



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

Juegos Serios Computacionales para el Aprendizaje de las
Alfas del Núcleo del estándar Essence

T E S I S
QUE PARA OPTAR POR EL GRADO DE:
DOCTORA EN CIENCIAS (COMPUTACIÓN)

PRESENTA:
ERÉNDIRA MIRIAM JIMÉNEZ HERNÁNDEZ

Directora de Tesis:
Dra. Hanna Jadwiga Oktaba
Facultad de Ciencias, UNAM

Comité Tutor:
Dr. Mario Gerardo Piattini Velthuis
Escuela Superior de Informática, UCLM

Dra. Frida Díaz Barriga Arceo
Facultad de Psicología, UNAM

Ciudad Universitaria, CD. MX. Diciembre, 2020



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A mi esposo, por su amor.

A mi papá, por su apoyo incondicional.

A mi mamá, por su cariño.

Agradecimientos

En momentos como este, cuando se logra un objetivo personal y académico tan importante, es inevitable pensar en las personas que me apoyaron para que esto fuera posible. Debo reconocer que es difícil dejar a un lado los sentimientos y que las palabras se quedan cortas al expresar mi aprecio y gratitud.

Comenzaré, entonces, agradeciendo a la Universidad Nacional Autónoma de México y al Posgrado en Ciencia e Ingeniería de la Computación, por darme la oportunidad de estudiar en la mejor institución educativa de mi país.

Igualmente, quiero hacer un reconocimiento al Consejo Nacional de Ciencia y Tecnología (CONACyT), pues su apoyo económico me permitió realizar los estudios de doctorado.

Agradezco muy especialmente a mi tutora, la Dra. Hanna Oktaba, por compartir su conocimiento conmigo y apoyarme durante estos años de manera incondicional. Me considero afortunada de haber tenido una extraordinaria mamá académica.

Asimismo, estoy profundamente agradecida con mis co-tutores, la Dra. Frida Díaz Barriga y el Dr. Mario Piattini. Agradezco a la Dra. Frida, por haber creído en mí y en mi trabajo, por sus enseñanzas, ayuda y motivación. Mi más sincero agradecimiento al Dr. Mario, por brindarme su guía y apoyo incondicional.

También me siento muy agradecida con mis sinodales. Agradezco al Dr. Enrique Ruíz Velasco, por sus invaluable comentarios y recomendaciones que enriquecieron mi trabajo. Gracias al Dr. Fernando Gamboa (Q. E. P. D.), porque su “Aula del Futuro” y sus consejos inspiraron siempre mi trabajo. Agradezco a la Dra. Sandra Dinora Orantes, porque sus enseñanzas en la maestría fueron fundamentales para realizar esta investigación. Gracias, además, por seguir apoyándome.

Asimismo, agradezco a aquellos profesores que han sido mentores, siempre al tanto. A la Maestra Lupita Ibarguengoitia, por sus sabios y lindos consejos. También, al Dr. Juan Manuel García, a la Maestra Carol Martínez y al Dr. Octavio Ortiz, quienes me han apoyado desde mis estudios de licenciatura.

Agradezco, también, a quienes me ayudaron administrativamente para culminar con éxito esta etapa. Gracias al coordinador del Posgrado, el Dr. Javier Gómez, por todo el apoyo. De igual forma, agradezco a Lulú González y a Ceci Mandujano, por el cariño y la ayuda que siempre me brindaron.

Además, me siento agradecida con Goyi Romero, que me apoyó con la logística para realizar mi estancia de investigación en Ciudad Real, así como con los miembros del Grupo Alarcos por todas las facilidades concedidas, tanto a mí como a “mis delfines”.

Estoy agradecida con mi ex-alumnos que confiaron en mi y trabajaron durante dos veranos de investigación. Gracias a Alan Revillagigedo, Arturo López, Sergio Flores y Daniel Bárcenas.

De igual manera, agradezco a Pamela Hernández, Francisco Aguilar, Maricruz Chávez, Jessica Comparán, José Mendoza, Jorge Díaz y Romario Alvarado, por haber confiado en mí como tutora para cursar un verano de investigación.

Están, además, aquellos quienes amablemente me prestaron sus voces para los juegos serios computacionales desarrollados. Extiendo mi gratitud a Melba Guzmán, Laura N. Alvarado, Trycia Bazinet, Karina Orozco, Yuritzi Aguilar, Karla Martínez, Jesús Regalado, Rigoberto Quezada, René Orozco, Fernanda Núñez y Claudia Fabián.

Finalmente, debo agradecer a mi familia, quienes siempre han estado a mi lado apoyándome. Gracias infinitas por haberme dado la fuerza necesaria para lograr esta meta tan anhelada.

Agradezco a mi esposo, mi compañero de vida, que siempre estuvo ahí para apoyarme cuando mis ánimos decaían. Te amo.

Me siento muy agradecida con mi papá, por la ayuda y el apoyo incondicional desde siempre. Gracias por inspirarme a ser mejor cada día.

Agradezco a mi mamá, quien siempre está al pendiente de mi. Gracias por el cariño y el apoyo.

Gracias a mi hermana, porque su orientación estadística fue esencial en mi investigación, pero sobretodo por ser mi mejor amiga.

También, gracias mi hermano, por ser un ejemplo de valentía y superación.

Y a Bender, por su compañía.

Eréndira Miriam Jiménez Hernández

Tabla de contenidos

I	Generalidades	1
1.	Introducción	2
1.1	Antecedentes y motivación	2
1.1.1	Essence	3
1.1.2	Alfas del núcleo de Essence	4
1.1.3	Juegos serios	7
1.2	Hipótesis y objetivo	9
1.3	Apoyos económicos de la investigación	11
1.4	Estructura de la tesis	11
2.	Método de investigación	14
2.1	Métodos y metodologías de investigación	14
2.2	Investigación-Acción Técnica (TAR)	15
2.3	Método de Mapeo Sistemático de Estudios (SMS)	17
2.4	Experimentos	18
2.5	Estrategia de investigación de esta tesis	21
2.5.1	Primer ciclo de ingeniería	23
2.5.2	Segundo ciclo de ingeniería	23
2.5.3	Tercer ciclo de ingeniería	24
2.5.4	Cuarto ciclo de ingeniería	25
2.5.5	Quinto ciclo de ingeniería	25
2.5.6	Sexto ciclo de ingeniería	26
3.	Estado del arte	27
3.1	Mapeo sistemático de estudios de juegos serios para el aprendizaje de procesos de Ingeniería de Software	27
3.1.1	Planificar	27
3.1.1.1	Pregunta de investigación del mapeo sistemático de estudios	28

3.1.1.2	Fuentes y estrategia de investigación	28
3.1.1.3	Criterios de selección	29
3.1.1.4	Estrategia de extracción de información	30
3.1.1.5	Evaluación de calidad	33
3.1.2	Realizar	34
3.1.3	Reportar	35
3.2	Análisis de Fortalezas, Oportunidades, Debilidades y Amenazas	45
3.2.1	Fortalezas	46
3.2.2	Oportunidades	46
3.2.3	Debilidades	47
3.2.4	Amenazas	48
3.2.5	Área de oportunidad	49

II Juegos Serios 50

4. Fundamentos Pedagógicos	51
4.1 Teorías Pedagógicas	51
4.1.1 Conductismo	51
4.1.2 Cognitivismo	52
4.1.3 Constructivismo	52
4.1.3.1 Teoría de la Psicología genética	53
4.1.3.2 Teoría de la Psicología cognitiva	53
4.1.3.3 Constructivismo social	54
4.1.4 Conectivismo	54
4.1.5 Humanismo	55
4.2 Teorías Psicopedagógicas Lúdicas	55
4.2.1 Teoría de Johan Huizinga	55
4.2.2 Teoría de Karl Groos	55
4.2.3 Teoría de Lev Vigotsky	56
4.2.4 Teoría de Jean Piaget	56
4.2.5 Teoría de Paul Gee	57

4.3	Taxonomía de Bloom	61
4.3.1	Revisión de la Taxonomía de Bloom	62
4.4	Modelos de Aprendizaje	64
4.4.1	Modelo VARK	64
4.4.2	Modelo de Aprendizaje Experiencial	65
4.5	Modelos de Diseño Instruccional	67
4.5.1	Modelo de Jonassen	67
4.5.2	Modelo ARCS	70
4.6	Modalidades de Aprendizaje	72
4.6.1	Aprendizaje Presencial	72
4.6.2	Aprendizaje Virtual	73
4.6.3	Aprendizaje Combinado	73
4.6.4	Aprendizaje Móvil	74
4.6.5	Aula invertida	74
5.	Metodología para construir juegos serios computacionales	76
5.1	Metodologías de desarrollo de software	76
5.2	Metodologías de desarrollo de videojuegos	77
5.3	Metodologías de desarrollo de juegos serios	78
5.4	Motivación para diseñar una metodología para construir juegos serios computacionales	78
5.5	Metodología para construir juegos serios computacionales	79
5.5.1	Descripción	79
5.5.2	Alcance	79
5.5.3	Filosofía	79
5.5.4	Roles	80
5.5.5	Etapas y procesos	81
5.5.5.1	Pre producción	83
5.5.5.2	Producción	88
5.5.5.3	Post producción	88
5.5.6	Juegos serios computacionales construidos	89
5.5.6.1	Juego serio computacional para aprender la simplificación de funciones Booleanas	90

5.5.6.2	Juego serio computacional para aprender operaciones aritméticas con fracciones matemáticas	92
5.5.6.3	Juego serio computacional para aprender recorridos de árboles	92
5.5.6.4	Juego serio computacional para aprender grafos	93
6.	Juegos serios computacionales para el aprendizaje de las Alfas de Essence	95
6.1	Alhaspot	96
6.1.1	Descripción	96
6.1.2	Características	96
6.1.3	Tecnologías empleadas	97
6.1.4	Niveles del juego	97
6.2	Alphalingo	101
6.2.1	Descripción	101
6.2.2	Características	101
6.2.3	Tecnologías empleadas	101
6.2.4	Módulos del juego	102
6.2.5	Tipos de ejercicios	103
6.3	Alphadecision	106
6.3.1	Descripción	106
6.3.2	Características	106
6.3.3	Tecnologías empleadas	107
6.3.4	Modalidades de juego	107
6.3.4.1	Jugador virtual	108
6.3.4.2	Jugador real	109
6.3.4.3	Certificación	109
III	Validación	112
7.	Validación de los juegos serios computacionales	113

7.1	Alphaspot	113
7.1.1	Evaluación de la motivación	113
7.2	Alphalingo	114
7.2.1	Método de evaluación del aprendizaje	114
7.2.1.1	Diseño del Experimento	114
7.2.1.2	Materiales	115
7.2.1.3	Procedimiento	115
7.2.2	Experimento	116
7.2.2.1	Ejecución	116
7.2.2.2	Análisis estadístico	116
7.2.3	Réplica del Experimento	119
7.2.3.1	Ejecución	119
7.2.3.2	Análisis estadístico	119
7.2.4	Meta-análisis	121
7.2.5	Evaluación de la motivación en el aprendizaje	123
7.2.6	Amenazas a la validez	123

IV Conclusiones 125

8. Conclusiones 126

8.1	Análisis de objetivos de investigación	126
8.2	Producción	128
8.2.1	Artículos en Revistas Internacionales	128
8.2.2	Artículos de Congreso	129
8.2.3	Capítulos de Libros	129
8.2.4	Carteles	130
8.2.5	Conferencias magistrales	130
8.2.6	Talleres impartidos	130
8.2.7	Estancias de investigación	131
8.2.8	Formación de recursos humanos	131
8.3	Trabajos a futuro	132

Referencias	133
--------------------	------------

V Apéndices	144
--------------------	------------

Apéndice A – Alfás de Essence	145
Apéndice B – Formulario de Extracción de Información del Mapeo Sistemático de Estudios	156
Apéndice C – Lista de Estudios Primarios del Mapeo Sistemático de Estudios	158
Apéndice D – Mapeo de Estudios Primarios	164
Apéndice E – ISO/IEC 29110 Deployment Package	166
Apéndice F – Motores de videojuego	167
Apéndice G – Duolingo	170
Apéndice H – Cuestionario para la evaluación de la motivación	171
Apéndice I – Cuestionario para la evaluación diagnóstica del aprendizaje (Pre-prueba)	174
Apéndice J – Cuestionario para la evaluación final del aprendizaje (Post-prueba)	177
Apéndice K – Cuestionario para identificar las características de los participantes en el experimento	181
Apéndice L – Resultados de la evaluación diagnóstica del aprendizaje (Pre-prueba)	183
Apéndice M – Resultados de la evaluación final del aprendizaje (Post-prueba)	187
Apéndice N – Resultados de la evaluación de la motivación (Post-prueba)	191
Apéndice Ñ – Características de los participantes en el experimento	198
Apéndice O – Acrónimos	200

Lista de figuras

Figura 1-1	Áreas de interés del núcleo de Essence (OMG, 2014).	4
Figura 1-2	Alfas del núcleo de Essence (OMG, 2014).	5
Figura 2-1	Iteraciones del TAR (Wieringa y Morali, 2012).	16
Figura 2-2	Ciclo de ingeniería del TAR (Wieringa y Morali, 2012).	17
Figura 2-3	Proceso para realizar un Mapeo Sistemático de Estudios (Kitchenham y Charters, 2007; Genero et al., 2014)	18
Figura 2-4	Proceso para realizar un Experimento (Wohlin <i>et al.</i> , 2012)	21
Figura 2-5	Proceso de investigación en esta tesis	22
Figura 3-1	Proceso de selección de estudios	34
Figura 3-2	Procesos del ciclo de vida de software que enseñan los juegos serios	37
Figura 3-3	Tipos de juegos serios en los estudios primarios	38
Figura 3-4	Audiencia de los juegos serios	38
Figura 3-5	Esquema de interacción de los juegos serios	39
Figura 3-6	Medios/Materiales empleados por los juegos serios	39
Figura 3-7	Objetivo de los juegos serios	40
Figura 3-8	Estilo de aprendizaje empleado en los juegos serios	40
Figura 3-9	Niveles de aprendizaje objetivo de los juegos serios	41
Figura 3-10	Resultados obtenidos al usar los juegos serios	41
Figura 3-11	Tipos de publicación de los estudios primarios	42
Figura 3-12	Talleres a los que pertenecen los estudios primarios	42
Figura 3-13	Congresos a los que pertenecen los estudios primarios	43
Figura 3-14	Revistas a las que pertenecen los estudios primarios	44
Figura 3-15	Estudios primarios publicados en las Bases de Datos por año	45
Figura 4-1	Taxonomía de Bloom original (Bloom, 1956) y revisada (Anderson y Krathwohl, 2001)	64
Figura 4-2	Estilos sensoriales de aprendizaje (Hawk y Shah, 2007)	65
Figura 4-3	Estilos de aprendizaje experienciales (Kolb, 1984)	67
Figura 4-4	Entornos de Aprendizaje Constructivistas (Jonassen, 1991)	69

Figura 4-5	Proceso ARCS para el diseño de entornos motivacionales de trabajo y aprendizaje (Keller, 1987)	71
Figura 5-1	Etapas y procesos de la metodología para construcción de juegos serios computacionales	82
Figura 5-2	Reuniones de desarrollo del concepto	84
Figura 5-3	Proceso de enseñanza-aprendizaje del equipo	84
Figura 5-4	Reuniones de análisis y diseño del juego serio	85
Figura 5-5	Diagrama de navegación/transición	86
Figura 5-6	Diagrama de Flujo	87
Figura 5-7	Gestión del Proyecto	87
Figura 5-8	Juego serio computacional para aprender simplificación de funciones Booleanas	90
Figura 5-9	Ranking de MiniBool	91
Figura 5-10	Juego serio computacional para aprender operaciones aritméticas con fracciones matemáticas	92
Figura 5-11	Juego serio computacional para aprender recorridos de árboles	93
Figura 5-12	Juego serio computacional para aprender grafos	94
Figura 6-1	Alphaspot en un dispositivo móvil	97
Figura 6-2	Narrativa inicial de Alphaspot	98
Figura 6-3	Nivel tutorial de Alphaspot	98
Figura 6-4	Núcleo del estándar Essence en Alphaspot	99
Figura 6-5	Nivel de acción del Alfa “Forma de Trabajar”	100
Figura 6-6	Nivel de evaluación en Alphaspot	100
Figura 6-7	Puntos de experiencia y progreso del aprendizaje	103
Figura 6-8	Primer tipo de ejercicios en Alphalingo	104
Figura 6-9	Segundo tipo de ejercicios en Alphalingo	104
Figura 6-10	Tercer tipo de ejercicios en Alphalingo	105
Figura 6-11	Cuarto tipo de ejercicios en Alphalingo	106
Figura 6-12	Modalidades de juego en Alphadecision	108
Figura 6-13	Jugador virtual de Alphadecision	108
Figura 6-14	Jugadores reales en Alphadecision	109
Figura 6-15	Calendarización de certificación en Alphadecision	110

Figura 6-16	Certificación en Alphadecision	110
Figura 6-17	Estadísticas de Alphadecision	111
Figura 6-18	Panel de administración de Alphadecision	111
Figura 7-1	Diagrama de caja comparativo de Pre-prueba/Post-prueba del Experimento y la Réplica.	122
Figura E-1	Arquitectura de ISO/IEC 29110 Deployment Package	166

Lista de tablas

Tabla 3-1	Fuentes de información seleccionadas	29
Tabla 3-2	Cadena de búsqueda	29
Tabla 3-3	Criterios de Inclusión	30
Tabla 3-4	Criterios de Exclusión	30
Tabla 3-5	Cuestionario de evaluación de calidad	33
Tabla 3-6	Resultados del mapeo sistemático de estudios	35
Tabla 3-7	Fortalezas de los estudios primarios	46
Tabla 3-8	Oportunidades de los estudios primarios	47
Tabla 3-9	Debilidades de los estudios primarios	47
Tabla 3-10	Amenazas de los estudios primarios	48
Tabla 4-1	Componentes y subcomponentes del modelo ARCS	70
Tabla 5-1	Plan del Proyecto	88
Tabla 6-1	Competencias específicas de cada módulo de Alphalingo	102
Tabla 7-1	Evaluación de los factores de motivación de Alphapot	114
Tabla 7-2	Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Pre-prueba (E)	117
Tabla 7-3	Prueba de Levene para la homogeneidad de varianzas y prueba T de Student de la Pre-prueba (E)	117
Tabla 7-4	Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Post-prueba (E)	118
Tabla 7-5	Prueba de Levene para la homogeneidad de varianzas y prueba T de Student de la Post-prueba (E)	119
Tabla 7-6	Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Pre-prueba (R)	120
Tabla 7-7	Prueba de Levene para la homogeneidad de varianzas y prueba T de Student de la Pre-prueba (R)	120
Tabla 7-8	Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Post-prueba (R)	120
Tabla 7-9	Prueba de Levene para la homogeneidad de varianzas y prueba T de Student de la Post-prueba (R)	121
Tabla 7-10	Resultados del meta-análisis	122

Tabla 7-11	Promedios de los factores de motivación del experimento y la réplica	123
Tabla L-1	Respuestas correctas del cuestionario para la evaluación diagnóstica del aprendizaje	183
Tabla L-2	Calificaciones de acuerdo al número de respuestas correctas	184
Tabla L-3	Calificaciones obtenidas por los participantes del grupo de control en la Pre-prueba	185
Tabla L-4	Calificaciones obtenidas por los participantes del grupo experimental en la Pre-prueba	186
Tabla M-1	Respuestas correctas del cuestionario para la evaluación final del aprendizaje	187
Tabla M-2	Calificaciones obtenidas por los participantes del grupo de control en la Post-prueba	188
Tabla M-3	Calificaciones obtenidas por los participantes del grupo experimental en la Post-prueba	189
Tabla N-1	Resultados de la evaluación de la motivación en el grupo de control	192
Tabla N-2	Promedios de los factores de motivación en el grupo de control	194
Tabla N-3	Resultados de la evaluación de la motivación en el grupo experimental	195
Tabla N-4	Promedios de los factores de motivación en el grupo experimental	197
Tabla Ñ-1	Características de los participantes en el experimento	198

Resumen

La Ingeniería de Software es un área de las Ciencias de la Computación que proporciona técnicas, prácticas, métodos y procesos para desarrollar, mantener y operar software. Este amplio cuerpo de conocimiento, se encuentra “esencializado” en el estándar internacional “Essence”, el cual permite, a través de sus “Alfas”, mejorar la calidad de software, por medio de un marco de pensamiento holístico, reflexivo, orientado a objetos, no prescriptivo y agnóstico, que se puede integrar con otros estándares.

Aunque las Alfas de Essence concentran el conocimiento de la Ingeniería de Software, el comprenderlas no resulta intuitivo para los estudiantes de carreras afines a la Computación, ni para los profesionales que laboran en entidades de software. Por esta razón, como parte de la investigación presentada en esta tesis, se desarrollaron tres juegos serios computacionales para facilitar el aprendizaje de las Alfas de Essence. Los juegos fueron construidos siguiendo una metodología propia, la cual también se incluye en este documento.

De este modo, los juegos serios computacionales desarrollados, brindan a los usuarios la posibilidad de práctica constante, con tolerancia al error, y mantienen la atención de los estudiantes por medio de estímulos que consideran los diferentes estilos de aprendizaje sensoriales. Asimismo, propician un aprendizaje activo y reflexivo, que a su vez permite lograr un aprendizaje significativo.

Los juegos construidos fueron probados por profesionales de software y evaluados formalmente, mostrando mejores resultados al ser comparados con el método de aprendizaje tradicional presencial. Lo que proporciona evidencia que sugiere que el uso de juegos serios computacionales en el aprendizaje de la Ingeniería de Software permite obtener resultados positivos.

Abstract

Software Engineering is an area of Computer Science that provides techniques, practices, methods and processes to develop, maintain and operate software. This broad body of knowledge is “essentialized” in the international standard “Essence”, which allows, through its “Alphas”, improving the quality of software, by means of a holistic, reflective, object-oriented, non-prescriptive and agnostic framework, which can be integrated with other standards.

Although the Essence Alphas concentrate the knowledge of Software Engineering, understanding them is not intuitive for students of careers related to Computing, neither for professionals who work in software entities. For this reason, as part of the research presented in this thesis, three serious games based on computer were developed to facilitate the learning of the Essence Alphas. The games were constructed following an own methodology, which is also included in this document.

In this way, the computer-based serious games developed provide to the users the possibility of constant practice, with tolerance for error, and maintain the apprentices’ attention through stimuli that consider the different sensory learning styles. Thus, these serious games promote an active and reflective learning, which in turn allows attaining a meaningful learning.

The serious games based on computer were used by software professionals, as part of a formal experiment that was carried out to evaluate the learning and the motivation that can be attain, showing higher results when they compared to the face-to-face learning method. Providing, therefore, evidence that suggests that the use of computer-based serious games in the learning of Software Engineering allows obtaining positive results.

Parte I
Generalidades

Capítulo 1

Introducción

*“No hay enseñanza sin investigación
ni investigación sin enseñanza”*
Paulo Freire (1921-1997)

Este capítulo presenta los antecedentes y la motivación de esta tesis, así como su hipótesis, objetivo, apoyos económicos y la forma en que está organizada.

1.1 Antecedentes y motivación

En la actualidad, el software se ha convertido en una herramienta indispensable para la sociedad, pues ha permitido la creación y expansión de tecnologías, las cuales han ayudado a mejorar la calidad de vida de las personas. Dada su importancia, la Ingeniería de Software (IS) ha diseñado técnicas, prácticas, métodos y procesos para apoyar el desarrollo, mantenimiento y operación del software.

Empero, aprender IS resulta ser una actividad poco trivial, debido a que el cuerpo de conocimiento de esta área es muy amplio. Con el objetivo de dar solución a esta problemática, en 2014, el organismo internacional de estandarización en tecnologías de la información llamado OMG (por sus siglas en inglés, *Object Management Group*), publicó el estándar llamado “*Essence*”, que “esencializa” el conocimiento empírico de la IS por medio de sus “Alfas”.

Las Alfas de *Essence* proporcionan un marco de pensamiento holístico, reflexivo, orientado a objetos, no prescriptivo y agnóstico, que permite mejorar la calidad del software (Morales-Trujillo y Oktaba, 2016; Péraire y Sedano, 2014; Simonette *et al.*, 2016; Zapata-Jaramillo y Montoya-Pérez, 2016; Castro-Rojas *et al.*, 2016). No obstante estos beneficios, las Alfas no han sido del todo adoptadas por las personas de la comunidad de IS; esto debido, principalmente, a que no poseen el tiempo suficiente para aprender sobre las Alfas, dada la sobrecarga de actividades que impera en las entidades de desarrollo de

software y, a la poca familiaridad que se tiene para estudiar estándares técnicos.

Lo anterior motivó a buscar la forma de facilitar el aprendizaje de las Alfas de Essence a los profesionales de software, encontrando en los juegos serios la solución, dados los resultados positivos que han sido reportados al ser empleados en el aprendizaje de varios temas en diversas áreas (de-Marcos, García-López y García-Cabot, 2016).

1.1.1 Essence

La especificación “Essence – Núcleo y Lenguaje para Métodos de Ingeniería de Software versión 1” (Essence), es un estándar internacional que fue publicado por el OMG en el año 2014. Su versión más reciente corresponde a la 1.2, la cual que fue publicada en el 2018.

Essence se compone de dos partes:

- A. **Un núcleo**, conformado por los elementos esenciales de la IS.
- B. **Un lenguaje**, con el que es posible definir prácticas y métodos.

El núcleo, que contiene lo efectivo y escalable de la IS, se organiza en tres áreas de interés (véase Figura 1-1), cada una de las cuales se centra en un aspecto específico de la IS:

1. **Cliente:** comprende todo lo relacionado con el uso real y la explotación del sistema de software que será desarrollado.
2. **Solución:** comprende todo lo relacionado con la especificación y el desarrollo del sistema de software.
3. **Esfuerzo:** comprende todo lo relacionado con el equipo y la forma en la que se realiza el trabajo.

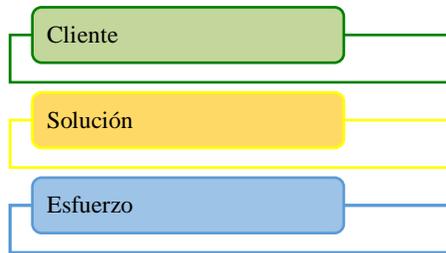


Figura 1-1. Áreas de interés del núcleo de Essence (OMG, 2014).

Asimismo, el núcleo está compuesto de tres tipos de elementos esenciales para llevar a cabo el trabajo de la IS:

1. **Alfas:** que representan las cosas con las que se trabaja.
2. **Espacios de Actividades:** que representan las cosas que se hacen.
3. **Competencias:** que representan las capacidades requeridas.

1.1.2 Alfás del núcleo de Essence

Las Alfás del núcleo de Essence son las representaciones de las cosas esenciales con las que se trabaja en un proyecto de software. Proporcionan descripciones de las cosas que un equipo va a administrar, producir y utilizar en el proceso de desarrollo, mantenimiento y soporte de software. Siendo de especial interés porque:

- Capturan los conceptos esenciales de la IS,
- Permiten evaluar y dar seguimiento al progreso y estado de cualquier proyecto de software y,
- Proporcionan una base común para definir métodos y prácticas de IS.

En la Figura 1-2 se muestran las Alfás que conforman el núcleo de Essence, las cuales están divididas de acuerdo al área de interés al que pertenecen. Las siete Alfás son: Oportunidad, Involucrados, Requerimientos, Sistema de Software, Trabajo, Equipo y Forma de Trabajar.

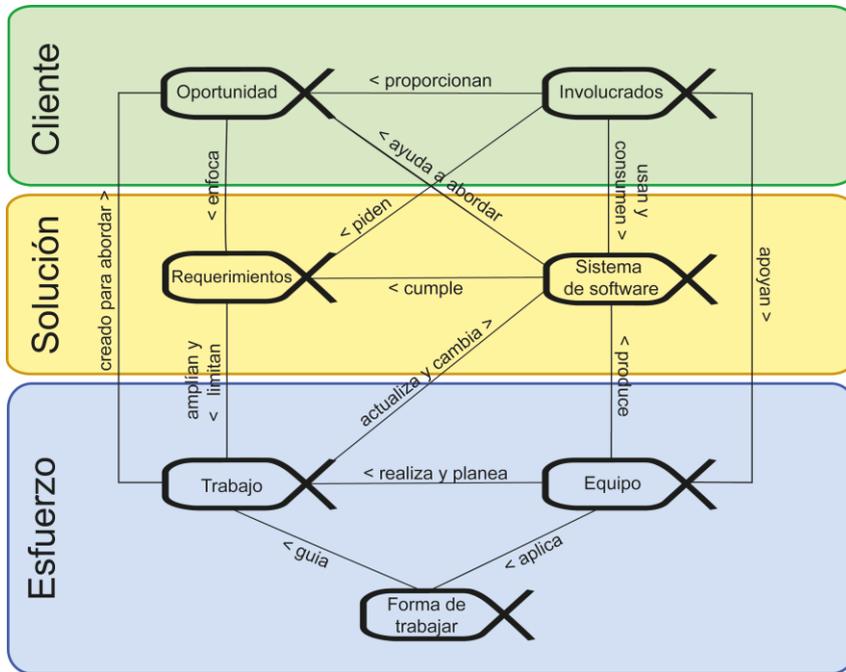


Figura 1-2. Alfes del núcleo de Essence (OMG, 2014).

Las Alfes del núcleo de Essence se componen de estados secuenciales. A su vez, cada estado contiene una lista de verificación con indicadores críticos, los cuales deben monitorearse con el objetivo de determinar el estado actual de un proyecto y progresar.

Es así que en área de interés del Cliente existen dos Alfes:

- **Oportunidad:** El conjunto de circunstancias que hace apropiado desarrollar o modificar un sistema de software. Los estados que componen esta Alfa son:
 - a. Identificada
 - b. Solución necesaria
 - c. Valor establecido
 - d. Viable
 - e. Abordada
 - f. Beneficio percibido

- **Involucrados:** Las personas, grupos u organizaciones quienes afectan o son afectados por el desarrollo o modificación de un sistema de software. Los estados que componen esta Alfa son:
 - g. Reconocidos
 - h. Representados
 - i. Implicados
 - j. De acuerdo
 - k. Satisfechos para el despliegue
 - l. Satisfechos en uso

En el área de interés de la Solución existen dos Alfas:

- **Requerimientos:** Lo que el sistema de software tiene que hacer para afrontar la oportunidad y satisfacer a los involucrados. Los estados que componen esta Alfa son:
 - a. Concebidos
 - b. Acotados
 - c. Coherentes
 - d. Aceptables
 - e. Abordados
 - f. Cumplidos

- **Sistema de software:** Sistema compuesto por software, hardware y datos. Los estados que componen esta Alfa son:
 - a. Arquitectura seleccionada
 - b. Demostrable
 - c. Utilizable
 - d. Listo
 - e. Operacional
 - f. Retirado

Finalmente, en el área de interés del Esfuerzo existen tres Alfas:

- **Trabajo:** La actividad que implica realizar un esfuerzo mental y/o físico con el fin de construir el sistema de software. Los estados que componen esta Alfa son:

- a. Iniciado
 - b. Preparado
 - c. Comenzado
 - d. Bajo control
 - e. Concluido
 - f. Cerrado
- **Equipo:** El grupo de personas que participan activamente en el desarrollo, mantenimiento, entrega o soporte de un sistema de software. Los estados que componen esta Alfa son:
 - a. Sembrado
 - b. Formado
 - c. Colaborando
 - d. Ejecutando
 - e. Suspendido
 - **Forma de trabajar:** El conjunto adaptado de prácticas y herramientas empleadas por un equipo para guiar y realizar su trabajo. Los estados que componen esta Alfa son:
 - a. Principios establecidos
 - b. Base establecida
 - c. En uso
 - d. En marcha
 - e. Trabajando bien
 - f. Retirada

Las listas de verificación de cada uno de los 41 estados pertenecientes a las 7 Alfas del núcleo de Essence, pueden apreciarse en el Apéndice A.

1.1.3 Juegos serios

El término “juego serio” fue definido por primera vez en el libro “*Serious Games*” (1970), escrito por Clark Abt, como aquél juego que “*tiene un propósito educativo explícito y cuidadosamente planeado, porque no está pensado para ser jugado únicamente por diversión*”.

Uno de los primeros juegos denominado como serio, es “T.E.M.P.E.R” que fue diseñado para estudiar el conflicto de la Guerra Fría (Abt Associates, 1965), siendo la milicia de Estados Unidos pionera en el uso de este tipo de juegos para sus entrenamientos (Sawyer y Rejeski, 2002). Desde entonces, los juegos serios han sido utilizados en diversos contextos, como la educación, salud, mercadotecnia, política, etc. Dirigidos a una gran variedad de público, como estudiantes de diferentes niveles educativos, profesionistas y consumidores de productos y servicios. Haciendo uso de elementos no tecnológicos y tecnológicos, como computadoras de escritorio y portátiles, consolas de videojuego, tabletas y celulares inteligentes; con la finalidad de promover la construcción del conocimiento, el desarrollo de capacidades y la comercialización.

Estos “*juegos que no tienen entretenimiento, disfrute o diversión como su fin principal*” (Michael y Chen, 2005), pueden clasificarse de acuerdo a su objetivo en tres tipos:

1. ***Edutainment***: surge de la contracción de las palabras en inglés “*education*” y “*entertainment*”, es decir, educación y entretenimiento (Landers y Callan, 2011). Consiste en atraer la atención de los estudiantes con elementos divertidos durante el proceso de aprendizaje (Okan, 2003). El término *edutainment* fue usado por primera vez en 1948, por “*Walt Disney Productions*”, para describir la serie de documentales llamada “*True-Life Adventures*”, que incluía un enfoque lúdico para que los niños pudiesen aprender temas relacionados con la naturaleza mientras se divertían (Barrier, 2008). Más tarde, en 1973, fue usado por Robert Heyman en sus documentales producidos para la “*National Geographic Society*” (Aksakal, 2015). Actualmente, es posible encontrar varios juegos serios de *edutainment* que facilitan el proceso de aprendizaje tanto de niños como de adultos (Kill *et al.*, 2012).
2. ***Advergaming***: surge de la contracción de las palabras en inglés “*advertising*” y “*gaming*”, es decir, publicidad y juego (Smith *et al.*, 2014). Son juegos que sirven como medios de comunicación o publicitarios, con los que se pretende fomentar la comercialización de una marca, producto, organización o idea (Kim, 2006). Algunos

ejemplos de *advergaming* son: la aplicación móvil “*Dumb ways to die*”, desarrollada para la red ferroviaria “*Metro Trains*” en Australia, con el objetivo de promover el uso del tren (Metro Trains Melbourne, 2012) y el videojuego “*America’s Army*”, diseñado para aumentar el número de jóvenes reclutados en el ejército de Estados Unidos (Department of Defense, 2002).

3. **Simuladores:** pretenden replicar experiencias, comportamientos y sensaciones que ocurren en determinadas situaciones, casos o máquinas en entornos de riesgo cero (Daungcharone, 2017). Los primeros juegos serios de simulación surgieron en 1960, con la misión de entrenar mejor a los pilotos de aviones (Hill y Tolk, 2017). Un ejemplo de este tipo de juegos serios es “*Battlezone/Bradley Trainer*”, creado por la compañía de videojuegos Atari® en 1980, para el adiestramiento de artilleros de tanques de guerra (Kent, 2001).

Particularmente, los juegos serios han sido muy empleados en la educación debido a que:

- Motivan el aprendizaje, superando la resistencia natural a participar en procesos repetitivos (Jensen, 2006).
- Agilizan los procesos de aprendizaje (Wang *et al.*, 2017).
- Potencian la creatividad (Dibona, 2004).
- Mejoran la retención de la información (Callaghan *et al.*, 2015).
- Incentivan a seguir aprendiendo (Cai *et al.*, 2009).
- Son seguros, porque recrean una parte de la realidad pero sin ningún peligro (Kasvi, 2000).

1.2 Hipótesis y objetivo

Al identificar los beneficios que podrían brindar las Alfás del núcleo del estándar Essence a los profesionales de software, de sus problemáticas para aprender sobre ellas y de conocer el potencial didáctico de los juegos serios en diversas áreas, se planteó la siguiente pregunta de investigación:

¿Podrían obtener mejores resultados los profesionales de software al aprender sobre las Alfás del núcleo del estándar Essence por medio del uso de juegos

serios computacionales, que por medio del método de aprendizaje tradicional presencial?

Lo que condujo a formular la siguiente hipótesis de investigación:

H₀: No existe diferencia significativa al aprender sobre las Alfás del núcleo del estándar Essence por medio del uso de juegos serios computacionales, que por medio del método de aprendizaje tradicional presencial.

H₁: Existe diferencia significativa al aprender sobre las Alfás del núcleo del estándar Essence por medio del uso de juegos serios computacionales, que por medio del método de aprendizaje tradicional presencial.

Para probar la hipótesis, se planteó como Objetivo Principal (OP) de investigación:

Construir juegos serios computacionales para facilitar el aprendizaje de las Alfás del núcleo del estándar Essence a profesionales de software.

Del cual se derivan los siguientes Objetivos Específicos (OEs):

OE1. Diseñar una metodología para la construcción de juegos serios computacionales.

OE2. Construir juegos serios computacionales que permitan el aprendizaje de las Alfás del núcleo del estándar Essence.

OE3. Realizar al menos un experimento formal con profesionales que laboren en una entidad de software, para probar si al menos uno de los juegos serios computacionales proporcionan una mejora significativa en el aprendizaje de las Alfás del núcleo del estándar Essence, en comparación con el método de aprendizaje tradicional presencial.

OE4. Realizar una prueba de hipótesis estadística para determinar si el aprendizaje de las Alfás del núcleo del estándar Essence en profesionales es significativamente mejor usando juegos serios, que usando el método de aprendizaje tradicional presencial.

