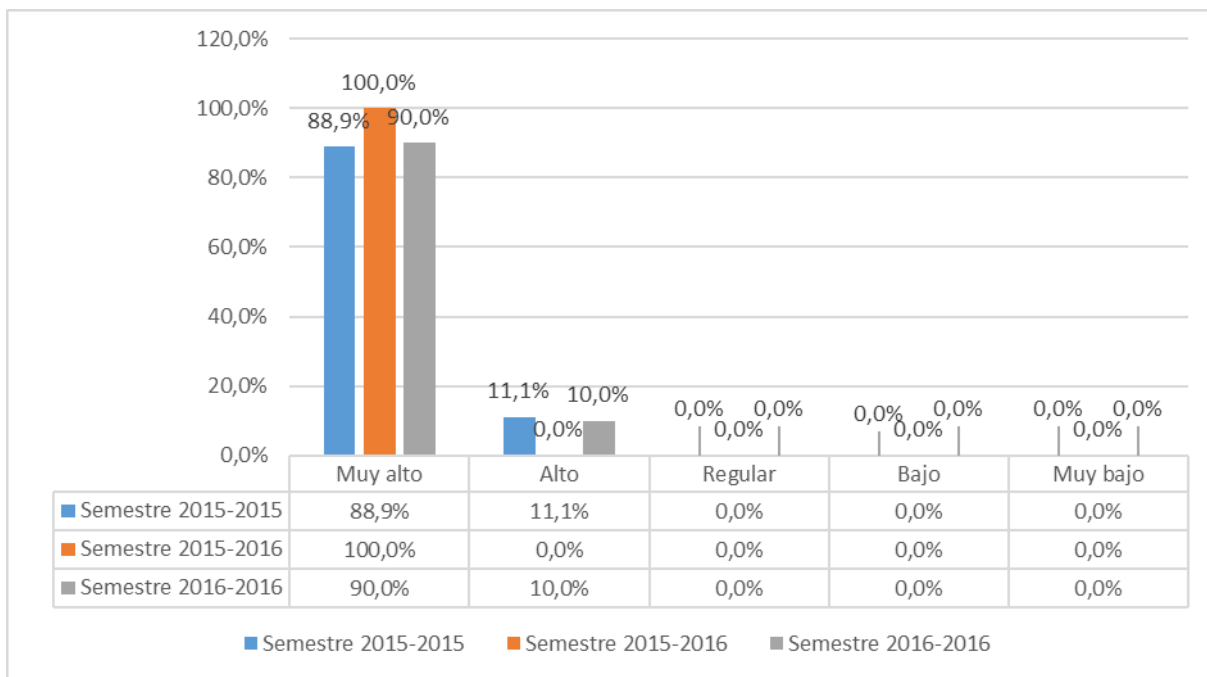


Figura 74. Grado de motivación en el aprendizaje de Programación I
Fuente: elaboración propia

En la Figura 74 se representa la opinión de los estudiantes sobre la clasificación del grado de motivación en el aprendizaje de Programación I. La motivación en los estudiantes de forma general fue positiva. En el segundo semestre, el 100,0 % de los estudiantes expresó un grado de motivación muy alto; en el primer y tercer semestre, solo un 11,1 % y un 10,0 % dijeron que el grado de motivación era alto, no existiendo estudiantes que manifestaran un grado regular, bajo o muy bajo de motivación.

P10. Motivación en el aprendizaje de Programación I del segundo semestre comparado con el primero.

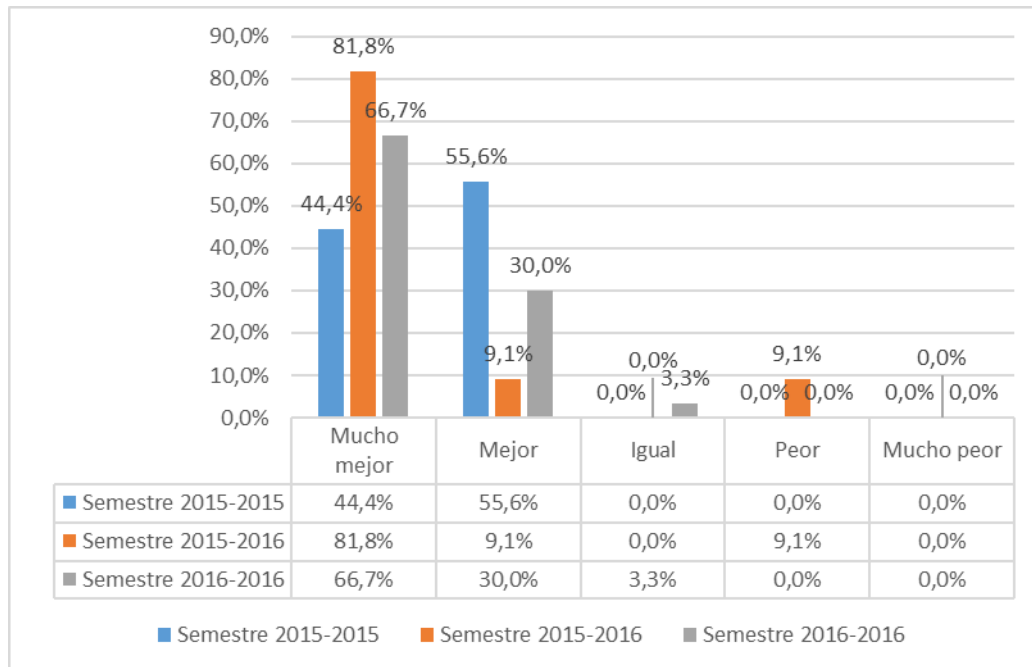


Figura 75. Grado de motivación en el aprendizaje del segundo semestre comparado con el primero
Fuente: elaboración propia

La presente pregunta de la encuesta es de suma importancia ya que hace que los estudiantes comparen la metodología de aprendizaje del segundo semestre y la del primero. Según la Figura 75, en los tres semestres se ve como los estudiantes afirman que la metodología gamificada con el uso de un entorno *e-learning* es mucho mejor y mejor porque les permite realizar las actividades en cualquier momento y desde cualquier lugar sin necesidad de tener instalada alguna herramienta en el computador y en caso de tener dudas pueden consultarlas virtualmente por la plataforma a otros compañeros de estudio. Además, son motivados por los premios e insignias y otras dinámicas y mecánicas de juego que se aplican en el curso. Solo en el segundo semestre un 9,1 % expresó que el grado de motivación fue peor.

P11. ¿Considera que ha mejorado su nivel de conocimiento de programación en este semestre?

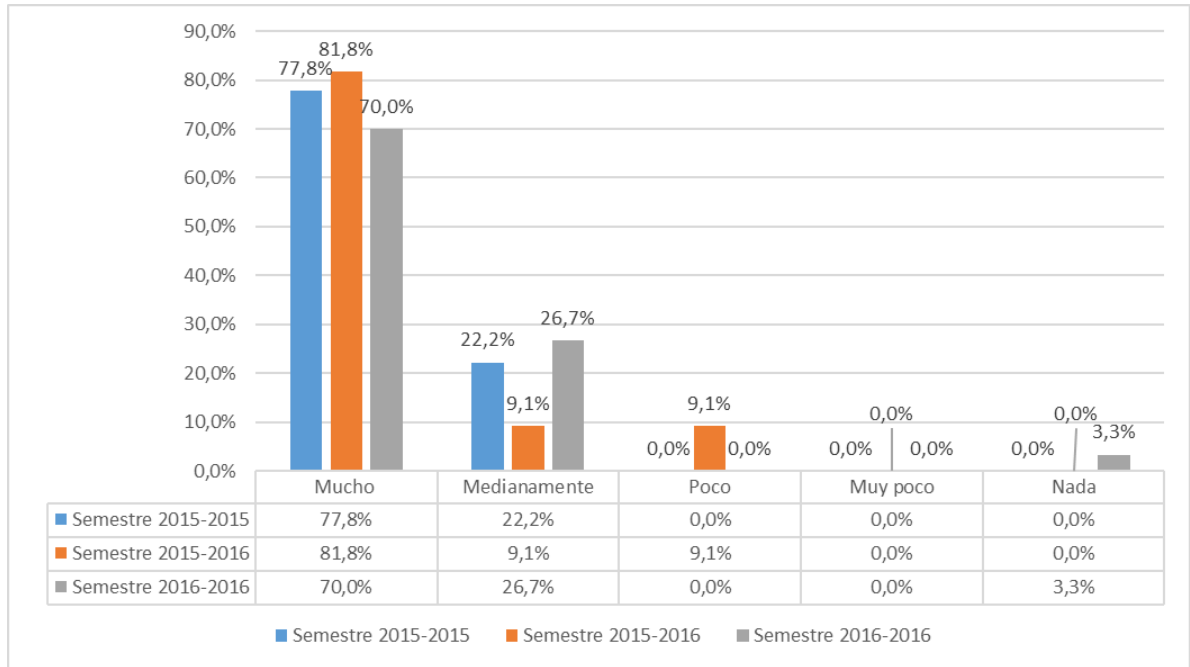


Figura 76. Nivel de conocimiento de programación
Fuente: elaboración propia

Con respecto a la valoración de conocimientos adquiridos, en los tres semestres, el 70 % o más opinaron que los resultados académicos fueron mejores y que su nivel de conocimientos de programación son más elevados gracias a la dedicación que le prestaban a la asignatura, pues cuando se practica de forma autónoma y motivados se adquieren mejor los contenidos (se aprende más). Solo un estudiante en el tercer semestre consideró que sus conocimientos no mejoraron nada.

P12. Califique su satisfacción general del curso gamificado de Programación I.

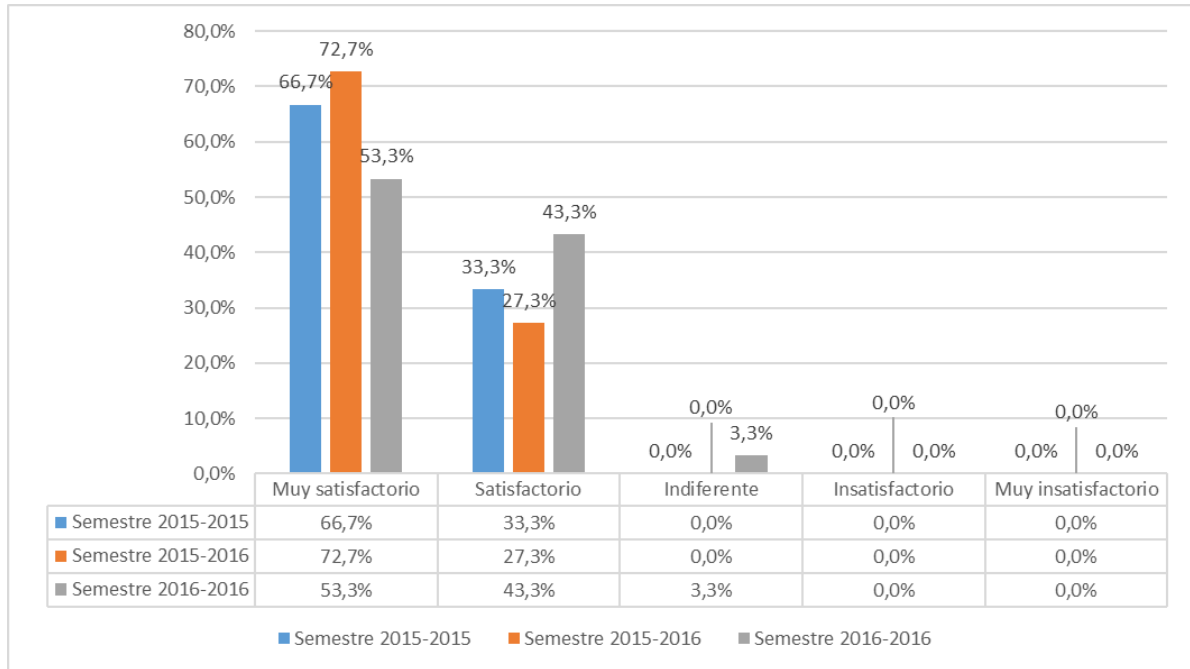


Figura 77. Nivel de satisfacción general del curso gamificado
Fuente: elaboración propia

La calificación que dan los alumnos al curso recibido en los tres semestres es positiva pues los niveles de satisfacción fueron de muy satisfactorio y satisfactorio, excepto un estudiante del último semestre que expresó que le era indiferente; por lo que casi el 100,0 % de los estudiantes fueron motivados para aprender a programar y quedaron completamente satisfechos con las estrategias de gamificación aplicadas en Programación Mágica I.