OE5. Diseñar un instrumento de evaluación para conocer y comparar el nivel de motivación de los profesionales de software al aprender sobre las Alfas del núcleo del estándar Essence, a través de los juegos serios computacionales y del método de aprendizaje tradicional presencial.

1.3 Apoyos económicos de la investigación

Esta investigación se realizó con el apoyo económico de los siguientes programas y proyectos de investigación:

- Beca nacional y mixta otorgada por el Consejo Nacional de Ciencia y Tecnología (CONACyT) de México.
- Programa de Apoyo a Estudiantes de Posgrado (PAEP), financiado por la Universidad Nacional Autónoma de México (UNAM).
- Programa Interinstitucional para el Fortalecimiento de la Investigación y el Posgrado del Pacífico (Programa Delfín), a través del XX y XXI Verano de la Investigación Científica y Tecnológica del Pacífico.
- Programa de Verano de la Investigación Científica de la Academia Mexicana de Ciencias (AMC).
- Proyecto GINSENG (TIN2015-70259-C2-1-R) (MINECO/FEDER) financiado por el Fondo Europeo de Desarrollo Regional y el Ministerio de Economía y Competitividad de España.

1.4 Estructura de la tesis

Este documento consta de 5 partes, 8 capítulos y 6 apéndices. El contenido se describe a continuación:

- Parte I – Generalidades: comprende la introducción (Capítulo 1), el método de investigación empleado para cumplir con los objetivos planteados (Capítulo 2) y el estado del arte del área de investigación (Capítulo 3).

- Parte II – Juegos serios: presenta las teorías y modelos pedagógicos considerados en la investigación, (Capítulo 4), la metodología diseñada para la construcción de juegos serios (Capítulo 5); así como los tres juegos serios construidos para el aprendizaje de las Alfas de Essence (Capítulo 6).
- Parte III – Validación: contiene las validaciones y experimentos efectuados con dos de los juegos serios construidos para el aprendizaje de las Alfas del núcleo de Essence (Capítulo 7).
- Parte IV – Conclusiones: presenta las aportaciones, los resultados y el trabajo a futuro de la investigación (Capítulo 8).
- Parte V – Apéndices: consiste de:
 - Apéndice A – Alfas de Essence
 - Apéndice B – Formulario de Extracción de Información del Mapeo Sistemático de Estudios
 - Apéndice C – Lista de Estudios Primarios del Mapeo Sistemático de Estudios
 - Apéndice D – Mapeo de Estudios Primarios
 - Apéndice E – ISO/IEC 29110 Deployment Package
 - Apéndice F – Motores de videojuego
 - Apéndice G – Duolingo
 - Apéndice H – Cuestionario para la evaluación de la motivación
 - Apéndice I – Cuestionario para la evaluación diagnóstica del aprendizaje (Pre-prueba)
 - Apéndice J – Cuestionario para la evaluación final del aprendizaje (Post-prueba)
 - Apéndice K – Cuestionario para identificar las características de los participantes en el experimento
 - Apéndice L – Resultados de la evaluación diagnóstica del aprendizaje (Pre-prueba)
 - Apéndice M – Resultados de la evaluación final del aprendizaje (Post-prueba)

Apéndice N – Resultados de la evaluación de la motivación (Post-prueba)

Apéndice Ñ – Características de los participantes en el experimento

Apéndice O – Acrónimos

Capítulo 2

Método de investigación

*“Si no conozco una cosa
la investigaré”*
Louis Pasteur (1822- 1895)

Este capítulo presenta la metodología y métodos de investigación empleados para alcanzar los objetivos planteados en esta tesis.

2.1 Métodos y metodologías de investigación

Un método de investigación es *“un conjunto de postulados, reglas y normas para el estudio y la solución de problemas específicos de investigación, que son institucionalizados por la comunidad científica”* (Bernal, 2006). De una forma más general, un método de investigación se refiere al *“conjunto de procedimientos que haciendo uso de los instrumentos o técnicas necesarias, examina y soluciona un problema o conjunto de problemas de investigación”* (Bunge, 1979).

Los métodos de investigación se clasifican de acuerdo al nivel de evidencia que proporcionan en dos tipos (Genero *et al.*, 2014):

- **Métodos primarios:** permiten obtener evidencia empírica sobre un tema de interés. Dentro de este tipo de métodos se encuentran los experimentos, las encuestas, los estudios de caso, etc., los cuales pueden clasificarse a su vez como cualitativos o cuantitativos, dependiendo del tipo de datos que manejan (Runeson *et al.*, 2012)
- **Métodos secundarios:** permiten recopilar de forma sistemática y rigurosa los estudios primarios relacionados con una pregunta de investigación específica, con la finalidad de sintetizar la evidencia existente que responda a dicha pregunta. Dentro de esta clasificación se encuentran las revisiones sistemáticas de literatura y los mapeos sistemáticos de estudios.

Por su parte, una metodología de investigación es un conjunto de aspectos operativos del proceso de investigación (Lara-Muñoz, 2011), es decir, se ocupa del estudio de los métodos de investigación (Kaplan, 2001).

Una de las metodologías más utilizadas en la Ingeniería de Software es la Investigación-Acción conocida en inglés como *Action-Research (AR)* (Genero *et al.*, 2014), la cual surgió después de la Segunda Guerra Mundial como una forma de relacionar el enfoque experimental de las ciencias sociales con programas de acción social (Lewin, 1946). Años más tarde, fue empleada por Wood-Harper (1985) en investigaciones del área de Computación.

En 2012, surgió una variante de la AR, llamada Investigación-Acción Técnica o mejor conocida por sus siglas en inglés como TAR (*Technical Action-Research*), que está dirigida por un artefacto, en contraste con la AR tradicional, que está dirigida por problema (Wieringa y Morali, 2012).

2.2 Investigación-Acción Técnica (TAR)

El TAR (*Technical Action-Research*, Investigación-Acción Técnica) es una metodología de investigación que permite identificar una problemática, resolviéndola mediante el diseño de un artefacto concreto, el cual es probado en condiciones ideales, perfeccionado y luego probado en condiciones reales, para finalmente extraer las lecciones aprendidas (Engelsman y Wieringa, 2012). De este modo, el TAR permite cerrar la brecha entre las idealizaciones hechas al diseñar un artefacto y las condiciones concretas de práctica que ocurren en los problemas del mundo real (Wieringa y Morali, 2012).

El TAR propone la existencia de ciclos de ingeniería, que pueden iterarse tantas veces sean necesarias. En cada ciclo se crea un artefacto, al que se le somete a pruebas ideales, para posteriormente probarlo en condiciones más realistas, hasta que permita resolver problemas concretos (véase Figura 2-1).

Para facilitar la comprensión de los ciclos de ingeniería, se pueden comparar con lo que ocurre cuando se desarrollan nuevos medicamentos: primero se crean, después se prueban en condiciones idealizadas en un laboratorio, luego con voluntarios y eventualmente con pacientes.

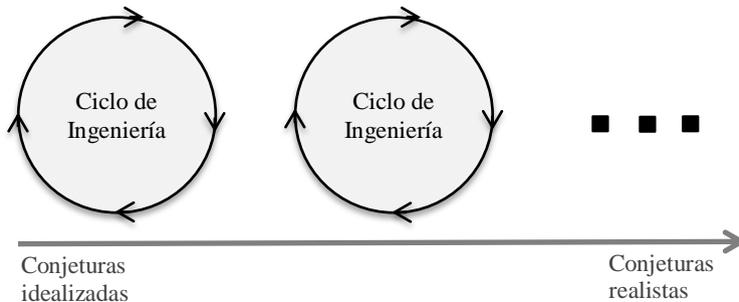


Figura 2-1. Iteraciones del TAR (Wieringa y Morali, 2012).

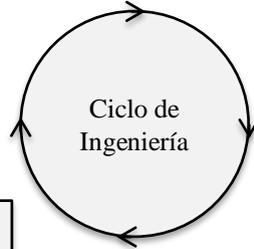
Cada ciclo de ingeniería del TAR (Wieringa y Morali, 2012) consta de cuatro fases (véase Figura 2-2):

1. **Investigación del problema.** Consiste en: a) identificar a los *stakeholders* o involucrados y sus objetivos, b) investigar los fenómenos relacionados con el problema y, c) evaluar los fenómenos para determinar si concuerdan con los objetivos de los involucrados.
2. **Diseño del tratamiento.** Consiste en diseñar un artefacto que interactúe con el contexto de un problema. Puede tratarse de software, hardware o entidades conceptuales como métodos y técnicas.
3. **Validación del diseño.** Consiste en validar el artefacto diseñado, por medio de la respuesta a dos cuestiones: 1) ¿cuáles serán los efectos esperados del artefacto diseñado en el contexto del problema? y, 2) ¿qué tan bien satisfarán estos efectos los criterios planteados (valor esperado de los efectos)? Asimismo, antes de pasar a la siguiente fase, la implementación, se deben analizar dos aspectos: a) el comportamiento del tratamiento planteado comparado con otros posibles (compensaciones) y, b) la sensibilidad, es decir, analizar si seguiría siendo efectivo y útil el tratamiento si el problema cambia.
4. **Implementación del tratamiento y evaluación.** Consiste en colocar el artefacto en el entorno y en evaluar los efectos, valor y sensibilidad al contexto del problema.

4. Implementación del tratamiento y evaluación

3. Validación del tratamiento

- A. ¿Efectos esperados en el contexto?
- B. ¿Evaluación de efectos?
- C. ¿Compensaciones?
- D. ¿Sensibilidad?



1. Implementación de la evaluación/ Investigación del problema

- A. ¿Involucrados, metas, criterios?
- B. ¿Fenómenos?
- C. ¿Evaluación?

2. Diseño del tratamiento

Figura 2-2. Ciclo de ingeniería del TAR (Wieringa y Morali, 2012).

2.3 Método de Mapeo Sistemático de Estudios (SMS)

Un mapeo sistemático de estudios, conocido en inglés por sus siglas SMS (*Systematic Mapping of Studies*), es un método que permite identificar, evaluar y resumir la evidencia disponible sobre un tema específico de investigación (Genero *et al.*, 2014), con la finalidad de conocer nuevos hallazgos y proponer ideas para futuras investigaciones (Zhang y Ali Babar, 2013). Un SMS se considera un estudio secundario y cada uno de los estudios individuales seleccionado, se denomina estudio primario (Kitchenham *et al.*, 2009).

Los SMS son revisiones de literatura (Kitchenham *et al.*, 2011) que se planifican formalmente y se ejecutan de manera sistemática y metódica, por lo que pueden ser replicados de forma independiente (Fernandez *et al.*, 2011).

El proceso para realizar un SMS, propuesto por Kitchenham (2002), está compuesto de tres fases secuenciales (véase Figura 2-3):

1. **Planificar.** El objetivo de esta fase es plantear el protocolo del SMS, el cual contiene todos los aspectos que harán que el mapeo sea sistemático y riguroso.

2. **Realizar.** En esta fase se pone en práctica lo planificado en el protocolo y se obtienen los resultados que responden a las preguntas de investigación definidas.
3. **Reportar.** Esta fase consiste en elaborar un informe que contenga todo el proceso del SMS, incluyendo las incidencias ocurridas para que pueda ser replicable.

Cada fase cuenta con varias tareas secuenciales, las cuales suelen iterarse varias veces cuando se está refinando el protocolo del SMS.

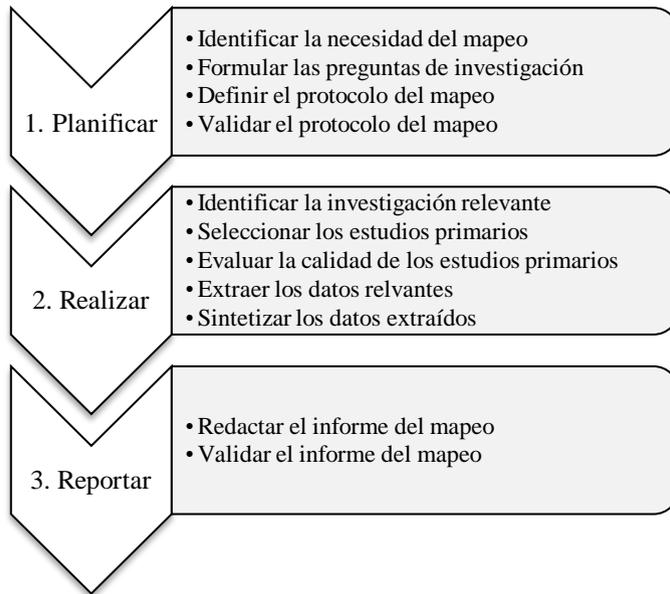


Figura 2-3. Proceso para realizar un Mapeo Sistemático de Estudios (Kitchenham y Charters, 2007; Genero *et al.*, 2014)

2.4 Experimentos

Un experimento es un método de investigación empírico que manipula una variable, llamada independiente o factor, perteneciente a un entorno o

fenómeno que se desea estudiar, para medir el efecto que tiene sobre otra variable, denominada variable dependiente (Genero *et al.*, 2014).

Los experimentos son apropiados para investigar diferentes aspectos, como confirmar teorías, probar que cierta relación se cumple, probar la presión de ciertos modelos, validar métricas, etc. (Hannay *et al.*, 2007). Es por esto que, el poder de los experimentos radica en demostrar si una afirmación, que es planteada como hipótesis, es verdadera o falsa y bajo qué situaciones lo es.

Los experimentos se pueden clasificar en: a) experimentos controlados, donde los sujetos se asignan de manera aleatoria y, b) cuasi-experimentos, donde no se usa la aleatorización.

Para llevar a cabo un experimento, según el proceso propuesto por Wohlin *et al.* (2012), es necesario realizar 5 actividades (véase Figura 2.4):

1. **Definición del alcance.** En esta actividad debe definirse la hipótesis; no es necesario que se precise formalmente, pero sí debe ser clara. Asimismo, a partir del problema a resolver, se establecen las metas y objetivos del experimento.
2. **Planificación.** En esta actividad se establecen las bases para realizar el experimento. Para ello, primero se define el contexto, por ejemplo, si se realizará en un ambiente estudiantil o en un entorno laboral. Después, se establece formalmente la hipótesis del experimento, incluyendo la hipótesis nula y la alternativa. Asimismo, se determinan las variables independientes o de entrada y las dependientes o de salida, incluyendo la escala de medición. Además, se identifican los sujetos de estudio y se selecciona un diseño de experimento adecuado. Para finalmente, evaluar la validez de los resultados.

Respecto a la validez del experimento, existen cuatro tipos que deben considerarse:

- a. **Interna,** se refiere a la validez dentro del entorno determinado y la confiabilidad de los resultados.

- b. Externa, esta validez permite conocer qué tan generales son los hallazgos.
 - c. De constructo, se refiere a la validez del instrumento de medida, para saber si se está midiendo lo que se dice medir.
 - d. De conclusión, se enfoca en la relación del tratamiento con el resultado del experimento.
3. **Operación.** Esta actividad se realiza en tres pasos: 1) la preparación, donde se elaboran los materiales, 2) la ejecución, donde debe cuidarse que el experimento se realice de acuerdo a lo planeado, incluyendo la recopilación de datos y, 3) la validación de datos obtenidos.
4. **Análisis e interpretación.** El primer paso de esta actividad consiste en tratar de entender los datos obtenidos por medio de estadísticas descriptivas, para contar con una panorámica general de la información. El siguiente paso, consiste en eliminar datos atípicos y en reducir el número de variables. Para después seleccionar la prueba estadística, aplicarla y determinar si la hipótesis fue rechazada o aceptada, interpretando los resultados del experimento.
5. **Presentación y difusión.** Esta actividad consiste en presentar los resultados en algún documento, por ejemplo, en un artículo de investigación. Para facilitar futuras réplicas del experimento, debe incluirse toda la información necesaria en dicho documento.

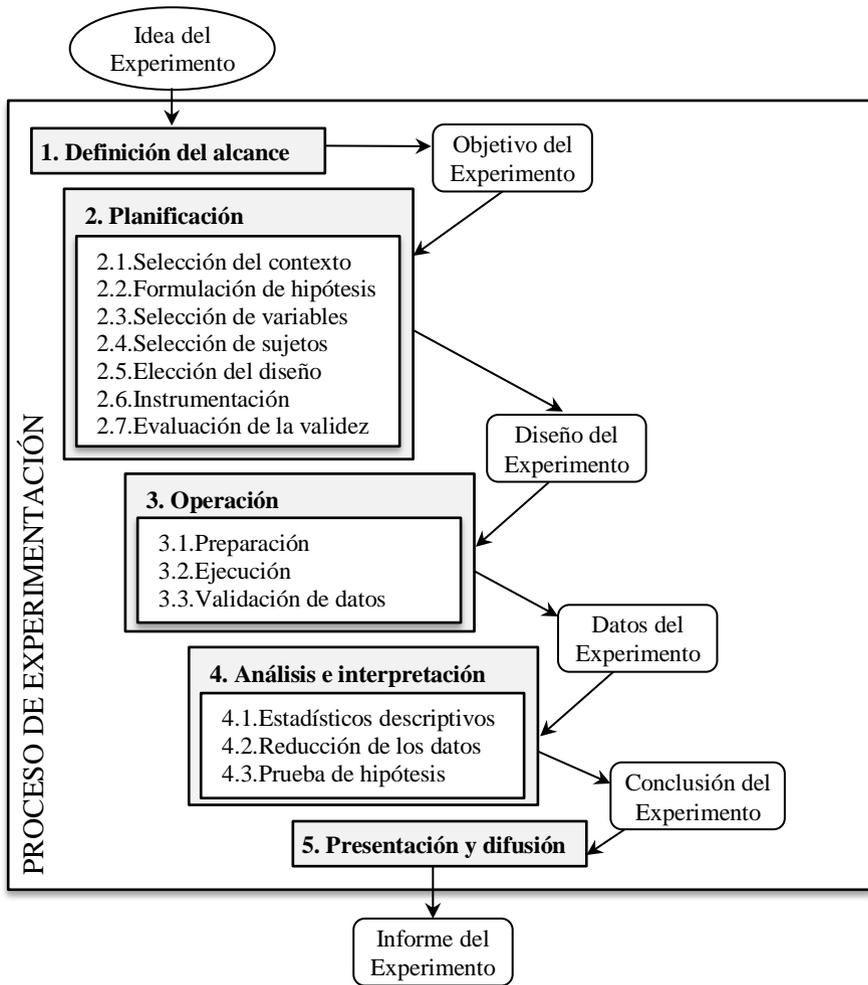


Figura 2-4. Proceso para realizar un Experimento (Wohlin *et al.*, 2012)

2.5 Estrategia de investigación de esta tesis

El proceso de investigación de esta tesis fue guiado por el TAR, incorporando la realización de un mapeo sistemático de estudios y un experimento. El proceso constó de 6 ciclos de ingeniería (véase Figura 2-5).

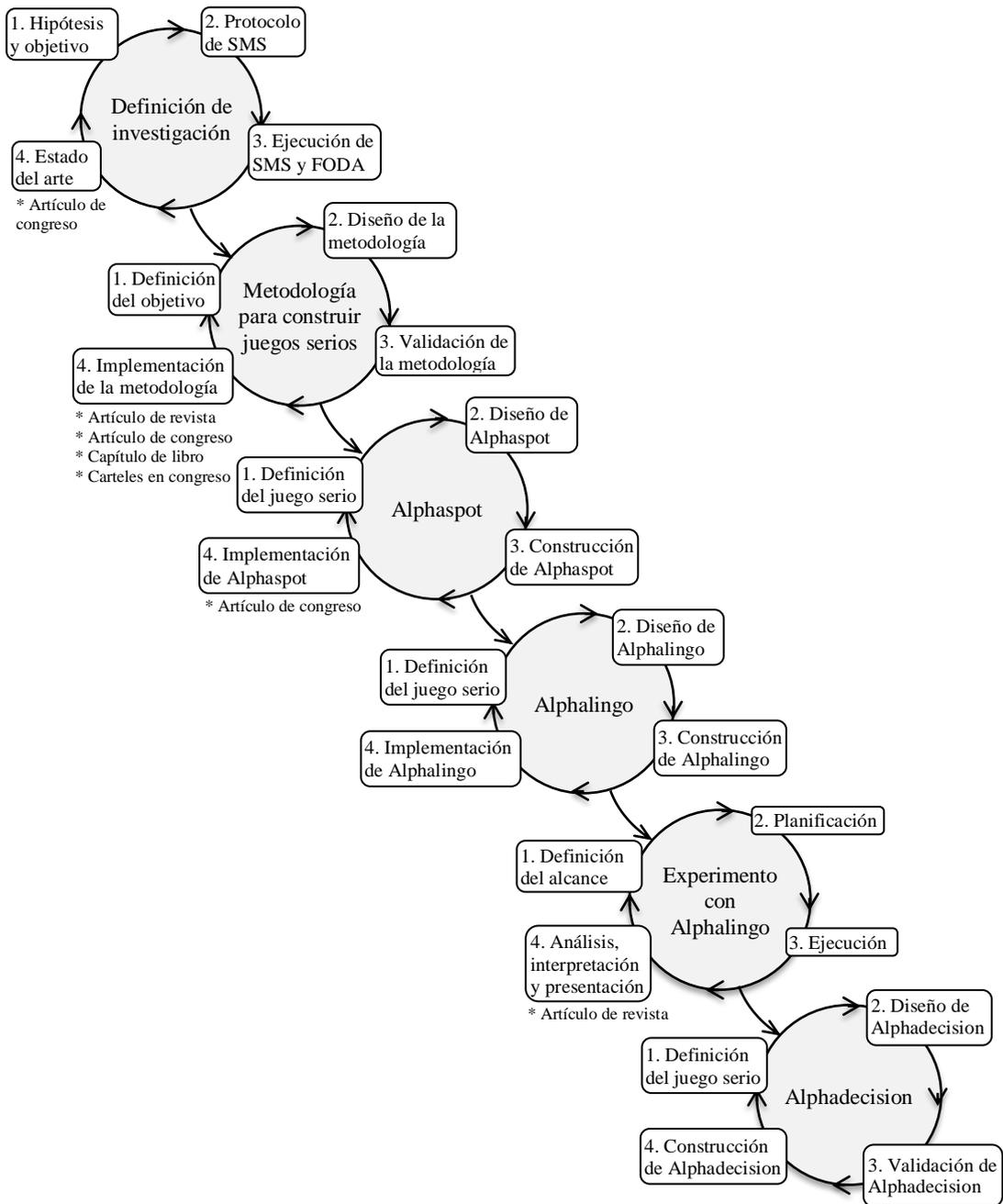


Figura 2-5. Proceso de investigación en esta tesis

2.5.1 Primer ciclo de ingeniería

El primer ciclo de ingeniería consistió en plantear la investigación. Para ello, en la fase 1, se definieron: la pregunta, la hipótesis, el objetivo general y los objetivos específicos de la investigación.

En la segunda fase de este ciclo, se elaboró el protocolo para realizar el SMS, con el objetivo de conocer el estado del arte de los juegos serios existentes para el aprendizaje de los procesos de Ingeniería de Software, los cuales se clasificaron haciendo uso del estándar ISO/IEC 12207 (IEEE, 2008).

En la fase 3, se llevó a cabo lo planteado en el protocolo del SMS. A partir de los estudios primarios identificados, se realizó un análisis por medio de la técnica conocida como FODA (Fortalezas, Oportunidades, Debilidades y Amenazas), lo que permitió identificar el área de contribución de la investigación (Humphrey, 2010; Pahl y Richter, 2009).

Finalmente, en la fase 4, se implementó la investigación, al validar que la idea planteada de forma inicial representa una aportación al área de contribución identificada. Los resultados de este ciclo fueron reportados por medio de un artículo (Jiménez-Hernández, Oktaba, Díaz-Barriga y Piattini, 2017c), que fue presentado y publicado en las memorias del “*5th International Conference in Software Engineering Research and Innovation*” (CONISOFT).

2.5.2 Segundo ciclo de ingeniería

En la primera fase de este ciclo, se definió el objetivo y el alcance de la metodología para construir juegos serios. En la fase 2 se diseñó la metodología a partir del análisis de los elementos incluidos en las metodologías empleadas en el desarrollo de algunos de los juegos serios pertenecientes a los estudios primarios del SMS (primer ciclo de ingeniería).

En la fase 3, se revisó la metodología diseñada. Para validarla se realizaron algunos cambios en su proceso, agregando los procesos y las etapas del paquete de despliegue de la norma ISO/IEC 29110 (2015).

Por último, se implementó la metodología para construir algunos juegos serios con estudiantes de nivel superior que participaron en el XX Verano de Investigación Científica y del Pacífico en la UNAM. Los juegos fueron presentados en carteles (Jiménez-Hernández *et al.*, 2015) en el “Primer Coloquio Internacional de Experiencias Educativas Mediadas por Tecnología”. En el mismo evento, se presentó y publicó un artículo sobre la metodología, el cual fue incluido en el libro “Construcción de buenas prácticas mediadas por tecnología” (Jiménez-Hernández y Oktaba, 2016). Posteriormente, se amplió la documentación de la metodología a través de la redacción de un capítulo del libro “*Software Engineering: Methods, Modeling, and Teaching, 4*” (Jiménez-Hernández *et al.*, 2016b).

Finalmente, uno de los juegos serios computacionales desarrollados como parte del verano de investigación, cuyo objetivo era facilitar el aprendizaje del álgebra booleana en alumnos de carreras relacionadas con la computación, fue publicado en la revista indexada “*Computer Applications in Engineering Education*” (Jiménez-Hernández, Oktaba, Díaz-Barriga y Piattini, 2020a).

2.5.3 Tercer ciclo de ingeniería

El primer juego serio computacional construido para facilitar el aprendizaje de las Alfas del núcleo del estándar Essence, se desarrolló en el tercer ciclo de ingeniería. Para lo cual, en la fase 1 se definieron los objetivos y el alcance del juego.

En la fase 2, se elaboró el diseño del juego serio de tipo videojuego llamado “Alphaspot” (Jiménez-Hernández *et al.*, 2015), usando como guía las prácticas descritas en la metodología resultante del segundo ciclo de ingeniería. Como parte del trabajo de diseño, se realizó la traducción en español del núcleo del estándar Essence, por tratarse del contenido de enseñanza (Jiménez-Hernández, 2018).

En la fase 3, se construyó Alphaspot. Y en la última fase, este juego serio fue probado por 12 profesionales de software que laboraban en una empresa de Morelia, Michoacán. Esto a través del instrumento de evaluación de la motivación, conocido por sus siglas en inglés como IMI (*Intrinsic Motivation Inventory*) (Yamabe y Nakajima, 2013).

Alphaspot fue presentado y publicado (Jiménez-Hernández *et al.*, 2016a), como parte de las memorias del “4th International Conference in Software Engineering Research and Innovation” (CONISOFT).

2.5.4 Cuarto ciclo de ingeniería

A partir de los resultados y las lecciones aprendidas del tercer ciclo de ingeniería con Alphaspot, se establecieron los objetivos para desarrollar un nuevo juego serio computacional para el aprendizaje de las Alfás del núcleo del estándar Essence, esto como parte de la fase 1 de este ciclo.

La fase 2, consistió en el diseño del nuevo juego serio computacional llamado Alphalingo (Jiménez-Hernández *et al.*, 2016c). Para lo cual, se usó como referencia la aplicación móvil para aprender idiomas llamada Doulingo© (Duolingo, 2013). Como parte del trabajo de diseño, se realizó la traducción de las Alfás de Essence al francés, la cual se encuentra disponible en (Jiménez-Hernández, 2018). Esto para poder proporcionar el contenido educativo a los aprendices en tres idiomas: inglés, español y francés.

En la fase 3, se construyó Alphalingo. Y en la última fase, este juego serio fue probado por algunos profesionales de software, quienes evaluaron su experiencia por medio del IMI. Esta información fue empleada como parte de la prueba piloto previa a la experimentación formal.

2.5.5 Quinto ciclo de ingeniería

El quinto ciclo de ingeniería consistió en realizar un experimento formal con una réplica, para probar estadísticamente si al usar Alphalingo con profesionales de software se obtenían mejores resultados que por medio del método de aprendizaje tradicional presencial. Para ello, en la fase 1 se definieron el alcance y los objetivos de la experimentación.

Asimismo, en la fase 2, se planificó tanto el experimento, como su la réplica. En la fase 3, se realizó la experimentación con profesionales de software en dos empresas de Guadalajara, Jalisco, donde se formaron dos grupos aleatoriamente: experimental, que aprendió por medio de Alphalingo y, de control, que aprendió por medio del método tradicional presencial.

Finalmente, en la fase 4 de este ciclo, se realizó el análisis y la interpretación de los resultados del experimento y su réplica, lo cual se reportó por medio de un artículo en la revista indexada “*IEEE Latin America Transactions*” (Jiménez-Hernández, Oktaba, Díaz-Barriga y Piattini, 2020b).

2.5.6 Sexto ciclo de ingeniería

En la fase 1 del último ciclo de ingeniería, se establecieron los objetivos y el alcance para desarrollar un nuevo juego serio computacional para el aprendizaje de las Alfas del núcleo del estándar Essence a un nivel superior, esto de acuerdo a la revisión de la Taxonomía de Bloom de Anderson y Krathwohl (2001). Para lo cual, se consideraron los resultados y las lecciones aprendidas del quinto ciclo de ingeniería con Alphalingo.

En la segunda fase, se diseñó la dinámica del juego serio llamado Alphadecision (Jiménez-Hernández *et al.*, 2017d), para lo que fue necesario, entre otras tareas, elaborar las 41 tarjetas con sus listas de verificación, que corresponden a los 41 estados de las Alfas; este material se encuentra disponible en (Jiménez-Hernández, 2018).

En la fase 3, se validó la dinámica y el contenido de Alphadecision por medio de un taller impartido en el CONISOFT 2017. Con los resultados obtenidos en la experiencia, se construyó el juego serio computacional basado en Web Alphadecision.

Capítulo 3

Estado del arte

*“El hombre nunca sabe de lo que
es capaz hasta que lo intenta”*
Charles Dickens (1812-1870)

Este capítulo presenta el estado del arte, el cual fue identificado por medio de un mapeo sistemático de estudios.

Además, contiene el resultado del análisis de las fortalezas, debilidades, oportunidades y amenazas, que se realizó sobre los estudios primarios seleccionados en el mapeo.

3.1 Mapeo sistemático de estudios de juegos serios para el aprendizaje de procesos de Ingeniería de Software

El método de investigación empleado para realizar este mapeo sistemático de estudios, o mejor conocido por sus siglas en inglés como SMS (*Systematic Mapping of Studies*), propuesto por Kitchenham y Charters (2007), comprende tres fases principales: planificar, realizar y reportar.

A continuación se presentan los resultados obtenidos en las tres fases del SMS realizado.

3.1.1 Planificar

En esta fase se planteó la pregunta de investigación del SMS, se seleccionaron las fuentes y la estrategia de investigación.

Además, se establecieron los criterios de inclusión/exclusión, la estrategia de extracción de la información y la evaluación de calidad de los estudios; todo esto como parte del protocolo de investigación del SMS.

3.1.1.1 Pregunta de investigación del mapeo sistemático de estudios

El SMS surgió a partir de la siguiente pregunta de investigación:

¿Qué juegos serios existen para el aprendizaje de los procesos del ciclo de vida de software?

La pregunta principal se dividió en sub-preguntas, con el objetivo de clasificar y resumir el conocimiento relacionado. Las sub-preguntas se presentan a continuación:

1. ¿Qué procesos del ciclo de vida de software se pueden aprender a través de juegos serios?
2. ¿Qué tipo de juegos serios se emplean para aprender los procesos del ciclo de vida de software?
3. ¿Cuál es la audiencia/público objetivo de los juegos serios diseñados para aprender procesos del ciclo de vida de software?
4. ¿Cuál es el esquema de interacción de los juegos serios diseñados para aprender procesos del ciclo de vida de software?
5. ¿Cuáles son los medios o materiales que emplean los juegos serios diseñados para aprender procesos del ciclo de vida de software?
6. ¿Cuál es el objetivo de los juegos serios diseñados para aprender procesos del ciclo de vida de software?
7. ¿Qué estilos de aprendizaje contemplan los juegos serios diseñados para aprender procesos del ciclo de vida de software?
8. ¿Cuál es el nivel de aprendizaje que pretenden alcanzar los juegos serios diseñados para aprender procesos del ciclo de vida de software?
9. ¿Cuál es el resultado que pretenden lograr los juegos serios diseñados para aprender procesos del ciclo de vida de software?

3.1.1.2 Fuentes y estrategia de investigación

En la Tabla 3-1, se encuentra la lista de Bases de Datos (BDs) electrónicas que fueron seleccionadas como fuentes de información.

Tabla 3-1. Fuentes de información seleccionadas

Base de Datos Electrónica	Sitio Web
Springer (SP)	www.springer.com
Science Direct (SD)	www.sciencedirect.com
Wiley Online Library (WILEY)	onlinelibrary.wiley.com
IEEE Xplore (IEEE)	ieeexplore.ieee.org
Scopus (SC)	www.scopus.com
ACM Digital Library (ACM)	dl.acm.org

La cadena de búsqueda empleada, se encuentra en la Tabla 3-2, la cual se compone de los sinónimos en inglés de los tres conceptos principales: Juego, Educación y Procesos del Ciclo de Vida de Software. Los sinónimos de un mismo concepto se unieron por medio del operador lógico “O”, en inglés *OR*; a su vez, los sinónimos de un concepto completo se unieron con otro a través de un “Y”, en inglés *AND*.

Tabla 3-2. Cadena de búsqueda

Concepto	Cadena de búsqueda
Game	(game OR play OR ludic OR serious game OR simulation)
	AND
Education	(education OR learning OR teaching OR training OR edutainment)
	AND
SLCPs	((software OR system) AND ((process) OR (engineering OR development OR project OR management OR design OR quality OR requirements)))

La búsqueda comprendió los estudios publicados en idioma inglés, entre el año 2000 y el 2016.

3.1.1.3 Criterios de selección

Los criterios de inclusión y exclusión aplicados se encuentran en la Tabla 3-3 y Tabla 3-4 respectivamente.

Tabla 3-3. Criterios de Inclusión

No.	Criterio de inclusión
1	Artículos que cumplan con la cadena de búsqueda.
2	Artículos de congreso, revista y talleres.
3	Artículos escritos en inglés.
4	Artículos publicados entre el año 2000 y diciembre de 2016.

Tabla 3-4. Criterios de Exclusión

No.	Criterio de exclusión
1	Estudios duplicados, misma investigación en publicaciones diferentes.
2	Artículos en formato de diapositivas, carteles, libros y reportes técnicos.

3.1.1.4 Estrategia de extracción de información

La estrategia de extracción de la información se diseñó a partir de las sub-preguntas planteadas en las Sección 3.1.1.1.

Con respecto a los procesos del ciclo de vida de software de los juegos serios (Sub-Pregunta 1), los estudios se clasificaron conforme a norma la ISO/IEC 12207 (2008), de la siguiente manera:

- a. Procesos del ciclo de vida del sistema: si el juego serio permite el aprendizaje de todos los procesos incluidos entre a.1 y a.4.
 - a.1. Procesos de acuerdo.
 - a.2. Procesos del proyecto.
 - a.3. Procesos técnicos.
 - a.4. Procesos organizativos de habilitación de proyectos.
- b. Procesos específicos del software: si el juego serio permie el aprendizaje de todos los procesos incluidos entre b.1 y b.3.
 - b.1. Procesos de implementación del software.
 - b.2. Procesos de soporte del software.
 - b.3. Procesos de reutilización de software.

Con respecto al tipo de juegos serios (Sub-Pregunta 2), los estudios se clasificaron de la siguiente manera:

- a. Edutainment: si el juego serio consistía en una combinación de educación y entretenimiento (Paraskeva *et al.*, 2010).
- b. Juego de entrenamiento: si el juego serio mejoraba las capacidades o aprendizajes de personas empleadas en una empresa de software (Prensky, 2007).
- c. Simulador: si el juego serio involucraba a los participantes en circunstancias simuladas con el objetivo de aprender algo (Warr y Eamonn, 2015).
- d. Juego persuasivo: si el juego influenciaba a los jugadores para tomar acciones (Visch *et al.*, 2013).
- e. Juego organizativo-dinámico: si el juego permitía aprender sobre la dinámica de organización en tres niveles: individual, grupal y cultural (Curz-Cunha, 2012).

Con respecto a la audiencia de los juegos serios (Sub- Pregunta 3), los estudios se clasificaron de la siguiente manera:

- a. Profesionales principiantes.
- b. Profesionales expertos.
- c. Estudiantes.

Con respecto al esquema de interacción de los juegos serios (Sub-Pregunta 4), los estudios se clasificaron de acuerdo a las comunidades de práctica (Wenger, 2002), de la siguiente forma:

- a. Esquema colaborativo.
- b. Esquema competitivo.
- c. Esquema individualista.

Con respecto a los medios y/o materiales empleados en los juegos serios (Sub-Pregunta 5), los estudios se clasificaron de la siguiente manera:

- a. Virtual: si el juego serio requería del uso de una computadora, tableta, teléfono, etc.

- b. Material: si el juego serio empleaba objetos físicos, como tarjetas o papeletas.
- c. Actuación: si el juego implicaba la actuación de un rol.
- d. Combinación de virtual y material.
- e. Combinación de virtual y actuación.
- f. Combinación de actuación y material.

Con respecto al objetivo de los juegos serios (Sub-Pregunta 6), los estudios se clasificaron de la siguiente manera:

- a. Educar.
- b. Modificar el comportamiento.

Con respecto a los estilos de aprendizaje empleados en los juegos serios (Sub-Pregunta 7), los estudios se clasificaron de acuerdo al modelo VARK (Hawk y Shah, 2007), de la siguiente forma:

- a. Visual.
- b. Auditivo.
- c. Lectura/Escritura.
- d. Kinestésico.
- e. Combinación de Lectura/Escritura y Kinestésico.
- f. Combinación de Lectura/Escritura y Visual.
- g. Combinación de Visual y Kinestésico.

Con respecto a los niveles de aprendizaje de los juegos serios (Sub-Pregunta 8), los estudios se clasificaron de acuerdo a la revisión de la Taxonomía de Bloom de Anderson y Krathwohl (2001), de la siguiente forma:

- a. Recordar.
- b. Comprender.
- c. Aplicar.
- d. Analizar.
- e. Evaluar.
- f. Crear.

Con respecto a los resultados que se obtienen con los juegos serios (Sub-Pregunta 9), los estudios se clasificaron de acuerdo al modelo KSA (Peterson *et al.*, 1999), de la siguiente forma:

- a. Conocimiento.
- b. Destrezas.
- c. Habilidades.
- d. Otras características: si después del uso de un juego serio, los jugadores modificaban o adquirirían nuevos intereses o experiencias.
- e. Combinación de conocimiento y habilidades.

3.1.1.5 Evaluación de calidad

Para evaluar la calidad de los estudios, se diseñó un cuestionario con cuatro afirmaciones (véase Tabla 3-5). Las tres primeras afirmaciones permitieron evaluar el diseño de la investigación, la cuarta permitió evaluar el rigor y la credibilidad de los estudios y la quinta permitió evaluar la relevancia.

Cada afirmación fue evaluada por medio de una escala de Likert de tres puntos (Likert, 1932), donde “1” significa “en desacuerdo”, 2 significa “parcialmente de acuerdo” y 3 significa “de acuerdo”.

El promedio de las calificaciones de las cinco afirmaciones del cuestionario, permitía calcular un porcentaje de calidad de cada artículo. De este modo, los estudios que tuvieron una calidad menor al 80% fueron excluidos.

Tabla 3-5. Cuestionario de evaluación de calidad

Afirmación a evaluar	Evaluación		
	1	2	3
1. El estudio presenta una descripción detallada del juego serio.			
2. El estudio proporciona pautas sobre cómo debe aplicarse el juego serio.			
3. El estudio describe claramente qué proceso del ciclo de vida del software se puede aprender con el juego serio.			
4. El estudio ha sido publicado en una revista o congreso relevante.			
5. El estudio muestra claramente los resultados de la aplicación del juego serio.			

3.1.2 Realizar

En esta fase, se llevó a cabo lo planificado en el protocolo del SMS. En la Figura 3-1 se muestra el proceso de selección con el número de artículos incluidos en cada etapa.

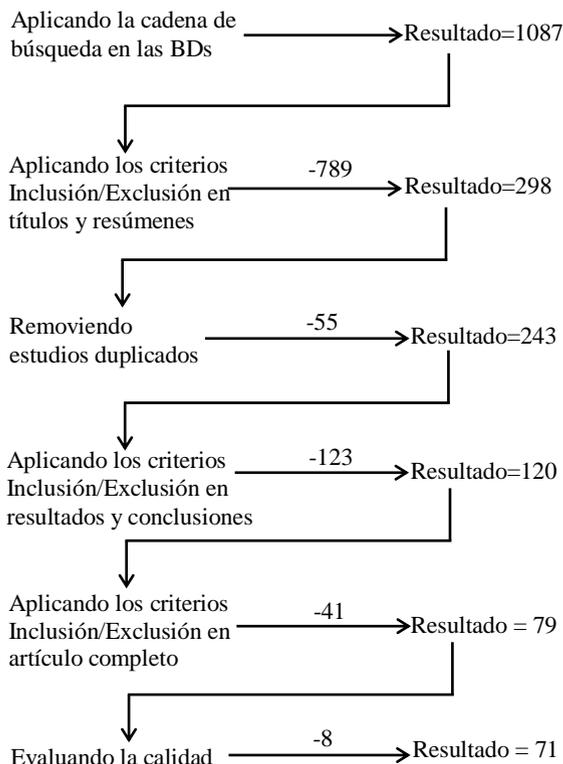


Figura 3-1. Proceso de selección de estudios

Las búsquedas que se realizaron en las seis BDs electrónicas, condujeron al hallazgo de 1087 estudios, de los cuales, después de aplicar los criterios de inclusión/exclusión en los títulos y resúmenes, se eliminaron 789, quedando en esta etapa 298 artículos.

De los 298 artículos, se removieron 55, debido a que se trataban de la misma investigación publicada en diferentes artículos. De los 243 artículos restantes,

se aplicaron los criterios de inclusión/exclusión en las secciones de resultados y conclusiones de los artículos, quedando 120 artículos en esta etapa.

Los 120 artículos fueron leídos en su totalidad para extraer la información que permitiera completar el formulario del Apéndice B. Aplicando los criterios de selección quedaron un total de 79 artículos, de los cuales se removieron 8, dado que después de evaluar la calidad, ésta era menor al 80% establecido. De este modo, se seleccionaron 71 estudios primarios (véase Apéndice C).

La clasificación de los estudios de acuerdo al formulario de extracción y la evaluación de calidad de cada estudio primario se encuentra en el Apéndice D.

3.1.3 Reportar

En la Tabla 3-6, se muestran los resultados del SMS realizado, los cuales fueron calculados a partir de la contabilización de los estudios primarios conforme a cada posible respuesta.

Tabla 3-6. Resultados del mapeo sistemático de estudios

Sub-Pregunta de investigación	Posibles respuestas	Resultados	
		Número	Porcentaje
1. ¿Qué proceso del ciclo de vida de software se aprende con el juego serio?	a. Procesos del ciclo de vida del sistema	30	42.2%
	a.1. Procesos de acuerdo	1	1.4%
	a.2. Procesos del proyecto	2	2.8%
	a.3. Procesos técnicos	11	15.5%
	a.4. Procesos organizativos de habilitación de proyectos	16	22.5%
	b. Procesos específicos del software	41	57.8%
	b.1. Procesos de implementación del software	27	38%
	b.2. Procesos de soporte del software	13	18.4%
	b.3. Procesos de reutilización de software	1	1.4%
2. ¿Cuál es el tipo de juego serio?	a. Edutainment	25	35.2%
	b. Juego de entrenamiento	7	9.9%
	c. Simulador	18	25.4%
	d. Juego persuasivo	9	12.7%

	e. Juego organizativo-dinámico	12	16.9%
3. ¿Cuál es la audiencia del juego serio?	a. Profesionales principiantes	7	9.9%
	b. Profesionales expertos	4	5.6%
	c. Estudiantes	60	84.5%
4. ¿Cuál es el esquema de interacción del juego serio?	a. Colaborativo	21	29.6%
	b. Competitivo	12	16.9%
	c. Individualista	38	53.5%
5. ¿Qué medios/materiales emplea el juego serio?	a. Virtual	49	69%
	b. Material	8	11.3%
	c. Actuación	7	9.9%
	d. Virtual y material	2	2.8%
	e. Virtual y actuación	3	4.2%
	f. Actuación y material	2	2.8%
6. ¿Cuál es el objetivo del juego serio?	a. Educar	55	77.5%
	b. Modificar el comportamiento	16	22.5%
7. ¿Qué estilo de aprendizaje emplea el juego serio?	a. Visual	10	14.1%
	b. Auditivo	0	0%
	c. Lectura/Escritura	8	11.3%
	d. Kinestésico	16	22.5%
	e. Lectura/Escritura y Kinestésico	15	21.1%
	f. Lectura/Escritura y Visual	17	23.9%
	g. Visual y Kinestésico	5	7%
8. ¿Cuál es el nivel de aprendizaje objetivo del juego serio?	a. Recordar	10	14.1%
	b. Comprender	26	36.6%
	c. Aplicar	12	16.9%
	d. Analizar	17	23.9%
	e. Evaluar	6	8.5%
	f. Crear	0	0%
9. ¿Cuál es el resultado que se obtiene con el juego serio?	a. Conocimiento	27	39.1%
	b. Destrezas	0	0%
	c. Habilidades	11	15.9%
	d. Otras características	2	2.9%
	e. Conocimiento y habilidades	31	44.9%

La Figura 3-2 muestra el número de juegos serios contabilizados según el proceso de ciclo de vida de software que enseñan. Los procesos relacionados con la implementación de software son el contenido de aprendizaje más común (27 estudios), la razón es que están asociados con la enseñanza de: a) elementos básicos de programación, como arreglos, tipos de datos, operadores, ciclos, etc., b) paradigmas de programación, como orientado a objetos, estructurado, etc., c) estructuras de datos, como árboles, colas y pilas, y d) lenguajes de programación, como C#, C++, .Net y HTML5. Todos estos temas forman parte del proceso de construcción de software, que pertenece a los procesos de implementación del software.

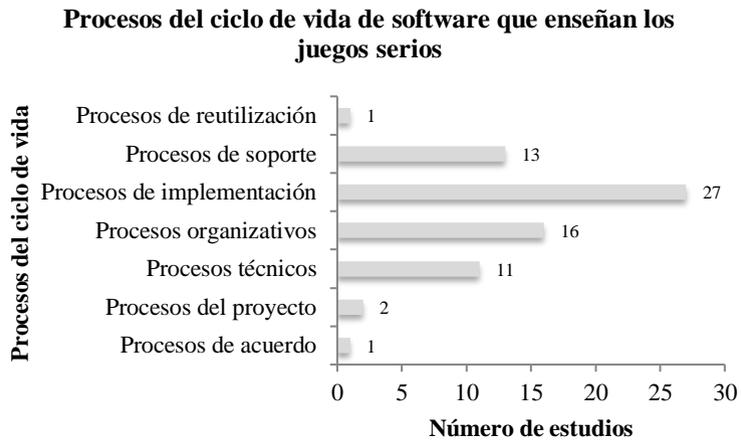


Figura 3-2. Procesos del ciclo de vida de software que enseñan los juegos serios

La Figura 3-3 muestra los resultados obtenidos en la Sub-Pregunta 2, la cual tiene que ver con el tipo de juego serio. El tipo de juego edutainment fue el que más implementado (25 estudios), mientras que el menos empleado fue el juego de entrenamiento (7 estudios).

Tipos de juegos serios de los estudios primarios

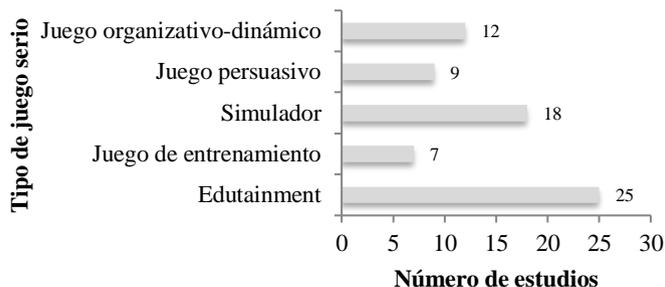


Figura 3-3. Tipos de juegos serios de los estudios primarios

En la Figura 3-4 se puede observar el número de juegos serios clasificados de acuerdo a su audiencia objetivo. Estos resultados se relacionan con los obtenidos respecto al tipo de juegos serios, debido a que el tipo menos encontrado fue el de entrenamiento, el cual está dirigido a capacitar/enseñar a personas laborando. En la misma gráfica se muestra que la mayoría de los juegos serios fueron diseñados para ser usados por estudiantes (60 estudios).

Público objetivo de los juegos serios

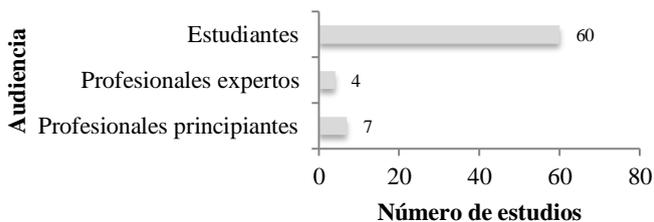


Figura 3-4. Audiencia de los juegos serios

Los resultados de los estudios primarios clasificados con base a las respuesta para la Sub-Pregunta 4, se presentan en la Figura 3-5. La gráfica muestra que la dinámica de interacción más empleada en los juegos serios es la individualista (38 estudios) y la menos usada es la competitiva (12 estudios). Estos resultados de nuevo están relacionados con la pregunta anterior, debido a el esquema

individualista permite calificar/evaluar a los estudiantes, porque como se observó, la mayoría de los juegos están dirigidos a estudiantes.

Dinámicas de interacción en los juegos serios

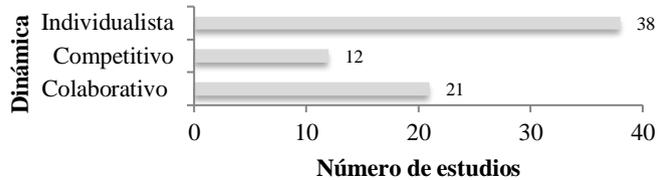


Figura 3-5. Esquema de interacción de los juegos serios

En la mayoría de los estudios primarios, los juegos serios fueron diseñados para ser jugador por medio de una computadora, siendo el medio virtual el más empleado (49 estudios). Y como lo muestra la Figura 3-6, los medios menos empleados son los que corresponden a combinaciones de actuación con medios materiales (2 estudios), y la combinación de material con virtual (2 estudios).

Medios de aprendizaje de los juegos serios

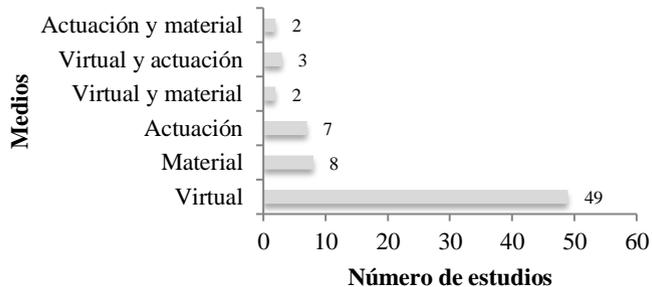


Figura 3-6. Medios/Materiales empleados por los juegos serios

Con respecto al objetivo de los juegos serios, se encontró que en la mayoría de ellos (55 estudios), el objetivo principal era educar (véase Figura 3-7). Esto está relacionado con los resultados anteriores, debido a que fueron diseñados para ser empleados por estudiantes.

Objetivo de los juegos serios

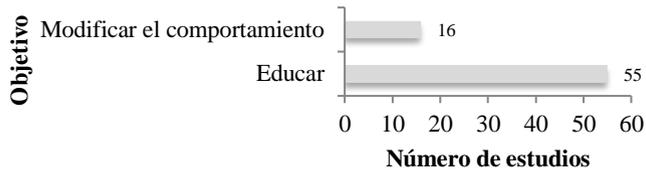


Figura 3-7. Objetivo de los juegos serios

La Figura 3-8 muestra que el estilo de aprendizaje que más se empleó en el diseño de los juegos serios pertenecientes a los estudios primarios, fue una combinación de lectura/escritura con visual (17 estudios), seguido por el estilo de aprendizaje kinestésico (16 estudios primarios). En los estudios seleccionados en el SMS, no se encontraron juegos que emplearan solamente el estilo auditivo.

Estilos de aprendizaje considerados en los juegos serios

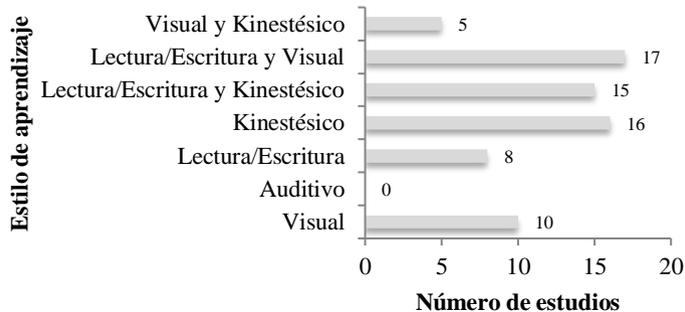


Figura 3-8. Estilo de aprendizaje empleado en los juegos serios

Con respecto a los artículos clasificados según las respuestas para la Sub-Pregunta 8, no se encontraron juegos serios que permitieran alcanzar el nivel más alto de aprendizaje, es decir “crear” (véase Figura 3-9). El nivel de aprendizaje que se encontró era obtenido con mayor frecuencia era el de “comprender” (26 estudios), seguido por el nivel de “analizar” (17 estudios).

Nivel del aprendizaje considerado en los juegos serios

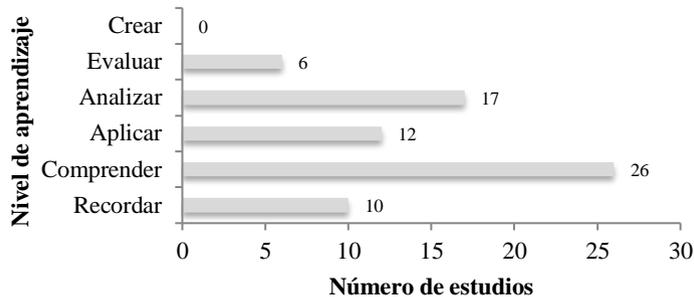


Figura 3-9. Niveles de aprendizaje objetivo de los juegos serios

Finalmente, de acuerdo a las posibles respuestas para la Sub-Pregunta 9, los juegos serios existentes permiten la obtención de conocimiento y habilidades en la mayoría de los casos (31 estudios).

Resultado que pretenden alcanzar los juegos serios

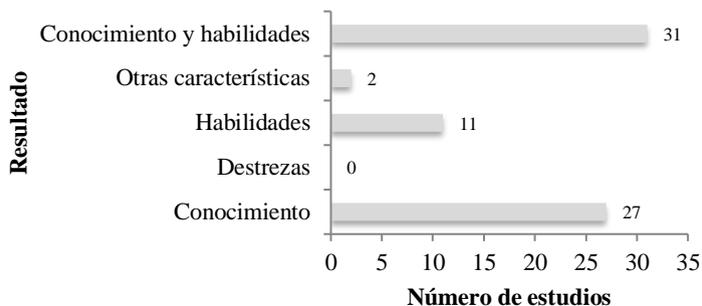


Figura 3-10. Resultados obtenidos al usar los juegos serios

En la Figura 3-11 se puede apreciar el porcentaje de estudios primarios clasificados de acuerdo al tipo de publicación que se estableció podría incluirse en el SMS. De este modo, se seleccionaron 5 artículos de taller (7%), 30 artículos de revista (42%) y 36 artículos de congreso (51%).

Tipos de publicaciones

■ Revistas ■ Conferencias ■ Talleres

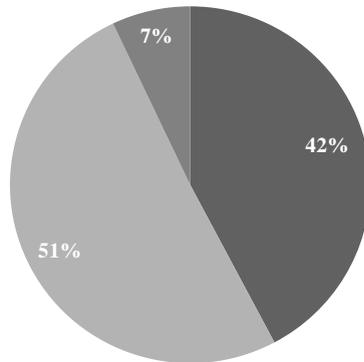


Figura 3-11. Tipos de publicación de los estudios primarios

La Figura 3-12 muestra los talleres a los que pertenecen los 5 estudios seleccionados de este tipo de publicación. Siendo el “*International Workshop on Games and Software Engineering (GAS)*” el que cuenta con más estudios.

Talleres a los que pertenecen los estudios primarios

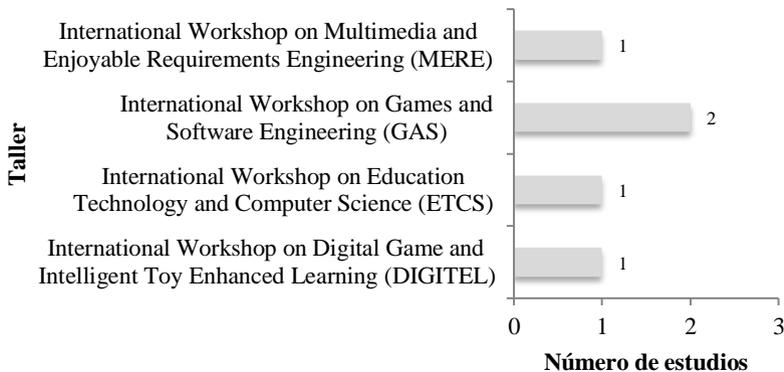


Figura 3-12. Talleres a los que pertenecen los estudios primarios

En la Figura 3-13 se muestra el número de estudios seleccionados en el SMS por congreso, siendo los congresos “*Frontiers in Education Conference (FIE)*” y “*Conference on Software Engineering Educational and Training (CSEE&T)*” a los que pertenecen más estudios primarios.

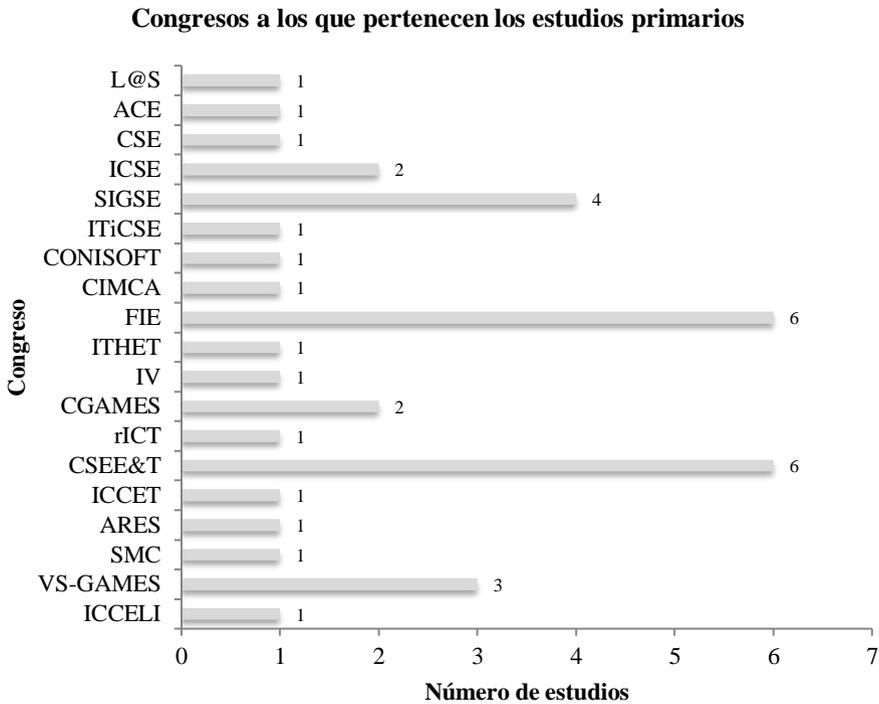


Figura 3-13. Congresos a los que pertenecen los estudios primarios

Respecto de las revistas, en la Figura 3-14 se puede observar que las dos revistas a las que pertenecen un mayor número de estudios primarios son: “*Computers & Education*” y “*Education Transaction Technologies*”.

Revistas a las que pertenecen los estudios primarios

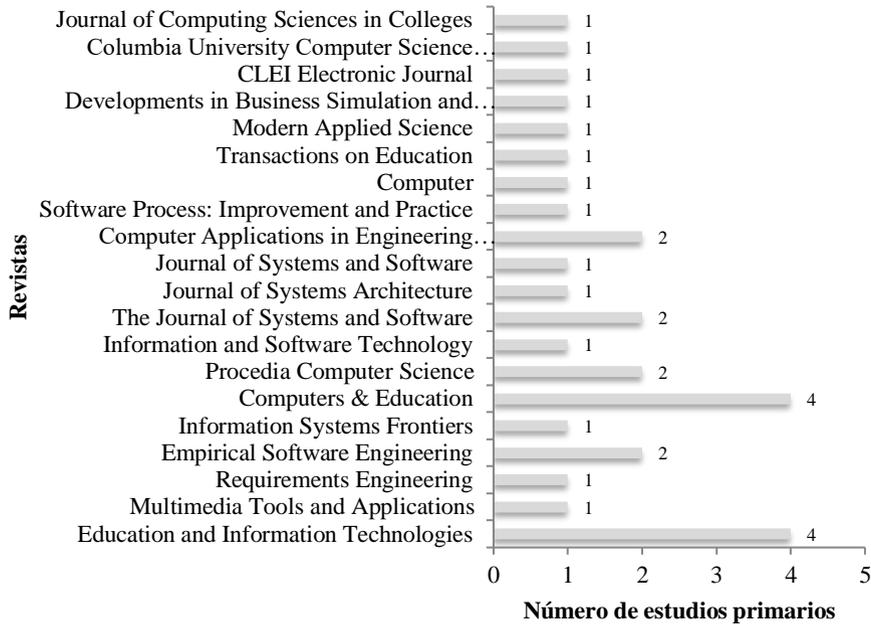


Figura 3-14. Revistas a las que pertenecen los estudios primarios

La Figura 3-15 presenta el número de estudios primarios publicados por año en cada Base de Datos electrónica seleccionada. Los resultados denotan un creciente interés por el desarrollo de juegos serios para el aprendizaje de procesos del ciclo de vida de software a partir del año 2005. En la misma gráfica, se puede observar que la Base de Datos electrónica con más estudios primarios fue IEEE (29 artículos), y en la que se encontraron menos estudios primarios fue Wiley (3 artículos).

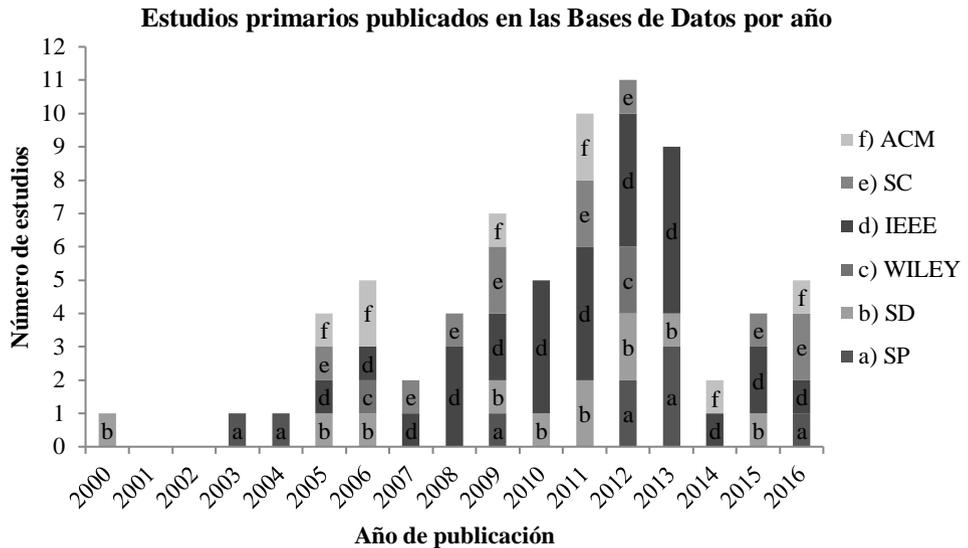


Figura 3-15. Estudios primarios publicados en las Bases de Datos por año

3.2 Análisis de Fortalezas, Oportunidades, Debilidades y Amenazas

Un análisis FODA es una técnica de investigación que permite estudiar la situación de un proyecto, institución o persona, con el fin de proponer una estrategia a futuro (Humphrey, 2010). Este tipo de análisis ha sido empleado en diversas áreas como la política, economía, etc. (Smith y Christensen, 1951; Fine, 2009).

Consta de los siguientes pasos (Humphrey, 2010):

1. **Análisis interno.** Se realiza por medio de la identificación de las debilidades y fortalezas.
2. **Análisis externo.** Se hace a través de la identificación de las amenazas y oportunidades.

3. **Determinación de la estrategia a emplear.** Consiste en corregir las debilidades, afrontar las amenazas, mantener las fortalezas y explorar las oportunidades.

3.2.1 Fortalezas

Las fortalezas identificadas en cada Sub-Pregunta de investigación, se presentan en la siguiente Tabla:

Tabla 3-7. Fortalezas de los estudios primarios

Sub-Pregunta	Fortaleza
1	Existen juegos serios para aprender casi todos los procesos del ciclo de vida de software.
2	Existen juegos de todos los tipos (edutainment, de entrenamiento, simulador, persuasivo y organizacional).
3	Existen varios juegos serios que guían sus esfuerzos hacia la enseñanza.
4	La mayoría de los juegos serios existentes (38%) tienen un esquema de interacción individualista, lo que permite evaluar con mayor precisión el aprendizaje de cada jugador.
5	La mayoría de los juegos serios se distinguen por el uso de la computadora como medio para enseñar (69%); la audiencia de los juegos está familiarizado con las computadoras.
6	En el 77.5% de los juegos serios, el objetivo principal es educar.
7	En los juegos serios encontrados, se emplearon casi todos los estilos de aprendizaje.
8	En el 67.7% de los estudios de juegos serios, es posible aprender desde el nivel de “recordar” hasta el nivel de “analizar”.
9	La mayoría de los estudios de juegos serios permiten a los jugadores adquirir conocimiento y habilidades (44.9%).

3.2.2 Oportunidades

Las oportunidades identificadas en cada Sub-Pregunta de investigación, se presentan en la siguiente Tabla:

Tabla 3-8. Oportunidades de los estudios primarios

Sub-Pregunta	Oportunidad
1	Diseñar juegos serios que permitan aprender procesos de acuerdo, de reutilización de software y de proyectos de software, porque son los menos explotados.
2	Crear juegos serios persuasivos o de entrenamiento, porque son los más adecuados para la formación de profesionales de software.
3	Diseñar juegos serios para profesionales de software, ya que existen pocos juegos serios para ellos.
4	Diseñar juegos serios para aprender procesos de software por medio de un esquema de interacción colaborativa, ya que es más compatible con la forma en que se crea el software en el mundo real.
5	Diseñar juegos serios que combinen el uso de computadoras, materiales físicos y actuación. Esto permitiría a los jugadores tener diferentes opciones con respecto al aprendizaje de procesos de software.
6	Diseñar juegos serios con los que sea posible cambiar el comportamiento, porque este es uno de los mayores problemas durante la adopción de nuevas prácticas.
7	Construir juegos serios que empleen todos los estilos de aprendizaje (visual, auditivo, de lectura/escritura y kinestésico).
8	Diseñar juegos que permitan alcanzar los niveles más altos de aprendizaje (evaluar y crear).
9	Crear juegos serios que permitan adquirir tanto conocimiento, como habilidades y otras características.

3.2.3 Debilidades

Las debilidades identificadas en cada Sub-Pregunta de investigación, se presentan en la siguiente Tabla:

Tabla 3-9. Debilidades de los estudios primarios

Sub-Pregunta	Debilidad
1	Sólo se encontró un estudio primario de un juego serio para aprender los procesos de acuerdo y otro para los procesos de reutilización de software. No se encontró algún juego serio que permitiera aprender todos los procesos del ciclo de vida de software.
2	Sólo un 9.9% de los estudios seleccionados son juegos serios de entrenamiento y un 12.7% de tipo persuasivo.

- 3 Sólo un 15.5% de los estudios primarios está dirigidos a profesionales de software (principiantes y expertos).
- 4 En la práctica, el software se construye en un equipo. Sólo el 29.6% de los estudios de juegos serios permiten el aprendizaje de los procesos de software en un entorno colaborativo.
- 5 Existen pocos estudios de juegos serios que combinen diversos medios y materiales para enseñar sobre los procesos de software. Sólo en el 2.8% de los estudios se combinan el medio virtual y la actuación, otro 2.8% combinan la actuación y materiales físicos.
- 6 Hay pocos estudios de juegos serios dirigidos a cambiar comportamientos (22.5%).
- 7 No se encontraron estudios de juegos serios que enseñaran procesos de software de forma auditiva, ni que involucren todos los tipos de aprendizaje (visual, auditivo, de lectura/escritura y kinestésico).
- 8 No se encontraron estudios de juegos serios que permitieran alcanzar el nivel superior de aprendizaje “crear”, y sólo una minoría permite alcanzar el nivel de “evaluar” (8.5%).
- 9 No existen estudios de juegos serios que permitan la adquisición de habilidades relacionadas con los procesos de software. Y sólo un 2.9% de los estudios permiten adquirir otras características a los jugadores, como interés.
-

3.2.4 Amenazas

Las amenazas identificadas en cada Sub-Pregunta de investigación, se presentan en la siguiente Tabla:

Tabla 3-10. Amenazas de los estudios primarios

Sub-Pregunta	Amenaza
1	Como se menciona en Abrahamsson <i>et al.</i> (2003), cada vez se usan más los métodos ágiles en la industria de software, esto desplaza el uso de procesos de desarrollo de software tradicionales como la ISO/IEC 12207.
2	Si se desarrollan juegos serios con la intención de reforzar conocimientos o persuadir, debe tomarse en cuenta la resistencia natural que existe en las personas, ya que implica cambiar la forma habitual de realizar sus tareas de trabajo (Bogost, 2007).
3	No es tarea sencilla hacer ver el valor que representa la capacitación de empleados a los directores/propietarios de las entidades de software, y menos aún cuando se realiza a través del juego

- (Johannesson y Blacklund, 2007).
- 4 Es más difícil evaluar el aprendizaje de un grupo de personas, que el de una sola (Hawk y Shah, 2007).
 - 5 Si se desarrollan juegos virtuales serios, será difícil "competir" con los gráficos de la industria de los videojuegos (Johannesson y Blacklund, 2007).
 - 6 Es complejo cambiar el comportamiento, porque implica cambiar la personalidad, los intereses, etc. (Peterson *et al.*, 1999).
 - 7 No es trivial diseñar juegos serios para el aprendizaje de procesos de software que contengan todos estilos de aprendizaje (Hawk y Shah, 2007).
 - 8 La taxonomía propuesta por Anderson y Krathwohl es jerárquica (2001), por lo que los juegos serios deben considerar que para alcanzar los niveles más altos de aprendizaje, es necesario adquirir niveles previos.
 - 9 Los jugadores pueden adquirir conocimientos, habilidades y otras características, pero es necesario que adquieran competencias genéricas y no sólo competencias específicas, lo cual no es una tarea trivial (González y Wagenaar, 2003).
-

3.2.5 Área de oportunidad

El análisis de las Fortalezas, Oportunidades, Debilidades y Amenazas de los estudios primarios permitió encontrar el área de contribución siguiente:

Construir juegos serios de tipo de entrenamiento (capacitación), dirigidos a profesionales de software, que faciliten el aprendizaje de los procesos de acuerdo, procesos de proyecto o bien, el ciclo de vida de software en su totalidad. Que consideren todos los estilos de aprendizaje y que permitan alcanzar los niveles superiores de la revisión de la Taxonomía de Bloom.

El proceso llevado a cabo para identificar el estado del arte y el área de oportunidad fue reportado por medio de un artículo de congreso (Jiménez-Hernández, Oktaba, Díaz-Barriga y Piattini, 2017c).

Parte II

Juegos Serios

Capítulo 4

Fundamentos Pedagógicos

*“El juego es el trabajo
de la infancia”*

Jean Piaget (1896–1980)

Este capítulo presenta las teorías, modelos y principios pedagógicos que fueron objeto de revisión en el trabajo de investigación de esta tesis.

4.1 Teorías Pedagógicas

Las teorías pedagógicas son marcos conceptuales que describen los procesos mediante los cuales se consigue el aprendizaje. Las diversas teorías, enmarcadas en distintas corrientes del pensamiento, intentan entender, anticipar y regular la conducta por medio de estrategias que faciliten la adquisición y retención del conocimiento (Spector, 2016).

A continuación se presentan algunas teorías de aprendizaje y perspectivas psicológicas relevantes.

4.1.1 Conductismo

Los principales representantes del conductismo son: John B. Watson (1878–1958), Burrhus F. Skinner (1904–1990) y Edward L. Thorndike (1874–1949).

Las teorías conductistas centran el aprendizaje en cosas directamente observables. La idea principal del conductismo comprende lo siguiente:

- El aprendizaje es un proceso que consiste en una serie de intentos y errores graduales, los cuales se efectúan hasta que se logra el aprendizaje.
- La clave del éxito en el aprendizaje es el refuerzo.
- El aprendizaje involucra una secuencia de estímulo-respuesta.

4.1.2 Cognitivismo

Los principales representantes del cognitivismo son: Jerome S. Bruner (1915–2016) y Robert M. Gagné (1916–2002).

En la psicología cognitiva, el aprendizaje se concibe como la adquisición de conocimiento: el alumno es un procesador que recibe información, realiza operaciones cognitivas sobre ella y la almacena en la memoria.

Según el cognitivismo, el aprendizaje no es una secuencia de estímulo-respuesta, sino la formación de estructuras cognitivas. Los estudiantes no sólo reciben estímulos mecánicamente y reaccionan pasivamente, sino que procesan los estímulos y determinan las respuestas adecuadas.

4.1.3 Constructivismo

Los principales representantes del constructivismo son Jean Piaget (1896–1980), David P. Ausubel (1918–2008) y Lev Vygotsky (1896–1934).

El constructivismo es un enfoque pedagógico que centra el aprendizaje en el estudiante, quien es concebido como un agente activo de su propio aprendizaje, capaz de construir conocimiento (Díaz-Barriga y Hernández-Rojas, 2010). Desde el constructivismo, el aprendizaje puede entenderse de la siguiente manera (Coll, 2001):

- El aprendizaje está o debería estar centrado en el aprendiz.
- El aprendizaje es el proceso mediante el cual los alumnos construyen activamente la representación psicológica interna.
- El proceso de aprendizaje consta de dos aspectos: la reorganización y reconstrucción de conocimientos antiguos y la construcción significativa de nuevos conocimientos.
- El aprendizaje no es sólo un comportamiento individualizado, sino también un comportamiento social y centrado en el lenguaje; el aprendizaje requiere comunicación y cooperación.
- El aprendizaje efectivo requiere fuentes apropiadas para apoyar la construcción de significado.

4.1.3.1 Teoría de la Psicología genética

Esta teoría, propuesta por Jean William Fritz Piaget (1896-1980), estudia la génesis del conocimiento y los procesos que intervienen en ella, sosteniendo que la competencia cognitiva depende del desarrollo intelectual, el cual evoluciona a través de cuatro etapas (Piaget, 1970): a) etapa sensorio-motriz, que comienza al nacer y se prolonga por dos años; b) etapa pre-operacional, que comprende de los 2 a los 6 años; c) etapa de las operaciones concretas, entre los 6 y 12 años y, d) etapa de las operaciones formales, de 12 a 16 años. Estas etapas caracterizan lo que las personas pueden o no pueden aprender de acuerdo a su edad, lo que permite diseñar estrategias específicas que favorezcan el aprendizaje.