5.3.3. Comprobación de hipótesis

Para determinar si la implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje para desarrollar tareas autónomas contribuye en la mejora del rendimiento de los estudiantes, en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, se comprueba la hipótesis relacionada con el rendimiento académico de los estudiantes mediante test de hipótesis.

Para esto, se toman las notas finales de los estudiantes en cada semestre y se someten a un test de Kolmogorov-Smirnov para comprobar la normalidad de la misma y la viabilidad de utilizar test paramétricos.

En caso de normalidad se usa el test t-Student para evaluar las diferencias entre las medias de las notas de los grupos comparados (tradicional y gamificado). Cuando la variable medida resulta tener una distribución que no es normal, se aplica el test no paramétrico para muestras independientes U-Mann-Whitney (comparación de dos grupos: normal vs. gamificado), complementándose con la prueba de la mediana para k muestras. Se realiza además una comparación entre los grupos que utilizaron la metodología gamificada por cada semestre, utilizándose para ello el test no paramétrico de Kruskal-Wallis (comparación de k muestras), complementándose de igual forma con la prueba de la mediana para k muestras.

1) Planteamiento de la hipótesis

H₀ = El rendimiento académico de los estudiantes en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador no es significativamente diferente entre los estudiantes que emplean la propuesta y aquellos que no la siguen.

H₁ = El rendimiento académico de los estudiantes en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador es significativamente diferente entre los estudiantes que emplean la propuesta y aquellos que no la siguen.

2) Decisión

La regla de decisión es: si $p \leq 0,05$ se rechaza H₀.

3) Aplicación del test

a) Test de normalidad de Kolmogorov-Smirnov

Tabla 18. Test de normalidad de Kolmogorov-Smirnov

Semestres	Z de Kolmogorov-Smirnov	
	Estadístico de prueba	p (sig. asintótica bilateral)
Semestre abril-septiembre 2015	0,223	0,025
Semestre octubre 2015-marzo 2016	0,112	0,200
Semestre abril-septiembre 2016	0,263	0,000

Fuente: elaboración propia

Según el test de normalidad de Kolmogorov-Smirnov se tienen los siguientes resultados:

- En los semestres abril-septiembre 2015 y abril-septiembre 2016, la distribución de las notas finales de los estudiantes no se ajustan a distribuciones normales pues $p \leq 0,05$.
- En el semestre octubre 2015-marzo 2016, $p > 0,05$ por lo que las notas finales de los estudiantes se distribuyen normalmente, con lo que en este caso se justifica el uso de pruebas paramétricas.

b) Test aplicados

Inicialmente, se evalúa si existen diferencias significativas entre los estudiantes que cursan bajo la metodología normal y la gamificada del semestre octubre 2015-marzo 2016. Para constatar lo mencionado se aplica una prueba t-Student para evaluar las medias de las notas en ambos grupos, ya que se había probado la normalidad de dicha variable.

Tabla 19. Test t-Student de las notas del semestre octubre 2015-marzo 2016

Estrategias	$\bar{X} \pm D.E$	T	P	IC 95 %
Normal	18,45 \pm 7,71	1,057	0,076	[-0,67;12,46]
Gamificada	24,35 \pm 8,13			

Fuente: elaboración propia

- **Interpretación:** según los resultados encontrados al aplicar el test, no existen diferencias significativas entre los grupos en este semestre pues $p > 0,05$.

En un segundo momento, se evalúan las diferencias entre los grupos bajo ambas metodologías en el semestre abril-septiembre 2016. Para este caso, al no ser normal la distribución de la variable, se aplica un test no paramétrico para muestras independientes U-Mann-Whitney. La prueba de suma de rangos Wilcoxon, o prueba de Wilcoxon-Mann-Whitney, comprueba la

heterogeneidad de dos muestras ordinales bajo los supuestos de independencia de ambos grupos y variables ordinales o continuas. Como complemento se realiza la prueba de la mediana para k muestras, que evalúa las diferencias entre las medianas de las notas en ambos grupos. Los resultados se exponen a continuación:

Tabla 20. Test no paramétrico de las notas del semestre abril-septiembre 2016

Estadísticos y test utilizados	Metodología de enseñanza	
	Normal	Gamificada
$\bar{X} \pm D.E$	26,88 ± 7,9	32,58 ± 5,0
Mediana	29,0	32,8
Mann-Whitney U	740,500	
Estadístico estandarizado	-3,587	
Significación asintótica bilateral (p)	0,000	
Mediana de grupos	30,615	
Test con corrección de Yates (Chi cuadrado)	12,262	
Significación asintótica bilateral (p)	0,000	

Fuente: elaboración propia

- **Interpretación:** al aplicar tanto el test de rangos U-Mann-Whitney como el test de las medianas, se constata la diferencia de los dos grupos ($p \leq 0,05$), por lo que podemos afirmar que existe una diferencia significativa entre las notas del grupo bajo la estrategia de aprendizaje normal y las notas del grupo bajo la estrategia gamificada.

Finalmente, con el objetivo de evaluar si existen diferencias significativas en las notas de los grupos gamificados en los semestres: abril-septiembre 2015 (semestre 1), octubre 2015-marzo 2016 (semestre 2) y abril-septiembre 2016 (semestre 3), se aplica el test no paramétrico de Kruskal-Wallis que compara la distribución de más de dos grupos. Al igual que en el caso anterior los resultados se complementan con el test de la mediana para k muestras. Los resultados se muestran a continuación:

Tabla 21. Test no paramétrico de las notas de los semestres 1, 2 y 3

Estadígrafos y test utilizados	Gamificado semestre 1	Gamificado semestre 2	Gamificado semestre 3
$\bar{X} \pm D.E$	21,72 \pm 9,3	26,88 \pm 7,9	32,58 \pm 5,0
Mediana	21,7	29,0	32,8
Anova de 1-factor de Kruskal-Wallis	20,090		
Significación asintótica bilateral (p)	0,000		
Mediana de grupos	30,700		
Test	7,133		
Significación asintótica bilateral (p)	0,028		

Fuente: elaboración propia

- **Interpretación:** los resultados obtenidos tras aplicar el test de Kruskal-Wallis y el test de medianas para k muestras verifican que existe una diferencia de grupos ($p \leq 0,05$), por lo que se puede decir que los resultados de las notas de los estudiantes bajo la metodología de enseñanza gamificada fueron diferentes en los tres semestres donde fue aplicada. Además, podemos constatar que el promedio de las notas en el semestre 2 con respecto al semestre 1 fue mayor, e igualmente las notas del semestre 3 fueron mayores a las de los dos semestres anteriores, existiendo una mejoría paulatina en cada semestre donde se aplica la propuesta.

4) Comprobación de hipótesis

De esta manera, se puede decir que la implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje para desarrollar tareas autónomas contribuirá al rendimiento de los estudiantes, en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

CONCLUSIONES

Las tecnologías de la información y de las comunicaciones en la era digital que vivimos son un factor fundamental para fomentar el conocimiento, pues permiten producirlo, almacenarlo, comunicarlo, estructurarlo, registrarlo, documentarlo y compartirlo desde todas partes y sin límites de tiempo, es decir, son contenedoras y productoras de conocimiento.

Las TIC han encaminado el proceso de conocimiento de los estudiantes porque posibilitan la colaboración, comunicación, análisis y creatividad como instrumentos del conocimiento; son la vía para lograr mayor interés y motivación de los estudiantes, lo que hace que este proceso sea atractivo, aprendan en menor tiempo, y maximicen el desempeño académico.

En el ámbito de la educación, las discusiones teórico-filosóficas se centran en la necesidad de optimizar la calidad de la enseñanza que se brinda a las nuevas generaciones. En Ecuador, la mayor preocupación está generada por los escasos resultados académicos que obtienen los estudiantes en asignaturas básicas como matemáticas y lengua materna [65]. Por lo que es necesario investigar sobre estrategias para el aprendizaje e innovaciones pedagógicas que incentiven y aumenten la eficacia de la educación presencial con la ayuda de la tecnología [68].

En particular, en las universidades la inclusión de las TIC es un hecho estratégico para el desarrollo del proceso de enseñanza-aprendizaje, pues estas ofrecen nuevas vías de aprendizaje, transforman el rol del docente y cuestionan los métodos pedagógicos actuales. Internet permite acceder a una gran cantidad de recursos académicos, el docente cede su acción transmisora de conocimientos y centra su trabajo en el aprendizaje [67, 66]. Por tanto, la repercusión de las TIC se ha generalizado siendo considerada una puerta abierta al conocimiento y a las transformaciones sociales.

Como afirma Negrete [70]: «en las altas casas de estudios que son normalmente presenciales se hace imprescindible la necesidad de asumir las TIC dentro de su patrón educativo»; es importante contar con modelos pedagógicos actualizados, concentrados en el estudiante, donde estos asuman el protagonismo de un aprendizaje significativo y no como actores pasivos frente al reto que simboliza una educación innovadora en esta época.

En la educación superior ecuatoriana, la virtualización se identifica en la manifestación de actores, información, conocimientos, procesos y objetos asociados a actividades de enseñanza-aprendizaje, investigación, extensión y gestión, así como en instrumentos cuya manipulación permite al usuario realizar diversas operaciones a través de Internet, tales como aprender mediante la interacción con cursos virtuales, suscribirse a foros, consultar documentos en una biblioteca virtual, o comunicarse multidireccionalmente entre docente y estudiante [70].

Para una mejor reestructuración de la integración de las TIC en el ámbito educativo, el Estado ecuatoriano creó varias leyes específicas para respaldar el proceso de enseñanza universitaria, donde establece obligaciones referentes a la actualización del sistema educacional, a la implementación de equipos tecnológicos de acuerdo a las particularidades de las diferentes instituciones educativas, y la capacitación al docente en el uso de las TIC para que este integre dichos conocimientos en sus asignaturas [69].

Normalmente las universidades han apoyado la enseñanza presencial con la virtualidad a través del *e-learning* o enseñanza virtual a través de los entornos o plataformas virtuales de aprendizaje (LMS), donde el profesor y los estudiantes se acompañen de sitios en los que pueden acceder a todos los recursos académicos, haciendo que desaparezcan las barreras espacio-tiempo, pero siempre el alumno es el centro, y el profesor deja de ser un mero comunicador de contenidos para ser un guía que orienta y ayuda en los procesos formativos; la comunicación siempre será constante con contenidos actualizados. Los LMS son aplicaciones informáticas que apoyan el proceso de enseñanza presencial, además permiten desarrollar las enseñanzas semipresencial y no presencial [52, p. 51, 41].

El uso de plataformas para entornos *e-learning* tiene como objetivo principal permitir la creación y gestión de los espacios de aprendizaje por Internet, donde docentes y estudiantes pueden comunicar e interactuar con información durante el proceso de formación, lo que aporta ventajas como: la motivación del estudiante, el fomento del proceso individual de aprendizaje, la retroalimentación inmediata y el acceso no lineal a la información [42].

Dicho espacio es el que permite al alumnado establecer competencias y motivarse por el curso, aunque se debe esclarecer que el docente es la persona más importante para lograr que los estudiantes se motiven y superen satisfactoriamente el mismo con notas relevantes [43].

Uno de los principales problemas que enfrentan los docentes es conseguir de los estudiantes una adecuada motivación y compromiso en el proceso de aprendizaje. Los docentes deben tener un papel activo como motivadores para el aprendizaje, poniendo a disposición de los estudiantes todos los recursos y estrategias que favorezcan la motivación intrínseca para el estudio, y crear un clima adecuado que fomente la participación y el interés por la asignatura (motivación extrínseca). La motivación por tanto constituye un condicionante decisivo del aprendizaje y el rendimiento académico [95].

La motivación en el aprendizaje de programación es también un factor determinante; los factores intrínsecos que contribuyen en el aprendizaje de programación son la actitud individual, las expectativas, y las metas desafiantes; los factores extrínsecos son una clara dirección, la recompensa y el reconocimiento, el castigo y la presión social, y la competencia [86].

En el aprendizaje de programación la motivación del estudiante es baja, debido principalmente, a la novedad de la asignatura, a la dificultad de aprender conceptos abstractos y su posterior uso en la resolución de problemas, y a la falta de capacidad de trabajo continuo [9]. El desarrollo de buenas habilidades de programación, por lo general requiere de mucha práctica, que debe ser realizada fuera del aula, y que no puede ser sostenida a menos que los estudiantes estén adecuadamente motivados [86]. A los estudiantes les gusta más la práctica que la teoría, pero no les gusta realizar trabajos autónomos; existe entonces una falta de motivación por el autoaprendizaje [15].

En este contexto, los docentes deben buscar estrategias, métodos y herramientas para motivar a los estudiantes que son «nativos digitales» y diseñar actividades centradas en los estudiantes para el desarrollo de sus competencias [6]. Se debe plantear métodos y herramientas para mantener e incrementar la motivación por los trabajos y tareas fuera del aula [15], así como hacer del aprendizaje de programación una tarea emocionante y agradable [86].

Surge entonces la gamificación como apoyo al proceso de enseñanza-aprendizaje, que sistematiza estrategias y procedimientos a la hora de impartir las clases, o en la realización de actividades de aprendizaje guiando a los estudiantes en la acumulación de conocimientos y competencias a partir de novedosas técnicas que les generen un aprendizaje más relevante. Se

ha usado con dos propósitos principales: fomentar el aprendizaje de conductas deseadas, y promover la participación de los estudiantes en actividades de aprendizaje [29]. Entre las muchas definiciones de este concepto, respecto a la educación puede asumirse que «la gamificación en educación es el uso de los elementos de juego (dinámicas, mecánicas y componentes) en un ambiente de aprendizaje, generalmente apoyado en las TIC» [30], con el objetivo de que los estudiantes logren apropiarse de forma adecuada de ciertos resultados, como la satisfacción, la persistencia, y los logros académicos [29].

Es el caso puntual de la gamificación aplicada a la educación, integra al proceso educacional algunas dinámicas que se basan en recompensas, logros, competencia y el estatus al que puede llegar el estudiante según su esfuerzo y motivación, y las mecánicas de juego que potencien el interés de los estudiantes, en el contexto educativo y profesional, tales como acumular puntos, escalar niveles, premios, regalos y otros [1].

Los autores Lee, Ceyhan, Jordan-Cooley y Sung [1] exponen que a partir de la gamificación pueden desarrollarse herramientas que ofrezcan al estudiante la oportunidad de aprender a través de una solución motivadora. En tanto, como instrumento educativo, resulta interesante debido a que «la gamificación puede ser una estrategia de gran alcance que promueva la educación entre las personas y un cambio de comportamiento» [1, p. 15].

Si tenemos en cuenta que la gamificación está dirigida a la inducción de la motivación en el acto del aprendizaje, entonces podemos relacionarla con los buenos resultados de los estudiantes, tales como: mejoras académicas, satisfacción y persistencia.

Respecto al aprendizaje de programación, la gamificación es también una buena alternativa para mejorar la motivación y el compromiso de los estudiantes en la ejecución de tareas autónomas; es un método eficaz para lograr cambios positivos en su comportamiento y actitud hacia el aprendizaje.

La gamificación con la aplicación de mecánicas y dinámicas de juego en un ambiente *e-learning*, puede ser aplicada en el proceso de aprendizaje como actividades que tienen el propósito de lograr ciertos objetivos de aprendizaje, aumentar la motivación de los estudiantes y que participen en un agradable entorno competitivo con otros alumnos [96]; además influye

en la mejora de su rendimiento académico, principalmente en la participación de las tareas en ambientes virtuales y en la asistencia, y en menor medida en las calificaciones [8, 30]. El *e-learning* es adecuado para una fácil y eficaz integración de la gamificación, así como también ayuda al docente a tener un sitio con todos los recursos académicos.

Moodle es una de las plataformas más populares de aprendizaje, diseñada para proporcionarles a educadores, administradores y estudiantes un sistema integrado único, robusto y seguro para crear ambientes de aprendizaje personalizados [53]. Tiene varias características y funcionalidades destinadas a facilitar el proceso de aprendizaje usando gamificación, entre ellas: avatares, visibilidad del progreso de los estudiantes, visualización del resultado de cuestionarios, niveles, retroalimentación, insignias, tabla de posiciones, actividades condicionales para restringir el acceso a contenido, etc. [15].

En este contexto, los autores Guo, Chai y Qian [37] refieren que para enseñar los conceptos de programación a los estudiantes de una manera más visual e interactiva que de la manera en que se hace, puede utilizarse como soporte al sistema convencional de *e-learning* una interfaz de aprendizaje basada en juegos.

Watson, Li y Lau [36] combinan juegos con tareas de programación tradicionales para hacerlas más entretenidas y atractivas para los alumnos, de tal manera que el estudiante puede aprender a programar completando las tareas. Proponen un método para la enseñanza de conceptos de programación mediante la generación de diferentes formas de retroalimentación correctiva con una interfaz de motivación basada en el juego para la visualización de los conceptos de codificación y la ejecución de código. A diferencia de los materiales de clase tradicionales, este concepto apoyado en las mecánicas de juego que ofrece la gamificación permite a los estudiantes desarrollar rápidamente una comprensión de los conceptos de codificación a través de un conjunto de ejemplos claros y visuales.

Por otra parte, el uso de esta estrategia ofrece una fuerte herramienta de motivación, donde los educandos pueden ver resultados inmediatos de la ejecución de código sin tener que dominar una gran cantidad de conocimientos. En este sentido, Anderson y McLoughlin [38] plantean que el escenario de juego también se puede personalizar, lo que permite mayor variedad de ejemplos y el aumento adicional de la motivación.

Como mencionan los investigadores Beltrán, Sánchez y Rico [84], la implementación del uso de la gamificación como apoyo en la educación y específicamente en la programación debe ser usada en un ambiente *e-learning* que promueva y ayude en el proceso de aprendizaje.

Arenas [63], para promover la motivación intrínseca, describe un modelo de aprendizaje de programación usando técnicas de gamificación. Se basa en el Gamification Framework D6 propuesto por Werbach [39]. Realiza un prototipo de una herramienta computacional que incluye algunos de los elementos del modelo usando WordPress CMS y el *plugin* BadgeOS para demostrar la factibilidad técnica de su implementación. Propone que el prototipo puede funcionar como base para un desarrollo posterior que incluya todos los elementos de juego y pueda ser utilizado en un salón de clases como herramienta de apoyo al aprendizaje [63].

Cacho y Rodrigo [58] proponen un diseño usando un conjunto de elementos y componentes de juego, para gamificar una plataforma de aprendizaje de código abierto, en particular Moodle, y crearon un prototipo sobre la base de este diseño. Plantean realizar la medición de la eficacia de su prototipo en un experimento con maestros y alumnos [58].

En este sentido, como conclusión al marco teórico y tecnológico que fundamenta la presente investigación, se partió de conceptos establecidos para cambiar una realidad que se hace difícil, y es el hecho de que los estudiantes de programación no tienen suficiente motivación, pues las clases pueden llegar a ser monótonas o muy densas, por lo que el cansancio puede vencerlos, por el no gusto de realizar tareas fuera del aula; se asume entonces la necesidad del uso de estrategias metodológicas y tecnológicas con la finalidad exclusiva de cambiar la visión sobre el proceso educativo como un proceso tradicional, a partir de la introducción de la gamificación en el desarrollo del aprendizaje, de combinar el placer con el aprender; esto se traduce en una renovación de las técnicas de motivación, en utilizar lo agradable de los juegos en un contexto menos deseado, como lo es el estudio diario.

La gamificación en un ambiente *e-learning* usando Moodle y otras herramientas complementarias, con sus aportes y la combinación de lo divertido y eficaz a la vez, es la respuesta más acertada que se ha encontrado pues ayuda en el incremento de la motivación por parte de los estudiantes, así como en la interacción profesor-estudiante.

Del contexto en el que se desarrolla la enseñanza de Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, esta se imparte en el primer semestre, en las carreras de Ingeniería en Informática, Matemática, Computación Gráfica y Diseño Industrial, en donde el alumno aprende los fundamentos de la programación y los conocimientos iniciales para diseñar algoritmos, que son implementados en programas simples de computador utilizando NetBeans o Eclipse como instrumentos para desarrollar *software* en lenguaje Java, con el apoyo de las distintas técnicas algorítmicas de la programación estructurada y de la programación orientada a objetos (atributos y métodos) [15].

El modelo de enseñanza de Programación I antes de nuestra intervención se apoyaba en el uso de la plataforma virtual Sakai, como un repositorio de material didáctico, para el envío de trabajos y prácticas, y para rendir pruebas o exámenes teóricos. Además, esta plataforma no fue inicialmente pensada para el aprendizaje de programación, por tanto, no permitía la codificación de programas en la misma plataforma, ni ejecutarlos, menos aún evaluarlos en forma automática, ya que no cuenta con módulos que permitan hacerlo, ni tampoco con herramientas para que el estudiante interactúe con los docentes y otros estudiantes en forma colaborativa.

La presente investigación tiene como objetivo principal presentar una propuesta de estrategias metodológicas y tecnológicas para la realización de tareas autónomas en la asignatura Programación I con el uso de la gamificación implementada en la plataforma virtual Moodle. Para esto se establecieron tres objetivos canalizados a partir de un aparato metodológico consistente, donde en primer lugar (objetivo 1) se analizó la metodología de enseñanza existente en la modalidad presencial de la asignatura Programación I, para identificar los principales factores que influyen en el proceso de aprendizaje de los estudiantes. Con este diagnóstico en el objetivo 2 se diseñó una nueva propuesta de estrategias y técnicas aplicando la gamificación para la realización de tareas autónomas en la asignatura Programación I. Este diseño se implementó en la plataforma Moodle que cuenta con varios módulos (Progress Bar, Ranking Block y VPL entre los principales) que fueron instalados y configurados. Por último, en el objetivo 3 se evalúa la actividad pedagógica del uso de la propuesta durante tres semestres en que el autor impartió la asignatura.

A continuación, se muestran las conclusiones a los resultados obtenidos en los tres objetivos de la investigación:

Conclusiones objetivo 1

Del primer objetivo de la investigación se arrojaron las siguientes conclusiones a partir del estudio cuantitativo:

- Existen problemas en el proceso de enseñanza-aprendizaje de Programación I entre los periodos octubre 2009-marzo 2010 y octubre 2014-marzo 2015 donde se hizo el estudio, debido al alto porcentaje de estudiantes que suspendieron (46,9 %), y muchos de ellos no vuelven a tomar la asignatura; mientras que el porcentaje de aprobados fue del 53,1 %, de ellos el 29,0 % no tomaron Programación II.
- A modo de resumen del análisis cuantitativo, después del análisis realizado por el investigador se encontró que las variables: género, tipo de provincia y tipo de colegio no influyeron en que el estudiante apruebe o no Programación I. Con respecto al semestre se muestra que hasta el periodo octubre 2012-marzo 2013, con excepción del semestre octubre 2011-marzo 2012, el porcentaje de estudiantes suspensos fue mayor que el de aprobados. Desde el semestre octubre 2012-marzo 2013 en adelante, la relación fue la contraria.
- La variable con mayor influencia en el aprendizaje de los estudiantes, que decide estadísticamente si los estudiantes aprueban o no la asignatura Programación I, es el docente que imparte la materia, específicamente la metodología y herramientas usadas, así como la exigencia de los docentes y que la dificultad de los exámenes no es equitativa. También se evidenció que en Programación II el profesorado tiene gran influencia en la cantidad de estudiantes aprobados y suspensos.

Las conclusiones obtenidas a partir del estudio cualitativo son las siguientes:

- En los resultados del análisis realizado a la encuesta a los estudiantes, se obtuvo que se sintieron conformes con la metodología tradicional aplicada en las clases, pero no así con los trabajos autónomos orientados por el docente. A los alumnos les gusta más la práctica que la teoría, pero no están satisfechos con las tareas que deben realizar fuera

del aula, viéndose la necesidad de proponer estrategias que faciliten y motiven el autoaprendizaje de Programación I.