Para Piaget, el desarrollo intelectual es un proceso de auto estructuración del conocimiento, que inicia con el descubrimiento de algo, generando conflictos cognitivos, los cuales son resueltos mediante las actividades intelectuales propias de un individuo, estructurando el conocimiento y generando un estado de equilibrio. Asimismo, refiere que los mecanismos de asimilación y acomodación, son los que determinan la evolución del hombre. Con la asimilación, se produce la interiorización de los conceptos y, por medio de la acomodación, los conceptos se modifican para recibir nuevos conocimientos y eventos.

4.1.3.2 Teoría de la Psicología cognitiva

Esta teoría, propuesta por David Ausubel (1918-2008), sostiene que el aprendizaje debe ser significativo para el alumno. Ausubel parte de la existencia de una estructura cognitiva jerárquica, que organiza todos los conocimientos adquiridos. De este modo, cuando se aprende algo nuevo, entra en la estructura cognitiva y al unirse con el conocimiento existente, se forma el significado (Ausubel, 1968).

El aprendizaje significativo se aloja en la memoria a largo plazo, mientras que el aprendizaje no significativo, se ubica en la memoria de corto plazo, por lo que se desvanece con mayor facilidad.

Como medio de enseñanza, Ausubel propone el uso de mapas mentales, los cuales permiten estructurar el conocimiento de lo general a lo particular, y representar las relaciones entre los conceptos e ideas (Ausubel, 1976).

4.1.3.3 Constructivismo social

Esta teoría fue propuesta por Lev Semiónovich Vygotsky (1896-1934), quien es considerado, por su prolífica obra y muerte temprana, como “el Mozart de la Psicología”.

Vygotsky, en su obra más importante “Pensamiento y lenguaje”, sugiere que el aprendizaje es en gran medida un proceso social. Que el ser humano adquiere conocimientos y experiencias del medio donde se encuentra inmerso (Vigotsky, 1993).

También plantea la existencia de “funciones mentales elementales”: atención, sensación, percepción y memoria. Las cuales evolucionan en procesos mentales más complejos o en las llamadas “funciones mentales superiores”, como el pensamiento crítico, la toma de decisiones y el razonamiento. Vigotsky sostiene que estas funciones cognitivas, tienen un origen social, que varía de una cultura a otra, dependiendo de las creencias, valores y herramientas de adaptación intelectual prevalecientes.

Un concepto importante en esta teoría es la “Zona de Desarrollo Próximo”, que es una estructura de apoyo basada en el aprendizaje guiado (por un experto) y cooperativo (entre un grupo de personas).

4.1.4 Conectivismo

Los principales representantes del conectivismo son George Siemens (1970-) y Stephen Downes (1959-).

El conectivismo enfatiza el papel del contexto social y cultural. Sostiene que el aprendizaje es como una red con nodos y conexiones, donde cada nodo puede tratarse de información, datos, sentimientos e imágenes. De este modo, el conocimiento se estructura como una red y el aprendizaje es un proceso de reconocimiento de patrones (Siemens, 2005).

4.1.5 Humanismo

Los principales representantes del humanismo son Abraham Maslow (1908–1970) y Carl Rogers (1902–1987).

El humanismo propone que cada persona tiene la capacidad de desarrollar su propio potencial y motivación. Los individuos pueden elegir libremente su propia dirección y valor de desarrollo. El humanismo investiga cómo crear un buen entorno para que los alumnos perciban el mundo desde su punto de vista y desarrollen una comprensión del mundo, con el objetivo de alcanzar el nivel más alto de autorrealización.

4.2 Teorías Psicopedagógicas Lúdicas

4.2.1 Teoría de Johan Huizinga

En 1938, Johan Huizinga (1872-1945) resalta, en su obra titulada “Homo Ludens”, la importancia del juego en la vida del hombre.

Huizinga concibe el juego como una actividad tan esencial como la reflexión (*Homo Sapiens*) y el trabajo (*Homo Faber*), planteando la caracterización del hombre como un *Homo Ludens*. Sostiene que el juego es más antiguo que la cultura, ya que, la cultura nace y se desarrolla en el juego.

Define al juego como “*una acción o una actividad voluntaria, realizada en ciertos límites fijados de tiempo y lugar, según una regla libremente aceptada, pero completamente imperiosa y provista de un fin en sí, acompañada de un sentimiento de tensión y de alegría*” (Huizinga, 1949).

4.2.2 Teoría de Karl Groos

Para Karl Groos (1861-1946), el juego es un ejercicio preparatorio para la vida, expresado de otro modo, planteó el juego como escuela de vida.

Groos se inspiró en la obra de Charles Darwin (1809-1882), conocida como “El origen de las especies” (Darwin, 1859; Darwin, 1877), para crear su obra

“El juego de los animales” (1902), donde plantea que los animales juegan para aprender ciertas conductas, y sobrevivir.

En su teoría, distingue varias categorías de juegos:

- Juego de experimentación, basado en el instinto de destrucción sistemática de objetos.
- Juegos de locomoción, que tiene como base un cambio de lugar.
- Juegos cinegéticos, que se realizan con una presa real, imaginaria o inanimada.
- Juegos de combate, que tienen como base establecer estructuras jerárquicas dentro de un grupo.
- Juegos arquitectónicos, que tienen como base la construcción.
- Juegos de coquetería o reproductivos.
- Juegos tróficos, donde se finge criar o alimentar a otro ser vivo.
- Juegos imitativos y de curiosidad.

4.2.3 Teoría de Lev Vigotsky

Lev Vigotsky sugiere que el juego surge como una necesidad de reproducir la interacción con los demás, donde el ser humano participa activamente pero no solo (Vigotsky, 1993).

Establece que el juego es una actividad social, en la que debido a la cooperación con otros individuos, se desempeñan roles complementarios. También propone que el juego es simbólico, porque permite transformar objetos por medio de la imaginación, en algo distinto.

4.2.4 Teoría de Jean Piaget

Jean Piaget, en su teoría psicogenética, plantea que el juego forma parte de la inteligencia del niño (Piaget, 1952), dado que representa la asimilación de la realidad en cada periodo evolutivo (sensorio-motriz, pre-operacional, de las operaciones concretas y de las operaciones formales).

Sostiene que existen tres estructuras básicas del juego, que evolucionan de acuerdo al pensamiento humano: a) el juego como simple ejercicio, parecido a

como lo hacen los animales; b) el juego simbólico, que es abstracto y ficticio y; c) el juego regulado, que es colectivo, resultado de un acuerdo grupal.

4.2.5 Teoría de Paul Gee

Paul Gee (1948-) plantea en su teoría, que los videojuegos no sólo son una fuente de entretenimiento, sino también un modelo a seguir en las escuelas. Considera que una buena enseñanza, debe incluir muchas de las características de los videojuegos como “*la interactividad, posibilidades de práctica constante, tolerancia al error, incorporación de una nueva identidad, sentido de logro*”, metacognición (autorregulación del proceso de aprendizaje), motivación constante, entre otras” (Gee, 2003).

Gee sostiene que los textos existentes relacionados con los videojuegos, como manuales y sitios Web, no logran comprenderse la mayoría de las veces hasta que se relacionan con experiencias vividas, es decir, hasta que se juega. Y que esta práctica ocurre en los textos usados en la educación tradicional, donde éstos no significan mucho para los alumnos hasta que se relacionan con experiencias.

Es así que, cuando una persona aprende a jugar un videojuego, aprende un nuevo alfabeto técnico especializado que nace de generalizaciones abstractas. Al jugar, las personas realizan en sus cerebros simulaciones de imágenes, acciones y experiencias, lo que proporciona guías para tomar decisiones.

Gee centra su teoría en 36 principios de aprendizaje presentes en los videojuegos, que pueden ser empleados en el aula:

1. **Principio de aprendizaje activo y crítico.** El entorno de enseñanza en el aula debe alentar el aprendizaje activo y crítico, sustituyendo al aprendizaje pasivo, lo que le permitirá al alumno participar y reflexionar, tal como sucede en los videojuegos.
2. **Principio de diseño.** Reside en conocer y apreciar los principios de diseño en videojuegos para usarlos en el aula.

3. **Principio semiótico.** Implica identificar, comprender y apreciar las relaciones entre los sistemas de símbolos, los cuales pueden ser palabras, imágenes, acciones, artefactos, etc.
4. **Principio de dominio semiótico.** Consiste en establecer el dominio o área donde se emplearán los sistemas de símbolos (del principio semiótico), para transmitir el contenido que será objeto de aprendizaje.
5. **Pensamiento de meta-nivel sobre el principio de dominio semiótico.** Radica en que el aprendizaje les permita a los estudiantes inferir las relaciones que existen entre los múltiples dominios semióticos.
6. **Principio de moratoria psicosocial.** En los videojuegos las consecuencias del mundo real no existe, lo que les permite a los jugadores tomar mayores riesgos. De forma similar, los entornos de aprendizaje en el aula debe permitirle al alumno realizar múltiples intentos hacia un objetivo determinado, lo que fomentará su participación.
7. **Principio de aprendizaje comprometido.** Consiste en buscar la participación comprometida del alumno, asemejando la forma en la que los videojuegos logran comprometer a los jugadores para completar una misión.
8. **Principio de identidad.** Se basa en la idea de que los estudiantes poseen una identidad del mundo real, una virtual y una proyectiva.
9. **Principio de autoconocimiento.** Se sustenta en la premisa de un proceso auto reflexivo, donde el alumno aprende sobre sí mismo y es capaz de identificar el rango en el que puede potenciar sus habilidades.
10. **Amplificación del principio de entrada.** Consiste en identificar cuánto esfuerzo necesita el alumno para recibir una recompensa.
11. **Principio de logro.** Se basa en que el alumno necesita recompensas intrínsecas adaptadas al nivel, esfuerzo y dominio del contenido educativo.

12. **Principio de práctica.** Sugiere que los estudiantes necesitan de práctica, que los comprometa y los motive, en el contexto de aprendizaje.
13. **Principio de aprendizaje continuo.** Reside la concepción de que el alumno debe pasar por varios ciclos de aprendizaje.
14. **Principio del régimen de competencia.** Consiste en que el aprendizaje resulte desafiante para el alumno, pero no imposible de lograr.
15. **Principio de prueba.** Se fundamenta en la idea de que el aprendizaje se logra al comprometerse, reflexionando sobre las acciones realizadas, formulando hipótesis, probando las hipótesis y aceptando o reconsiderando las hipótesis.
16. **Principio de rutas múltiples.** Radica en brindarles a los alumnos varias opciones para lograr el aprendizaje, tomando en cuenta sus fortalezas, debilidades y estilos de aprendizaje específicos.
17. **Principio de significado situado.** Radica en que todos los significados de los símbolos (principio semiótico) deben situarse en experiencias para el alumno.
18. **Principio del texto.** Sostiene que los textos no sólo se comprenden al entender las palabras, sino por medio de la experiencia.
19. **Principio intertextual.** Postula que los estudiantes comprenden los textos a través de la experiencia, y son capaces de relacionar los significados de varios textos.
20. **Principio multimodal.** Reside en la posibilidad de aprender de diferentes modos, además de palabras. Por ejemplo, a través de sonidos e imágenes.
21. **Principio de inteligencia material.** Plantea que las tecnologías, los objetos materiales y el entorno contienen información, a la que un alumno puede acceder por medio de la interacción.

22. **Principio de conocimiento intuitivo.** Establece que no todo el conocimiento se puede verbalizar, como el conocimiento que se adquiere al practicar una tarea.
23. **Principio del subconjunto.** Sostiene que el aprendizaje tiene lugar primero, en un subconjunto simplificado del dominio real.
24. **Principio incremental.** Establece que los estudiantes crean conexiones en etapas iniciales y fáciles, las cuales les ayudan en etapas posteriores (más difíciles).
25. **Principio de muestra concentrada.** Reside en la importancia de que los alumnos adquieran experiencias con conceptos y acciones fundamentales.
26. **Principio de habilidades básicas ascendentes.** Sostiene que las habilidades básicas se aprenden en contexto.
27. **Principio de información explícita a demanda.** Establece que la información debe proporcionarse al alumno en momentos cruciales para maximizar el aprendizaje.
28. **Principio de descubrimiento.** Consiste en exponer poco al alumno de forma explícita, incentivándolo a que explore y descubra por sí mismo.
29. **Principio de transferencia.** Radica en la posibilidad que tienen los estudiantes de aplicar el aprendizaje de etapas anteriores, en etapas posteriores.
30. **Principio de modelos culturales acerca del mundo.** Plantea que los alumnos deben reflexionar sobre sus modelos culturales de una manera “externa” a sus identidades del mundo real.
31. **Principio de modelos culturales acerca del aprendizaje.** Sostiene que los estudiantes pueden reflexionar sobre sus modelos culturales con respecto al aprendizaje y a sí mismos, como aprendices de una forma “externa” a sus identidades del mundo real.

32. **Principio de modelos culturales acerca del dominio semiótico.** Sostiene que los estudiantes pueden reflexionar sobre sus modelos culturales con respecto a dominios particulares de los que están aprendiendo, de una manera “externa” a sus identidades del mundo real.
33. **Principio distribuido.** Establece que los textos, las herramientas, las personas y la tecnología están conectados en red, permitiendo que la información se distribuya entre ellos.
34. **Principio de dispersión.** Plantea la posibilidad de que el alumno pueda compartir su conocimiento con otros estudiantes, a los cuales quizás nunca conocerá cara a cara.
35. **Principio de afinidad del grupo.** Sostiene que los estudiantes se agrupan de acuerdo a identidades, objetivos y prácticas compartidas en lugar de por raza, sexo o nacionalidad.
36. **Principio de persona involucrada.** Establece que el alumno contribuye al juego y no es sólo un consumidor pasivo.

4.3 Taxonomía de Bloom

En 1956, el psicólogo y pedagogo estadounidense Benjamin Bloom (1913-1999), publicó una Taxonomía que muestra una visión global del proceso educativo, la cual se compone de tres dimensiones: afectiva, psicomotora y cognitiva.

La taxonomía de Bloom del dominio cognitivo, clasifica las habilidades mentales en seis niveles de complejidad jerárquicos, asumiendo que *“el aprendizaje a niveles superiores depende de la adquisición de conocimientos y habilidades a niveles inferiores”* (Bloom, 1956).

Los seis niveles propuestos en orden ascendente son:

1. **Conocimiento.** Se adquiere por medio de la recolección de información; por lo que en este nivel la persona es capaz de: definir, describir, memorizar, nombrar, recitar y listar la información obtenida.
2. **Comprensión.** En este nivel, se entiende el significado del conocimiento adquirido; por lo que la persona es capaz de: expresar, parafrasear, resumir, inferir y distinguir.
3. **Aplicación.** En este nivel se puede aplicar lo aprendido en situaciones nuevas; de este modo, se tiene la capacidad de: aplicar, demostrar, examinar, cambiar, clasificar, experimentar, usar, resolver, construir y calcular.
4. **Análisis.** En este nivel se pueden encontrar patrones, organizar las partes, reconocer significados ocultos, identificar componentes, relacionar las conjeturas, hipótesis, evidencias o estructuras de alguna pregunta o aseveración. Por lo que se tiene la capacidad para: separar, ordenar, explicar, comparar, seleccionar, inferir, analizar y contrastar.
5. **Síntesis.** En este nivel se puede construir una estructura o patrón con diversos elementos, generalizar a partir de datos suministrados, relacionar conocimiento con diversas áreas y predecir conclusiones derivadas. De este modo, se tiene la capacidad para: combinar, integrar, reordenar, crear, diseñar, inventar, modificar, desarrollar y generalizar.
6. **Evaluación.** En este nivel es posible hacer juicios sobre el valor de las ideas o materiales, por lo que la persona es capaz de: juzgar, medir, recomendar, decidir, concluir, seleccionar, criticar, comparar y valorar.

4.3.1 Revisión de la Taxonomía de Bloom

En el año 2001, Lorin Anderson (1945-) y David Krathwohl (1921-2016) realizaron una revisión de la Taxonomía de Bloom, con el objetivo de ayudar a los docentes a comprender e implementar currículos basados en estándares.

Esta nueva versión de la Taxonomía, proporciona un conjunto completo de clasificaciones para los procesos cognitivos del aprendiz. La clasificación de los objetivos de instrucción, usando esta taxonomía, ayuda a determinar los niveles de aprendizaje incluidos en una unidad de aprendizaje o lección.

Los seis niveles en orden ascendente de la Revisión de la Taxonomía de Bloom, según Anderson y Krathwohl son:

1. **Recordar.** Capacidad de recuperar conocimiento relevante desde la memoria a largo plazo.
2. **Comprender.** Habilidad para construir significado a partir del material didáctico, como la lectura o las explicaciones del docente.
3. **Aplicar.** Capacidad para llevar a cabo o usar un proceso aprendido en una situación específica.
4. **Analizar.** Capacidad para descomponer el conocimiento en sus partes constituyentes y determinar cómo se relacionan con su estructura global.
5. **Evaluar.** Consta de comprobación y crítica. Este nivel se ubica en la cúspide de la taxonomía original.
6. **Crear.** Capacidad para llevar a cabo tareas creadoras, para juntar elementos de forma coherente y funcional, así como para reorganizar elementos dentro de un nuevo patrón o estructura. Este nivel estaba considerado parcialmente en el nivel de síntesis de la taxonomía original.

En la Figura 4-1 se muestran los niveles del dominio cognitivo de la Taxonomía de Bloom y de la Revisión de Anderson y Krathwohl, donde es posible observar que los nombres de los niveles en la propuesta original corresponden a sustantivos, mientras que en la revisión se emplean verbos. También, se puede apreciar que los dos niveles de aprendizaje superior presentan un nuevo orden.

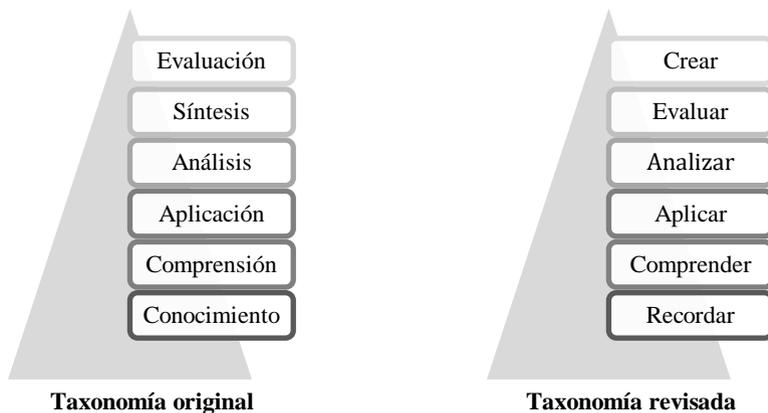


Figura 4-1. Taxonomía de Bloom original (Bloom, 1956) y revisada (Anderson y Krathwohl, 2001)

4.4. Modelos de Aprendizaje

4.4.1 Modelo VARK

En 1922, Neil Fleming (1939-) y Collen E. Mills (1930-) presentaron el modelo de preferencias sensoriales, conocido por sus siglas en inglés como VARK (*Visual, Auditory, Reading/Writing and Kinesthetic*; en español, Visual, Auditivo, Lectura/Escritura y Kinestésico) (véase Figura 4-2).

El Modelo VARK considera que las personas son diferentes y por tanto, aprenden de manera diferente, dadas las preferencias sensoriales que poseen para procesar la información (Hawk y Shah, 2007). Para conocer el estilo dominante de una persona, el modelo comprende un cuestionario con una serie de preguntas de opción múltiple (Fleming, 1995).

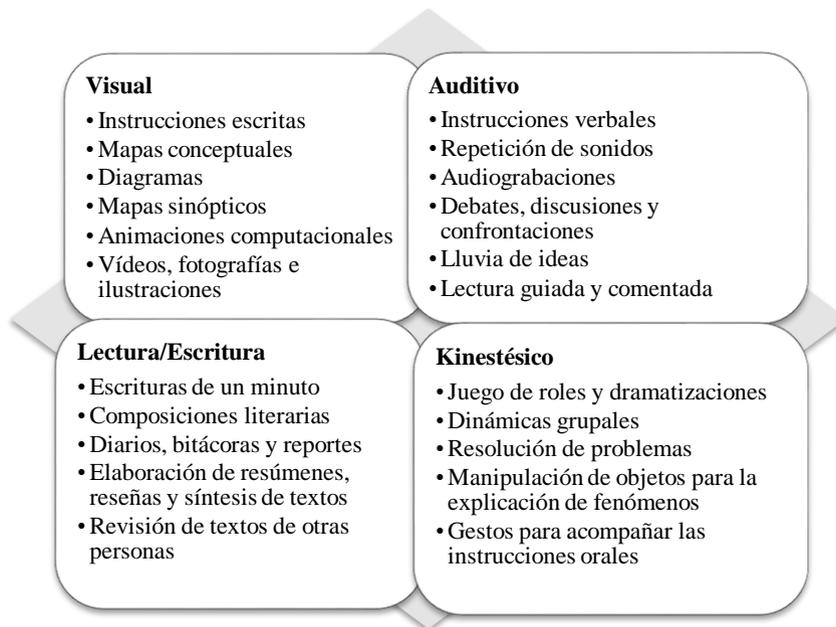


Figura 4-2. Estilos sensoriales de aprendizaje (Hawk y Shah, 2007)

4.4.2 Modelo de Aprendizaje Experiencial

En 1984, el psicólogo David Kolb (1939-) presentó el Modelo de Aprendizaje Experiencial, el cual sostiene que el aprendizaje depende de tres factores: la genética, las experiencias de vida y las exigencias del entorno (Kolb, 1984).

Para Kolb, el aprendizaje es un ciclo continuo que consta de cuatro etapas (véase Figura 4-3):

1. **Experiencia concreta.** En esta etapa la persona aprende a través de vivencias y experiencias (esto se relaciona con sentir); es decir, por medio del aprendizaje activo.
2. **Observación reflexiva.** En esta etapa el aprendiz revisa lo que hace o experimenta por medio de la reflexión (esto se relaciona con ver); es decir, a través del aprendizaje reflexivo.

3. **Conceptualización abstracta.** En esta etapa el aprendiz interpreta y relaciona los acontecimientos, mediante la conceptualización y la generalización (esto se relaciona con pensar); es decir, a través del aprendizaje teórico.
4. **Experimentación activa.** Esta etapa consiste en aplicar lo aprendido en una situación nueva (esto se relaciona con hacer); es decir, existe un aprendizaje pragmático.

El modelo de Kolb plantea la existencia de cuatro tipos de estilos de aprendizaje, basados en dos ejes: a) la percepción del contenido a aprender, que se realiza a través de la experiencia concreta o de la conceptualización abstracta y, b) el procesamiento del conocimiento, que se realiza por medio de la observación reflexiva o de la experimentación activa. Los estilos de aprendizaje experienciales son los siguientes (véase Figura 4-3):

1. **Divergente.** Este estilo se sitúa entre la experiencia concreta y la observación reflexiva (sentir y ver). Las personas que poseen este estilo de aprendizaje suelen ser emocionales, creativas, sociales, empáticas, abiertas, intuitivas y generan ideas espontáneas.
2. **Asimilador.** Este estilo se sitúa entre la observación reflexiva y la conceptualización abstracta (ver y pensar). Las personas con este estilo de aprendizaje destacan por su razonamiento inductivo, son reflexivas, poseen un pensamiento abstracto y les gusta investigar.
3. **Convergente.** Este estilo se sitúa entre la conceptualización abstracta y la experimentación activa (pensar y hacer). Las personas que poseen este estilo de aprendizaje destacan por su razonamiento deductivo, son analíticas, organizadas, herméticas, poco empáticas y gustan de la experimentación.
4. **Acomodador.** Este estilo se sitúa entre la experimentación activa y la experiencia concreta (hacer y sentir). Las personas con este estilo de aprendizaje tienden a resolver problemas de forma intuitiva y por medio del método de prueba y error, son sociales, organizados, aceptan retos, abiertos, espontáneos y flexibles.

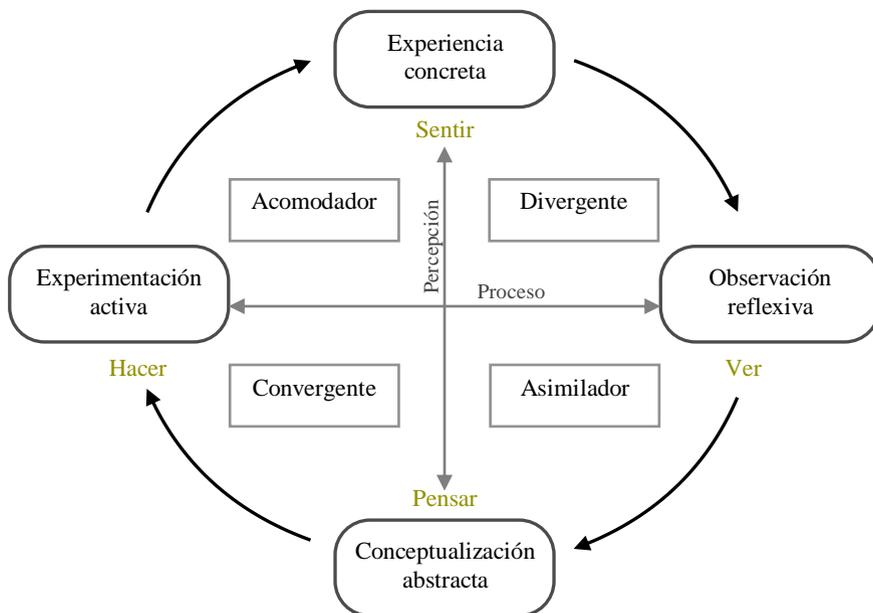


Figura 4-3. Estilos de aprendizaje experienciales (Kolb, 1984)

Para conocer el estilo de aprendizaje de una persona, el modelo comprende un cuestionario con una serie de preguntas de opción múltiple llamado “Inventario de Estilos de Aprendizaje” (en inglés, *Learning Style Inventory*) (Kolb, 1984).

4.5. Modelos de Diseño Instruccional

Los modelos de diseño instruccional son marcos conceptuales que permiten diseñar y desarrollar experiencias educativas de calidad (Merrill, 1996).

A continuación se presentan algunos modelos de diseño instruccional, que fueron considerados en el trabajo de investigación de esta tesis.

4.5.1 Modelo de Jonassen

El Modelo de diseño de Entornos de Aprendizaje Constructivistas (EAC) de David Jonassen (1947-2012), tiene como objetivo fomentar la solución de problemas y el desarrollo conceptual (Jonassen y Rorher-Murphy, 1999),

enfatisando el papel del aprendiz en la construcción del conocimiento (aprender haciendo).

El modelo EAC permite diseñar ambientes educativos apoyados por la tecnología, que comprometan al estudiante en la elaboración del conocimiento por medio de seis etapas, tal como se muestra en la Figura 4-4 (Jonassen, 1991):

1. **Identificar el problema, pregunta o proyecto**, con la intención de establecer el objetivo de aprendizaje que los alumnos resolverán. Para ello deben especificarse tres componentes: contexto, representación y espacio de manipulación. De este modo, el problema o proyecto dirige el aprendizaje del estudiante.
2. **Proporcionar casos relacionados**, con la finalidad de que los alumnos comprendan las cuestiones implícitas relacionadas con el problema, permitiendo: a) reforzar la memoria del aprendiz y, b) aumentar la flexibilidad cognitiva.
3. **Facilitar fuentes de información**, que les permitan a los alumnos elaborar sus modelos mentales y formular hipótesis sobre el problema planteado. Es recomendable que las fuentes de información sean valoradas y seleccionadas previamente por el docente.
4. **Brindar herramientas cognitivas**, que permitan a los alumnos representar y organizar la información pertinente y necesaria para resolver el problema planteado. Jonassen propone cuatro tipos de herramientas: a) herramientas de representación de problemas o ejercicios, b) herramientas para hacer modelos sobre el conocimiento estático y dinámico, c) herramientas de apoyo al rendimiento y, d) herramientas para recopilar información.
5. **Seleccionar herramientas de colaboración y conversación**, para ayudar a los alumnos a elaborar de forma conjunta un conocimiento socialmente compartido.
6. **Ofrecer apoyo contextual y social**, dado que para tener éxito en el aprendizaje, este debe adaptarse a los factores contextuales, para lo que

es necesario promover tres funciones cognitivas fundamentales en el aprendizaje: a) la exploración, b) la articulación y, c) la reflexión de lo que ha realizado.

Asimismo, el modelo EAC comprende tres actividades pedagógicas que apoyan el aprendizaje:

- A. **La modelización**, proporciona al aprendiz un ejemplo de rendimiento deseado. Jonassen plantea dos tipos: 1) modelización del comportamiento del rendimiento evidente y, 2) modelización cognitiva de los procesos cognitivos encubiertos.
- B. **La tutoría**, motiva a los alumnos al brindar respuestas y consejos, estimulando la reflexión y los procesos metacognitivos.
- C. **El refuerzo (andamiaje)**, proporciona modelos temporales para respaldar el aprendizaje y la representación de los alumnos más allá de sus capacidades.

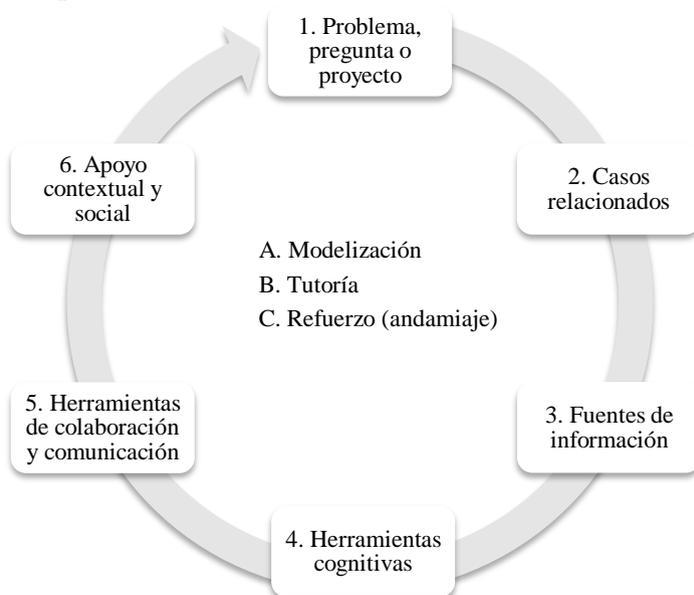


Figura 4-4. Entornos de Aprendizaje Constructivistas (Jonassen, 2000)

4.5.2 Modelo ARCS

Este modelo derivó de la síntesis sobre investigación la motivación humana de John Keller (1938-). Está compuesto por cuatro componentes, los cuales deben asegurarse en el siguiente orden: Atención (A), Relevancia (R), Confianza (C) y Satisfacción (S). La Atención, implica capturar el interés de los aprendices, estimulando su curiosidad por aprender. La Relevancia, reside en conocer las necesidades y metas personales de los estudiantes para lograr una actitud positiva. La Confianza, consiste en ayudar a los alumnos a creer y sentir que pueden controlar y tener éxito en el aprendizaje. Mientras que, la Satisfacción radica en reforzar el logro del aprendizaje con recompensas (Keller, 1987).

Cada componente del modelo ARCS está conformado por subcomponentes, los cuales contienen reflexiones que sirven como guías al momento de diseñar las estrategias educativas motivacionales (véase Tabla 4-1).

Tabla 4-1. Componentes y subcomponentes del modelo ARCS

Componente	Subcomponente	Reflexiones
Atención	Incitación perceptiva	¿Cómo puede capturarse el interés?
	Incitación a la investigación	¿Cómo puede estimularse una actitud de investigación?
	Variabilidad	¿Cómo puede mantenerse la atención?
Relevancia	Orientado a metas	¿Cómo pueden satisfacerse las necesidades del aprendiz? ¿Se conocen las necesidades?
	Familiaridad	¿Cómo puede vincularse la instrucción con las experiencias de los aprendices?
Confianza	Motivar coincidencias	¿Cuándo y cómo pueden brindarse responsabilidades?
	Requerimientos de aprendizaje	¿Cómo puede construirse una expectativa positiva de éxito?
	Oportunidad de éxito	¿Cómo puede mejorarse la experiencia de aprendizaje?
Satisfacción	Control personal	¿Cómo sabrán los aprendices que el éxito depende de su esfuerzo y habilidades?
	Consecuencias naturales	¿Cómo pueden brindarse oportunidades significativas para que los aprendices usen sus conocimientos recién adquiridos?
	Consecuencias positivas	¿Qué proporcionará refuerzo al éxito de los aprendices?
	Equidad	¿Cómo puede hacerse sentir a los aprendices que su esfuerzo y habilidades se reflejan positivamente en sus logros?

El proceso de diseño motivacional consta de diez pasos o actividades (Keller, 1987), tal como se muestra en la Figura 4-5. Los primeros dos pasos, residen en obtener información sobre contenido del curso y la audiencia. Los pasos 3 y 4, consisten en analizar la audiencia y los materiales de instrucción, lo que proporciona información de entrada para formular los objetivos y las evaluaciones motivacionales (paso 5). El paso 6, consiste en generar posibles soluciones o estrategias, las cuales suelen obtenerse por medio de lluvias de ideas u otro tipo de actividad abierta. Posteriormente (paso 7), se analizan y se eligen las estrategias más factibles y se integran en los materiales didácticos (paso 8). Una vez desarrollados los materiales de motivación, se implementa una prueba piloto o una primera oferta del curso (paso 9), para finalmente, llevar a cabo la evaluación formativa (paso 10).

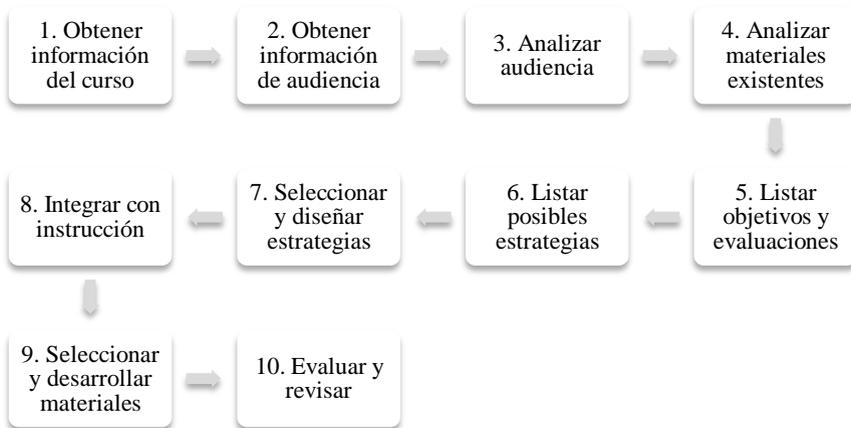


Figura 4-5. Proceso ARCS para el diseño de entornos motivacionales de trabajo y aprendizaje (Keller, 2010)

Asimismo, Keller proporciona dos instrumentos para evaluar el nivel de motivación percibido por los aprendices, de acuerdo a los factores ARCS:

1. **Encuesta de Interés del Curso** (en inglés, *Course Interest Survey*), que fue diseñada para medir las reacciones de los estudiantes durante el aprendizaje dirigido por un instructor. Consta de 34 preguntas: ocho ítems para el componente de Atención, nueve para la Relevancia, ocho para la Confianza y nueve para la Satisfacción. Cada pregunta se evalúa con una escala Likert que va del 1 al 5, donde 1 corresponde a

“No es cierto” y 5 corresponde a “Muy cierto” (Keller, 2006).

2. **Encuesta de Motivación de Materiales de Instrucción** (en inglés, *Materials Motivation Survey*), que fue diseñada para medir las reacciones a los materiales de instrucción auto dirigidos. Consta de 36 preguntas: los componentes de Relevancia y Confianza tienen 9 ítems, el componente de Satisfacción tiene 6 y el de Atención 12. Cada pregunta se evalúa con una escala Likert que va del 1 al 5, donde 1 corresponde a “Totalmente en desacuerdo” y 5 corresponde a “Totalmente de acuerdo” (Keller y Suzuki, 1988).

4.6. Modalidades de Aprendizaje

4.6.1 Aprendizaje Presencial

Esta modalidad, conocida en inglés como *Face-to-Face Learning*, consiste en desarrollar la educación en el aula, siendo ésta la manera tradicional de enseñanza, donde el profesor y los alumnos son los actores que intervienen en el proceso de aprendizaje (Rjaibi, Rabai y Limam, 2012).

Algunas ventajas del aprendizaje presencial son:

- Mantiene una relación estrecha entre los estudiantes y el profesor.
- Retroalimentación inmediata.
- El profesor puede conocer las necesidades de los alumnos con mayor facilidad.

Algunas desventajas del aprendizaje presencial son:

- Se requiere infraestructura, como un salón de clases, butacas, mesas, pizarrones, etc.
- Requiere que el profesor y los alumnos se trasladen para asistir a la escuela a una hora determinada; no hay flexibilidad.
- El proceso educativo está centrado en el profesor, quien dirige el aprendizaje y representa la fuente básica de información; por lo que es común que el estudiante actúe pasivamente como un oyente u observador.

4.6.2 Aprendizaje Virtual

Esta modalidad, conocida en inglés como *Electronic Learning (E-Learning)*, permite desarrollar la educación mediante la interacción del aprendiz con diversos medios electrónicos y el Internet (Shneiderman, 2003).

Dado que el docente y los estudiantes se encuentran físicamente separados, la comunicación entre ellos puede ocurrir de dos maneras: 1) sincrónica, permitiendo la comunicación en tiempo real, por ejemplo, con una vídeo conferencia, mensajería instantánea, etc. o, 2) asincrónica, permitiendo la comunicación de forma no simultánea, como podría ser por medio de foros, blogs o correos electrónicos.

Algunas ventajas del aprendizaje virtual son:

- La reducción del coste de traslados, alojamientos y materiales didácticos.
- Flexibilidad para aprender en cualquier momento y lugar.

Algunas desventajas del aprendizaje virtual son:

- Puede ser difícil para el profesor conocer las necesidades de los alumnos a distancia.
- Pueden existir dificultades de interacción con la tecnología.
- Si la comunicación es asincrónica, los estudiantes deben esperar para recibir retroalimentación.

4.6.3 Aprendizaje Combinado

Esta modalidad, conocida en inglés como *Blended Learning (B-Learning)*, combina el aprendizaje presencial y el aprendizaje virtual, permitiendo crear escenarios múltiples de aprendizaje por medio de actividades en el aula y de medios digitales (Zeng *et al.*, 2018).

Algunas ventajas del aprendizaje combinado son:

- Formación flexible, ya que el alumno puede elegir el lugar y la hora para estudiar.

- Mantiene la interacción entre los estudiantes, el profesor y el entorno durante las clases presenciales.
- Optimización del tiempo de aprendizaje presencial.

Algunas desventajas del aprendizaje combinado son:

- Pueden existir dificultades de interacción con la tecnología.
- Se deben cubrir gastos de traslados, alojamientos y materiales didácticos para las clases presenciales.

4.6.4 Aprendizaje Móvil

Esta modalidad, conocida en inglés como *Mobile Learning (M-Learning)*, permite desarrollar el aprendizaje mediante el uso de dispositivos móviles, como teléfonos inteligentes o tabletas (Garaj, 2010).

Algunas ventajas del aprendizaje móvil son:

- Flexibilidad para aprender en cualquier momento y lugar (ubicuidad).
- Portabilidad, debido a que los dispositivos móviles son pequeños, pueden llevarse a donde se desee con facilidad.
- Las nuevas generaciones de estudiantes (nativos digitales), suelen sentirse más motivados con esta modalidad de aprendizaje, debido a la familiaridad que tienen con el uso de dispositivos móviles.

Algunas desventajas del aprendizaje móvil son:

- Problemas de usabilidad, ya que los dispositivos móviles tienen pantallas pequeñas.
- Pueden existir dificultades de interacción con los dispositivos móviles.
- Algunos aprendices podrían verse afectados al no poseer los recursos económicos para adquirir un dispositivo móvil y ni tener acceso a Internet.

4.6.5 Aula Invertida

El aula invertida, conocida en inglés como *Flipped classroom*, es una modalidad de aprendizaje en la que se invierten el orden y el lugar en el que ocurren las actividades del modelo tradicional, es decir, en casa se adquieren los conocimientos por medio de materiales multimedia, como vídeos o audios,

mientras que en el aula se realizan los ejercicios y prácticas (tareas) (Nwokeji y Holmes, 2017).

Algunas ventajas del aula invertida son:

- Fomenta la participación del alumno en su propio aprendizaje, por lo que pasa de ser un sujeto pasivo a uno activo.
- Facilita la entrega y la revisión de tareas.
- Promueve la comprensión y la experimentación, lo que permite un aprendizaje más profundo y perdurable.
- Proporciona más tiempo para resolver dudas y consolidar conocimientos.

Algunas desventajas del aula invertida son:

- Requiere del compromiso de los estudiantes para adquirir los conocimientos en casa.
- Los estudiantes con menos recursos económicos pueden verse afectados al no contar con acceso a Internet, ni poseer dispositivos electrónicos (computadoras, tabletas, celular inteligentes, etc.).
- Los alumnos pueden presentar resistencia al cambio de paradigma.
- Implica mayor tiempo y esfuerzo por parte del docente para la preparación cuidadosa del material educativo.

Capítulo 5

Metodología para construir juegos serios computacionales

“La educación es el arma más poderosa que puedes usar para cambiar al mundo”
Nelson Mandela (1918-2013)

Este capítulo presenta la metodología para construir juegos serios computacionales que fue diseñada como parte de la investigación reportada en esta tesis.

Las primeras secciones de este capítulo exponen una síntesis de algunas metodologías para desarrollar software, videojuegos y juegos serios que fueron objeto de estudio.

5.1 Metodologías de desarrollo de software

Una metodología de desarrollo de software es un marco de trabajo (Paulk, 1993), compuesto de métodos y prácticas (Sommerville, 2015), que permite estructurar, planificar y controlar el proceso de desarrollo de software (Piattini y Garzás, 2010), para la entrega de un producto de calidad (Pressman, 2005).

Las metodologías de desarrollo de software han evolucionado a medida que las tecnologías y los paradigmas de programación lo han hecho. En la década de los 70s, el paradigma de programación predominante era el estructurado, donde lenguajes como Pascal, ALGOL y Ada eran muy populares. Una de las metodologías más empleadas en esta época, fue la creada por Yourdon (1976), donde el uso de Diagramas de Flujos de Datos, Diagramas de Contexto, Diagramas de Estados y Diagramas de Entidad-Relación, aparecían como una constante en el modelado del sistema.

Con el surgimiento del paradigma de programación orientado a objetos, nacieron las metodologías tradicionales, que facilitan la división del sistema en

subsistemas, fomentando el rehuso de componentes, brindando una estructura bien definida y resaltando la importancia de documentar. Como ejemplos de metodologías tradicionales, se pueden mencionar Métrica 3 (MAP, 2001) y RUP (*Rational Unified Process*, Proceso Racional Unificado) (Jacobson *et al.*, 1999).

En los 90s, como reacción a la excesiva burocratización de las metodologías tradicionales, surgieron las metodologías ágiles, siendo las más populares XP (*eXtreme Programming*, Programación Extrema) (Beck, 1999) y SCRUM (Schwaber y Beedle, 2001).

5.2 Metodologías de desarrollo de videojuegos

En un inicio los videojuegos no eran tan complejos, como en el caso de Tetris (1984) y Pac-Man (1980), pero con el tiempo fueron incorporando más y más elementos tecnológicos hasta convertirse en un producto de software sofisticado (Osborne *et al.*, 2014).

Para facilitar el desarrollo de los videojuegos, existen algunas metodologías, como la diseñada por Maxwell (2014), que propone la construcción de prototipos funcionales incrementales, guiados por un proceso compuesto de cuatro fases: pre producción, producción, pruebas y post producción.

Otra metodología para desarrollo de videojuegos es “Huddle”, una adaptación de SCRUM, diseñada para equipos conformados por 5-10 personas, los cuales deben desempeñar los roles clásicos de SCRUM, más el rol de “diseñador del juego”. El proceso de desarrollo de Huddle es iterativo, incremental y evolutivo, compuesto de tres fases: pre-producción, producción y post-mortem (Morales-Urrutia *et al.*, 2010).

Adicionalmente, se puede mencionar la metodología SUM, que combina prácticas de SCRUM y XP, diseñada para equipos conformados entre 3 y 7 personas, para proyectos de menos de 1 año de duración. SUM adopta los roles de SCRUM y añade el rol de “verificador beta”. Define un proceso compuesto de cinco fases iterativas e incrementales que se ejecutan secuencialmente, las cuales son: concepto, planificación, elaboración, beta y cierre (Acerenza *et al.*, 2009).

5.3 Metodologías de desarrollo de juegos serios

Para “*equilibrar los aportes pedagógicos con el placer de juego*”, Marne *et al.* (2011) proponen un marco conceptual para desarrollar juegos serios, que consta de 6 etapas o facetas:

- Faceta 1: Objetivos pedagógicos. Definición del dominio de referencia y de objetivos pedagógicos, incluyendo conceptos erróneos.
- Faceta 2: Simulación del dominio. Definición de un modelo formal del dominio de simulación.
- Faceta 3: Interacción con la simulación. Definición de interacciones con el modelo formal y de la metáfora intrínseca.
- Faceta 4: Problemas y progresión. Definición de la progresión en el juego serio.
- Faceta 5: Decoro. El decoro consta de elementos gráficos y sonoros.
- Faceta 6: Condición de utilización. Definición de las condiciones de utilización del juego serio.

Por otro lado, Yusoff *et al.* (2009) proponen una metodología basada en 7 etapas. Las dos primeras etapas, se centran en definir el dominio de enseñanza; las tres siguientes, se enfocan en el diseño de escenarios; la sexta etapa, consiste en evaluar los escenarios y; la última etapa, comprende la redacción de las especificaciones destinadas a los constructores del juego serio (diseñadores gráficos, diseñadores de sonido, grafistas, etc.).

5.4 Motivación para diseñar una metodología para construir juegos serios computacionales

Las metodologías descritas en las secciones anteriores de este capítulo, no fueron empleadas en la construcción de los juegos serios computacionales de la investigación reportada en esta tesis. Ya que las descritas en la Sección 5.1 proporcionan marcos para desarrollo de software genérico, no contemplando

elementos pedagógicos. Del mismo modo, las mencionadas en la Sección 5.2 facilitan la construcción de videojuegos, sin que tengan fines necesariamente educativos.

Por su parte las metodologías de la Sección 5.3 brindan directrices para construir juegos serios con enfoque pedagógico, pero no guían el desarrollo de software.

Por lo anterior, se decidió diseñar una metodología propia para construir de juegos serios computacionales, que proporcionara tanto directrices pedagógicas como tecnológicas.

5.5 Metodología para construir juegos serios computacionales

5.5.1 Descripción

Esta metodología es un marco de trabajo que permite estructurar, planificar y controlar el proceso de desarrollo de un juego serio computacional.

5.5.2 Alcance

La metodología fue diseñada para:

- Construir juegos serios computacionales sin fines comerciales. Por esta razón, no se contempla la existencia de un modelo de negocio.
- Desarrollar juegos serios basados en software de tipo:
 - Videojuegos.
 - Aplicaciones móviles.
 - Sitios Web.
- Proyectos cortos (menores a 4 meses).
- Equipos pequeños (4-7 integrantes).

5.5.3 Filosofía

La filosofía de esta metodología se centra en el jugador o aprendiz. De este modo, deben tomarse en cuenta las necesidades y gustos de los usuarios

durante todo el proceso de desarrollo, con la finalidad de garantizar la construcción de un juego serio computacional que motive el aprendizaje.

5.5.4 Roles

Esta metodología propone la existencia de cuatro roles principales:

1. **Investigador:** puede tratarse de un profesor u otra persona con la visión e ideas para construir el juego serio computacional.

Sus responsabilidades son:

- Compartir las ideas iniciales sobre juego serio para definir el concepto.
- Validar el concepto del juego serio.
- Aprobar el plan de proyecto.
- Participar en la evaluación del producto después de cada iteración.
- Validar el producto final.

2. **Administrador Pedagógico:** es la persona con conocimientos de Pedagogía que forma parte del equipo de desarrollo de juegos serios computacionales.

Sus responsabilidades son:

- Definir los objetivos pedagógicos del juego serio.
- Buscar información sobre el contenido educativo del juego serio.
- Aprender de forma autodidacta el contenido educativo que será incluido en el juego serio.
- Transmitir y compartir el conocimiento relacionado con el contenido educativo del juego serio al resto del equipo.
- Elaborar el diseño instruccional.
- Evaluar los artefactos pedagógicos construidos.

3. **Administrador del Proyecto:** es la persona con conocimientos de Ingeniería de Software que forma parte del equipo de desarrollo de juegos serios computacionales.

Sus responsabilidades son:

- Definir los requerimientos funcionales y no funcionales del juego serio computacional.
 - Estimar los recursos, el esfuerzo y la duración del proyecto.
 - Establecer el plan del proyecto, especificando las actividades a realizar y asignando las responsabilidades.
 - Dar seguimiento al plan del proyecto.
 - Coordinar el trabajo de los desarrolladores de software.
 - Coordinar la comunicación entre todos los involucrados.
 - Asegurar la calidad en el proceso y en el producto por medio de acciones para la mejora continua.
4. **Programadores de Software:** son las personas con habilidades y conocimientos de programación que forman parte del equipo de desarrollo de juegos serios computacionales.

Sus responsabilidades son:

- Ayudar a definir los aspectos técnicos para desarrollar el juego serio computacional.
- Codificar.
- Realizar pruebas unitarias en cada iteración.
- Realizar pruebas de usuario y de aceptación del producto.

Es importante mencionar que aunque se sugiere que una persona sea destinada a cumplir sólo un rol, en la práctica pueden hacerse las adecuaciones necesarias para desempeñar dos roles a la vez.

5.5.5 Etapas y procesos

Esta metodología para construir juegos serios computacionales se compone de tres etapas: *Pre producción*, *Producción* y *Post producción*. Además, comprende tres procesos: *Administración del Proyecto*, *Implementación de Software* e *Implementación Pedagógica* (véase Figura 5-1).

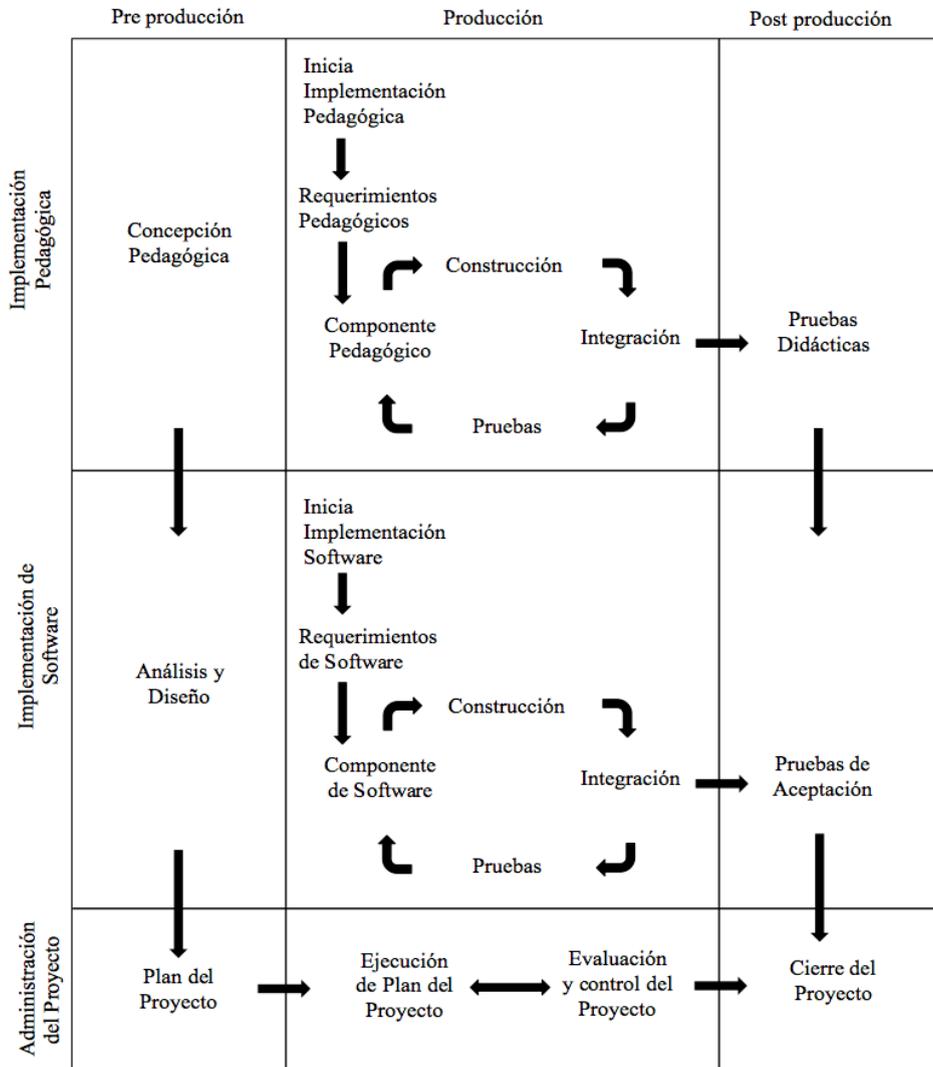


Figura 5-1. Etapas y procesos de la metodología para construcción de juegos serios computacionales

La arquitectura de la metodología propuesta, se basó en el estándar ISO/IEC 29110 *Deployment Package* (IEEE, 2008), el cual cuenta con perfiles y guías dirigidas a las entidades de software muy pequeñas, conocidas en inglés como “*Very Small Entities*”, las cuales pueden tratarse de empresas, organizaciones o proyectos que incluyan a lo más 25 personas (véase Apéndice E).

A continuación se presentan las etapas de la metodología que aquí se propone.

5.5.5.1 Pre producción

En esta etapa se realizan de manera secuencial las actividades de *Concepción Pedagógica, Análisis y Diseño*, así como la elaboración del *Plan del Proyecto*, las cuales corresponden a los procesos de *Implementación Pedagógica, Implementación de Software y Administración del Proyecto* respectivamente.

La primera de las actividades llamada *Concepción Pedagógica* (véase Figura 5-1), consiste en idear y plantear el juego serio computacional. Para lo cual es necesario bosquejar las ideas que se tienen del juego, considerando el contenido a enseñar, la audiencia y el tipo de juego (videojuego, aplicación móvil o sitio Web). Por lo que en este punto, deberían poderse responder las siguientes interrogantes:

- “¿Qué se podrá aprender al usar el juego serio computacional?” Con la finalidad de definir el objetivo general de enseñanza.
- “¿Quién usará este juego y dónde?” Con la intención de definir la audiencia o público objetivo.
- “¿Qué tipo de juego serio computacional sería mejor desarrollar, dadas las características del contenido educativo y la audiencia?” Para conocer qué tipo de desafíos y dinámicas de juegos deben incluirse.

Esta actividad la realizan el *Investigador* y el *Administrador Pedagógico* durante las *Reuniones de Desarrollo del Concepto* (véase Figura 5-2). De las cuales se obtiene como producto el *Documento de Especificación del Juego Serio Computacional*, el cual debe contener: el nombre del juego serio computacional, objetivo general de aprendizaje, objetivos específicos de

aprendizaje, público objetivo, tipo de juego serio computacional y el diseño instruccional.



Figura 5-2. Reuniones de desarrollo del concepto

Una vez concebido el juego serio computacional, el *Administrador Pedagógico* debe ampliar su nivel de conocimiento sobre el contenido educativo del juego y compartirlo por medio de clases al resto del equipo. Para que el equipo de trabajo estructure el conocimiento y logre un aprendizaje significativo, que le permita aportar ideas en el diseño del juego, se sugiere la elaboración de mapas conceptuales con la información aprendida, tal como lo plantea Ausubel (1976).

Finalmente, como parte de la primera actividad, la *Concepción Pedagógica*, el *Administrador Pedagógico* debe evaluar el aprendizaje para comprobar que todos los miembros del equipo dominan el tema que se enseñará por medio del juego serio computacional (véase Figura 5-3).

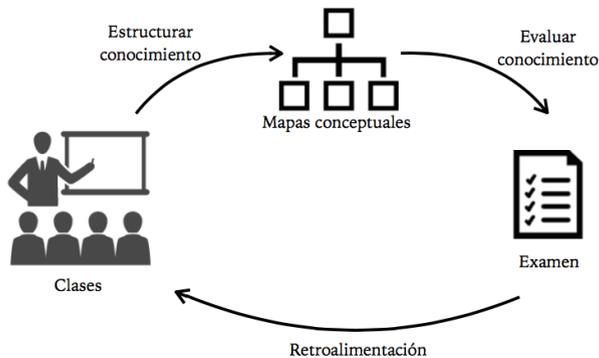


Figura 5-3. Proceso de enseñanza-aprendizaje del equipo

La segunda actividad de la fase de *Pre producción* se llama *Análisis y Diseño*. Consiste en realizar *Reuniones de Análisis y Diseño*, a las que asisten todos los integrantes del equipo, para llevar a cabo *Lluvias de Ideas* que les permitan diseñar la arquitectura del juego serio computación (véase Figura 5-4).

La técnica de *Lluvia de Ideas* consiste en:

1. Seleccionar un tema.
2. Generar la lluvia de ideas a través de la participación de los involucrados.
3. Realizar el análisis de las ideas.
4. Seleccionar las mejores ideas (factibles, viables y adecuadas).

Para obtener mejores resultados durante las *Lluvias de Ideas* en las *Reuniones de Análisis y Diseño*, se recomienda que:

1. El *Investigador* y el *Administrador Pedagógico* presenten ideas previamente analizadas al inicio de las reuniones.
2. Fomentar la participación de todos los asistentes.
3. Anotar las ideas de los asistentes en un lugar que sea visible para todos, como un pizarrón o cartulina, registrando los pros y contras de cada idea.

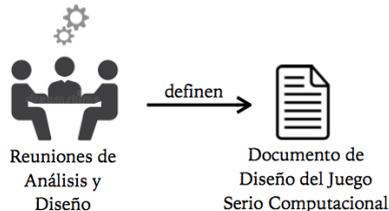


Figura 5-4. Reuniones de análisis y diseño del juego serio

Como resultado de la actividad de *Análisis y Diseño*, debe elaborarse el *Documento de Diseño del Juego Serio Computacional*, el cual especifica: las herramientas tecnológicas, lenguajes de programación, plataformas, personajes y guión gráfico (si es un videojuego), *Diagramas de Navegación/Transición* (si es una aplicación móvil o un sitio Web) y *Diagramas de Flujo*.

Si el juego serio computacional será de tipo:

a) Videojuego

Se recomienda hacer uso de un *motor de videojuego*. Para seleccionarlo se debe analizar y definir si el entorno gráfico será en dos dimensiones (2D) o tres dimensiones (3D), si se jugará en primera persona, si se jugará en una consola o en un dispositivo móvil, si se cuenta con el presupuesto para el pago de licencias, etcétera. En el Apéndice F, se muestran algunas de las características de los motores de videojuego más populares: Unity (Unity Technologies, 2018), Unreal Engine (Epic Games, 2018) y CryENGINE (Crytek GmbH, 2018).

Además, deben diseñarse los *personajes* y elaborarse un *guión gráfico* que muestre la secuencia de la narrativa o historia del videojuego. De este modo, será posible visualizar el videojuego para corroborar su diseño antes de codificarlo.

b) Aplicación móvil o sitio Web

Se debe modelar la estructura, el contenido y la apariencia del juego por medio de *Diagramas de Navegación/Transición* (véase Figura 5-5). Esto permitirá corroborar el diseño antes de pasar a la siguiente etapa.

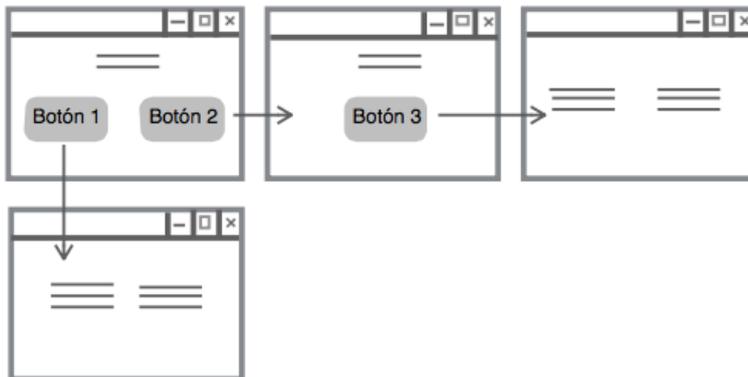


Figura 5-5. Diagrama de navegación/transición

Independientemente del tipo de juego serio computacional que se construirá, es conveniente definir la secuencia de operaciones o actividades que podrá realizar el jugador o aprendiz al interactuar con la herramienta por medio de un *Diagrama de Flujo*. En la Figura 5-6 se muestran los símbolos que se emplean para elaborar *Diagramas de Flujo* (Jiménez-Murillo, Jiménez-Hernández y Alvarado-Zamora, 2014).

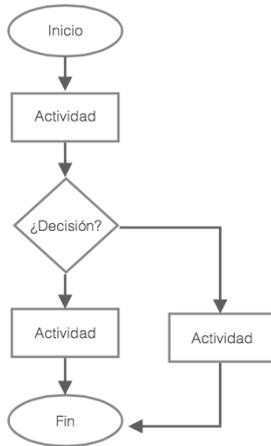


Figura 5-6. Diagrama de Flujo

La última actividad que se realiza en la etapa de *Pre producción*, es la de elaborar el *Plan del Proyecto*. El cual se plantea una vez que se cuenta con la aprobación del *Documento de Diseño del Juego Serio Computacional*, por parte de todos los miembros del equipo (véase Figura 5-7).

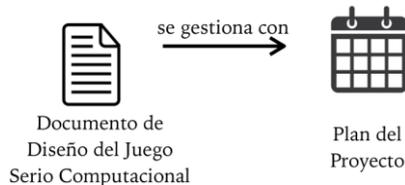


Figura 5-7. Gestión del Proyecto

El *Plan del Proyecto*, es elaborado por el *Administrador del Proyecto*, haciendo uso de la Tabla 5-1. De este modo, se enumeran las actividades a

realizar, estableciendo fecha, hora y lugar o forma de entrega, así como el o los responsables e involucrados en dichas actividades.

Tabla 5-1. Plan del Proyecto

No. Actividad	Fecha	Hora	Lugar	Actividad	Responsable	Involucrados	Notas
1							
2							
.				.			
n							

5.5.5.2 Producción

En esta etapa, las actividades correspondientes a los tres procesos de la metodología se realizan de forma paralela, con la guía que proporciona el *Plan del Proyecto*.

De este modo, el *Administrador del Proyecto* asigna a los *Programadores de Software* la creación de un módulo, quienes lo codifican, lo integran y realizan *pruebas de caja blanca* (Pressman, 2005) y *de caja negra* (Sommerville, 2015). Esto se ejecuta dentro de un ciclo iterativo e incremental, tal como propone McConnell en sus mejores prácticas metodológicas en Ingeniería de Software (2006).

Las *pruebas de caja blanca* permiten verificar las funciones internas del módulo, y son realizadas por los *Programadores de Software*, mientras que las *pruebas de caja negra*, validan el cumplimiento de los requisitos funcionales, y son realizadas por los *Programadores de Software*, con la colaboración del *Administrador Pedagógico* y el *Administrador del Proyecto*.

5.5.5.3 Post producción

En esta etapa, se realizan de manera secuencial las *Pruebas Didácticas*, *Pruebas de Aceptación* y el *Cierre del Proyecto*, que corresponden a los procesos de *Implementación Pedagógica*, *Implementación de Software* y *Administración del Proyecto* respectivamente.

Como parte de las *Pruebas Didácticas*, el *Investigador* y el *Administrador Pedagógico*, deben corroborar que el juego serio computacional construido permite el aprendizaje de los temas planteados, es decir, que se cumple con lo establecido en el *Documento de Especificación del Juego Serio Computacional*.

En el caso de las *Pruebas de Aceptación*, el *Investigador*, el *Administrador Pedagógico* y el *Administrador del Proyecto*, deben realizar las *pruebas de usuario*, las cuales pueden realizarse con algunos voluntarios pertenecientes al público objetivo definido. En caso de encontrarse algún defecto, este debe ser reparado, para después regresar a esta misma etapa y probar el juego serio computacional.

Una vez que el juego serio computacional ha sido aceptado, se realiza la actividad final de *Cierre del proyecto*, que consiste en elaborar y entregar el *manual técnico*, junto con el juego serio computacional desarrollado.

5.5.6 Juegos serios computacionales construidos

El marco de trabajo descrito en este capítulo, fue empleado en el desarrollo de cuatro juegos serios computacionales como parte de la investigación de esta tesis. Uno de ellos tiene por objetivo facilitar el aprendizaje de la simplificación de funciones Booleanas y los otros tres, fueron diseñados para aprender las Alfas del núcleo del estándar Essence. El primer juego, se presenta en la siguiente sub-sección (5.5.6.1) y los otros tres, al apoyar el aprendizaje del contenido que es objeto de estudio de esta tesis, se describen con más detalle en el Capítulo 6.

La metodología diseñada ha sido utilizada no sólo por los autores de la misma, sino también por los profesores del grupo de Investigación Educativa “Karani”, del Tecnológico Nacional de México (TecNM) campus Morelia, quienes a partir de las guías que proporciona el marco de trabajo, construyeron junto con sus alumnos, tres juegos serios computacionales: dos juegos para facilitar el aprendizaje de temas relacionados con la materia de Matemáticas Discretas (árboles y grafos) y un juego para apoyar a los niños de primaria en el aprendizaje de operaciones aritméticas con fracciones matemáticas. Estos juegos serios también se presentan a continuación como sub-secciones.

5.5.6.1 Juego serio computacional para aprender la simplificación de funciones Booleanas

Este juego serio computacional, está basado en un sitio Web llamado MiniBool (Jiménez-Hernández *et al.*, 2019) y fue diseñado para reforzar el aprendizaje relacionado con la simplificación de funciones Booleanas. Se construyó para ser usado por los estudiantes que cursan la materia de Matemáticas Discretas, la cual forma parte del plan de estudios de las carreras afines a la Computación. Sin embargo, dado que este tema es fundamental en el diseño de circuitos digitales, también puede ser utilizado por los alumnos de carreras relacionadas con Eléctrica y Electrónica.

Esta herramienta permite la simplificación de expresiones Booleanas de 3, 4, 5, ... n variables, las cuales pueden ser generadas tanto por el software mismo, como declaradas por el alumno. Estas funciones pueden representarse de tres maneras, como tablas de verdad, mapas de Karnaugh y expresiones Booleanas (véase Figura 5-8).

Variables: 3 4 5 n

Solve as: Sum of products Products of sums

Data in: Function True table Karnaugh map

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0

CD

AB	00	01	11	10
00	1	1	0	1
01	0	1	0	1
11	1	1	1	0
10	0	1	0	1

Write your answer:

F=

Possible solutions:

$A'B'D + CD + A'CD + B'CD + ABC + ABD$

$A'B'C + CD + A'CD + B'CD + ABC + ABD$

Figura 5-8. Juego serio computacional para aprender simplificación de funciones Booleanas

De este modo, cuando un alumno introduce su respuesta con la simplificación de una función, MiniBool le proporciona todas las soluciones simplificadas óptimas (porque puede haber más de una). En caso de que su respuesta sea correcta, obtiene puntos, los cuales dependen de la complejidad del problema.

Los puntos que obtienen los estudiantes los posicionan en un ranking (véase Figura 5-9), el cual compara el desempeño de todos los usuarios. De este modo, se incluye gamificación en MiniBool (Nowostawski, McCallum y Mishra, 2018) para acelerar el proceso de aprendizaje (Wang *et al.*, 2017), mejorar la retención del información (Domínguez, Saenz-de-Navarrete, de-Marcos, Fernández-Saenz, Pagés y Martínez-Herráiz, 2013) e incrementar la motivación por aprender (Callaghan *et al.*, 2015).

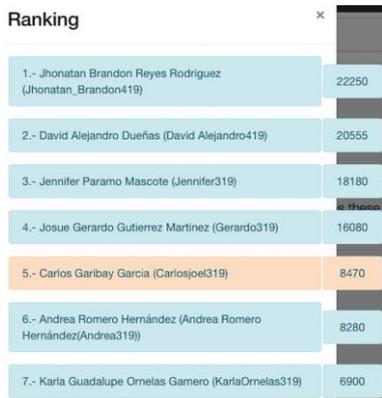


Figura 5-9. Ranking de MiniBool

Este juego serio computacional de tipo Web, presenta la información en tres idiomas: español, inglés y francés. Fue probado por medio de un experimento formal con estudiantes de la materia de Matemáticas Discretas en el TecNM campus Morelia. Los resultados estadísticos obtenidos muestran que los estudiantes que usaron MiniBool para reforzar su aprendizaje obtuvieron mejores calificaciones, que aquellos que lo reforzaron de forma tradicional, usando lápiz y papel para resolver ejercicios. Asimismo, los resultados de la evaluación de la motivación, reflejaron que con MiniBool es posible obtener mejores niveles de Atención, Relevancia y Satisfacción, que con el método de reforzamiento tradicional (Jiménez-Hernández, Oktaba, Díaz-Barriga y Piattini, 2020a).

5.5.6.2 Juego serio computacional para aprender operaciones aritméticas con fracciones matemáticas

Este juego serio computacional, de tipo videojuego, fue desarrollado con la colaboración del grupo de investigación Karani (Jiménez-Murillo, 2016), con el objetivo de apoyar el aprendizaje de las operaciones aritméticas con fracciones matemáticas en niños de primaria.

Consta de cuatro niveles, los cuales corresponden a las operaciones aritméticas básicas: suma, resta, multiplicación y división (Jiménez-Hernández *et al.*, 2017b). En cada nivel se presenta un conjunto de ejercicios que involucra la operación aritmética en cuestión, así que el alumno debe encontrar los valores numéricos que resuelvan correctamente las operaciones planteadas (véase Figura 5-10).



Figura 5-10. Juego serio computacional para aprender operaciones aritméticas con fracciones matemáticas

5.5.6.3 Juego serio computacional para aprender recorridos de árboles

Este juego serio computacional, de tipo videojuego, fue desarrollado por los miembros del grupo Karani (Jiménez-Murillo, 2016), con el objetivo de apoyar a los alumnos de la materia de Matemáticas Discretas en el aprendizaje de recorridos de árboles.

Consta de tres niveles, los cuales tienen una escenografía que representa la estructura de los diferentes tipos de árboles: binarios, ternarios y cuaternarios (Jiménez-Murillo, 2015). De este modo, el alumno debe visitar en Pre-orden, In-orden y Post-Orden cada árbol. Si el recorrido realizado por el aprendiz es correcto, obtiene puntos que le permiten adquirir nuevas habilidades al personaje principal del videojuego (véase Figura 5-11).



Figura 5-11. Juego serio computacional para aprender recorridos de árboles

5.5.6.4 Juego serio computacional para aprender grafos

Este juego serio de tipo videojuego, también fue construido con la colaboración del grupo de investigación Karani (Jiménez-Hernández et al., 2017a).

El objetivo de este juego es que los estudiantes de la materia de Matemáticas Discretas puedan aprender los conceptos básicos de los grafos: vértices, aristas, lados, lados paralelos, lazo y valencia (Jiménez, 2015).

Comprende un nivel en forma de laberinto, en el que los jugadores debe leer y responder las preguntas que se les presentan para poder avanzar hasta la salida (véase Figura 5-12).



Figura 5-12. Juego serio computacional para aprender grafos

Capítulo 6

Juegos serios computacionales para el aprendizaje de las Alfas de Essence

*“Quien se atreve a enseñar,
nunca debe dejar de aprender”*
John Cotton Dana (1856-1929)

Este capítulo presenta los tres juegos serios computacionales que fueron desarrollados con el objetivo de facilitar el aprendizaje de las Alfas del núcleo del estándar Essence, los cuales fueron construidos siguiendo el marco de trabajo descrito en el Capítulo 5.

Los referentes teóricos y psicopedagógicos del diseño didáctico de los juegos serios construidos, se ubican en las aproximaciones constructivistas y humanistas derivadas de la psicología de la educación virtual (Coll y Monero, 2008; Díaz-Barriga, 2005). De este modo, el aprendiz es considerado como un sujeto activo, crítico y comprometido en su propio aprendizaje y las tecnologías son como consideradas artefactos culturales mediadores del aprendizaje en los planos intra e interpsicológicos. Por lo que en los juegos serios desarrollados, se propone un diseño educativo flexible, centrado en el aprendiz, en sus necesidades e intereses, con el objetivo de privilegiar la construcción de conocimiento.

Uno de los principios educativos presentes en los tres juegos es el de la retroalimentación, de manera que el sistema instruccional está en constante diálogo con el aprendiz, actualizando la información relacionada con su progreso y desempeño. Además, se tomaron en cuenta los distintos estilos de aprendizaje sensoriales (Hawk y Shah, 2007), para atraer y mantener la atención de los usuarios, propiciando aprendizaje activo (Drudi *et al.*, 2013).

Es así que los juegos serios computacionales construidos promueven la motivación intrínseca y la conducta autotélica (persistencia y satisfacción en la actividad por el interés de la misma) (Marcano, 2008), permitiendo a los

aprendices desarrollar las siguientes competencias genéricas: capacidad de análisis, síntesis, organización, planificación, resolución de problemas, razonamiento crítico y toma de decisiones, en un proyecto de software.

6.1 Alhaspot

6.1.1 Descripción

Alhaspot es el primer juego serio computacional construido como parte de la investigación reportada en esta tesis. Es un videojuego que tiene como objetivo facilitar el aprendizaje de las Alfas de Essence, sus estados y listas de verificación (Jiménez-Hernández *et al.*, 2015).

Con Alhaspot el aprendiz será capaz de:

- Comprender la arquitectura del núcleo de Essence.
- Entender qué son las Alfas del núcleo de Essence y cómo se emplean para evaluar el estado y progreso en un proyecto de software.
- Recordar los estados de las Alfas de Essence y sus listas de verificación.
- Identificar el estado de un Alfa.

6.1.2 Características

Alhaspot fue diseñado para ser utilizado por profesionales laborando en entidades de software, aunque podría ser jugado por estudiantes del área de Computación también.

Algunas características del juego serio computacional son:

- Es gratuito.
- Está disponible en idioma español e inglés.
- Es multiplataforma, ya que puede ser ejecutado en computadoras con Windows© y Mac OS©, así como en dispositivos móviles con Android© (véase Figura 6-1).



Figura 6-1. Alphaspot en un dispositivo móvil

6.1.3 Tecnologías empleadas

Para el desarrollo de Alphaspot se emplearon las siguientes tecnologías:

- El motor de videojuego Unity 5.
- El lenguaje de programación C#.
- El programa de creación gráfica y animación 3ds MAX.
- El editor de gráficos Adobe Photoshop.
- El reproductor multimedia Adobe Flash.
- El programa de edición de audio Audacity.

6.1.4 Niveles del juego

Alphaspot consta de 4 tipos de niveles:

1. Nivel de introducción.

En Alphaspot sólo existe un nivel de este tipo, el cual se muestra cuando se abre el videojuego.