- Los docentes señalan que se debe estandarizar el contenido y el alcance de la asignatura a través del cumplimiento del sílabo, así como el grado de exigencia en todos los cursos que reciben la materia en una misma carrera tiene que ser similar mediante un examen unificado. Además, se necesita reforzar los contenidos, guías, prácticas de laboratorio y los recursos para el proceso de enseñanza-aprendizaje y mejorar el acceso a Internet, laboratorios y el uso de la plataforma virtual.
- La motivación en los estudiantes se encuentra muy baja debido principalmente a la dificultad de la asignatura; a medida que la complejidad de los ejercicios aumenta, no pueden resolverlos fácilmente y les causa desmotivación, factor importante que se encuentra estrechamente vinculado con el compromiso del aprendizaje de Programación I y las calificaciones que obtienen los alumnos. Este factor debe ser elevado en todos los estudiantes que estén matriculados en un curso de Programación I para evitar que se retiren o suspendan.
- Aprender a programar es sustancial para todo profesional que desee ejercer las carreras de Ingenierías: Informática, Computación Gráfica, Matemática y Diseño Industrial. En tanto, el 94,2 % de los estudiantes expresaron que es muy importante aprender a programar, aunque es una materia compleja de aprender.
- Entre las soluciones citadas por los docentes y estudiantes para cambiar la situación existente se destaca la puesta en práctica de técnicas innovadoras y motivadoras que permitan comprender el contenido de la asignatura Programación I y adquirir en mejor forma las competencias de la misma. Es necesario aplicar una metodología con la que indistintamente de la complejidad de los ejercicios, los estudiantes se encuentren motivados para hacerlos y presten interés al aprendizaje de la programación.
- Con relación al uso de herramientas tecnológicas que se usan en el proceso de aprendizaje, se pudo observar que los estudiantes, por una parte, aceptan el uso de las tecnologías, pero por otro lado expresan que no son estandarizadas, no se usan correctamente, ni están generalizadas entre los docentes. Pero sí es un importante elemento que las tecnologías de la información y las comunicaciones sean aceptadas

por los estudiantes y docentes, pues avala las acciones realizadas por el investigador en los objetivos 2 y 3 del presente trabajo.

- La plataforma virtual Sakai como apoyo en el aprendizaje de la asignatura Programación I se usó siempre o frecuentemente en un 32,9 %, lo que evidencia que es una herramienta que debe ser mejorada y estandarizada en su uso.

En este sentido el autor de la presente investigación, sin pretender solventar todos los problemas detectados, propuso el uso de la gamificación en un ambiente *e-learning* para influir en la motivación y el compromiso con el desarrollo de las tareas autónomas, y como consecuencia mejorar las calificaciones de los alumnos, en la asignatura Programación I.

La propuesta incluye la utilización de la plataforma Moodle y herramientas complementarias que permiten asignar puntos, insignias, tableros de clasificación, evaluación automática y retroalimentación en línea para el desarrollo de ejercicios y que los estudiantes vean su calificación una vez subida la solución de los mismos. Los docentes de esta manera podrán controlar el proceso de aprendizaje de los estudiantes en forma más efectiva, por lo que el 83,4 % de los profesores están totalmente de acuerdo con que la gamificación en un ambiente *e-learning* puede ayudar a mejorar el proceso de aprendizaje de la asignatura Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

Conclusiones objetivo 2

Del segundo objetivo de la investigación se arrojaron las siguientes conclusiones:

- Se diseñaron estrategias de gamificación a partir de los objetivos y resultados del aprendizaje, y de las competencias que los estudiantes deben adquirir en Programación I. Además, se tomaron en cuenta los capítulos que posee la unidad Fundamentos de Java, desglosando los mismos en subcapítulos (contenidos a vencer) y clasificándolos según las dinámicas y mecánicas de la gamificación.
- Se detallaron las tareas autónomas a ser gamificadas en cada uno de los niveles (capítulos) y subniveles (subcapítulos), así como también su distribución en el tiempo y

los puntos de experiencia que se ganan en cada tarea, y la estrategia de transformación de los puntos de experiencia a la nota final del curso.

- Para el diseño de la asignatura gamificada se usó el Framework D6 propuesto por Werbach [39], posibilitando a las instituciones educativas diseñar el proceso docente de forma motivadora y flexible, y que los estudiantes aprendan de manera divertida. Así como también el modelo para la gamificación de actividades educativas planteado por González y Mora [6].
- Mediante el uso de dinámicas y mecánicas de la gamificación propuestas por Kapp [19], se logró promover el aprendizaje e incentivar a los alumnos por la programación en un ambiente tecnológico y virtual.
- La plataforma Moodle cuenta con diversos módulos, como asignación de puntos, insignias, tableros de clasificación, despliegue de recursos mediante condiciones (niveles), retroalimentación en línea para ejercicios Java (autoevaluación de los ejercicios), entre otros que permiten gamificar un curso [84].
- Se implementaron las estrategias de gamificación diseñadas en la plataforma virtual Moodle, creando la plataforma llamada Moodle gamificado, donde se instalaron y configuraron varios *plugins* complementarios destinados a gamificar Moodle, entre ellos Progress Bar, Ranking Block y Virtual Programming Lab.
- Para acceder a los niveles y subniveles progresivamente se usó la funcionalidad de Moodle para habilitar contenido de acuerdo a condiciones que pueden ser configuradas.
- La plataforma virtual Moodle permitió gamificar las tareas autónomas en un entorno virtual divertido para tener el dinamismo y la facilidad que necesitan los estudiantes en busca de desarrollar el autoaprendizaje, el interés y la motivación por la programación.
- Las dinámicas, mecánicas y componentes de juego que se implementaron en Moodle son:
 - **Dinámicas:** recompensas, estatus, competencia, logros, expresión e identidad propia, altruismo y ayuda, retroalimentación.
 - **Mecánicas y componentes:** puntos, insignias o medallas, tabla de clasificaciones, barra de progreso, niveles, avatares, retroalimentación automática, colaboración, misiones y retos, bienes virtuales, regalos o premios.

- El entorno virtual Moodle para el proceso de enseñanza-aprendizaje es una poderosa herramienta para ser integrada con las estrategias de gamificación. También ayuda al docente a tener un sitio con todos los recursos académicos y así aprovechar el tiempo en atención personalizada a los estudiantes con problemas en la asignatura, ya que las tareas autónomas son evaluadas automáticamente por la plataforma mediante pruebas unitarias.

Las estrategias de gamificación en Moodle fueron usadas durante tres semestres: abril-septiembre 2015 (semestre 1), octubre 2015-marzo 2016 (semestre 2) y abril-septiembre 2016 (semestre 3), donde impartió el autor la asignatura Programación I. Con un enfoque de mejora continua, el investigador fue adquiriendo la experiencia necesaria para realizar mejoras y cambios metodológicos y tecnológicos hasta llegar a la propuesta óptima que se evalúa en el objetivo 3.

Conclusiones objetivo 3

Del tercer objetivo de la investigación se tienen las siguientes conclusiones a partir del estudio cuantitativo:

En los tres semestres donde se usaron las estrategias de gamificación en Moodle gamificado, se observa un excelente cumplimiento de las tareas autónomas.

- En el primer semestre, el 90,0 % de los estudiantes lograron el nivel superior (Vida) y el avatar de archimago. El 50,0 % tuvo el máximo de insignias, es decir, completó satisfactoriamente todas las tareas en cada nivel. El 90,0 % de los estudiantes alcanzó más de 88,90 de puntos de experiencia, que equivalen a 8,89 en la nota de tareas autónomas.
- El 100 % de los estudiantes alcanzaron el máximo nivel (Vida), con el avatar de archimago en el semestre 2, y ganaron el total de insignias, es decir, realizaron correctamente todos los retos de hechicería en cada uno de los niveles. Todos los estudiantes tuvieron más de 1000 puntos de experiencia que les da una nota final en tareas autónomas de 10,0.

- En el semestre 3, la totalidad de los estudiantes alcanzaron el mayor de los niveles (Vida), con el avatar de archimago. El 81,3 % ganaron todas las insignias automáticas. El 90,6 % tuvieron más de 1000 puntos de experiencia que les da una nota final en tareas autónomas de 10,0.

De acuerdo con los resultados obtenidos en los tres semestres en la realización de las tareas fuera del aula, se concluye que se logró que los estudiantes manifestaran su interés genuino por la programación y que desarrollaran una motivación intrínseca que los incitara a ejercitar, estudiar y realizar las actividades y tareas planificadas, para de esta forma adquirir mayor nivel de conocimientos. El uso de las estrategias de gamificación y la plataforma Moodle gamificada influyó en la motivación de los alumnos por la ejecución de las tareas autónomas, y fue fundamental para el cumplimiento de las mismas.

Las notas finales de los alumnos mejoraron considerablemente con la aplicación de las estrategias de gamificación en Moodle.

- Los resultados obtenidos por los alumnos durante la implementación progresiva de las estrategias de gamificación aplicadas en Programación I por el investigador fueron satisfactorios alcanzando en el primer semestre abril-septiembre 2015 un porcentaje de aprobados del 35,3 %, en el semestre octubre 2015-marzo 2016 el 38,5 % aprobaron y en el último semestre implementado, abril-septiembre 2016, aprobó el 97,0 % de los estudiantes, alcanzando un alto progreso. Además, se identificó que se retiraban menos estudiantes, en el primer semestre se retiró el 41,2 %, en el segundo un 23,1 % y en el tercero solo un 3,0 %.
- En el semestre 2, el porcentaje de alumnos aprobados fue ligeramente mayor en la metodología gamificada (38,5 %) que en la normal (33,3 %), en el semestre 3 la diferencia fue mayor, el 97,0 % aprobaron usando la metodología gamificada y el 77,4 % la normal.
- El 3,0 % de los estudiantes se retiraron en el semestre 3 con la metodología gamificada y el 9,7 % con la tradicional; en el semestre 2, el 23,1 % se retiró con la metodología gamificada y el 50,0 % con la tradicional.

- Luego de implementar las mejoras prácticas adquiridas sobre el uso de las estrategias de gamificación en Moodle gamificado, los mejores resultados se tuvieron en el semestre 3 y se encontró una alternativa óptima al problema planteado en la investigación sobre la baja motivación de los alumnos por la realización de las tareas autónomas; el 97,0 % aprobaron, y solo el 3,0 % se retiró; de quienes terminaron la asignatura, el 100,0 % aprobó.

Las conclusiones conseguidas a partir del estudio cualitativo son las siguientes:

- El uso de Moodle gamificado implicó que el estudiante esté pendiente día a día de los ejercicios de cada uno de los niveles y subniveles. Obtener los puntos de experiencia, recompensas, insignias y las calificaciones es cuestión de esfuerzo y voluntad. En los semestres estudiados la valoración de muy bueno fue del 77,8 %, 100,0 % y 93,3 % respectivamente.
- Moodle gamificado como apoyo en el proceso de aprendizaje de Programación I fue mucho mejor y mejor comparado con Sakai, con porcentajes por encima del 85,0 %. Esto debido a que Moodle cuenta con varios *plugins* propios y externos que dinamizan mucho más el curso, como obtener en tiempo real la evaluación de las tareas autónomas de programación, así como obtener insignias, identificarse con un avatar, el tablero de clasificación y ver cómo se va evolucionando en cada nivel.
- Los estudiantes valoraron en general la metodología de gamificación en el ambiente *e-learning* como muy buena, con porcentajes superiores al 80,0 % en los tres semestres. También más del 90,0 % de los alumnos afirmaron que esta metodología fue mucho mejor y mejor que la metodología tradicional. Los alumnos lograron desarrollar sus conocimientos de programación en Java realizando los ejercicios de programación una y otra vez hasta que conseguían que funcionaran correctamente, no se daban por vencidos.
- El aprendizaje de programación fue divertido, así lo afirmaron la mayoría de los estudiantes quienes opinaron estar totalmente de acuerdo y de acuerdo, porque en esta nueva experiencia existía competencia entre los estudiantes y a la vez colaboración entre ellos. Todos los estudiantes realizaban las tareas autónomas sin que existiera una

orden del profesor, sino con el propósito de obtener mayor puntuación y premios para superar a sus compañeros.

- Las estrategias de gamificación y la plataforma gamificada influyeron en la motivación de los estudiantes, más del 90,0 % en los tres semestres calificaron su motivación como muy alta y alta. Además, en un porcentaje superior al 90,0 % de los casos manifestaron que su motivación fue mucho mejor y mejor que en el primer hemisemestre donde se usó la metodología tradicional. La nueva propuesta les permite realizar las actividades en cualquier momento y desde cualquier lugar sin necesidad de tener instalada alguna herramienta en el computador y en caso de tener dudas pueden consultarlas virtualmente por la plataforma a otros compañeros de estudio o al docente. También son motivados por los premios e insignias y otras dinámicas y mecánicas de juego que se aplican en el curso.
- Todas las dinámicas y mecánicas aplicadas en las estrategias de gamificación favorecen la motivación del aprendizaje de Programación I. Las dinámicas mejor evaluadas fueron «los logros» y «las recompensas». Entre las mecánicas de juego más valoradas estuvieron las «misiones o retos», «puntos», «desbloqueo de contenidos» y «retroalimentación».
- Prácticamente el 100 % de los estudiantes estuvieron motivados para aprender a programar y quedaron completamente satisfechos con las estrategias de gamificación aplicadas en Programación Mágica I; solo un estudiante del último semestre expresó que le eran indiferentes.

Como resumen de las conclusiones de los análisis cuantitativo y cualitativo se tiene lo siguiente:

- La plataforma virtual Moodle permitió gamificar las tareas autónomas y se logró tener dinamismo y facilidad para el autoaprendizaje; se influyó también en el interés y motivación de los estudiantes por la asignatura Programación I.
- Fue una experiencia novedosa para los estudiantes en el aprendizaje de programación Java, haciéndoles estar pendientes todo el tiempo de la realización de los ejercicios en cada nivel, ya que obtener los puntos, recompensas, insignias y buenas calificaciones necesitaba esfuerzo y voluntad.

- Tiende a ser una manera única y diferente (fuera de lo común) para aprender las bases de la programación, la cual ayudó a comprender todos los contenidos del curso, debido a que debían resolver los ejercicios (con evaluación automática y retroalimentación inmediata) y demás tareas autónomas, para obtener las recompensas, insignias y puntos, competir con los demás compañeros y poder ganarles al aumentar su puntuación en el tablero de clasificación.
- Las dinámicas y mecánicas de juego pudieron ser aplicadas en el proceso de aprendizaje, como actividades que tienen la intención de lograr ciertos objetivos de aprendizaje, influir en la motivación de los alumnos y en su participación en un agradable y divertido entorno de competencia con otros compañeros.
- El uso de estrategias de gamificación, la aplicación de mecánicas y dinámicas de juego y los procesos que pueden desarrollarse en un entorno *e-learning* influyeron en que los alumnos desarrollasen actividades autónomas y se motivasen por la asignatura Programación I, incidiendo positivamente en el rendimiento académico de los estudiantes.

Recordemos la hipótesis planteada en la presente investigación: «La implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje para desarrollar tareas autónomas influye sobre la motivación de los estudiantes y contribuye a mejorar su rendimiento académico, en la asignatura Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador». En las conclusiones de los análisis cuantitativo y cualitativo del objetivo 3 se determina que las estrategias de gamificación en Moodle gamificado influyen en la motivación de los estudiantes por la realización de tareas autónomas.

Para determinar estadísticamente que la propuesta permitió mejorar en el rendimiento académico de los estudiantes de Programación I, se procedió a realizar la comprobación de la hipótesis alternativa: «El rendimiento académico de los estudiantes en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador es significativamente diferente entre los estudiantes que emplean la propuesta y aquellos que no la siguen».

Los resultados obtenidos en las notas finales para los grupos que estudiaron con la metodología tradicional (grupo de control) y la metodología gamificada (grupo experimental) en el semestre octubre 2015-marzo 2016, mediante la prueba t-Student por ser una distribución normal, arrojaron que no existen diferencias entre ambos grupos pues $p > 0,05$.

Luego se evaluó la diferencia de las notas finales entre los grupos bajo ambas metodologías en el semestre abril-septiembre 2016, donde se usaron las mejoras prácticas encontradas por el investigador de acuerdo a las experiencias tenidas en los semestres anteriores. Para este caso, al no ser normal la distribución de la variable, se aplicó el test no paramétrico para muestras independientes U-Mann-Whitney. Al aplicar la prueba se constata la diferencia de los dos grupos ($p \leq 0,05$), por lo que se rechaza la hipótesis nula en favor de la hipótesis alternativa descrita anteriormente, es decir, podemos afirmar que existe una diferencia significativa entre las notas del grupo bajo la estrategia de aprendizaje normal y las notas del grupo bajo la estrategia gamificada.

Finalmente, al aplicar el test de Kruskal-Wallis y el test de medianas para k muestras a los semestres donde se usó la metodología y plataforma gamificadas, se verifica que existe una diferencia de grupos ($p \leq 0,05$), por lo tanto, los resultados de las notas finales de los estudiantes bajo esta metodología fueron diferentes en los tres semestres donde fue aplicada. Además, como el promedio de las notas en el semestre 2 con respecto al semestre 1 fueron mayores, e igualmente las notas del semestre 3 fueron mayores a las de los dos semestres anteriores, se comprueba que existió una mejoría paulatina en cada semestre donde se aplicó la propuesta.

Como conclusión final de la propuesta implementada, se expone que la implementación de estrategias de gamificación sobre una plataforma virtual de aprendizaje no solo influyó en la motivación de los estudiantes para desarrollar tareas autónomas, sino que también contribuyó a tener mejores calificaciones mejorando así su rendimiento académico, en la asignatura de Programación I de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

Limitaciones

Las condiciones o limitaciones que deben tenerse en cuenta para el funcionamiento adecuado de la propuesta como apoyo al aprendizaje de Programación I son las siguientes:

- La imposibilidad de tener una única solución tecnológica y metodológica con el apoyo de la gamificación, debido a que cada docente tiene libertad de cátedra y puede usar los recursos y herramientas que le parezcan más adecuados.
- La predisposición y compromiso del docente para aplicar la nueva propuesta ya que conlleva cambios metodológicos y el uso de la plataforma gamificada para la realización de las tareas fuera del aula; deben dedicar tiempo a supervisar el progreso de los estudiantes, revisar las actividades que realizan y responderles oportunamente, así como asegurarse de que los alumnos están alcanzando el nivel adecuado.
- El docente debe incentivar constantemente a los estudiantes en la realización de tareas autónomas y en su participación en la plataforma. El docente es el factor fundamental para lograr una motivación adecuada en sus estudiantes.
- El docente debe repensar los ejercicios prácticos de programación para que puedan ser evaluados automáticamente por la plataforma. Debe dedicar tiempo para ello.
- Todas las tareas autónomas se realizan en la plataforma, los estudiantes pueden tener la problemática de no poseer un computador o dispositivo móvil para acceder al curso o simplemente no contar con conexión a Internet para acceder al mismo.

En consecuencia, el diseño de la propuesta está condicionado y limitado por decisiones estratégicas, aspectos metodológicos y herramientas tecnológicas adoptadas previamente por la universidad y sus docentes.

La solución propuesta fue implementada bajo la cultura y realidad de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador, y si tomamos en cuenta las limitaciones anotadas anteriormente, los resultados obtenidos no necesariamente pueden ser generalizados en todos los casos.

LÍNEAS DE INVESTIGACIÓN FUTURAS

Luego de concluida la investigación, surgen las líneas de investigación que a continuación se muestran:

- En los resultados obtenidos en el objetivo 1 de la investigación se evidencia que el alto porcentaje de estudiantes suspensos en Programación I está influenciado por otros factores o aspectos relacionados con el nivel de exigencia de los docentes y la generalización de la metodología y las herramientas utilizadas en las clases.
- Desarrollo de estudios sobre la influencia de la motivación y otros aspectos psicosociales, como factores determinantes en el aprendizaje de programación y asignaturas afines en la UCE.
- Creación de actividades autónomas para la enseñanza de programación con fines específicos mediante el uso de los entornos virtuales, con el propósito de que los estudiantes puedan desarrollar su aprendizaje en temáticas vinculadas a la carrera que estudian.
- Determinar los problemas que son frecuentes en el modelo de enseñanza apoyado en las TIC y su influencia en el aprendizaje de Programación I en los estudiantes de la UCE.
- Aplicar las estrategias de gamificación y Moodle gamificado en todos los cursos donde se imparte Programación I y en asignaturas relacionadas (como Programación II o Estructura de datos), con el objetivo de unificar la enseñanza de programación en la universidad.
- Diseño y aplicación de estrategias de gamificación en las diversas asignaturas que se imparten en la UCE, haciendo uso de la plataforma Moodle gamificado.
- Desarrollo de planes de actualización de los docentes en el uso de las TIC y en nuevas metodologías que favorezcan el proceso de enseñanza-aprendizaje en la Universidad Central de Ecuador.

BIBLIOGRAFÍA

- [1] J. Lee, P. Ceyhan, W. Jordan-Cooley y W. Sung, «GREENIFY: A Real-World Action Game for Climate Change Education,» *Simulation and Gaming. Sage journals*, vol. 44, nº 2-3, pp. 185-189, 2013.
- [2] M. Oviedo y F. Ortiz Uribe, «La enseñanza de la programación,» Academias de Humanidades de la UPIICSA, México DF, 2012.
- [3] A. García, A. Lías, Á. Mahíllo y R. M. Pinero, Abandono de primer año en la Ingeniería Informática, Madrid: Universidad Politécnica de Madrid, 2014.
- [4] T. Jenkins, «The motivati3n of students of programming,» University of Kent, Canterbury, 2001.
- [5] A. Steinmann, B. Bosch y D. Aiassa, «Motivaci3n y Expectativas de los estudiantes por aprender Ciencias en la Universidad,» *RMIE*, vol. 18, nº 57, pp. 585-598, 2013.
- [6] C. Gonz3lez y A. Mora, «T3cnicas de gamificaci3n aplicadas en la docencia de Ingenier3a Inform3tica,» *Revista de Investigaci3n en Docencia Universitaria de la Inform3tica*, vol. 8, nº 1, pp. 29-40, 2014.
- [7] K. Huotari y J. Hamari, «Defining Gamification - A Service Marketing Perspective,» 16th International Academic Mind Trek Conference, Tampere, 2012.
- [8] C. I. Muntean, «Raising engagement in e-learning through gamification,» Babes-Bolyai University, Cluj-Napoca , 2013.
- [9] M. Salam3, J. Camps, C. Vallesp3, D. Vernet, X. Llor3, E. Bernad3, J. Garrell y X. Gonz3lez, «Iniciativas para motivar a los alumnos de Programaci3n,» Universitat Ramon Llull, Barcelona, 2001.
- [10] P. Garc3a y F. Angulo, «Revista Interuniversitaria de Formaci3n,» *Revista Interuniversitaria de Formaci3n del profesorado*, vol. 17, nº 1, p. 14, 2003.

- [11] C. Sosa y M. M. Rosero, «Modelo de Enseñanza y su relación con los procesos metacognitivos en programación de sistemas,» *Revista Educación en Ingeniería*, vol. 9, nº 17, p. 12, 2014.
- [12] A. Ferreira y G. A. Rojo, «Enseñanza de la programación,» *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, vol. 1, nº 1, pp. 1-8, 2006.
- [13] J. J. Kaasboll, «Exploring didactic models for programming,» University of Oslo, Oslo-Noruega, 1999.
- [14] S. Volet y C. P. Lund, «Metacognitive Instruction in Introductory Computer Programming: A Better Explanatory Construct for Performance than Traditional Factors,» *Journal of Educational Computing Research*, vol. 10, nº 4, pp. 297-329, 1994.
- [15] J. Beltrán, H. Sánchez y M. Rico, «Análisis cuantitativo y cualitativo del aprendizaje de Programación I en la Universidad Central del Ecuador,» Universidad Central, Quito, 2015.
- [16] C. Andrades Benítez, «ELE Internacional,» ELE Internacional, [En línea]. Available: <http://www.eleinternacional.com/gamificacion/>. [Último acceso: 12 06 2016].
- [17] J. Almazán, «Slideshare,» TicMind, 27 02 2015. [En línea]. Available: <https://es.slideshare.net/CarolEnLaNube/ticmind-gamificacion-salesforcedevgroupmadrid-primera-reunion>. [Último acceso: 21 06 2016].
- [18] G. Zichermann y C. Cunningham, «Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps,» Cambridge, MA: O'Reilly Media, Boston, 2011.
- [19] K. Kapp, «The Gamification of Learning and Instruction: Game-Based Methods and Strategies for Training and Education,» John Wiley & Sons, San Francisco, 2012.
- [20] J. Hamari y J. Koivisto, «Social motivations to use gamification: an empirical study of gamifying exercise,» Proceedings of the 21st European Conference on Information Systems, Utrecht, 2013.