La primera vez que se ejecuta Alphaspot, el usuario está obligado a ver la narrativa que se presenta por completo, después es opcional (véase Figura 6-2).

Por medio de este nivel se explica el reto del videojuego y se expone la historia, arquitectura y modo de uso del estándar Essence.



Figura 6-2. Narrativa inicial de Alhaspot

2. Nivel tutorial.

En el videojuego, sólo hay un nivel de este tipo, el cual funciona como el nivel de introducción: la primera vez que se ejecuta Alhaspot es obligatorio que el usuario juegue este nivel, siendo opcional después.

Con este nivel los jugadores aprenderán a desenvolverse adecuadamente en el videojuego (véase Figura 6-3).



Figura 6-3. Nivel tutorial de Alhaspot

3. Niveles de acción.

Alphaspot cuenta con 41 niveles de este tipo, es decir, existe un nivel para cada uno de los estados de las Alfas, los cuales cuentan con una ambientación diferente entre sí.

El menú principal para estos niveles se diseñó considerando el principio de dominio semiótico de Paul Gee (2003). De este modo, el jugador puede apreciar, identificar y comprender las relaciones entre los sistemas de símbolos del videojuego con el contenido de aprendizaje. En la Figura 6-4, pueden apreciarse los 7 portales que representan a las Alfas, cuyos los colores coinciden con los empleados en el estándar para identificar a las áreas de interés. Asimismo, cada Alfa comprende un conjunto de niveles que deben jugarse secuencialmente, ya que es la manera en la que evolucionan los estados según el estándar Essence.



Figura 6-4. Núcleo del estándar Essence en Alphaspot

En cada uno de los niveles, el jugador debe encontrar los puntos de la lista de verificación de ese estado (nivel), los cuales se van contabilizando por medio de un marcador en la parte superior derecha (véase Figura 6-5).

Cuando el jugador encuentra todos los puntos de la lista de verificación de un nivel, obtiene la tarjeta del estado correspondiente, la cual puede ser consultada las veces que el jugador lo desee.



Figura 6-5. Nivel de acción del Alfa “Forma de Trabajar”

4. Nivel de evaluación.

Cuando el jugador completa los 41 niveles de acción o aventura, se muestra un vídeo donde los personajes que aparecen en el nivel de introducción, agradecen al jugador por haber conseguido las 41 tarjetas de los estados de las Alfas del núcleo del estándar Essence.

Después, se activa un último nivel que le permitirá al aprendiz evaluar el conocimiento adquirido por medio de preguntas con respuestas de opción múltiple (véase Figura 6-6).



Figura 6-6. Nivel de evaluación en Alphaspot

6.2 Alphalingo

6.2.1 Descripción

Alphalingo es el segundo juego serio computacional construido como parte de la investigación realizada. Es una aplicación móvil que tiene como objetivo facilitar el aprendizaje de las Alfabetas de Essence (Jiménez-Hernández *et al.*, 2016c). Fue diseñado usando como referencia la aplicación móvil para aprender idiomas conocida como Duolingo© (véase Anexo G).

6.2.2 Características

Alphalingo puede ser usado tanto por profesionales laborando en entidades de software, como por estudiantes de carreras afines al área de Computación.

Algunas características de este juego serio computacional son:

- Es gratuito.
- Está disponible en idioma español, inglés y francés.
- Puede jugarse en dispositivos móviles con Android©.

6.2.3 Tecnologías empleadas

Alphalingo fue desarrollado empleando las siguientes tecnologías:

- El lenguaje de etiquetas HTML5.
- El lenguaje de diseño gráfico CSS.
- El lenguaje de programación JavaScript.
- El conjunto de componentes Ionic Framework.
- El conjunto de componentes AngularJS.
- El conjunto de componentes Bootstrap.
- El entorno de desarrollo de aplicaciones móviles Apache Cordova.
- El editor de gráficos Adobe Photoshop.
- El programa de edición de audio Audacity.

6.2.4 Módulos del juego

El contenido educativo de la aplicación móvil se encuentra organizado en nueve módulos, con el objetivo de que los profesionales puedan desarrollar las competencias específicas descritas en la Tabla 6-1.

Tabla 6-1. Competencias específicas de cada módulo de Alphalingo

Módulo	Competencias específicas
1	Comprender y aplicar el estándar Essence en un proyecto de software.
2	Interpretar el concepto de alfa y accionar el núcleo con las alfas de Essence en un proyecto de software.
3	Identificar, establecer y abordar la oportunidad dentro de un proyecto de software usando el Alfa Oportunidad.
4	Reconocer, implicar y establecer acuerdos con los involucrados en un proyecto de software aplicando el Alfa Involucrados.
5	Plantear, acotar, abordar y cumplir los requerimientos en un proyecto de software empleando el Alfa Requerimientos.
6	Seleccionar una arquitectura y conducir el desarrollo de un sistema de software en un proyecto, aplicando el Alfa Sistema de Software.
7	Preparar, iniciar y dar seguimiento al esfuerzo que se realiza en un proyecto de software usando el Alfa Trabajo.
8	Conformar y lograr la colaboración en un equipo dentro de un proyecto de software aplicando el Alfa Equipo.
9	Establecer y poner en marcha el conjunto de prácticas y herramientas para realizar las actividades necesarias en un proyecto de software, empleando el Alfa Forma de Trabajar.

Cuando un jugador realiza un ejercicio en la aplicación móvil, esta le brinda retroalimentación inmediata (Fleming, 1987), lo que le permite al usuario saber si su respuesta fue correcta o incorrecta. Cuando la respuesta es correcta, el aprendiz obtiene “puntos de experiencia”, los cuales se otorgan dependiendo de la complejidad del ejercicio resuelto (véase Figura 6-7). De esta manera, la herramienta incluye en su diseño gamificación, lo que permite aumentar la motivación de los aprendices (Romero *et al.*, 2016) superar la resistencia natural a participar en actividades repetitivas (Challco *et al.*, 2015) acelerar el proceso de aprendizaje (Paim y Barbosa, 2016) y mejorar la retención de la información (González, Gómez y Echeverri, 2017).

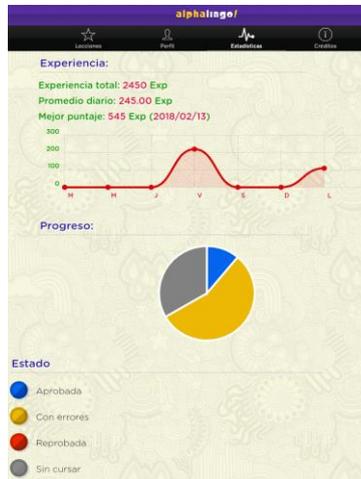


Figura 6-7. Puntos de experiencia y progreso del aprendizaje

Además, la propuesta educativa incluye aprendizaje espaciado (Ebbinghaus, 1885) al distribuir el tiempo de estudio en periodos cortos y, aprendizaje para el dominio (Block *et al.*, 1971) al solicitar que cada módulo sea aprobado con al menos un 80% de aciertos para poder continuar con otro nivel, lo que permite perfeccionar el nivel de aprendizaje.

6.2.5 Tipos de ejercicios

Con la intención de reforzar el conocimiento, Alphalingo presenta un mismo contenido educativo por medio de diferentes ejercicios, los cuales pueden ser percibidos sensorialmente por el aprendiz de cuatro maneras: visualmente, auditivamente, kinestésicamente, o por medio de la lectura/escritura (Hawk y Shah, 2007). Dependiendo del tipo de estilo de aprendizaje se diseñaron 4 tipos de ejercicios:

1. Primer tipo de ejercicios.

Esta clase de ejercicios presentan el contenido educativo de dos maneras: auditivamente y por medio de la lectura. De este modo, el usuario debe repetir el contenido, ya sea de forma escrita o verbalmente (véase Figura 6-8).

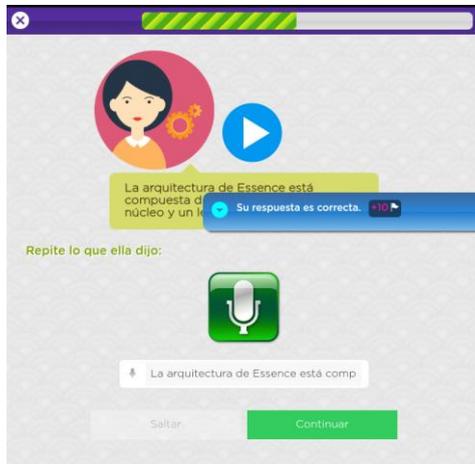


Figura 6-8. Primer tipo de ejercicios en Alphalingo

2. Segundo tipo de ejercicios.

Este tipo de ejercicios presentan un enunciado que contiene uno o dos espacios en blanco, los cuales deben completarse seleccionando las opciones correctas (véase Figura 6-9).

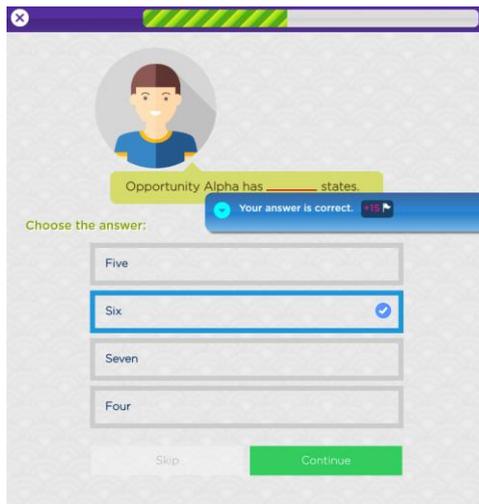


Figura 6-9. Segundo tipo de ejercicios en Alphalingo

3. Tercer tipo de ejercicios.

El tercer tipo muestra una pregunta relacionada con el contenido, cuya respuesta debe seleccionarse a partir de las posibles opciones que despliegan la información de forma visual y textual (véase Figura 6-10).

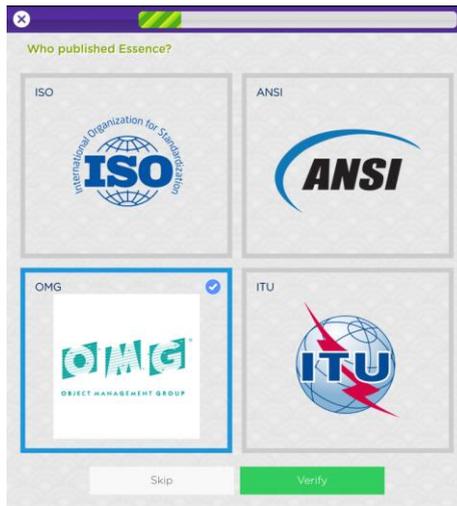


Figura 6-10. Tercer tipo de ejercicios en Alphalingo

4. Cuarto tipo de ejercicios.

El último tipo de ejercicios también expone una pregunta al usuario, pero a diferencia de la clase anterior de ejercicios, la respuesta no consiste en seleccionar un solo elemento, sino que se deben elegir y arrastrar al área indicada todas las palabras que conforman la respuesta correcta.



Figura 6-11. Cuarto tipo de ejercicios en Alphalingo

6.3 Alphadecision

6.3.1 Descripción

Alphadecision es el tercer juego serio computacional que fue desarrollado como parte del trabajo de investigación realizado. Es un sitio Web que tiene como objetivo facilitar el aprendizaje de las Alfas del núcleo del estándar Essence (Jiménez-Hernández *et al.*, 2017d).

La estrategia didáctica de este juego consiste en que el aprendiz:

- Analice individualmente escenarios de desarrollo de software, para determinar, con la ayuda de unas tarjetas impresas, cuál es el estado de un Alfa específico.
- Comparta el análisis realizado con otras personas.
- Plantee y comparta nuevos escenarios de desarrollo de software para ser analizados comunitariamente.

6.3.2 Características

Alphadecision puede ser usado tanto por profesionales de software como por estudiantes de carreras afines al área de Computación.

Algunas características de este juego serio computacional son:

- Es gratuito.
- Está disponible en idioma español e inglés.
- Es portable, ya que puede jugarse desde cualquier dispositivo móvil o computadora con cualquier sistema operativo.

6.3.3 Tecnologías empleadas

En el desarrollo de Alphadecision se emplearon las siguientes tecnologías:

- El lenguaje de etiquetas HTML5.
- El lenguaje de diseño gráfico CSS.
- El lenguaje de programación JavaScript.
- El editor de gráficos Adobe Photoshop.
- El sistema de gestión de bases de datos relacional MySQL.
- El servidor Apache.

6.3.4 Modalidades de juego

Alphadecision posee tres modalidades de juego: jugador virtual, jugador real y certificación (véase Figura 6-12). Para los cuales, se requiere del uso de una baraja con las cartas de los estados de las Alfas de Essence, la cual se le proporciona al usuario al momento de registrarse en la plataforma educativa.

Este juego emplea gamificación en sus tres modalidades, con la intención de recompensar y motivar el aprendizaje del usuario (Romero *et al.*, 2016). En las modalidades de jugador virtual y real se hace a través de insignias (medallas de oro y plata) y de puntos, que otorgan un lugar dentro de un ranking que compara el puntaje de todos los usuarios. Mientras que en la modalidad de certificación, el aprendiz puede obtener un diploma de participación.

A continuación se describen las tres modalidades de juego.



Figura 6-12. Modalidades de juego en Alphadecision

6.3.4.1 Jugador virtual

En esta modalidad, el usuario debe analizar el Alfa relacionado con cuatro casos prácticos de desarrollo de software, para determinar el estado de dicha Alfa. Esto lo hace interactuando con un jugador virtual (véase Figura 6-13).

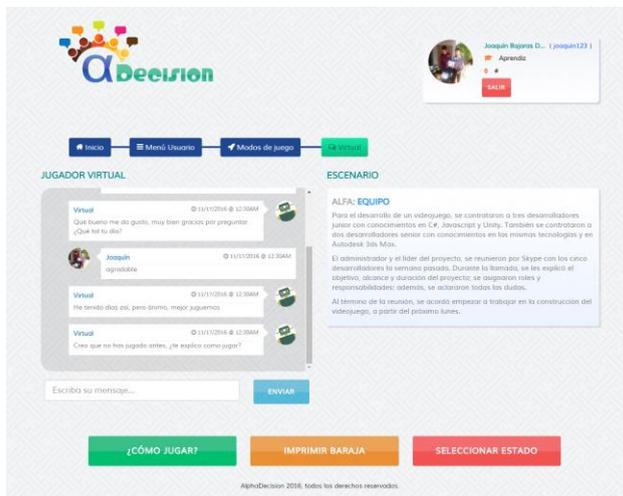


Figura 6-13. Jugador virtual de Alphadecision

6.3.4.2 Jugador real

Una vez que el usuario ha completado el análisis de los escenarios planteados en la modalidad anterior, se desbloquea esta modalidad, en la cual el usuario debe analizar un total de tres casos prácticos por medio de la interacción con otros jugadores en tiempo real (véase Figura 6-14).

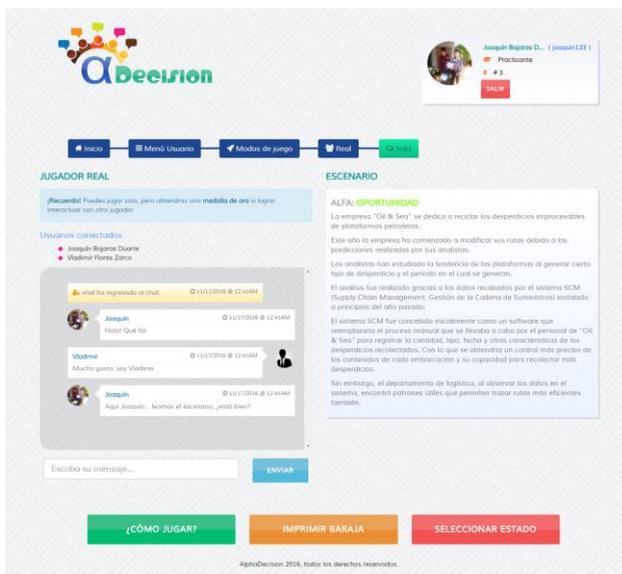


Figura 6-14. Jugadores reales en Alphadecision

6.3.4.3 Certificación

Después de haber concluido las dinámicas solicitadas en las modalidades virtual y real, los usuarios pueden participar en esta modalidad de certificación o evaluación del conocimiento. Para ello, deben inscribirse o solicitar una fecha de certificación con antelación, la cual será confirmada por el administrador del sistema o evaluador (véase Figura 6-15).



Figura 6-15. Calendarización de certificación en Alphadecision

El día de la certificación, los aprendices deben analizar un escenario de desarrollo de software para así ser evaluados por el administrador, quien proporcionará retroalimentación sobre sus respuestas y dará el resultado de la evaluación; en caso de aprobar, recibirán su diploma de participación (véase Figura 6-16).

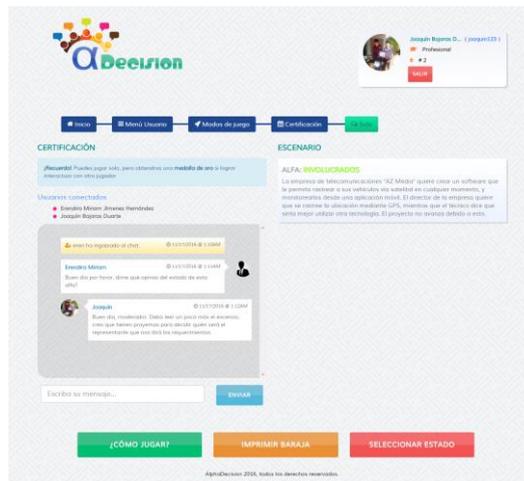


Figura 6-16. Certificación en Alphadecision

Finalmente, es importante mencionar que por medio de Alphadecision, el administrador del sistema puede dar seguimiento al aprendizaje de los usuarios por medio de la consulta de algunos parámetros estadísticos, tal como puede apreciarse en la Figura 6-17.

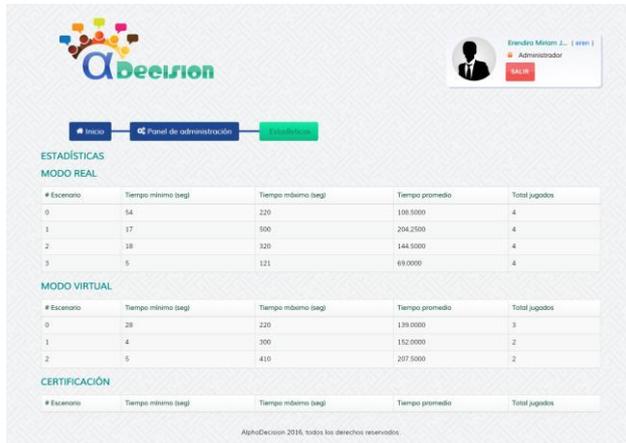


Figura 6-17. Estadísticas de Alphadecision

Asimismo, el administrador cuenta con los permisos para dar de alta, baja o actualizar información referente a usuarios y escenarios, los cuales pueden ser propuestos por los aprendices (véase Figura 6-18).



Figura 6-18. Panel de administración de Alphadecision

Parte III
Validación

Capítulo 7

Validación de los juegos serios computacionales

*“Sólo la propia y personal experiencia
hace al hombre sabio”*
Sigmund Freud (1856 - 1939)

Este capítulo presenta los resultados de la evaluación de la motivación percibida por profesionales de software al aprender sobre las Alfas del núcleo del estándar Essence por medio de Alphaspot.

Además, muestra los resultados de la evaluación de la motivación y del aprendizaje de las Alfas, que se obtuvieron por medio de un Experimento formal con una Réplica, con profesionales que laboraban en dos empresas de software, quienes probaron Alphalingo.

7.1 Alphaspot

7.1.1 Evaluación de la motivación

Alphaspot fue probado por 12 profesionales laborando en una empresa de software en Morelia, Michoacán.

Los participantes jugaron en promedio 2 horas durante 5 días (10 horas en total). Todos completaron los niveles de acción o aventura del videojuego, obteniendo las 41 cartas con los estados de las Alfas del núcleo del estándar Essence. Asimismo, once de los doce participantes aprobaron el nivel final (de evaluación) en el primer intento.

De este modo, después de jugar Alphaspot, los participantes respondieron un cuestionario (véase Apéndice H), con el objetivo de conocer el nivel de motivación en el aprendizaje de las Alfas del núcleo de Essence, que generó el uso de juego serio computacional.

El cuestionario fue diseñado a partir de la Encuesta de Motivación de Materiales Didácticos (EMMD) (Loorbach *et al.*, 2015), que permite valorar los factores ARCS (Atención, Relevancia, Confianza y Satisfacción) por medio de 36 preguntas, las cuales se evalúan con una escala Likert que va de 1 a 5, donde 1 corresponde a “Totalmente en desacuerdo” y 5 a “Totalmente de acuerdo”.

Los resultados obtenidos respecto a los cuatro factores de motivación ARCS, se presentan en la Tabla 7-1. Donde se puede observar que el factor con la mejor evaluación corresponde a la Atención (4.14), seguido por la Satisfacción (4.02), la Relevancia (3.93) y la Confianza (3.86).

Tabla 7-1. Evaluación de los factores de motivación de Alphaspot

Participante	Factor de Motivación			
	Atención	Relevancia	Confianza	Satisfacción
1	4.33	4.02	3.87	4
2	4.05	4	4.02	4
3	4.25	3.87	3.78	4.02
4	3.87	3.87	3.2	3.87
5	4.16	3.65	4	4.21
6	4.21	3.98	3.87	4
7	4.03	4.03	4	3.87
8	4.02	4	4.05	4.02
9	4.33	4.18	4.02	4.33
10	3.98	3.65	3.42	3.87
11	4.35	4.05	4.05	4
12	4.08	3.87	4	4
Promedio	4.14	3.93	3.86	4.02

7.2 Alphalingo

7.2.1 Método de evaluación del aprendizaje

7.2.1.1 Diseño del Experimento

El aprendizaje obtenido por medio de Alphalingo, fue evaluado a través de un Experimento (E) formal de tipo Pre-prueba/Post-prueba grupo de control aleatorizado (Cohen *et al.*, 2007) que consistió en comparar estadísticamente el cambio observado entre un Grupo de Control (GC) y un Grupo Experimental (GE) (Cook y Campbell, 1986).

Asimismo, siguiendo los principios para la replicación de estudios en IS propuesto por Brooks *et al.* (1995), se realizó una Réplica (R) con el objetivo de incrementar la fiabilidad en la validez de la conclusión del Experimento.

De este modo, se administraron dos evaluaciones escritas tanto en el Experimento como en la Réplica:

- a) Un examen diagnóstico (Pre-prueba), que permitió comparar el conocimiento previo de los participantes del GC y del GE sobre las Alfas del núcleo del estándar Essence (véase Apéndice I).
- b) Un examen de conocimientos (Post-prueba), que permitió comparar el cambio observado después de la experimentación entre ambos grupos, donde el GC aprendió sobre las Alfas empleando el método tradicional presencial (clases) y el GE usando Alphalingo (véase Apéndice J).

Los participantes en el Experimento y la Réplica fueron divididos aleatoriamente para formar los Grupos de Control y Experimental.

7.2.1.2 Materiales

Los exámenes aplicados en ambos grupos (GC y GE), tanto en la Pre-prueba como en la Post-prueba, estaban compuestos por 12 preguntas con cuatro posibles respuestas, de las cuales sólo una era la correcta.

Ambos exámenes eran similares respecto a su contenido y dificultad. Fueron calificados por el mismo instructor usando la misma escala (0 a 100).

7.2.1.3 Procedimiento

Tanto el Experimento como la Réplica se llevaron a cabo en un total de 11 días, de la siguiente manera:

- **El primer día**, el instructor administró el examen diagnóstico (Pre-prueba) a ambos grupos (GC y GE). Así como un cuestionario para conocer las características de los participantes (véase Apéndice K).

- **Del segundo al décimo día**, un mismo instructor impartió clases al GC, exponiendo el contenido correspondiente a un módulo de Alphalingo por día (véase Tabla 6-1 del Capítulo 6). Las clases tuvieron lugar en una sala de reuniones de la empresa de software, con una duración de 20 minutos cada una.

En este mismo periodo de tiempo, del día 2 al 10, los participantes del GE usaron Alphalingo, completando los mismo módulos de la Tabla 6-1 (Capítulo 6), pero a su propio ritmo y en el lugar elegido por ellos, como en su casa, oficina, etc.

- **El undécimo día**, el instructor aplicó el examen final para evaluar el conocimiento adquirido en ambos grupos (Post-prueba).

7.2.2 Experimento

7.2.2.1 Ejecución

El Experimento se llevó a cabo siguiendo el método descrito en la Sección 7.2.1, con 60 profesionales que trabajaban en la empresa de software en Guadalajara, México. Los cuales fueron divididos aleatoriamente en dos grupos (cada uno conformado por 30 personas): el Grupo de Control del Experimento GC(E) y el Grupo Experimental del Experimento GE(E).

El GC(E) estaba integrado por 27 hombres y 3 mujeres, mientras que el GE(E) por 26 hombres y 4 mujeres.

7.2.2.2 Análisis estadístico

Los resultados del examen diagnóstico (véase Apéndice L) y del examen final (véase Apéndice M) del Experimento, se analizaron estadísticamente para determinar la diferencia entre el conocimiento de ambos grupos, GC(E) y GE(E), antes y después de la experimentación.

De este modo, primero se analizaron los resultados de la Pre-prueba. Para lo cual se calculó la media, la desviación estándar y el error estándar de ambos grupos (véase Tabla 7-2). Asimismo, se comprobaron los dos supuestos: a)

normalidad, por medio de una prueba Shapiro-Wilk (véase Tabla 7-2) y, b) homocedasticidad, por medio de la prueba de Levene (véase Tabla 7-3). Ambas pruebas se realizaron con un nivel de confianza de 95%.

Tabla 7-2. Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Pre-prueba (E)

Grupo	Medidas Estadísticas				Shapiro-Wilk	
	N	Media	Desv. Est.	Error Est.	Estadístico	<i>p</i> -valor
GC(E)	30	22.77	12.165	2.221	0.958	.279
GE(E)	30	21.93	10.606	1.936	0.944	.115

Desv. Est. – Desviación Estándar
Error Est. – Error Estándar

Una vez comprobados los dos supuestos (normalidad y homogeneidad de las varianzas), se realizó la prueba *t* de Student con un nivel de confianza de 95%, considerando las siguientes hipótesis en la Pre-prueba (E):

H₀: No existe diferencia significativa entre las medias de las calificaciones del conocimiento inicial del GC(E) y las del GE(E).

H_a: Existe diferencia significativa entre las medias de las calificaciones del conocimiento inicial del GC(E) y las del GE(E).

Los resultados de la prueba *t* de Student se muestran en la Tabla 7-3, los cuales permiten afirmar que no existe diferencia significativa entre las medias de las calificaciones del conocimiento previo del GC(E) y las de GE(E), ya que el *p*-valor es .778, que es mayor a 0.05. Esto significa que ni los participantes del GC(E) ni los del GE(E) tuvieron ventaja respecto a su conocimiento inicial sobre las Alfas del núcleo del estándar Essence antes de la experimentación.

Tabla 7-3. Prueba de Levene para la homogeneidad de varianzas y prueba *t* de Student de la Pre-prueba (E)

	Prueba de Levene			Prueba <i>t</i> de Student			
	F	<i>p</i> -valor	<i>t</i>	GL	C		<i>p</i> -valor
					I	S	
AIV	0.304	.584	0.283	58	-5.06	6.73	.778

AIV – Asumiendo Igualdad de Varianzas

GL – Grados de Libertad

IC – Intervalo de Confianza del 95% (Inferior/Superior)

Después, se analizaron los resultados de la Post-prueba de forma similar a los de la Pre-prueba. Las medidas estadísticas del GC(E) y del GE(E) en la Post-prueba, así como el resultado de la prueba de normalidad, se muestran en la Tabla 7-4. Por su parte, el resultado de la prueba de homogeneidad de las varianzas se presenta en la Tabla 7-5.

Tabla 7-4. Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Post-prueba (E)

Grupo	Medidas Estadísticas				Shapiro-Wilk	
	N	Media	Desv. Est.	Error Est.	Estadístico	<i>p</i> -valor
GC(E)	30	70.83	12.365	2.258	0.949	.163
GE(E)	30	80.57	10.569	1.930	0.943	.109

Desv. Est. – Desviación Estándar

Error Est. – Error Estándar

Una vez que se corroboró el cumplimiento de los supuestos de normalidad y homocedasticidad con un nivel de confianza de 95%, se realizó la prueba *t* de Student con el mismo nivel del confianza, considerando las siguientes hipótesis en la Post-prueba (E):

*H*₀: No existe diferencia significativa entre las medias de las calificaciones finales del GC(E) y las del GE(E).

*H*_a: Existe diferencia significativa entre las medias de las calificaciones finales del GC(E) y las del GE(E).

Los resultados de la prueba *t* de Student (Tabla 7-5) muestran que el *p*-valor es .002, el cual es menor a 0.05, por lo que se rechaza la hipótesis nula y se acepta la hipótesis alternativa. Lo que permite afirmar que los participantes del GE(E), que aprendieron por medio de Alphalingo, fueron capaces de obtener una calificación significativamente mayor (*media*=80.57), que los participantes del GC(E), que aprendieron por medio del método tradicional de enseñanza-aprendizaje (*media*=70.83).

Tabla 7-5. Prueba de Levene para la homogeneidad de varianzas y prueba t de Student de la Post-prueba (E)

	Prueba de Levene		Prueba t de Student				
	F	p-valor	t	GL	C		p-valor
					I	S	
AIV	0.624	.443	-3.277	58	-15.6	-3.78	.002

AIV – Asumiendo Igualdad de Varianzas

GL – Grados de Libertad

IC – Intervalo de Confianza del 95% (Inferior/Superior)

7.2.3 Réplica del Experimento

7.2.3.1 Ejecución

Con el objetivo de verificar que los resultados obtenidos en el Experimento no fueran producto de un error de muestreo causado por un error Tipo I, se realizó una Réplica considerando las mismas hipótesis de investigación.

La Réplica se llevó a cabo con 64 profesionales que laboraban en otra empresa de software en Guadalajara, México. Los participantes se dividieron aleatoriamente en dos grupos (cada uno conformado por 32 personas): el Grupo de Control de la Réplica GC(R) y el Grupo Experimental de la Réplica GE(R). Sobre las características de los participantes, el GC(R) estaba conformado por 28 hombres y 4 mujeres, mientras que el GE(R) por 27 hombres y 5 mujeres.

7.2.3.2 Análisis estadístico

Los resultados obtenidos en la Réplica también se analizaron estadísticamente, para determinar la diferencia entre el conocimiento de los dos grupos, GC(R) y GE(R), tanto antes como después de la experimentación (Pre-prueba/Post-prueba).

Las medidas estadísticas del GC(R) y GE(R) en la Pre-prueba, se encuentran en la Tabla 7-6, donde también se muestran los resultados de la prueba de normalidad. Mientras que en la Tabla 7-7, se pueden observar los resultados de la prueba de homogeneidad de las varianzas.

Tabla 7-6. Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Pre-prueba (R)

Grupo	Medidas Estadísticas				Shapiro-Wilk	
	N	Media	Desv. Est.	Error Est.	Estadístico	<i>p-valor</i>
GC(E)	32	22.94	10.800	1.909	0.940	.073
GE(E)	32	22.13	10.285	1.818	0.942	.087

Desv. Est. – Desviación Estándar
Error Est. – Error Estándar

Al haberse comprobado el cumplimiento de los dos supuestos (normalidad y homocedasticidad) con un nivel de confianza del 95%, se realizó una prueba *t* de Student con el mismo nivel de confianza, considerando las mismas hipótesis del Pre-test del Experimento. Los resultados de esta prueba se muestran en la Tabla 7-7, donde puede apreciarse que *p-valor* es .759, que es mayor a 0.05, lo que permite afirmar que ninguno de los grupos poseía ventaja sobre el otro respecto al conocimiento inicial de las Alfas del núcleo de Essence.

Tabla 7-7. Prueba de Levene para la homogeneidad de varianzas y prueba *t* de Student de la Pre-prueba (R)

	Prueba de Levene		Prueba <i>t</i> de Student				
	F	<i>p-valor</i>	<i>t</i>	GL	C		<i>p-valor</i>
					I	S	
AIV	0.019	.890	0.308	62	-4.45	6.08	.759

AIV – Asumiendo Igualdad de Varianzas

GL – Grados de Libertad

IC – Intervalo de Confianza del 95% (Inferior/Superior)

Las medidas estadísticas de los resultados de la Réplica en la Post-prueba, se encuentran en la Tabla 7-8, la cual también incluye los resultados de la prueba de normalidad.

Tabla 7-8. Medidas estadísticas y prueba de normalidad Shapiro-Wilk de la Post-prueba (R)

Grupo	Medidas Estadísticas				Shapiro-Wilk	
	N	Media	Desv. Est.	Error Est.	Estadístico	<i>p-valor</i>
GC(E)	32	70.34	12.346	2.183	0.953	.180
GE(E)	32	81.78	10.279	1.817	0.937	.062

Desv. Est. – Desviación Estándar
Error Est. – Error Estándar

Mientras que la Tabla 7-9 presenta los resultados de la prueba de homocedasticidad y de la prueba t de Student con error del 5%. Esta última prueba se realizó considerando las mismas hipótesis de la Post-prueba en el Experimento y los resultados muestran un p -valor de .00015, el cual es menor a 0.05, por lo que se rechaza la hipótesis nula y se acepta la alternativa. Lo que permite afirmar que los participantes del GE(R) obtuvieron mejores resultados en el aprendizaje de las Alfas del núcleo de Essence ($media=81.78$), que los participantes del GC(R), donde la media es 70.34.

Tabla 7-9. Prueba de Levene para la homogeneidad de varianzas y prueba t de Student de la Post-prueba (R)

	Prueba de Levene		Prueba t de Student				
	F	p -valor	t	GL	C		p -valor
					I	S	
AIV	1.027	.315	-4.027	62	-17.1	-5.76	.00015

AIV – Asumiendo Igualdad de Varianzas
 GL – Grados de Libertad
 IC – Intervalo de Confianza del 95% (Inferior/Superior)

Los resultados estadísticos del Experimento y su Réplica, se calcularon usando el software IBM SPSS Statistics©.

7.2.4 Meta-análisis

La información obtenida en el Experimento y en la Réplica fue sintetizada por medio de un meta-análisis, en el que se estimó el tamaño del efecto individual de cada estudio, así como su potencia estadística, con la finalidad de determinar el tamaño del efecto global.

En primer lugar, puede apreciarse que tanto en el Experimento como en la Réplica se obtuvieron resultados significativos, ya que el p -valor de la prueba t de Student en el Post-test es menor a .05 (véase Tabla 7-10). Además, en ambos casos la potencia estadística es mayor a .80, por lo que la probabilidad estimada de cometer error Tipo II en las afirmaciones se reduce a .103 en el Experimento y .023 en la Réplica ($1-\beta$). De forma similar, la medida g de Hedges de cada estudio es mayor a 0.80, lo que se considera como un tamaño de efecto grande, de acuerdo con Cohen (Cohen, 1988).

Tabla 7-10. Resultados del meta-análisis

Estudio	<i>p</i>-valor	Potencia	<i>g</i>
Experimento	.002	.897	0.835
Réplica	.00015	.977	0.994
Tamaño del efecto global			0.912

Así, el tamaño del efecto global (E y R) es de 0.912, el cual es mayor a 0.80, lo que indica que el alcance de los hallazgos es grande (Cohen J., 1988). En consecuencia, puede afirmarse que Alphalingo tiene un grado de utilidad grande en el aprendizaje de las Alfas del núcleo del estándar Essence.

Finalmente, por medio del diagrama de la Figura 7-1, puede apreciarse cómo se distribuyen las calificaciones iniciales y las finales (E y R). En los resultados de la Pre-prueba se observa que no hay diferencia significativa entre el GC y el GE. Por otra parte, en los resultados de la Post-Prueba, se puede ver la diferencia superior del GE sobre el GC.

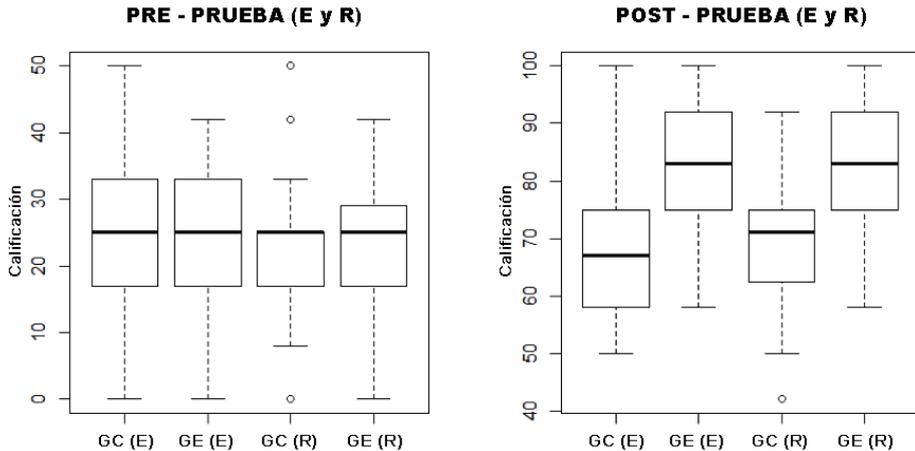


Figura 7-1. Diagrama de caja comparativo de Pre-prueba/Post-prueba del Experimento y la Réplica.

Estos resultados se obtuvieron usando el software G*Power©.

7.2.5 Evaluación de la motivación en el aprendizaje

En la Post-prueba del Experimento y la Réplica, los participantes de ambos grupos (GC y GE) respondieron un cuestionario basado en el EMMD (véase Apéndice H), con la intención de comparar el nivel de motivación en función de los cuatro factores ARCS.