- [21] J. Vassileva, «Motivating Participation in Social Computing Applications: A User Modeling Perspective,» *User Modeling and User-Adapted Interaction*, vol. 22, nº 1, pp. 177-201, 2012.
- [22] C. Albrecht, *The game of happiness. Gamification of positive activity interventions*, Maastricht University: Maastricht, 2012.
- [23] R. Tejeiro, M. Pelegrina del Río y J. Gómez, «Efectos psicosociales de los videojuegos,» *Comunicación*, vol. 1, nº 7, pp. 235 -250, 2009.
- [24] J. Díaz y Y. Troyano, «El Potencial de la Gamificación Aplicado al Ámbito Educativo,» pp. 1-9, 2013.
- [25] K. Werbach y D. Hunter, *For the Win: How Game Thinking Can Revolutionize Your Business*, Wharton: Digital Press, 2012.
- [26] F. Teixes, *Gamificación: fundamentos y aplicaciones*, Barcelona: Editorial UOC, 2014.
- [27] S. Padilla, F. Halley y J. Chantler, «Improving Product Browsing whilst Engaging Users,» *Digital Engagement 11*, pp. 15-17, 2011.
- [28] Csikszentmihalyi, «Gamificación, el negocio de la diversión,» *BBVA Innovation Edge*, vol. 3, nº 1, pp. 1- 65, 2012.
- [29] M. Ibañez, A. Di Seis y C. Delgado, «Gamification for engaging Computer Science Students in Learning Activities,» *IEEE Transactions on learning technologies*, vol. 7, nº 3, pp. 291-301, 2014.
- [30] J. Simoes, R. Díaz y A. Fernández, «A social gamification framework for a K-6 learning platform,» *Computer in Human Behavior*, vol. 29, nº 2, pp. 345-353, 2013.
- [31] E. Dale, *Audio-visual Methods in Teaching*, 15ta ed., New York: Dryden Press, 2016.
- [32] I. Fernández, «Juego serio: gamificación y aprendizaje,» *Comunicación y Pedagogía*, pp.

43-48, 2015.

- [33] A. Dávila, «Efectos de algunas tecnologías educativas digitales sobre el rendimiento académico en matemáticas,» *Compendium*, vol. 10, n° 8, pp. 21-36, 2014.
- [34] W. Cook, «Training Today,» 2012. [En línea]. Available: <http://www.trainingmag.com/content/>. [Último acceso: 15 12 2016].
- [35] J. Bennedsen y M. Caspersen, «Failure Rates in Introductory Programming, ACM,» *SIGCSE Bulletin*, vol. 39, n° 2, pp. 32-36, 2014.
- [36] C. Watson, F. Li y R. Lau, «Learning Programming Languages through Corrective Feedback and Concept Visualisation,» *Lecture Notes in Computer Science*, vol. 7048, pp. 11-20, 2011.
- [37] M. Guo, T. Chai y K. Qian, « Design of Online Runtime and Testing Environment for Instant Java Programming Assessment,» *Proc. Information Technology*, pp. 1102-1106, 2010.
- [38] E. Anderson y L. McLoughlin, *Critters in the Classroom: A 3D Computer-Game-Like Tool for Teaching Programming to Computer Animation Students*, Proc., New York: ACM SIGGRAPH, 2014.
- [39] K. Werbach, «Gamification Design Framework,» UVIC, Pennsylvania, 2012.
- [40] M. Guerrero, D. Gramoú y J. C. Caiza, «Revisión de herramientas de apoyo en el proceso de enseñanza-aprendizaje de Programación,» *Revista Politécnica*, vol. 35, n° 1, pp. 1-9, 2015.
- [41] J. M. A. Díaz González, *El uso de las plataformas de enseñanza virtual para impartir la asignaturas jurídicas*, Sevilla: Universidad de Sevilla, 2012, p. 12.
- [42] M. d. R. Gerrero, *Metodologías activas y aprendizaje por descubrimiento. Las TIC y la Educación*, Bogotá: DidactyTab, 2014.

- [43] A. Fernández, J. Cristóbal, A. Navarro y F. Pampillón, «El campus virtual en la universidad Complutense de Madrid,» Universidad Complutense de Madrid, Madrid, 2008.
- [44] A. Bartolomé, «Blended Learning. Conceptos básicos,» *Revista de Medios y Educación*, vol. 23, nº 3, pp. 7-20, 2004.
- [45] J. Adell, «World Wide Web: Un Sistema Hipermedia Distribuido Para La Docencia Universitaria,» Ediciones Alfar, España, 2002.
- [46] M. Ally, F. Lin, R. McGreal y B. Woo, *An Intelligent Agent for Adapting and Delivering Electronic Course Materials to Mobile Learners*, Athabasca : Athabasca University, 2014.
- [47] A. Bates, *T-learning Study. A study into TV-based interactive learning to the home, Final Report*, Reino Unido: pjb Associates, 2003.
- [48] M. Morfi, «U-learning aprendizaje donde quieras que estés,» *Learningreview*, 2011. [En línea]. Available: <http://www.learningreview.es/plataformas-de-e-learning/1971-u-learning-aprendizaje-donde-quiera-que-estes>. [Último acceso: 23 septiembre 2016].
- [49] M. Villar, «EVO I.T.,» 02 02 2011. [En línea]. Available: <http://blog.evoit.com/2011/03/nuevas-tendencias-p-learning-y-u-learning/>. [Último acceso: 14 07 2016].
- [50] L. Massimino, «Un nuevo Paradigma Educativo: del e-learning al cloud learning (c-Learning). El Conocimiento en la Nube,» *Marketing-diseño*, 11 octubre 2011. [En línea]. Available: <http://www.lauramassimino.com/featured/un-nuevo-paradigma-educativo-del-e-learning-al-cloud-learning-c-learning-el-conocimiento-en-la-nube>. [Último acceso: 23 septiembre 2016].
- [51] I. Calderón, «Desarrollo de una Metodología para la Creación de Objetos de Aprendizaje en el Modelo B-Learning y Aplicación en una Materia de la Escuela de Ingeniería de

Sistemas,» Riobamba-Ecuador, 2012.

- [52] J. Silva Quiroz y S. Bogoña, Diseño y moderación de entornos virtuales de aprendizaje (EVA), Santiago de Chile: Editorial UOC, 2015.
- [53] Moodle, «Moodle,» Moodle, [En línea]. Available: https://docs.moodle.org/all/es/Acerca_de_Moodle. [Último acceso: 21 12 2016].
- [54] Intef, «Moodle, Plataforma de aprendizaje,» Instituto Nacional de Educación, Formación Profesional Universitaria, [En línea]. Available: http://www.ite.educacion.es/formacion/materiales/184/cd/M1_introduccion/pedaggicamente_hablando.html. [Último acceso: 11 9 2016].
- [55] Moodle, «Moodle,» Moodle, 07 10 2006. [En línea]. Available: <https://moodle.com/>. [Último acceso: 5 15 2016].
- [56] Moodle, «Moodle,» Moodle, 10 09 2015. [En línea]. Available: <https://docs.moodle.org/>. [Último acceso: 2 12 2016].
- [57] Moodle, «Moodle,» Moodle, 21 09 2014. [En línea]. Available: <https://docs.moodle.org/all/es/Filosof%C3%ADa>. [Último acceso: 21 12 2016].
- [58] G. P. Cacho y M. T. Rodrigo, «Gamifying Moodle: Integrating Game Elements Into an Open-Source Learning Management System,» pp. 22-29, 2014.
- [59] D. Thiébaud, «Automatic evaluation of computer programs using Moodle's virtual programming lab (VPL) plug-in,» *Journal of Computing Sciences in Colleges*, pp. 145-151, 2015.
- [60] J. A. Villalobos, «Proyecto CUIP2 – Una solución integral al problema de enseñar y aprender a programar,» Universidad de Los Andes, Bogotá, 2009.
- [61] E. Larsen, «Game of Exams. Creating a gamified learning platform to teach university curriculum in programming,» University of Oslo, Oslo, 2013.

- [62] L. Martínez y M. Duarte, «Exploring Gamification to Teach Computer Science to Non-computer Science Related Major,» de *SIGCSE '16 Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, Memphis Tennessee, 2016.
- [63] R. J. Arenas, «Modelo para la Motivación del Aprendizaje de la Programación utilizando Gamification,» Pontificia Universidad Javeriana, Bogotá, 2014.
- [64] K. Behnke, *GAMIFICATION IN INTRODUCTORY COMPUTER SCIENCE*, Boulder Colorado: ATLAS Institute, 2015.
- [65] F. A. Briones, J. A. Rivas, A. E. Viteri y L. M. Estrada, «La Educación en el Ecuador, Situación y Propuesta del Sistema de Vouchers,» Escuela Superior Politécnica del Litoral, Guayaquil, 2011.
- [66] R. Fernández, E. Carballos y M. Delavaut, «Un modelo de autoaprendizaje con integración de las TIC y los métodos de gestión del conocimiento,» *RIED*, vol. 11, nº 2, pp. 137-149, 2008.
- [67] G. Meneses Benítez, «Las TICs en la Universidad,» Universitat Rovira I Virgili, Cataluña, 2007.
- [68] A. Bustos, «Estrategias didácticas para el uso de TIC's en la docencia universitaria presencial. Un manual para los ciudadanos del Ágora,» Pontificia Universidad Católica de Valparaíso, Valparaíso, 2011.
- [69] J. F. Cuenca Saavedra, «Uso del Internet como herramienta en el proceso de aprendizaje en los estudiantes de Comunicación Social,» Universidad Central del Ecuador, Quito, 2013.
- [70] P. F. Negrete, «TIC y Educación,» PUCE, Quito, 2013.
- [71] A. Laurencio y P. Farfán, «Experiencia de integración de las TIC en el proyecto educativo de la UPS del Ecuador,» Universidad Politécnica Salesiana, Quito, 2013.

- [72] M. A. Poaquiiza, «El uso de LMS y su aplicación en el PEA de la Carrera de Docencia en Informática de la Facultad de Ciencias Humanas y de la Educación de la Universidad Técnica de Ambato de la ciudad de Ambato provincia de Tungurahua,» Universidad Técnica de Ambato, Ambato, 2014.
- [73] C. Marcillo y M. Marcillo, «Entornos virtuales de aprendizaje: Diagnóstico para la implementación de entornos virtuales basados en software libre en la carrera de Ingeniería en sistemas computacionales de la Universidad Estatal del Sur de Manabí, Ecuador,» de *A Gestão do Conhecimento e os Novos Modelos de Universidade. XIV Coloquio Internacional de Gestao Universiária–CIGU*, Florianópolis – Santa Catarina, 2014.
- [74] G. Samaniego, L. Marqués y M. Gisbert, «El profesorado universitario y el uso de Entornos Virtuales de aprendizaje,» *Campus Virtuales*, vol. 4, n° 2, pp. 50-58, 2015.
- [75] Asamblea Nacional Constituyente, «Constitución Política de la República del Ecuador 2008,» Monsalve, Quito, 2008.
- [76] Asamblea Nacional de Ecuador, « Ley Orgánica de Educación Superior,» Editora Nacional, Quito, 2010.
- [77] R. D. Moreno, «El presente Modelo Educativo fue aprobado por el Honorable Consejo Universitario en sesión ordinaria,» UCE, Quito, 2013.
- [78] S. Cadena y R. A. Enríquez, «Diseño e implementación de una Universidad Abierta caso Universidad Central del Ecuador,» UCE, Quito, 2015.
- [79] Vicerrectorado Académico De Investigación Y Posgrado, «Sílabo,» Universidad Central, Quito, 2016.
- [80] Netbeans, «Netbeans,» Netbeans, [En línea]. Available: <https://netbeans.org/features/platform/index.html>. [Último acceso: 12 1 2017].
- [81] T. M. Cantor, «Importancia de las tareas dentro del entorno escolar,» Uniminuto, Bogotá

D.C, 2014.