Los resultados obtenidos en el Experimento (véase Apéndice N) y la Réplica, se resumen en la Tabla 7-11, donde se puede apreciar que las medias de todos los factores son mayores en el GE de ambos estudios (E y R), lo que refleja que los profesionales de software del grupo experimental estuvieron más motivados al usar el método aprendizaje basado en Alphalingo, que los profesionales del GC (E y R), quienes aprendieron por medio del método tradicional de enseñanza-aprendizaje.

Tabla 7-11. Promedios de los factores de motivación del Experimento y la Réplica

Factor de Motivación	Experimento		Réplica	
	GC	GE	GC	GE
Atención	3.92	4.33	3.34	4.21
Relevancia	4.06	4.25	3.25	4.16
Confianza	3.92	4.16	4.02	4.33
Satisfacción	3.68	4.21	3.99	4.35
Todos los factores	3.90	4.24	3.65	4.26

7.2.6 Amenazas a la validez

Cuando se lleva a cabo un experimento, es necesario evaluar si sus resultados son válidos (Wohlin *et al.*, 2012), para ello Cook y Campbell (1979) definieron cuatro tipos de amenazas que pueden comprometer la validez de un experimento: validez interna, validez externa, validez del constructo y validez de conclusión.

Respecto a la validez interna del Experimento, se evitó un efecto en la selección de los participantes al dividirlos aleatoriamente en los dos grupos (GC y GE). Además, se aseguró la equivalencia estadística de ambos grupos antes de la experimentación, al aplicarles un examen diagnóstico, que corroboró que ninguno de los grupos tenía una ventaja significativa en cuanto

al conocimiento previo relacionado con las Alfas del núcleo del estándar Essence. Asimismo, para minimizar un posible efecto en la instrumentación, los exámenes de la Pre-prueba/Post-prueba fueron aplicados en el mismo lugar, al mismo tiempo, calificados por el mismo instructor y con la misma escala de evaluación. Como ninguno de los participantes abandonó el Experimento, es posible afirmar que no hubo efecto de mortalidad.

Sin embargo, las amenazas a la validez interna no pueden considerarse del todo inexistentes, puesto que los participantes de ambos grupos (GC y GE) trabajaban en la misma empresa de software, por lo que pudo haber ocurrido una posible difusión de tratamientos.

En relación con la validez externa, como el Experimento sólo ha sido replicado una vez, no sería propio generalizar los resultados. En consecuencia, como apoyo a futuras Réplicas, la caracterización de los participantes del Experimento reportado (véase Apéndice Ñ) y los instrumentos de evaluación empleados, se encuentran disponibles en el sitio Web de Jiménez-Hernández (2018).

En el caso de la validez del constructo, se realizó una prueba piloto antes de llevar a cabo el Experimento, lo que permitió realizar los ajustes necesarios en los instrumentos de evaluación. Para evitar un efecto de aprendizaje, los exámenes de la Pre-prueba y la Post-prueba tenían una estructura similar, pero no igual. Asimismo, como las respuestas a las preguntas de los exámenes eran de opción múltiple, se empleó la distribución binomial para corroborar la probabilidad de aprobar los exámenes seleccionando al azar las respuestas, la cual se reduce a 0.0143.

Con respecto a las amenazas a la validez de conclusión, se evitó cualquier efecto en la confiabilidad del análisis estadístico, al realizar las pruebas de normalidad y homocedasticidad antes de efectuar la prueba t de Student, tanto en la Pre-prueba como en la Post-prueba. De la misma forma, se consideró un nivel de confianza del 95% al realizar todas las pruebas estadísticas.

Las amenazas identificadas para el Experimento, afectan del mismo modo a su Réplica, puesto que se siguió el mismo método de experimentación, con las mismas hipótesis de investigación y los resultados fueron analizados estadísticamente de la misma manera.

Parte IV
Conclusiones

Capítulo 8

Conclusiones

*“En la vida no hay cosas que temer,
sólo hay cosas que comprender”*
Marie Curie (1867-1934)

Este capítulo incluye las conclusiones y los resultados finales obtenidos de esta investigación. Por esta razón, se presenta un análisis de los objetivos de la tesis, se enlistan las publicaciones generadas y se plantean los trabajos a futuro.

10.1 Análisis de objetivos de investigación

En el Capítulo 1, se definió el Objetivo Principal de investigación como:

Construir juegos serios computacionales para facilitar el aprendizaje de las Alfás del núcleo del estándar Essence a profesionales de software.

La consecución del Objetivo Principal de investigación se basó en lograr los siguientes Objetivos Específicos (OEs):

OE1. Diseñar una metodología para la construcción de juegos serios computacionales.

Capítulo 4: Fundamentos Pedagógicos. Presenta una síntesis de las teorías, modelos y principios pedagógicos que fueron objeto de revisión.

Capítulo 5: Metodología para construir juegos serios computacionales. Presenta la metodología que fue diseñada a partir del estudio de los Fundamentos Pedagógicos del *Capítulo 4* y de la revisión de las metodologías para desarrollar software, videojuegos y juegos serios incluida en el *Capítulo 5*.

OE2. Construir juegos serios computacionales que permitan el aprendizaje de las Alfás del núcleo del estándar Essence.

Capítulo 3: Estado del arte. Presenta el estado del arte de la investigación, que fue identificado por medio de un mapeo sistemático de estudios y un análisis de las fortalezas, debilidades, oportunidades y amenazas, que se realizó sobre los estudios primarios seleccionados en el mapeo.

Capítulo 6: Juegos serios computacionales para el aprendizaje de las Alfas de Essence. Presenta los juegos serio computacionales que fueron construidos siguiendo el marco de trabajo diseñado en el *Capítulo 5*, los cuales consideraron los resultados y el análisis realizado en el *Capítulo 3*.

OE3. Realizar al menos un experimento formal con profesionales que laboren en una entidad de software, para probar si al menos uno de los juegos serios computacionales proporcionan una mejora significativa en el aprendizaje de las Alfas del núcleo del estándar Essence, en comparación con el método de aprendizaje tradicional presencial.

Capítulo 7: Validación de los juegos serios computacionales. Presenta los resultados de la evaluación de la motivación y del aprendizaje de las Alfas del núcleo del estándar Essence, que se obtuvieron por medio de un Experimento formal con una Réplica, con profesionales que laboraban en dos empresas de software, quienes probaron el juego serio computacional Alphalingo.

OE4. Realizar una prueba de hipótesis estadística para determinar si el aprendizaje de las Alfas del núcleo del estándar Essence en profesionales de software es significativamente mejor usando juegos serios, que usando el método de aprendizaje tradicional presencial.

Capítulo 7: Validación de los juegos serios computacionales. Los resultados de la prueba de hipótesis realizada con la información obtenida del Experimento y su Réplica, reflejan que se obtienen mejores resultados en el aprendizaje de las Alfas del núcleo del estándar Essence con profesionales de software, al emplear juegos serios computacionales (Alphalingo), que usando método tradicional presencial (clases).

OE5. Diseñar un instrumento de evaluación para conocer y comparar el nivel de motivación de los profesionales de software al aprender sobre las Alfas del núcleo del estándar Essence, a través de los juegos serios computacionales y del método de aprendizaje tradicional presencial.

Capítulo 7: Validación de los juegos serios computacionales. Presenta el cuestionario diseñado para evaluar la motivación. Así como los resultados que muestran que los profesionales de software que utilizaron el juego serio computacional Alphalingo, se sintieron más motivados al aprender sobre las Alfas del núcleo del estándar Essence, que los que aprendieron a través del método tradicional presencial.

Los resultados obtenidos en los dos estudios reportados en el *Capítulo 7*, el Experimento y su Réplica, reflejan que en todo los aspectos valorados respecto al aprendizaje y la motivación de los aprendices, el método basado en juegos serios computacionales superó al método tradicional presencial. Lo que permite afirmar que se logró construir un sistema instruccional efectivo, acorde a los principios pedagógicos del constructivismo y humanismo, en los cuales, el aprendizaje activo, autorregulado y orientado a la solución de problemas con un tratamiento lúdico y práctico, permite el aprendizaje de contenidos y habilidades propios de un campo profesional.

Por lo anterior, puede decirse que la innovación está presente en esta investigación, que atiende a la configuración de nuevos escenarios de aprendizaje, los cuales emergen de nuevos ambientes o ecologías de enseñanza-aprendizaje (Coll, 2013), donde es indispensable vincular el aprendizaje formal e informal a través de prácticas flexibles y lúdicas, que generen sentido y significado ante los retos de la sociedad de la información.

10.2 Producción

Como parte del trabajo realizado, se obtuvieron los siguientes productos de investigación y se desarrollaron las siguientes actividades:

10.2.1 Artículos en Revistas Internacionales

- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., y Piattini, M. Using web-based gamified software to learn Boolean algebra simplification in a blended learning setting. *Computer Applications in Engineering Education*, 1-21. ISSN: 1099-0542, 2019 Impact Factor: 0.856. <https://doi.org/10.1002/cae.22335>, 2020.

- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., y Piattini, M. Using an Educational Mobile Application for Learning the Essence 1.2 Kernel Alphas. *IEEE Latin America Transactions*, 1-9. ISSN: 1548-0992, 2019 Impact Factor: 0.804. Aceptado el 2 de Agosto del 2020, por publicarse.

10.2.2 Artículos de Congreso

- Jiménez-Hernández, E. M., Oktaba, H., Piattini, M., y Díaz-Barriga, F. *Methodology to Construct Educational Video Games in Software Engineering*. 2016 4th International Conference in Software Engineering Research and Innovation (CONISOFT). 2016, Puebla, México. 110-114, doi: 10.1109/CONISOFT.2016.25.
- Jiménez-Hernández, E. M., Oktaba, H., Piattini, M., y Díaz-Barriga, F. *Serious Games When Used to Learn Software Processes: An Analysis from a Pedagogical Perspective*. 2017 5th International Conference in Software Engineering Research and Innovation (CONISOFT). 2017, Mérida, México. 194-203, doi: 10.1109/CONISOFT.2017.00031.

10.2.3 Capítulos de Libros

- Jiménez-Hernández, E. M., y Oktaba, H. “Experiencia práctica: usando el modelo de Jonassen para el desarrollo de juegos serios virtuales” en *Construcción de buenas prácticas educativas mediadas por tecnología*, A. Flores-Hernández, F. Díaz-Barriga, y M. A. Rigo-Lemini, Eds. Puebla, México: Benemérita Universidad Autónoma de Puebla, 2016, cap. 14, págs.97-101. ISBN: 978-607-525-210-0.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagiedo-Tulais, A. M., y López-Guzmán, A. “Using ISO/IEC 29110 Deployment Package to construct educational video games in software engineering” en *Software Engineering: Methods, Modeling, and Teaching*, vol. 4, C. M. Zapata-Jaramillo, C. E. Durango-Vanegas, y W. Perdomo-Charry, Eds. Bogotá, Colombia: Universidad de San Buenaventura, 2017, cap. 2, págs. 27-40. ISBN: 978-958-8928-49-4.

10.2.4 Carteles

- Jiménez-Hernández, E. M., Aguilar-Landero, F., Oktaba, H., y Díaz-Loaiza, J. *Ardilla Ekorn: un videojuego para el aprendizaje del núcleo de Essence*. Coloquio Internacional de Experiencias Educativas Mediadas por Tecnología, Puebla, México, 2015.
- Jiménez-Hernández, E. M., Hernández-García, P., Oktaba, H., y Alvarado-Sánchez, R. *Essence's Labyrinth: un juego serio virtual para la enseñanza de las actividades y competencias del núcleo de Essence*. Coloquio Internacional de Experiencias Educativas Mediadas por Tecnología, Puebla, México, 2015.

10.2.5 Conferencias magistrales

- “Evaluando la salud y el progreso de tu proyecto de software con Essence”, en Congreso Multidisciplinario en Ingeniería y Tecnologías para la Innovación Tékhne, 20 de Octubre del 2015, Morelia, México.
- “La importancia de las Aplicaciones Móviles en la Nueva Era Educativa Digital”, en 5th International Conference in Software Engineering Research and Innovation (CONISOFT), 27 de Octubre del 2017, Mérida, México.
- “Essence: a new OMG standard for Software Engineering”, en ProMove Innovation Days, 21 de Noviembre del 2017, Río de Janeiro, Brasil.

10.2.6 Talleres impartidos

- “Elaboración de Revisiones Sistemáticas de Literatura”. Duración: 4 horas. En el 3th International Conference in Software Engineering Research and Innovation, 27 de Abril de 2015, San Luis Potosí, México.

- “Un juego de cartas para evaluar la salud del software”. Duración: 4 horas. En el TSP Symposium, 23 de Febrero del 2016, Ciudad de México, México.
- “Evaluando la salud y el progreso de software con un juego de cartas”. Duración: 4 horas. En el 4th International Conference in Software Engineering Research and Innovation, 29 de Abril de 2016, Puebla, México.
- “Construction of Educational Video Games”. Duración: 4 horas. En el 5th International Conference in Software Engineering Research and Innovation, 25 de Octubre de 2017, Mérida, Yucatán, México.

10.2.7 Estancias de investigación

- Universidad de Castilla - La Mancha (UCLM), Ciudad Real, España. “Estrategias lúdicas para la enseñanza de Ingeniería de Software”. Asesor: Dr. Mario Piattini, Abril-Noviembre 2016.

10.2.8 Formación de recursos humanos

- Co-asesora durante el XX Verano de la Investigación Científica y Tecnológica del Pacífico, realizado en la Universidad Nacional Autónoma de México del 22 de junio al 7 de agosto de 2015, de los siguientes estudiantes:
 1. Alan Martin Revillagigedo Tulais
 2. Arturo López Guzmán
 3. Sergio Vladimir Flores Zarco
 4. Daniel Bárcenas Acosta
 5. Francisco Javier Aguilar Landero
 6. Pamela Isui Hernández García
 7. Maricruz Chávez Chávez
 8. Jessica Montserrat Comparán Carrillo
 9. José Alfredo Mendoza Heredia
 10. Jorge Manuel Díaz Loaiza
 11. Romario Alvarado Sánchez

- Co-asesora durante el XXVI Verano de la Academia Mexicana de Ciencias (ACM), realizado en la Universidad de Castilla-La Mancha del 27 de junio al 26 de agosto de 2016, de los siguientes estudiantes:
 1. Alan Martin Revillagigedo Tulais
 2. Arturo López Guzmán
 3. Sergio Vladimir Flores Zarco
 4. Daniel Bárcenas Acosta
- Asesora de proyecto “Alhaspot: un juego serio computacional para aprender el estándar Essence”, en la X Edición del Concurso Latinoamericano de Proyectos de Cómputo (SOLACYT), Sede Regional México-Michoacán, el 20 de febrero de 2016.

10.3 Trabajos a futuro

Como a trabajo a futuro se pretende:

- Replicar la experimentación del juego serio computacional Alphalingo en otros contextos, como el educativo. Para conocer el nivel de aprendizaje y motivación que puede alcanzarse con estudiantes de carreras afines a la Computación.
- Realizar experimentación formal con los juegos serios computacionales Alhaspot y Alphadecision, tanto en entornos laborales como académicos.
- Formalizar la descripción de la metodología diseñada para construir juegos serios computacionales por medio del lenguaje Kuali-Beh, perteneciente al estándar Essence (Morales-Trujillo, Oktaba y Piattini, 2015).
- Aplicar la metodología para construir juegos serios computacionales diseñada en esta investigación, para desarrollar juegos que faciliten el aprendizaje de diversos contenidos educativos.

Referencias

- 2K Boston. (2009). *BioShock*. Recuperado el 3 de Febrero de 2018, de BioShock: <http://www.bioshock2game.com>
- Abrahamsson, P., Warsta, J., Siponen, M. y Ronkainen, J. (2003). *New directions on agile methods: a comparative analysis*. International Conference on Software Engineering (págs. 244-254). San Francisco: IEEE.
- Abt Associates. (1965). *Biography of Clark C. Abt*. Recuperado el 21 de Septiembre de 2018, de <http://www.abtassociates.com/About-Us/Our-History.aspx>
- Abt, C. (1970). *Serious games*. California: Viking Press.
- Acerenza, N. et al. (2009). Una Metodología para Desarrollo de Videojuegos. *Simposio Argentino de Ingeniería de Software (ASSE)*, (págs. 171-176). Mar del Plata.
- Aksakal, N. (2015). Theoretical View to The Approach of The Edutainment. *Procedia Social and Behavioral Sciences*, 1232-1239.
- Anderson, L. W. y Krathwohl, D. R. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. New York: Addison Wesley Longman.
- Atkinson, R. (1972). Optimizing the learning of a second-language vocabulary. *Journal of Experimental Psychology*, 96 (1), 124-129.
- Ausubel, D. P. (1968). *Educational Psychology: A Cognitive View*. New York: Holt, Rinehart and Winston.
- Ausubel, D. P. (1976). *Psicología educativa*. México: Trillas.
- BANDAI NAMCO Entertainment Inc. (1980). *Pac-Man Stories*. Recuperado el 20 de Enero de 2018, de Pac-Man: <http://pacman.com/en/>
- Barrier, M. (2008). *The Animated Man: A Life of Walt Disney*. California: University of California Press.
- Beck, K. (1999). *Extreme Programming Explained: Embrace Change*. Addison Wesley Longman.
- Bernal, C. A. (2006). *Metodología de la investigación* (2ª edición ed.). Ciudad de México: Pearson.
- Block, J., Airasian, P., Bloom, B. y Carroll, J. (1971). *Mastery Learning: Theory and Practice*. New York: Holt Rinehart & Winston.
- Bloom, B. (1956). *Taxonomy of Educational Objectives: The Classification of Educational Goals*. Philadelphia: David McKay Company, Inc.
- Bloom, K. y Shuell, T. (1981). Effects of massed and distributed practice on the learning and retention of second language vocabulary. *Journal of Educational Psychology*, 74, 245-248.

- Boelens, R. et al. (2018). The design of blended learning in response to student diversity in higher education: Instructors' views and use of differentiated instruction in blended learning. *Computers & Education*, 197-212.
- Bogost, I. (2007). *Persuasive games: The expressive power of videogames*. Massachusetts: The MIT Press.
- Brooks, A., Daly, J., Miller, M., y Wood, M. (1995). *Reporting of Experimental Results in Software Engineering*. University of Strathclyde, Dept. of Computer Science.
- Bunge, M. (1979). *La ciencia, su método y su filosofía*. Bogotá: Siglo XX.
- Cai, Y., Miao, C., Tan, A., Shen, Z. y Li, B. (2009). Creating an Immersive Game World with Evolutionary Fuzzy Cognitive Maps. *IEEE Journals & Magazines*, 30, 58-70.
- Callaghan, M., Savin-Baden, M., McShane, N., y Gómez, A. (2015). Mapping Learning and Game Mechanics for Serious Games Analysis in Engineering Education. *IEEE Transactions on Emerging Topics in Computing*, 77-83.
- Campo Santo. (2018). *Firewatch*. Recuperado el 14 de Enero de 2018, de Firewatch: <http://www.firewatchgame.com>
- Castro-Rojas, L. F., Montaña-Lince, S. y Espitia-Peña, E. (2016). *Goal Oriented Requirements Engineering supported by the SEMAT kernel*. 4th International Conference in Software Engineering Research and Innovation (CONISOFT) (págs. 65-70). Puebla: IEEE.
- Cepeda, N., Pashler, H., Vul, E., Wixted, J. y Rohrer, D. (2006). Distributed practice in verbal re- call tasks: A review and quantitative synthesis. *Psychological Bulletin*, 132 (3), 354.
- Chalco, G. C., Moreira, D., Bittencourt, I., Mizoguchi, R. y Isotani, S. (2015). Personalization of Gamification in Collaborative Learning Contexts using Ontologies. *IEEE Latin America Transactions*, 13 (6), 1995-2002.
- Cloud Imperium Games. (2018). *Star Citizen*. Recuperado el 6 de Marzo de 2018, de Star Citizen: <https://robertsspaceindustries.com/star-citizen/>
- Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Science*. New Jersey: Lawrence Erlbaum Hillsdale.
- Cohen, L., Manion, L. y Morrison, K. (2007). *Research Methods in Education*. London, UK: Routledge.
- Coll, C. (2001). "Constructivismo y educación. La concepción constructivista de la enseñanza y el aprendizaje". En C. Coll, J. Palacios, y A. Marchesi (Edits.), *Desarrollo psicológico y educación* (Vol. 2). Madrid: Alianza Editorial.
- Coll, C. (2013). El currículo escolar en el marco de la nueva ecología del aprendizaje. *Aula de innovación educativa*, 219, 31-36.
- Coll, C. y Monereo, C. (2008). *Psicología de la educación virtual*. Madrid: Morata.
- Cook, T. y Campbell, D. (1979). *Quasi-experimentation – Design and Analysis Issues for Field Settings*. Boston, MA, USA: Houghton Mifflin Company.

- Cook, T. y Campbell, D. (1986). The causal assumptions of quasi- experimental practice. *Synthese*, 68 (1), 141-180.
- Crytek GmbH. (2018). *Features*. Recuperado el 12 de Febrero de 2018, de CryEngine: <https://www.cryengine.com>
- CryTek. (2018). *Ryse: Son of Rome*. Recuperado el 5 de Marzo de 2018, de Ryse: Son of Rome: <http://www.crytek.com/games/ryse/overview>
- Curz-Cunha, M. (2012). *Handbook of Research on Serious Games as Educational, Business and Research Tools*. Pennsylvania: IGI Global.
- Darwin, C. (1859). *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. London: John Murray.
- Darwin, C. (1877). *El origen de las especies*. (E. Godínez y A. Zulueta, Trans.) España: Biblioteca Perojo.
- Daungharone, K. (2017). *Enhancement the computational thinking skills via the simulation game*. International Conference on Digital Arts, Media and Technology (ICDAMT) (págs. 195-199). Chiang Mai: IEEE.
- De-Marcos, L., García-López, E. y García-Cabot, A. (2016). On the effectiveness of game - like and social approaches in learning: Comparing educational gaming, gamification & social networking. *Computers & Education*, 95, 99-113.
- DeKeyser, R. (2008). "Implicit and explicit learning". En C. Doughty y M. Long, *The Handbook of Second Language Acquisition* (págs. 313–348). New Jersey: Blackwell Publishing Ltd.
- Department of Defense. (2002). *Proving grounds*. Recuperado el 5 de Marzo de 2008, de American's Army: <https://www.americasarmy.com>
- Deterding, S., Khaled, R., Nacke, L. y Dixon, D. (2011). *Gamification: Toward a Definition*. Conference on Human Factors in Computing Systems (págs. 1-4). Vancouver: ACM.
- Díaz Barriga, F. (2010). Principios de diseño instruccional de entornos de aprendizaje apoyados con TIC: Un marco de referencia sociocultural y situado. *Tecnología y comunicación educativas*, 20 (41), 4-16.
- Díaz-Barriga, F. y Hernández-Rojas, G. (2010). *Estrategias docentes para un aprendizaje significativo: una interpretación constructivista* (3ª edición ed.). Ciudad de México: McGraw-Hill.
- Dibona, C. (2004). *A conversation with Will Harvey*. ACM Queue , 21-27.
- Domínguez, A., Saenz-de-Navarrete, J., de-Marcos, L., Fernández-Saenz, L., Pagés, C. y Martínez-Herráiz, J. (2013). Gamifying learning experiences: Practical implications and outcomes. *Computers & Education*, 63, 380-392.
- Drudi, S., Litvack, F., Chesini, E., Castro, A., Pisetta, R. y Becerra, L. (2013). A Motivating Experience for Learning Alternating Current,. *IEEE Latin America Transactions*, 11 (1), 579-584.
- Duolingo. (2013). *Manifiesto*. Recuperado el 24 de Febrero de 2016, de Doulingo: www.duolingo.com

- Ebbinghaus, H. (1885). *Memory: A Contribution to Experimental Psychology*. New York: Columbia University.
- Engelsman, W. y Wieringa, R. (2012). “Goal-oriented requirements engineering and enterprises architecture: Two case studies and some lessons learned”. En B. Regnell y D. Damian (Edits.), *Requirements Engineering: Foundation of Software Quality* (Vol. 7195, págs. 306-320). Heidelberg: Springer Berlin.
- Epic Games. (2018). *Make something unreal*. Recuperado el 26 de Enero de 2018, de Unreal Engine: <https://www.unrealengine.com/en-US/what-is-unreal-engine-4>
- Fahy, P. J. (2004). “Media characteristics and online learning technology”. En T. Anderson y F. Elloumi (Edits.), *Theory and Practice of Online Learning* (págs. 137–171). Athabasca, Alberta, Canada: Athabasca University.
- Fernandez, A., Insfran, E. y Abrahão, S. (2011). Usability evaluation methods for web: A systematic mapping study. *Information and Software Technology*, 53, 789-817.
- Fine, L. (2009). The SWOT Analysis: Using your Strength to overcome Weaknesses, Using Opportunities to overcome Threats. 23-49. CreateSpace Independent Publishing Platform.
- Fleming, M. (1987). “Displays and communication”. En R. M. Gagne (Ed.), *Instructional technology foundations* (págs. 233- 260). Hillsdale, NJ: Erlbaum.
- Fleming, N. D. (1995). “I’m different; not dumb Modes of presentation (V.A.R.K.) in the tertiary classroom”. En A. Zelmer (Ed.), *Research and Development in Higher Education*, Proceedings of the 1995 Annual Conference of the Higher Education and Research Development Society of Australasia (HERDSA) (Vol. 18, págs. 308-313). Perth: HERDSA.
- Garaj, V. (2010). m-Learning in the Education of Multimedia Technologists and Designers at the University Level: A User Requirements Study. *IEEE Transactions on Learning Technologies*, 3 (1), 24-32.
- Gee, P. (2003). *What video games have to teach us about learning and literacy*. New York: Palgrave Macmillan.
- Genero, M., Cruz-Lemus, J. y Piattini, M. (2014). *Métodos de investigación en ingeniería de software*. Madrid, España: Ra-Ma.
- González, J. y Wagenaar, R. (Edits.). (2003). *Tuning Educational Structures in Europe*. Deusto: Universidad de Deusto.
- Gross, K. (1902). *Les Jeux des animaux*. Paris: Félix Alcan Éditeur.
- Hannay, J., Sjøberg, D. y Dybå, T. (2007). A systematic review of theory use in software engineering experiments. *IEEE Transactions on Software Engineering*, 33 (2), 87-107.
- Hawk, F. y Shah, A. (2007). Using Learning Style Instruments to Enhance Student Learning. *Decision Sciences Journal of Innovative Education*, 5, 1-19.
- Hernández, P., Alvarado, R., Jiménez-Hernández, E. M. y Oktaba, H. (2015). Cartel: Essence’s Labyrinth: un juego serio virtual para la enseñanza de las actividades y

- competencias del núcleo de Essence. Primer Coloquio Internacional de Experiencias Educativas Mediadas por Tecnología. Puebla.
- Hill, R. R. y Tolk, A. (2017). “A History of Military Computer Simulation”. En A. Tolk, J. Fowler, G. Shao y E. Yücesan (Edits.), *Advances in Modeling and Simulation. Simulation Foundations, Methods and Applications* (págs. 277-299). Cham, Switzerland: Springer.
- Huizinga, J. (1949). *Homo Ludens. A study of the play-element in culture*. London: Routledge & Kegan Paul.
- Humphrey, A. S. (2010). *Management consulting, Transformation of culture, SWOT analysis, Harvard University, SRI International, Investors in People*. (F. P. Miller, A. F. Vandome y J. McBrewster, Edits.) Alphascript Publishing.
- IEEE. (2008). *International Standard ISO/IEC 12207-Systems and software engineering-Software life cycle processes*. IEEE.
- inXile entertainment. (2018). *Wasteland*. Recuperado el 21 de Enero de 2018, de Wasteland 2: <https://wasteland2.inxile-entertainment.com>
- ISO/IEC 29110. (2015). *Deployment Package. Part 1- Technical Description , 1-48. International Organization for Standardization (ISO)*.
- Jacobson, I., Booch, G. y Rumbaugh, J. (1999). *The unified software development process*. Boston: Addison-Wesley Longman.
- Jensen, B. (2006). Responding to the enrollment crisis-Alternative strategies to increasing students’ interest in Computer Science. *Journal of Computing Science in Colleges*, 21 (4), 8-8.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagigedo, A. et al. (2015). *Alphaspot*. Recuperado el 16 de Febrero de 2016, de Alphagames: alphagames.mx/Alphaspot
- Jiménez-Hernández, E. y Oktaba, H. (2016). “Experiencia práctica: usando el modelo de Jonassen para el desarrollo de juegos serios virtuales”. En A. Flores, F. Díaz-Barriga y M. Rigo (Edits.), *Construcción de buenas prácticas mediadas por tecnología* (págs. 97-101). Puebla: Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagigedo, A. et al. (2016a). *Methodology to construct educational video games in software engineering*. International Conference in Software Engineering Research and Innovation (CONISOFT) (págs. 110-114). Puebla: IEEE.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagigedo, A. et al. (2016b). “Using ISO/IEC 29110 Deployment Package to construct educational video games in software engineering”. En *Software Engineering: Methods, Modeling, and Teaching* (Zapata-Jaramillo, C.M., Durango-Vanegas, C. E. y Perdomo-Charry, W. Edits., Vol. 4, págs. 27-40). Medellín, Colombia: Editorial de la Universidad de San Buenaventura.

- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagigedo, A. *et al.* (2016c). *Alphalingo*. Recuperado el 3 de Julio de 2016, de Alphagames: alphagames.mx/alphalingo
- Jiménez-Hernández, E. M., Jiménez-Murillo, J. A., Ortíz, O., Oktaba, H., Díaz-Barriga, F., Piattini, M. *et al.* (2017a). *El viaje de León Grafíel*. Recuperado el 29 de Septiembre de 2020, de Grupo de Investigación Karani: <http://karani.mx/proyectos/leonegrafiel/>
- Jiménez-Hernández, E. M., Jiménez-Murillo, J. A., Ortíz, O., Oktaba, H., Díaz-Barriga, F., Piattini, M. *et al.* (2017b). *CatMat*. Recuperado el 29 de Septiembre de 2020, de Grupo de Investigación Karani: <http://karani.mx/proyectos/catmat/>
- Jiménez-Hernández, E., Oktaba, H., Piattini, M. y Díaz-Barriga, F. (2017c). *Serious games when used to learn software processes: An analysis from a pedagogical perspective*. 5th International Conference in Software Engineering Research and Innovation (págs. 194-203). Mérida: IEEE.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Revillagigedo, A. *et al.* (2017d). *Alphadecision*. Recuperado el 29 de Julio de 2017, de Alphagames: alphagames.mx/alphadecision
- Jiménez-Hernández, E. M. (2018). *E-Portafolio*. Recuperado el 24 de Enero de 2018, de Eréndira Jiménez: <http://www.erendirajimenez.mx>
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F., Piattini, M., Olvera, M., Arreygue, J. *et al.* (2019). *MiniBool*. Recuperado el 29 de Septiembre de 2020, de <http://minibool.com/>
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F. y Piattini, M. (2020a). Using web-based gamified software to learn Boolean algebra simplification in a blended learning setting. *Computer Applications in Engineering Education*, 1-21.
- Jiménez-Hernández, E. M., Oktaba, H., Díaz-Barriga, F. y Piattini, M. (2020b). Using an Educational Mobile Application for Learning the Essence 1.2 Kernel Alphas. *IEEE Latin America Transactions*, por publicarse.
- Jiménez-Murillo, J. A., Jiménez-Hernández, E. M. y Alvarado-Zamora, L. N. (2014). *Fundamentos de Programación Diagramas de flujo, Diagramas N-S, Pseudocódigo y Java*. Ciudad de México: Alfaomega.
- Jiménez-Murillo, J. A. (2015). *Matemáticas para la Computación*. Ciudad de México: Alfaomega.
- Jiménez-Murillo, J. A. (2016). *Karani*. Recuperado el 29 de septiembre de 2020, de Grupo de investigación Karani: <http://karani.mx/>
- Johannesson, S. M. y Blacklund, P. (2007). *Serious games—An overview*. Technical Report HS-IKI-TR-07-001.
- Jonassen, D. (1991). Evaluating Constructivist Learning. *Educational Technology*, 31 (9), 28-33.

- Jonassen, D. (2000). "El diseño de entornos constructivistas de aprendizaje". En C. Reigeluth, Diseño de la instrucción. *Teoría y modelos* (págs. 93-224). Madrid, España: Aula XXI Santillana.
- Jonassen, D. y Rorher-Murphy, L. (1999). Activity Theory as framework for designing constructivist learning environments. *Educational Technology: Research and Development*, 46 (1).
- Kaplan, A. (2001). *Métodos de investigación en psicopedagogía*. España: McGraw-Hill.
- Kasvi, J. J. (2000). "Not just fun and games - Internet games as a training medium". En P. Kymäläinen y L. Seppänen (Edits.), *Cosiga - Learning with computerized simulation games* (págs. 23-34). Espoo: HUT.
- Keller, J. (2006). *Development of two measures of learner motivation*. Tallahassee: Florida State University.
- Keller, J. M. (1987). The systematic process of motivational design. *Performance & Instruction*, 16 (9), 1-8.
- Keller, J. M. (2010). *Motivational Design for Learning and Performance: The ARCS Model Approach*. New York, USA: Springer.
- Keller, J. M. y Suzuki, K. (1988). "Use of the ARCS Motivation Model in Courseware Design". En D. Jonassen y D. Jonassen (Ed.), *Instructional designs for microcomputer courseware*. Hillsdale, N. J., USA: Lawrence Erlbaum Associates.
- Kent, S. (2001). *The ultimate history of video games*. New York: Three Rivers Pr.
- Kill, K., de Freitas, S., Arnab, S. y Lainema, T. (2012). The Design Principles for Flow Experience in Educational Games. *Procedia Computer Science*, 15, 78-91.
- Kim, A. (2006). The New Era of in-Game Advertising: The Fading Line between Fantasy and Reality. *Harris Interactive Trends & Tudes Newsletter*, 1-6.
- Kitchenham, B. y Charters, S. (2007). *Guidelines for performing systematic literature reviews in software engineering*. EBSE-2007-01, Keele University.
- Kitchenham, B. y Charters, S. (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Keele University, Software Engineering Group. Durham: EBSE-2007-01.
- Kitchenham, B. et al. (2009). Systematic literature reviews in software engineering - A systematic literature review. *Information and Software Technology*, 51 (1), 7-15.
- Kitchenham, B., Budgen, D. y Brereton, P. (2011). Using mapping studies as the basis for further research - A participant-observer case study. *Information and Software Technology*, 53 (6), 638-651.
- Kitchenham, B. et al. (2002). Preliminary guidelines for empirical research in Software Engineering. *IEEE Transactions in Software Engineering*, 28 (8), 721-734.
- Kolb, D. A. (1984). *Experiential Learning. Experience as the source of learning and development*. New Jersey: Prentice Hall.
- Landers, R. N. y Callan, R. C. (2011). "Casual social games as serious games: The psychology of gamification in undergraduate education and employee training". En

- A. Ma, A. Oikonomou y L. C. Jain (Edits.), *Serious Games and Edutainment Applications* (págs. 399-423). London: Springer.
- Lara-Muñoz, E. M. (2011). *Fundamentos de investigación - Un enfoque por competencias*. Ciudad de México: Alfaomega.
- Lewin, K. (1946). Action research and minority problems. *Journal of Social Issues*, 2, 34-46.
- Likert, R. (1932). A Technique for the Measurement of Attitudes. *Archives of Psychology*, 140, 1-55.
- Lui, R. W. y Au, C. H. (2017). *Designing simulation games for information systems education - A case study in teaching for digital marketing*. 6th International Conference on Teaching, Assessment, and Learning for Engineering (TALE) (págs. 290-295). Hong Kong: IEEE.
- MAP. (2001). *Métrica 3: guía de referencia*. Madrid: Tecnos.
- Marcano, B. (2008). Juegos serios y entrenamiento en la sociedad digital. *Revista Electrónica Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, 9 (3), 93-107.
- Marne, B., Huynh-Kim-Bang, B. y Labat, J. (2011). *Articuler motivation et apprentissage grâce aux facettes du jeu sérieux*. Conférence Environnements Informatiques pour l'Apprentissage Humain, (págs. 1-12). Mons.
- Maxwell, H. (2014). *The Game Production Handbook*. Burlington: Jones & Bartlett Learning .
- McConell, S. (2006). *The Ten Most Important Ideas in Software Engineering*. Construx.
- Melton, A. (1970). The situation with respect to the spacing of repetitions and memory. *Journal of Verbal Learning and Verbal Behavior*, 9, 596-606.
- Merrill, D., Drake, L., Lacy, M., Pratt, J. y Group, I. R. (1996). Reclaiming instructional design. *Education Technology*, 3 (5), 5-7.
- Metro Trains Melbourne. (Noviembre de 2012). *Dumb ways to die – PSA*. Recuperado el 3 de Marzo de 2018, de Dumb ways to die: <http://www.dumbwaystodie.com>
- Michael, D. y Chen, S. (2005). *Serious Games: Games That Educate, Train, and Inform*. Cengage Learning Ptr.
- Morales-Trujillo, M. E. y Oktaba, H. (2016). *Using ESSENCE ALPHAs in a CMMI level 5 software development organization*. Federated Conference on Computer Science and Information Systems (FedCSIS) (págs. 1531-1538). Gdansk: IEEE.
- Morales-Urrutia, G. A. et al. (2010). Procesos de desarrollo para videojuegos. *CULCyT*, 7 (36/37), 25-39.
- Niantic, Inc. (2016). Pokémon GO. Recuperado el 12 de Enero de 2018, de Pokémon GO: <https://www.pokemongo.com/es-la/>
- Nwokeji, J. y Holmes, T. (2017). The impact of learning styles on student performance in flipped pedagogy. 2017 IEEE Frontiers in Education Conference (FIE) (págs. 1-7). Indianapolis: IEEE.