- [82] M. Gallegos, «La importancia del autoaprendizaje en la escuela,» *Colegio Guillermo León Valencia*, p. 1, 10 8 2012.
- [83] J. Santrock, *Psicología de la educación*, México: Mc Graw-Hil, 2002.
- [84] J. Beltrán, H. Sánchez y M. Rico, «Incrementar la motivación en el aprendizaje de Fundamentos de Programación Java usando Moodle Gamificado,» de *11ª Conferencia Ibérica de Sistemas y Tecnologías de Información*, Gran Canaria - España, 2016.
- [85] M. González, «Los estilos de enseñanza y aprendizaje como soporte de la actividad docente,» *Estilos de Aprendizaje*, vol. 11, nº 11, p. 23, 2013.
- [86] A. Yacob y M. Yazid, «Assessing level of motivation in learning programming among engineering students,» de *The International Conference on Informatics and Applications (ICIA2012)*, Malaysia, 2012.
- [87] A. Chevtchenko, «Gamified Education. Introducing Game Elements into the School Environment to Enhance Student Motivation and Performance,» Erasmus University Rotterdam, Rotterdam, 2013.
- [88] J. Beltrán, «Aprendizaje Gamificado,» 4 Marzo 2016. [En línea]. Available: <http://www.aprendizajegamificado.com>. [Último acceso: 14 Enero 2017].
- [89] Moodle, «Moodle,» Moodle, [En línea]. Available: https://moodle.org/plugins/block_progress. [Último acceso: 2017 12 3].
- [90] W. Mano, «Moodle,» Moodle, [En línea]. Available: https://moodle.org/plugins/block_ranking. [Último acceso: 14 1 2017].
- [91] VPL, «Virtual Programming Lab,» VPL, [En línea]. Available: <http://vpl.dis.ulpgc.es/>. [Último acceso: 14 01 2017].

- [92] Scrapfy, «Scrapfy,» Scrapfy, [En línea]. Available: <https://scrapfy.io/>. [Último acceso: 25 01 2017].
- [93] G. Bertalan y M. Rumler, «Moodle,» Moodle, [En línea]. Available: https://moodle.org/plugins/view.php?plugin=qtype_javaunittest. [Último acceso: 12 01 2017].
- [94] J. Beltrán, «Javaunittest plugin de Moodle como apoyo en el aprendizaje de Fundamentos de Programación en Java,» *Sarance*, nº 35, pp. 63-71, 2016.
- [95] C. Mas y M. Medinas, «Motivaciones por el estudio en universitarios,» *Anales de psicología*, vol. 23, nº 1, pp. 17-24, 2007.
- [96] G. Kyriakova, N. Angelova y L. Yordanova, «Gamification in education,» Trakia University, Stara Zagora, 2013.
- [97] Moodle, «Moodle,» Moodle, 20 08 2016. [En línea]. Available: https://docs.moodle.org/all/es/29/Instalaci%C3%B3n_de_Moodle. [Último acceso: 3 1 2017].
- [98] Moodle, «Moodle 2.9,» Moodle 2.9, [En línea]. Available: https://docs.moodle.org/all/es/Notas_de_Moodle_2.9#Requisitos_del_servidor. [Último acceso: 14 02 2017].

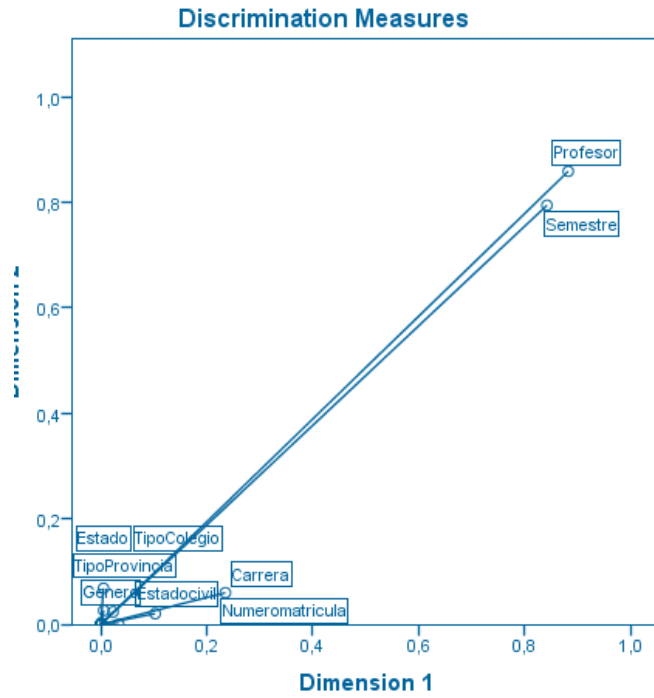
ANEXOS

Anexo 1. Resultados del análisis de componentes principales

Se presentan los resultados obtenidos en el SPSS v23, del análisis de componentes principales, confirmando los resultados obtenidos en el estudio cuantitativo del objetivo 1. Tanto en la tabla como en la figura que se muestran a continuación, podemos ver como las variables «semestres» y «profesores» son las de mayor influencia.

Discrimination Measures

	Dimension		Mean
	1	2	
Carrera	,235	,060	,148
Semestre	,843	,794	,818
Genero	,001	,003	,002
Estadocivil	,033	,000	,017
TipoProvincia	,006	,026	,016
TipoColegio	,023	,024	,023
Numeromatricula	,103	,020	,061
Estado	,005	,068	,037
Profesor	,883	,859	,871
Active Total	2,130	1,855	1,993
% of Variance	23,672	20,608	22,140



Anexo 2. Encuesta a docentes

Objetivo: analizar las estrategias de aprendizaje de Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

CUESTIONARIO

P1. Las siguientes herramientas tecnológicas que usted utiliza, ¿facilitan el aprendizaje de Programación ?

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

1= Si cumple

2= Cumple parcialmente, con esfuerzo

3= Cumple mínimamente

	1	2	3	
Plataforma virtual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV5I
Audio	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV6I
Video	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV7I
Internet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV8I
Software especializado	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV9I
Redes sociales	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV10I
Correo electrónico	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV11I

P2. Los siguientes métodos de enseñanza que Usted utiliza, ¿facilitan el aprendizaje de Programación ?

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

1= Si cumple

2= Cumple parcialmente, con esfuerzo

3= Cumple mínimamente

	1	2	3	
Exposición oral (clase magistral)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV12I
Ejercicios dentro de clase	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV13I
Conferencias (profesores invitados)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV14I
Prácticas de laboratorio	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV15I
Trabajos de investigación	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV16I
Talleres	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV17I
Exposición audiovisual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV18I
Ejercicios fuera del aula	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV19I
Lecturas obligatorias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV20I
Prácticas de campo/Resolución de problemas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV21I
Desarrollo de proyectos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV22I
Consulta en internet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV23I
Otras	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV24I

P3. ¿Cuáles considera son los principales problemas que enfrenta el estudiante en el aprendizaje de Programación ?

Escriba texto, sea tan amplio como desee

IV25I

P4. ¿Qué aspectos considera usted que faltan dentro de la metodología aplicada por la Facultad para un adecuado aprendizaje de Programación ?

Escriba texto, sea tan amplio como desee

IV26I

P5. ¿Cómo calificaría usted la metodología usada en el proceso de enseñanza - aprendizaje, de acuerdo a las competencias de Programación ?

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

1= Muy Bueno

2= Bueno

3= Regular

4= Malo

5= Muy Malo

	1	2	3	4	5	
Analizar y diseñar algoritmos sencillos utilizando fundamentos y técnicas de programación estructurada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV27I
Construir programas sencillos en lenguaje Java	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV29I

P6. ¿Considera que la motivación es un factor importante en el aprendizaje de Programación?

Seleccione sólo una opción

Totalmente de acuerdo	IV30I	<input type="radio"/>	1		En desacuerdo	IV30I	<input type="radio"/>	4
De acuerdo	IV30I	<input type="radio"/>	2		Totalmente en desacuerdo	IV30I	<input type="radio"/>	5
Indiferente	IV30I	<input type="radio"/>	3					

P7. ¿Considera que Gamificación usado en un ambiente eLearning puede ayudar en el proceso de enseñanza - aprendizaje de Programación ?

Gamificación. La Gamificación en la educación y la formación es la aplicación de recursos de los juegos (diseño, dinámicas, elementos, etc.) para modificar los comportamientos de los alumnos para que el resultado de la acción educativa o formativa se efectiva para ellos, para el impartido y para el promotor de ésta.

Totalmente de acuerdo	IV31I	<input type="radio"/>	1		En desacuerdo	IV31I	<input type="radio"/>	4
De acuerdo	IV31I	<input type="radio"/>	2		Totalmente en desacuerdo	IV31I	<input type="radio"/>	5
Indiferente	IV31I	<input type="radio"/>	3					

Anexo 3. Encuesta a estudiantes

El objetivo de la presente encuesta es analizar las estrategias de aprendizaje de Programación I en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

CUESTIONARIO

Generales

¡Escriba aquí las instrucciones para el llenado de esta Sección!

<p>Género <i>Seleccione sólo una opción</i></p> <p>Femenino <input type="radio"/> 1 V5 </p> <p>Masculino <input type="radio"/> 2</p>		<p>Edad? <i>Por favor escriba un número</i></p> <p># <input style="width: 80px;" type="text"/> V6 </p>											
<p>Colegio donde se graduó? <i>Seleccione sólo una opción</i></p> <p>Fiscal / Público <input type="radio"/> 1 V7 </p> <p>Particular / Privado <input type="radio"/> 2</p> <p>Municipal <input type="radio"/> 3 V7 </p>													
<p>Provincia? <i>Sea preciso en su respuesta</i></p> <p><input style="width: 100%;" type="text"/> V8 </p>		<p>Ciudad? <i>Sea preciso en su respuesta</i></p> <p><input style="width: 100%;" type="text"/> V9 </p>											
<p>Carrera? <i>Sea preciso en su respuesta</i></p> <p><input style="width: 100%;" type="text"/> V10 </p>		<p>Curso? <i>Sea preciso en su respuesta</i></p> <p><input style="width: 100%;" type="text"/> V11 </p>											
<p>Paralelo? <i>Sea preciso en su respuesta</i></p> <p><input style="width: 100%;" type="text"/> V12 </p>		<p>¿Fue la Universidad Central su primera opción para estudiar? <i>Seleccione sólo una opción</i></p> <p>Sí <input type="radio"/> 1 V13 </p> <p>No <input type="radio"/> 2</p>											
<p>¿La Carrera que estudia, fue su primera opción? <i>Seleccione sólo una opción</i></p> <p>Sí <input type="radio"/> 1 V14 </p> <p>No <input type="radio"/> 2 V14 </p>													
<p>Si su respuesta a la pregunta anterior es "No", ¿qué área del Conocimiento le gustaría estudiar? <i>Seleccione sólo una opción</i></p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">Programas generales <input type="radio"/> 1 V15 </td> <td style="width: 50%;">Educación <input type="radio"/> 6 V15 </td> </tr> <tr> <td>Humanidades y artes <input type="radio"/> 2</td> <td>Ciencias sociales, educación comercial y derecho <input type="radio"/> 7</td> </tr> <tr> <td>Ciencias <input type="radio"/> 3</td> <td>Ingeniería, industria y construcción <input type="radio"/> 8</td> </tr> <tr> <td>Agricultura <input type="radio"/> 4</td> <td>Salud y servicios sociales <input type="radio"/> 9</td> </tr> <tr> <td>Servicios <input type="radio"/> 5</td> <td>Sectores desconocidos no especificados <input type="radio"/> 10</td> </tr> </table>				Programas generales <input type="radio"/> 1 V15 	Educación <input type="radio"/> 6 V15 	Humanidades y artes <input type="radio"/> 2	Ciencias sociales, educación comercial y derecho <input type="radio"/> 7	Ciencias <input type="radio"/> 3	Ingeniería, industria y construcción <input type="radio"/> 8	Agricultura <input type="radio"/> 4	Salud y servicios sociales <input type="radio"/> 9	Servicios <input type="radio"/> 5	Sectores desconocidos no especificados <input type="radio"/> 10
Programas generales <input type="radio"/> 1 V15 	Educación <input type="radio"/> 6 V15 												
Humanidades y artes <input type="radio"/> 2	Ciencias sociales, educación comercial y derecho <input type="radio"/> 7												
Ciencias <input type="radio"/> 3	Ingeniería, industria y construcción <input type="radio"/> 8												
Agricultura <input type="radio"/> 4	Salud y servicios sociales <input type="radio"/> 9												
Servicios <input type="radio"/> 5	Sectores desconocidos no especificados <input type="radio"/> 10												
<p>¿Ha cursado alguna asignatura previa de Programación en el Colegio? <i>Seleccione sólo una opción</i></p> <p>Sí <input type="radio"/> 1 V16 </p> <p>No <input type="radio"/> 2 V16 </p>													
<p>Número de matrícula? <i>Seleccione sólo una opción</i></p> <p>Primera <input type="radio"/> 1 V17 </p> <p>Segunda <input type="radio"/> 2</p> <p>Tercera <input type="radio"/> 3 V17 </p>													
<p>Pasó el curso? <i>Seleccione sólo una opción</i></p> <p>Sí <input type="radio"/> 1 V18 </p> <p>No <input type="radio"/> 2 V18 </p>													

P7. ¿Cuáles son los aspectos metodológicos que usted considera favorecen más en el aprendizaje de Programación I?

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

- 1= Totalmente de acuerdo 2= De acuerdo
3= Indiferente 4= En desacuerdo
5= Totalmente en desacuerdo

	1	2	3	4	5	
Exposición oral (clase magistral)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV39I
Ejercicios dentro de clase	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV40I
Conferencias (profesores invitados)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV41I
Prácticas de laboratorio	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV42I
Trabajos de investigación	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV43I
Talleres	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV44I
Exposición audiovisual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV45I
Ejercicios fuera del aula	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV46I
Lecturas obligatorias	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV47I
Prácticas de campo/Resolución de problemas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV48I
Desarrollo de proyectos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV49I
Consulta en internet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV50I
Otras	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV51I

P8. Califique la contribución de la metodología usada para el aprendizaje de Programación I en las destrezas listadas en la siguiente tabla

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

- 1= Muy buena 2= Buena
3= Regular 4= Mala
5= Muy Mala

	1	2	3	4	5	
Analizar y diseñar algoritmos sencillos utilizando fundamentos y técnicas de programación estructurada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV52I
Construir programas sencillos en lenguaje Java	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV55I

P9. Considere en general que la metodología utilizada para el aprendizaje de Programación I es:

Seleccione sólo una opción

	IV56I		IV56I
Muy buena	<input type="radio"/> 1	Mala	<input type="radio"/> 4
Buena	<input type="radio"/> 2	Muy mala	<input type="radio"/> 5
Regular	<input type="radio"/> 3		

P10. ¿Cuáles son las formas de evaluar que usted considera favorecen más en el aprendizaje de Programación I?

A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala

- 1= Totalmente de acuerdo 2= De acuerdo
3= Indiferente 4= En desacuerdo
5= Totalmente en desacuerdo

	1	2	3	4	5	
Pruebas parciales	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV58I
Trabajos y tareas fuera del aula	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV59I
Participación en clase	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV60I
Informes de laboratorio	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV61I
Examen final	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV62I
Asistencia a prácticas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV63I
Trabajo en equipo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV64I
Otras	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV65I

P11. ¿Considera que la motivación es un factor importante en el aprendizaje de Programación I?

Seleccione sólo una opción

	IV66
Totalmente de acuerdo	<input type="radio"/> 1
De acuerdo	<input type="radio"/> 2
Indiferente	<input type="radio"/> 3
En desacuerdo	<input type="radio"/> 4
Totalmente en desacuerdo	<input type="radio"/> 5

P12. Califique su grado de motivación en el semestre en el aprendizaje de Programación I

Seleccione sólo una opción

	IV66
Muy alto	<input type="radio"/> 1
Alto	<input type="radio"/> 2
Regular	<input type="radio"/> 3
Bajo	<input type="radio"/> 4
Muy bajo	<input type="radio"/> 5

P13. ¿Considera que ha mejorado su nivel de conocimientos de Programación en este semestre?

Seleccione sólo una opción

	IV68I		IV68I
Mucho	<input type="radio"/> 1	Muy poco	<input type="radio"/> 4
Medianamente	<input type="radio"/> 2	Nada	<input type="radio"/> 5
Poco	<input type="radio"/> 3		

P14. Califique su satisfacción general del curso de Programación I

Seleccione sólo una opción

	IV69I		IV69I
Muy satisfecho	<input type="radio"/> 1	Insatisfecho	<input type="radio"/> 4
Satisfecho	<input type="radio"/> 2	Muy insatisfecho	<input type="radio"/> 5
Indiferente	<input type="radio"/> 3		

Anexo 4. Encuesta a estudiantes sobre las estrategias de gamificación y la plataforma virtual gamificada usadas en el aprendizaje de Programación I

El objetivo de la presente encuesta es analizar las estrategias de gamificación y la plataforma virtual gamificada usadas como apoyo al aprendizaje de Programación I, en la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central de Ecuador.

CUESTIONARIO

<p>P1. ¿Considera que el uso de la plataforma virtual Moodle Gamificado utilizado en el aprendizaje de Programación I fue?</p> <p style="text-align: center;"><i>Seleccione sólo una opción</i></p> <p style="text-align: right;">IV5I</p> <p>Muy bueno <input type="radio"/> 1</p> <p>Bueno <input type="radio"/> 2</p> <p>Regular <input type="radio"/> 3</p> <p>Malo <input type="radio"/> 4</p> <p>Muy malo <input type="radio"/> 5</p>	<p>P2. Considera que la plataforma virtual Moodle Gamificado comparado con la plataforma virtual Sakai, como apoyo en el aprendizaje de Programación I fue:</p> <p style="text-align: center;"><i>Seleccione sólo una opción</i></p> <p style="text-align: right;">IV6I</p> <p>Mucho mejor <input type="radio"/> 1</p> <p>Mejor <input type="radio"/> 2</p> <p>Igual <input type="radio"/> 3</p> <p>Peor <input type="radio"/> 4</p> <p>Mucho peor <input type="radio"/> 5</p>																																																	
<p>P3. Considera en general que la metodología "Gamificación en un ambiente e-Learning" usando utilizado en el segundo semestre como apoyo en el aprendizaje de Programación I fue:</p> <p style="text-align: center;"><i>Seleccione sólo una opción</i></p> <p style="text-align: right;">IV7I</p> <p>Muy buena <input type="radio"/> 1</p> <p>Buena <input type="radio"/> 2</p> <p>Regular <input type="radio"/> 3</p> <p>Mala <input type="radio"/> 4</p> <p>Muy mala <input type="radio"/> 5</p>	<p>P4. Considera que la metodología "Gamificación en un ambiente e-Learning" usado en el segundo semestre en el aprendizaje de Programación I, comparado con la metodología normal usada en el primer semestre fue:</p> <p style="text-align: center;"><i>Seleccione sólo una opción</i></p> <p style="text-align: right;">IV8I</p> <p>Mucho mejor <input type="radio"/> 1</p> <p>Mejor <input type="radio"/> 2</p> <p>Igual <input type="radio"/> 3</p> <p>Peor <input type="radio"/> 4</p> <p>Mucho peor <input type="radio"/> 5</p>																																																	
<p>P5. ¿Considera que aprender a programar usando la plataforma virtual y metodología gamificada fue divertido?</p> <p style="text-align: center;"><i>Seleccione sólo una opción</i></p>																																																		
<p style="text-align: right;">IV9I</p> <p>Totalmente de acuerdo <input type="radio"/> 1</p> <p>De acuerdo <input type="radio"/> 2</p> <p>Indiferente <input type="radio"/> 3</p>	<p style="text-align: right;">IV9I</p> <p>En desacuerdo <input type="radio"/> 4</p> <p>Totalmente en desacuerdo <input type="radio"/> 5</p>																																																	
<p>P6. Considera usted que la plataforma virtual Gamificada y la metodología utilizadas en el segundo semestre como apoyo en el aprendizaje de Programación I permiten:</p> <p style="text-align: center;"><i>A continuación se muestra una tabla con ciertos ítems en las filas y una escala en las columnas. Por favor valore cada uno de estos enunciados según la escala</i></p> <p>1= Totalmente de acuerdo 2= De acuerdo</p> <p>3= Indiferente 4= En desacuerdo</p> <p>5= Totalmente en desacuerdo</p>																																																		
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 70%;"></th> <th style="text-align: center;">1</th> <th style="text-align: center;">2</th> <th style="text-align: center;">3</th> <th style="text-align: center;">4</th> <th style="text-align: center;">5</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td>Motivan al estudiante</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV10I</td> </tr> <tr> <td>Hacen más fácil el aprendizaje</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV11I</td> </tr> <tr> <td>Permiten mantener el interés en clases</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV12I</td> </tr> <tr> <td>Favorecen la práctica</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV13I</td> </tr> <tr> <td>Optimizan el aprendizaje</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV14I</td> </tr> <tr> <td>Motivan el autoaprendizaje</td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: center;"><input type="radio"/></td> <td style="text-align: right;">IV15I</td> </tr> </tbody> </table>			1	2	3	4	5		Motivan al estudiante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV10I	Hacen más fácil el aprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV11I	Permiten mantener el interés en clases	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV12I	Favorecen la práctica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV13I	Optimizan el aprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV14I	Motivan el autoaprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV15I
	1	2	3	4	5																																													
Motivan al estudiante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV10I																																												
Hacen más fácil el aprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV11I																																												
Permiten mantener el interés en clases	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV12I																																												
Favorecen la práctica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV13I																																												
Optimizan el aprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV14I																																												
Motivan el autoaprendizaje	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	IV15I																																												

Anexo 5. Requerimientos de *software* y *hardware* para la instalación y configuración de la plataforma Moodle

La versión de Moodle que fue implementada es la 2.9. A continuación se anotan los requerimientos mínimos para su funcionamiento en un ambiente de producción. Estos podrán variar dependiendo de las combinaciones del *hardware* y *software* específicos, además del tipo de uso y la carga; los sitios muy concurridos muy probablemente requerirán recursos adicionales [97].

Requisitos de *software* [97, 98]:

- Moodle está desarrollado principalmente en GNU/Linux usando Apache, PostgreSQL/MySQL/MariaDB y PHP (también conocida como plataforma LAMP).
- Para la instalación de las últimas versiones de la plataforma Moodle es necesario que el servidor web cuente con una versión de PHP 5.5.x en adelante. PHP 7 no es soportado.
- Si se desea correr sobre un sistema operativo Windows, a partir de PHP 5.5 en adelante, también se necesita tener Visual C++ Redistributable for Visual Studio 2012 instalado.
- Ghostscript debe estar instalado para anotación de PDF.
- Para el envío de correo electrónico debe estar configurado el protocolo SMTP en Moodle.

Requisitos de *hardware* [97]:

- Espacio de disco: tanto cuanto usted necesite para almacenar sus materiales además del espacio para los archivos del sistema operativo y el programa Moodle. 5 GB es el espacio mínimo para correr un sitio de Moodle en producción.
- Espacio de disco para los respaldos: al menos los mismos 5 GB y preferentemente en una localización remota.
- Procesador: 1 GHz (mínimo), se recomienda doble núcleo de 2 GHz. Estas características pueden variar de acuerdo a los recursos usados.
- Memoria: 2 GB o más es lo recomendado. La cantidad de memoria RAM que se necesita depende de muchos factores, especialmente de cómo sea configurado y usado su sitio. Especial importancia tiene la cantidad de usuarios que usan concurrentemente

el sitio. Concurrente realmente significa procesos de servidor web en memoria al mismo tiempo (por ejemplo: usuarios interactuando con el sistema dentro de una ventana de unos pocos segundos), no significa personas ingresadas al sitio.

Requerimientos de la base de datos [98]:

En la siguiente tabla se listan los servidores de bases de datos soportados por Moodle; en el presente trabajo se usó MySQL.

Base de datos	Versión mínima	Recomendada
PostgreSQL	9.1	La estable más reciente
MySQL	5.5.31	La estable más reciente
MariaDB	5.5.31	La estable más reciente
Microsoft SQL Server	2008	La estable más reciente
Oracle Database	10.2	La estable más reciente

Requerimientos del cliente [98]:

En la tabla que se muestra a continuación están los navegadores soportados para un correcto funcionamiento del sitio de Moodle.

Navegador	Versión mínima	Recomendada
Google Chrome	30.0	La más reciente
Mozilla Firefox	25.0	La más reciente
Apple Safari	6	La más reciente
Microsoft Internet Explorer	9	La más reciente