- Nowostawski, M., McCallum, S. y Mishra, D. (2018). Gamifying research in software engineering. *Computer Applications in Engineering Education*, 26, 1641– 1652.
- Okan, Z. (2003). Edutainment: Is a Learning At Risk? *British Journal of Educational Technology*, 255-264.
- OMG. (2014). *ESSENCE - Kernel and language for software engineering methods*. MA: Object Management Group.
- Osborne, A., Coleman, G. y O'Connor, R. (2014). Software Development Processes for Games: A Systematic Literature Review. *Systems, Software and Services Process Improvement*, 425, 182-193.
- Pahl, N. y Richter, A. (2009). *SWOT Analysis - Idea, Methodology And A Practical Approach*. Norderstedt, Alemania: GRIN Verlag.
- Paim, C. y Barbosa, L. (2016). Octopus: A gamification model to aid in ubiquitous care of chronic diseases. *IEEE Latin America Transactions*, 14 (4), 1948-1958.
- Paraskeva, F., Mysirlaki, S. y Paggianni, A. (2010). Multiplayer online games as educational tools: Facing new challenges in learning. *Computers & Education*, 54, 498-505.
- Paulk, M. et al. (1993). *Capability Maturity Model for Software*. Pittsburgh: Carnegie Mellon University.
- Pavlik Jr, P. y Anderson, J. (2008). Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied*, 14 (2), 101-117.
- Péaire, C. y Sedano, T. (2014). *State-based monitoring and goal-driven project steering: field study of the SEMAT essence framework*. 36th International Conference on Software Engineering (ICSE), (págs. 325-334). Hyderabad.
- Peterson, N. et al. (1999). *An occupational information system for the 21st Century*. Washington D. C.: American Psychological Association.
- Piaget, J. (1952). *The origins of intelligence in children*. New York: International Universities Press.
- Piaget, J. (1970). *Psicología y Epistemología*. Buenos Aires: Emecé.
- Piattini, M. y Garzías, J. (2010). *Fábricas de software: experiencias, tecnologías y organización* (2ª edición ed.). Madrid: Ra-Ma.
- Prensky, M. (2007). *Digital Game-Based Learning*. Minnesota: Paragon House.
- Pressman, R. S. (2005). *Ingeniería de Software Un enfoque práctico* (6ª edición ed.). (J. E. Murrieta-Murrieta, E. Pineda-Rojas, & V. Campos-Olguín, Trans.) Ciudad de México: McGraw-Hill Interamericana.
- Ribeiro, C., Monteiro, M., Baalsrud, J., Pereira, J. y Antunes, T. (2016). *Sepsis Fast Track: A simulation game for Clinical education based on the Sepsis Fast Track protocol*. International Conference on Serious Games and Applications for Health (SeGAH) (págs. 1-8). Orlando: IEEE.
- Rjaibi, N., Rabai, B. A. y Limam, M. (2012). *Modeling the prediction of student's satisfaction in face-to-face learning: An empirical investigation*. International Conference on Education and e-Learning Innovations (págs. 1-6). Sousse: IEEE.

- Rocksteady Studios. (2009). *Batman: Arkham Asylum*. Recuperado el 3 de Febrero de 2018, de Batman: Arkham Asylum: <http://www.batmanarkhamasylum.com/start>
- Romero, J. A., García-Gaona, P. A. y Montenegro-Marín, C. (2016). Assessment model in a selection process based in gamification. *IEEE Latin America Transactions*, 14 (6), 2789-2794.
- Rumbaugh, J. (1990). *Object-Oriented Modeling and Design*. Upper Saddle River: Prentice Hall.
- Runeson, P., Host, M., Rainer, A. y Regnell, B. (2012). *Case study research in Software Engineering: Guidelines and examples*. Wiley.
- Sawyer, B. y Rejeski, D. (2002). *Serious Games: Improving Public Policy Through Game-based Learning and Simulation*. Washington, DC: Woodrow Wilson International Center for Scholars.
- Schwaber, K. y Beedle, M. (2001). *Agile Software Development With Scrum*. Englewood Cliff: Prentice Hall.
- Settles, B. y Meeder, B. (2016). *A Trainable Spaced Repetition Model for Language Learning*. 4th Annual Meeting of the Association for Computational Linguistics, (págs. 1848–1858). Berlin.
- Settles, B. y Meeder, B. (2016). *A Trainable Spaced Repetition Model for Language Learning*. 54th Annual Meeting of the Association for Computational Linguistics (págs. 1848–1858). Berlin: Association for Computational Linguistics.
- Shneiderman, B. (2003). *Human Needs and the New Computing Technologies*. MIT Press.
- Siemens, G. (2005). *Connectivism: learning as network-creation*. Astd Learning News.
- Simonette, M. J., Magalhães, M. E. y Spina, E. (2016). *PMBOK Five Process Groups and Essence Standard: Perfect Partners?* 4th International Conference in Software Engineering Research and Innovation (CONISOFT) (págs. 53-58). Puebla: IEEE.
- Smith, G. y Christensen, C. (1951). *Policy Formulation and Administration: A Casebook of Senior Management Problems in Business*. Richard d Irwin.
- Smith, M., Sun, W., Sutherland, J. y Mackie, B. (2014). Game Advertising: A Conceptual Framework and Exploration of Advertising Prevalence. *The Computer Games Journal*, 3 (1), 94-123.
- Sommerville, I. (2015). *Software Engineering* (10ª edición ed.). Londres: Pearson.
- Spector, J. M. (2016). *Foundations of educational technology: Integrative approaches and interdisciplinary perspectives* (2da. ed.). New York: Routledge.
- Take-Two Interactive Software, Inc. (2017). *Evolve*. Recuperado el 14 de Febrero de 2018, de Evolve game: <https://evolvegame.com>
- Tetris Holding. (1984). *About Tetris*. Recuperado el 24 de Enero de 2018, de Teris: <https://tetris.com/about-us>
- Unity Technologies. (2018). *Unity*. Recuperado el 12 de Enero de 2018, de Unity: <https://unity3d.com/es>

- Vigotsky, L. S. (1993). "Pensamiento y lenguaje". En L. S. Vigotsky, *Obras escogidas* (Vol. II). Madrid: Visor.
- Visch, V., Vejt, N., Anderiesen, H. y Van der Kooij, K. (2013). *Persuasive Game Design: A model and its definitions*. Conference on Changing Perspectives (págs. 107-110). París: ACM.
- Wang, Y., Rajan, P., Sankar, C. S. y Raju, P. K. (2017). Let Them Play: The Impact of Mechanics and Dynamics of a Serious Game on Student Perceptions of Learning Engagement. *IEEE Transactions on Learning Technologies*, 10 (4), 514-525.
- Warr, A. y Eamonn, O. (2015). *Understanding Design as a Social Creative Process. Conference on Creativity & Cognition United Kingdom* (págs. 118-127). Singapore: ACM.
- Wenger, E. (2002). *Cultivating Communities of Practice (Hardcover)*. Pennsylvania: Harvard Business Press.
- Wieringa, R. y Morali, A. (2012). "Technical Action Research as a Validation Method in Information Systems Design Science". En K. Peffers, R. M. y B. Kuechler (Edits.), *Design Science Research in Information Systems. Advances in Theory and Practice* (Vol. 7286, págs. 220-238). Heidelberg, Berlin: Lecture Notes in Computer Science.
- Wohlin, C. et al. (2012). *Experimentation in Software Engineering*. Heidelberg: Springer-Verlag.
- Wood-Harper, A. T. (1985). "Research methods in information systems: Using action research". En E. Mumford, R. Hirschheim, G. Fitzgerald y A. T. Wood-Harper (Edits.), *Research Methods in Information Systems*. Amsterdam: North-Holland.
- Yamabe, T. y Nakajima, T. (2013). Playful training with augmented reality games: case studies towards reality-oriented system design. *Multimedia Tools and Applications*, 62, 259-286.
- Yourdon, E. (1976). *Techniques of Program Structure and Design*. Upper Saddle River: Prentice Hall.
- Yusoff, A., Crowder, R., Gilbert, L. y Wills, G. (2009). *A Conceptual Framework for Serious Games*. International Conference on Advanced Learning Technologies (págs. 21-23). Riga: IEEE.
- Zapata-Jaramillo, C. M. y Montoya-Pérez, Y. (2016). *On the relationship of ISO/IEC 9126 metrics and the alpha states of the SEMAT kernel*. 4th International Conference in Software Engineering Research and Innovation (CONISOFT) (págs. 59-64). Puebla: IEEE.
- Zeng, X. et al. (2018). The construction and online/offline blended learning of small private online courses of principles of chemical engineering. *Computer Applications in Engineering Education*, 1519-1526.
- Zhang, H. y Ali Babar, M. (2013). Systematic reviews in software engineering: An empirical investigation. *Information and Software Technology*, 55 (7), 1341-1354.

Parte V
Apéndices

Apéndice A - Alfas de Essence

 **Oportunidad**

Identificada

- Se ha identificado una idea para mejorar la forma actual de hacer el trabajo, aumentar la presencia en el mercado o aplicar un sistema de software nuevo o innovador.
- Al menos uno de los involucrados desea invertir en comprender mejor la oportunidad y el valor asociado al abordarla.
- Los otros involucrados que comparten la oportunidad han sido identificados.

1/6

 **Oportunidad**

Solución necesaria

- Se han identificado los involucrados en la oportunidad y la solución.
- Se han establecido las necesidades de los involucrados que generan la oportunidad.
- Se han identificado los problemas subyacentes y las raíces de las causas.
- Se ha confirmado que se necesita una solución basada en software.
- Al menos se ha propuesto una solución basada en software.

2/6

 **Oportunidad**

Valor establecido

- Se ha cuantificado el valor de abordar la oportunidad en términos absolutos o en retornos o en ahorros por periodos de tiempo (por ejemplo, por año).
- Se entiende el impacto de la solución en los involucrados.
- Se entiende el valor que el sistema de software ofrece a los involucrados que financiarán y usarán el sistema.
- Son claros los criterios de éxito con los cuales serán evaluados los despliegues del sistema de software.
- Los resultados deseados tienen una solución clara y cuantificable.

3/6

 **Oportunidad**

Viable

- Se ha descrito una solución.
- Los indicios apuntan a que la solución puede ser desarrollada y desplegada dentro de las limitaciones.
- Los riesgos asociados con la solución son aceptables y manejables.
- El indicativo (estimado) de los costos de solución es menor que el valor esperado de la oportunidad.
- Las razones para desarrollar una solución basada en software son entendidas por todos los miembros del equipo.
- Está claro que la búsqueda de la oportunidad es viable.

4/6



Oportunidad

Abordable

- Está disponible un sistema usable que demuestra que aborda la oportunidad.
- Los involucrados están de acuerdo en que la solución disponible es digna de desplegarse.
- Los involucrados están convencidos de que la solución aborda la oportunidad.

5/6



Oportunidad

Beneficio percibido

- La solución ha comenzado a mostrar beneficios a los involucrados.
- El perfil de retorno de la inversión es por lo menos tan buena como se esperaba.

6/6



Involucrados

Reconocidos

- Se identificaron todos los grupos diferentes de involucrados que son, o serán afectados por el desarrollo y operación del sistema de software.
- Hay un acuerdo sobre los grupos de involucrados que serán representados. Como mínimo deben considerarse los grupos de involucrados que financian, usan, dan soporte y mantienen el sistema.
- Se han definido las responsabilidades de los representantes de los involucrados.

1/6



Involucrados

Representados

- Los representantes de los involucrados han acordado asumir sus responsabilidades.
- Los representantes de los involucrados están autorizados para llevar a cabo sus responsabilidades.
- Se ha acordado el esquema de colaboración entre los representantes de los involucrados.
- Los representantes de los involucrados apoyan y respetan la forma de trabajo del equipo.

2/6



Involucrados

Implicados

- Los representantes de los involucrados ayudan al equipo de acuerdo con sus responsabilidades.
- Los representantes de los involucrados proporcionan información y participan oportunamente en la toma de decisiones.
- Los representantes de los involucrados comunican con prontitud los cambios que sean relevantes para sus grupos de involucrados.

3/6



Involucrados

De acuerdo

- Los representantes de los involucrados han acordado cuáles son las expectativas mínimas para el próximo despliegue de sistema.
- Los representantes de los involucrados están satisfechos con su participación en el trabajo.
- Los representantes de los involucrados coinciden en que su aporte es valorado por el equipo y que son tratados con respeto.
- Los miembros del equipo están de acuerdo en que su aporte es valorado por los representantes de los involucrados y que son tratados con respeto.
- Los representantes de los involucrados están de acuerdo con la forma en que se equilibran sus diferentes prioridades y perspectivas, para proporcionar una dirección clara para el equipo.

4/6



Involucrados

Satisfechos para despliegue

- Los representantes de los involucrados proporcionan retroalimentación sobre el sistema desde la perspectiva de sus grupos de involucrados.
- Los representantes de los involucrados confirman que están de acuerdo en que el sistema está listo para su despliegue.

5/6



Involucrados

Satisfechos en uso

- Los involucrados están utilizando el nuevo sistema y proporcionan retroalimentación sobre sus experiencias.
- Los involucrados confirman que el nuevo sistema cumple con sus expectativas.

6/6

OX Requerimientos

Concebidos

- El conjunto inicial de involucrados están de acuerdo en que se producirá un nuevo sistema.
- Están identificados los involucrados que usarán el nuevo sistema.
- Están identificados los involucrados que financiarán el nuevo sistema.
- Hay una oportunidad clara que puede ser abordada por medio del nuevo sistema.

1/6

OX Requerimientos

Acotados

- Están identificados los involucrados implicados en el desarrollo del nuevo sistema.
- Los involucrados están de acuerdo en el propósito del nuevo sistema.
- Está claro qué es el éxito para el nuevo sistema.
- Los involucrados tienen un entendimiento compartido de la magnitud de la solución propuesta.
- Se ha acordado la forma en la que se describirán los requerimientos.
- Se han establecido los mecanismos para la administración de los requerimientos.
- Es claro el esquema de priorización.
- Las restricciones están identificadas y consideradas.
- Los supuestos están claramente establecidos.

2/6

OX Requerimientos

Coherentes

- Los requerimientos están capturados y compartidos con el equipo y los involucrados.
- El origen de los requerimientos es claro.
- El fundamento de los requerimientos es claro.
- Los requerimientos conflictivos están identificados y atendidos.
- Los requerimientos comunican las características esenciales del sistema a ser entregado.
- Los escenarios de uso más importantes para el sistema se pueden explicar.
- La prioridad de los requerimientos es clara.
- Se entiende el impacto de la implementación de los requerimientos.
- El equipo entiende lo que tiene que ser entregado y se compromete a entregarlo.

3/6

OX Requerimientos

Aceptables

- Los involucrados aprueban que los requerimientos describen una solución aceptable.
- La tasa de cambio de los requerimientos aceptados es relativamente pequeña y está bajo control.
- Es claro el valor proporcionado con la implementación de los requerimientos.
- Son claras las partes de la oportunidad satisfechas con los requerimientos.
- Los requerimientos son comprobables.

4/6

Requerimientos

Abordados

- Se han abordado suficientes requerimientos para que el sistema resultante sea aceptable para los involucrados.
- Los involucrados aceptan que los requerimientos son un reflejo exacto de lo que el sistema hace o no hace.
- El conjunto de los requerimientos implementados proporciona un valor claro a los involucrados.
- El sistema que implementa los requerimientos es aceptado por los involucrados, así que vale la pena ponerlo en operación.

5/6

Requerimientos

Cumplidos

- Los involucrados aceptan los requerimientos como una captura exacta de lo que ellos requieren para cumplir satisfactoriamente con la necesidad de un nuevo sistema.
- No hay elementos de requerimientos pendientes que impidan que el sistema sea aceptado para satisfacer totalmente los requerimientos.
- El sistema es aceptado por los involucrados, porque satisface los requerimientos plenamente.

6/6

Sistema de software

Arquitectura seleccionada

- Se han acordado los criterios que serán usados para elegir una arquitectura.
- Se han identificado las plataformas de hardware.
- Se han elegido los lenguajes de programación y tecnologías a usar.
- Se conocen los límites del sistema.
- Se han hecho decisiones significativas acerca de la organización del sistema.
- Se han realizado decisiones de compra, construcción y reutilización.
- Se han aceptado los riesgos técnicos clave.

1/6

Sistema de software

Demostrable

- Se han demostrado las características clave de la arquitectura.
- El sistema puede ser ejercitado y su rendimiento puede medirse.
- Se han demostrado las configuraciones críticas del hardware.
- Se han demostrado las interfaces críticas.
- Se ha demostrado la integración con otros sistemas existentes.
- Los involucrados más relevantes están de acuerdo en que la arquitectura demostrada es apropiada.

2/6

Sistema de software

Utilizable

- El sistema puede ser operado por los involucrados que lo usan.
- La funcionalidad proporcionada por el sistema ha sido probada.
- El desempeño del sistema es aceptable para los involucrados.
- El nivel de los defectos es aceptable para los involucrados.
- El sistema está completamente documentado.
- Se conoce el contenido liberado.
- El valor agregado que aporta el sistema es claro.

3/6

Sistema de software

Listo

- Está disponible la documentación de la instalación y de usuario.
- Los representantes de los involucrados aceptan que el sistema cumple su propósito.
- Los representantes de los involucrados quieren poner el sistema en operación.
- El soporte operativo está listo.

4/6

Sistema de software

Operacional

- El sistema ha sido puesto a disposición de los involucrados para que intenten usarlo.
- Por lo menos un ejemplo del sistema es completamente operacional.
- El sistema es totalmente compatible con los niveles de servicio acordados.

5/6

Sistema de software

Retirado

- El sistema ha sido reemplazado o descontinuado.
- El sistema ya no tiene soporte.
- No hay ningún involucrado "oficial" que aún use el sistema.
- No se producirán más actualizaciones al sistema.

6/6



Equipo

Sembrado

- Se ha definido la misión del equipo en términos de las oportunidades y los resultados.
- Se conocen las restricciones del funcionamiento del equipo.
- Se han establecido los mecanismos para hacer crecer al equipo.
- Está definida la composición del equipo.
- Se ha definido cualquier restricción relacionada con el dónde y cómo se llevará a cabo el trabajo.
- Las responsabilidades del equipo están descritas.
- Es claro el nivel de compromiso.
- Están definidas las competencias requeridas.
- Está definido el tamaño del equipo.
- Están definidas las reglas de gobernanza.

1/5



Equipo

Formado

- Se entienden las responsabilidades individuales.
- Suficientes miembros del equipo se han reclutado para habilitar el progreso del trabajo.
- Todos los miembros entienden cómo está organizado el equipo y cuáles son sus roles.
- Todos los miembros del equipo entienden cómo llevar a cabo su trabajo.
- Los miembros del equipo se han reunido (tal vez virtualmente) y están empezando a conocerse unos a otros.
- Los miembros del equipo entienden sus responsabilidades y cómo se alinean con sus competencias.
- Los miembros del equipo aceptan el trabajo.
- Se ha identificado cualquier colaborador externo (organizaciones, equipos e individuos).

2/5



Equipo

Colaborando

- El equipo está trabajando como una unidad cohesiva.
- La comunicación dentro del equipo es abierta y honesta.
- El equipo está enfocado en alcanzar la misión.
- Los miembros del equipo se conocen unos a otros.

3/5



Equipo

Ejecutando

- El equipo cumple constantemente sus compromisos.
- El equipo continuamente se adapta a los cambios de contexto.
- El equipo identifica y aborda los problemas sin ayuda externa.
- Se está logrando un progreso efectivo con un retroceso mínimo y poca reelaboración.
- El trabajo perdido y la posibilidad de hacer trabajo desperdiciado, son eliminados continuamente.

4/5

 **Equipo**

Retirado

- Las responsabilidades del equipo han sido entregadas o cumplidas.
- Los miembros del equipo están disponibles para ser asignados a otros equipos.
- Los miembros del equipo dejaron de hacer esfuerzos para completar la misión.

5/5

 **Trabajo**

Iniciado

- El resultado del trabajo iniciado es claro.
- Cualquier restricción sobre el desempeño del trabajo ha sido claramente identificada.
- Se conoce a los involucrados que financiarán el trabajo.
- Se ha identificado claramente el iniciador del trabajo.
- Se conoce a los involucrados que aceptarán los resultados.
- Es clara la fuente de financiamiento.
- Las prioridades del trabajo son claras.

1/6

 **Trabajo**

Preparado

- El compromiso está hecho.
- Se han estimado los costos y esfuerzos del trabajo.
- Se entiende la disponibilidad de recursos.
- Son claras las políticas de gobernanza y procedimientos.
- Se entiende la exposición a riesgos.
- Están definidos los criterios de aceptación y el cliente está de acuerdo con ellos.
- Se ha dividido el trabajo lo suficiente para comenzar un trabajo productivo.
- Se han identificado y priorizado las tareas por el equipo y los involucrados.
- Un plan de trabajo creíble está disponible.
- La financiación está disponible para iniciar el trabajo.
- El equipo o al menos algunos miembros del equipo están listos para comenzar el trabajo.
- Se han definido los puntos de integración y entrega.

2/6

 **Trabajo**

Comenzado

- Se ha iniciado el trabajo de desarrollo.
- El progreso del trabajo se está monitoreando.
- Se está dividiendo el trabajo en elementos de trabajo accionables con claras definiciones de criterios de terminación.
- Los miembros están aceptando y progresando tareas.

3/6



Trabajo

Bajo control

- Se están completando tareas.
- Trabajos no planeados, están bajo control.
- Los riesgos y el impacto que causarían (si llegasen a ocurrir) están bajo control, y la probabilidad de que se produzcan se ha reducido a niveles aceptables.
- Se han revisado las estimaciones para reflexionar sobre el desempeño del equipo.
- Están disponibles las mediciones para mostrar el progreso y la velocidad.
- El re-trabajo está bajo control.
- Las tareas son consistentemente completadas en tiempo y dentro de sus estimaciones.

4/6



Trabajo

Concluido

- Las tareas pendientes son de tipo administrativo o relacionadas con la preparación de la siguiente pieza de trabajo.
- Se han obtenido los resultados del trabajo.
- Los involucrados han aceptado el sistema de software resultante.

5/6



Trabajo

Cerrado

- Las lecciones aprendidas se han desglosado, registrado y discutido.
- Se han hecho disponibles las métricas.
- Todo ha sido archivado.
- Se ha conciliado y cerrado el presupuesto.
- El equipo ha sido liberado.
- No hay tareas pendientes, sin terminar.

6/6



Forma de trabajar

Principios establecidos

- Los principios y limitaciones son estipulados por el equipo.
- Están acordados los principios y limitaciones con los involucrados.
- Los involucrados están de acuerdo con las herramientas necesarias para el trabajo.
- Se dispone de una recomendación para el planteamiento que debe seguirse.
- Se entiende el contexto dentro del cual operará el equipo.
- Se conocen las restricciones que deben considerarse para seleccionar, adquirir y usar prácticas y herramientas.

1/6

Forma de trabajar

Base establecida

- Se han seleccionado las prácticas claves y herramientas que formarán la base de la forma de trabajar.
- Se han acordado suficientes prácticas en el equipo para comenzar el trabajo.
- Se han identificado todas las prácticas y herramientas que no son negociables.
- Se han analizado y entendido las brechas que existen entre las prácticas y herramientas necesarias y las que se disponen.
- Se han analizado y entendido las brechas que existen entre lo que se necesita ejecutar de la forma de trabajar y los niveles de capacidad del equipo.
- Las prácticas y herramientas seleccionadas se han integrado para construir una forma de trabajar utilizable.

2/6

Forma de trabajar

En uso

- Se están usando las prácticas y herramientas para hacer el trabajo real.
- Se inspecciona regularmente el uso de las prácticas y herramientas seleccionadas.
- Se están adaptando las prácticas y herramientas al contexto del equipo.
- El uso de las prácticas y herramientas es apoyado por el equipo.
- Están en uso los procedimientos para manejar la retroalimentación de la forma de trabajar del equipo.
- Las prácticas y herramientas apoyan la comunicación y colaboración.

3/6

Forma de trabajar

En marcha

- Las prácticas y herramientas están siendo utilizadas por todo el equipo para realizar su trabajo.
- Todos los miembros del equipo tienen acceso a las prácticas y las herramientas necesarias para hacer su trabajo.
- Todo el equipo está involucrado en la inspección y adaptación de la forma de trabajar.

4/6

Forma de trabajar

Trabajando bien

- Los miembros del equipo están generando avances según lo planeado mediante el uso y adaptación de la forma de trabajar para ajustarla a su contexto actual.
- El equipo aplica naturalmente las prácticas sin tener que pensar en ellas.
- Las herramientas soportan naturalmente la forma en la cual el equipo trabaja.
- El equipo continuamente refina el uso de las prácticas y herramientas.

5/6

Forma de trabajar

Retirada

- Ya no se usa la forma de trabajar del equipo.
- Se han compartido las lecciones aprendidas para su uso futuro.

6/6

Apéndice B - Formulario de Extracción de Información del Mapeo Sistemático de Estudios

Información del estudio	Descripción
1. Identificador del estudio	Identificador único de estudio
2. Fecha de extracción	Mes y año en el que se consultó el estudio
3. Referencia bibliográfica	Autor, Título, Revista/Congreso, Volumen, Número, Año y Páginas.
4. Objetivo del estudio	Objetivo(s) del estudio
Sub-Preguntas	Posibles Respuestas
1. ¿Qué proceso del ciclo de vida de software se aprende con el juego serio?	a. Procesos del ciclo de vida del sistema a.1. Procesos de acuerdo a.2. Procesos del proyecto a.3. Procesos técnicos a.4. Procesos organizativos de habilitación de proyectos b. Procesos específicos del software b.1. Procesos de implementación del software b.2. Procesos de soporte del software b.3. Procesos de reutilización de software
2. ¿Cuál es el tipo de juego serio?	a. Edutainment b. Juego de entrenamiento c. Simulador d. Juego persuasivo e. Juego organizativo-dinámico
3. ¿Cuál es la audiencia del juego serios?	a. Profesionales principiantes b. Profesionales expertos c. Estudiantes
4. ¿Cuál es el esquema de	a. Colaborativo

- interacción del juego serio? b. Competitivo
c. Individualista
5. ¿Qué medios/materiales emplea el juego serio? a. Virtual
b. Material
c. Actuación
d. Combinación de virtual y material
e. Combinación de virtual y actuación
f. Combinación de actuación y material
6. ¿Cuál es el objetivo del juego serio? a. Educar
b. Modificar el comportamiento
7. ¿Qué estilo de aprendizaje emplea el juego serio? a. Visual
b. Auditivo
c. Lectura/Escritura
d. Kinestésico
e. Combinación de Lectura/Escritura y Kinestésico
f. Combinación de Lectura/Escritura y Visual
g. Combinación de Visual y Kinestésico
8. ¿Cuál es el nivel de aprendizaje objetivo del juego serio? a. Recordar
b. Comprender
c. Aplicar
d. Analizar
e. Evaluar
f. Crear
9. ¿Cuál es el resultado que se obtiene con el juego serio? a. Conocimiento
b. Destrezas
c. Habilidades
d. Otras características (personalidad, interés, preferencias).
e. Combinación de conocimiento y habilidades
-

Apéndice C – Lista de Estudios Primarios del Mapeo Sistemático de Estudios

ID	Estudio
[S01]	Wendel,V., Gutjahr, M., Göbel, S. and Steinmetz, R. 2013. “Designing collaborative multiplayer serious games. Escape from Wilson Island - A multiplayer 3D serious game for collaborative learning in teams”. <i>Education and Information Technology</i> , Vol. 18, pp. 287-308. [SP].
[S02]	Couceiro, R., Papastergiou, M., Kordaki, M. and Veloso, A. 2013. “Design and evaluation of a computer game for the learning of Information and Communication Technologies (ICT)”. <i>Education and Information Technology</i> , Vol. 18, pp. 531-554. [SP].
[S03]	Martin-Dorta, N., Sanchez-Berriel, I., Bravo, M., Hernandez, J., Saorin, J. and Contero, M. 2013. “Virtual Blocks: a serious game for spatial ability improvement on mobile devices”. <i>Multimedia Tools and Application</i> . DOI: 10.1007/s11042-013-1652-0. [SP].
[S04]	Ruiz, I. and Gómez-Nieto, M. 2012. “Rolling: A new technique for the practical teaching in computer science university degree”. <i>Education and Information Technology</i> , Vol. 17, pp. 49-77. [SP].
[S05]	Milne, I. and Rowe, G. 2004. “OGRE: Three-Dimensional Program Visualization for Novice Programmers”. <i>Education and Information Technology</i> , Vol. 9, pp. 219-237. [SP].
[S06]	Atladdottir, G., Hvanberg, E. and Gunnarsdottir, S. 2012. “Comparing task practicing and prototype fidelities when applying scenario acting to elicit requirements”. <i>Requirements Engineering</i> , Vol. 17, pp. 157-170. [SP].
[S07]	Pfahl, D., Laitenberger, O., Dorsch, J. and Ruhe, G. 2003. “An Externally Replicated Experiment for Evaluating the Learning Effectiveness of Using Simulations in Software Project Management Education”. <i>Empirical Software Engineering</i> , Vol. 8, pp. 367-395. [SP].
[S08]	Gresse, C., Thiry, M. and Kochanski, D. 2009. “Empirical evaluation of an educational game on software measurement”. <i>Empirical Software Engineering</i> , Vol. 14, pp. 418-452. [SP].
[S09]	Allal-Chérif, O., Bidan, M. and Makhoulf, M. 2016. “Using serious games to manage knowledge and competencies: The seven-step development process”. <i>Empirical Software Engineering</i> , pp. 1-11. [SP].
[S10]	Domínguez, A., Saenz-de-Navarrete, J., de-Marcos, L., Fernández-Sanz, L., Pagés, C. and Martínez-Herráiz, J. 2013. “Gamifying learning experiences: Practical implications and outcomes”. <i>Computers & Education</i> , Vol. 63, pp. 380-392. [SD].

- [S11] Sancho-Thomas, P., Fuentes-Fernández, R. and Fernández-Manjón, B. 2009. "Learning teamwork skills in university programming courses". *Computers & Education*, Vol. 53, pp. 517-531. [SD].
- [S12] Liu, C., Cheng, Y. and Huang, C. 2011. "The effect of simulation games on the learning of computational problem solving". *Computers & Education*, Vol. 57, pp. 1907-1918. [SD].
- [S13] Fernandes, J., Duarte, D., Ribeiro, C., Farinha, C., Madeiras, J. and da Silva, M. 2012. "iThink: A game-based approach towards improving collaboration and participation in requirement elicitation". *Procedia Computer Science*, Vol. 15, pp.66-77. [SD].
- [S14] Gresse, C. Savi, R. and Ferreti, A. 2012. "DELIVER! - An educational game for teaching Earned Value Management in computing courses". *Information and Software Technology*, Vol. 54, pp. 286-298. [SD].
- [S15] Kazimoglu, C., Kiernan, M., Bacon, L. and MacKinnon, L. 2012. "Learning Programming at the Computational Thinking Level via Digital Game-Play". *Procedia Computer Science*, Vol. 9, pp.522-531. [SD].
- [S16] Baker, A., Navarro, E. and Van der Hoek, A. 2005. "An experimental card game for teaching software engineering processes". *The Journal of Systems and Software*, Vol. 75, pp.3-16. [SD].
- [S17] Dubinsky, Y. and Hazzan, O. 2006. "Using a role scheme to derive software project metrics". *Journal of Systems Architecture*, Vol. 52, pp. 693-699. [SD].
- [S18] Drappa, A. and Ludewig, J. 2000. "Quantitative modeling for the interactive simulation of software projects". *The Journal of Systems and Software*, Vol. 46, pp.113-122. [SD].
- [S19] Hainey, T., Connolly, T., Stansfield, M. and Boyle, E. 2010. "Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level". *Computer and Education*, Vol. 56, pp. 21-35. [SD].
- [S20] Ghanbari, H., Similä, J. and Markkula, J. 2015. "Utilizing online serious games to facilitate distributed requirements elicitation". *Journal of Systems and Software*, Vol. 109, pp. 32-49. [SD].
- [S21] Díaz, R., Fernández, A., Pazos, J. and Gil, A. 2012. "Collaborative and Role-Play Strategies in Software Engineering Learning With Web 2.0 Tools". *Computer Applications in Engineering Education*, Vol. 22, pp. 658-668. [WILEY].
- [S22] García, I. and Pacheco, C. 2012. "Using TSPi and PBL to Support Software Engineering Education in an Upper-Level Undergraduate Course". *Computer Applications in Engineering Education*, Vol. 22, pp. 736-749. [WILEY].
- [S23] de O. Barros, M., Dantas, A., Veronese, G. and Werner, C. 2006. "Model-driven game development: experience and model enhancements in software project management education". *Software Process: improvement and Practice*, Vol. 11, pp. 411-421. [WILEY].
- [S24] Wassila, D. and Tahar, B. 2012. "Using serious game to simplify algorithm

- learning”. Proc. of the International Conference on Education and e-Learning Innovations, (ICEELI ‘12), pp. 1-5. [IEEE].
- [S25] Labrador, E., García, O., Cubeles, A. and Fox, P. 2013. “Playing Entrepreneurship as a Memorable Experience for All Sorts of Talented Warriors”. Proc. of the 5th. International Conference on Games and Virtual Worlds for Serious Applications, (VS-GAMES ‘13), pp. 1-2. [IEEE].
- [S26] Mitamura, T., Suzuki, Y. and Oohori, T. 2012. “Serious games for learning programming languages”. Proc. of the International Conference on Systems, Man and Cybernetics, (ICSMC ‘12), pp. 1812-1817. [IEEE].
- [S27] Paliokas, I., Arapidis, C. and Mpimpitsos, M. 2011. “PlayLOGO 3D: A 3D Interactive Video Game for Early Programming Education: Let LOGO Be a Game”. Proc. of the 3rd. International Conference on Games and Virtual Worlds for Serious Applications, (VS-GAMES ‘11), pp. 24-31. [IEEE].
- [S28] Nerbraten, O. and Røstad, L. 2009. “hACMEgame: A Tool for Teaching Software Security”. Proc. of the International Conference on Availability, Reliability and Security, (ARES ‘09), pp. 811-816. [IEEE].
- [S29] Polytechnic, T. and Polytechnic, T. 2010. “Notice of Retraction Game-based Learning for Data Structures: A case study”. Proc. of the 2nd International Conference on Computer Engineering and Technology, (ICCET ‘10), pp. 718-721. [IEEE].
- [S30] Ab Hamid, S. and Yu, L. 2007, “Learn Programming by Using Mobile Edutainment Game Approach”. Proc. of the First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning, (DIGITEL ‘07), pp. 170-172. [IEEE].
- [S31] Tillmann, N., de Halleux, J. and Xie, T. 2011. “Pex4Fun: Teaching and learning computer science via social gaming”. Proc. of the 24th Conference on Software Engineering Education and Training, (CSEE&T ‘11), pp. 546-548. [IEEE].
- [S32] Pebriadi, P., Wuryandari, A. and Setijadi, A. 2013. “Design and implementation learning methods HTML5 on iCode application”. Proc. of the Joint International Conference on Rural Information & Communication Technology and Electric-Vehicle Technology, (rICT & ICeV-T ‘13), pp. 1-6. [IEEE].
- [S33] Masso, N. and Grace, L. 2011. “Shapemaker: A game-based introduction to programming”. Proc. of the 16th International Conference on Computer Games, (CGAMES 2011), pp. 168-171. [IEEE].
- [S34] Adamo-Villani, N., Haley-Hermiz, T. and Cutler, R. 2013. “Using a Serious Game Approach to Teach 'Operator Precedence' to Introductory Programming Students”. Proc. of the 17th International Conference on Information Visualization, (IV ‘13), pp. 523-526 [IEEE].
- [S35] Karapinar, A. and Zavrak, S. 2012. “Binary apple tree: A game approach to tree traversal algorithms”. Proc. of the International Conference on Information Technology Based Higher Education and Training, (ITHET ‘12), pp. 1-3. [IEEE].

- [S36] Rusu, A. Russell, R. Cocco, R. and DiNicolantonio, S. 2011. "Introducing Object Oriented Design Patterns through a Puzzle-Based Serious Computer Game". Proc. of the 41st ASEE/IEEE Frontiers in Education Conference, (FIE '11), pp. F1H-1 F1H-6. [IEEE].
- [S37] Chen, W., Wu, W., Wang, T. and Su, C. 2008. "Work in Progress-A Card-based Learning System for Software Engineering Education". Proc. of the 38th ASEE/IEEE Frontiers in Education Conference, (FIE '08), pp. T2A-12-T2A-13. [IEEE].
- [S38] Chen, W. 2010. "Work in Progress-An Investigation of Varied Game-Based Learning Systems in Engineering Education". Proc. of the 40th ASEE/IEEE Frontiers in Education Conference, (FIE '10), pp. S1G-1-S1G-2. [IEEE].
- [S39] Zuppiroli, S., Ciancarini, P. and Gabbrielli, M. 2012. "A role-play game for a software engineering lab: developing a product line". Proc. of the 25th IEEE Conference on Software Engineering Education and Training, (CSEE&T' 12), pp.13-22. [IEEE].
- [S40] Fernandes, J. and Sousa, S. 2010. "PlayScrum- A Card Game to Learn the Scrum Agile Method". Proc. of the 2nd. International Conference on Games and Virtual Worlds for Serious Applications, (VS-GAMES '10), pp. 52-59. [IEEE].
- [S41] Aaen, I., Vestergaard, J., Jensen, R. and Saxov, M. 2008. "Reviews in innovative software development-The Gauntlet game". Proc. of the International Conference on Computational Intelligence for Modeling Control & Automation, (CIMCA '08), pp. 1234-1239. [IEEE].
- [S42] Jain, A. and Boehm, B. 2006. "SimVBSE: Developing a Game for Value-Based Software Engineering". Proc. of the 19th Conference on Software Engineering Education and Training, (CSEET '06), pp. 103-114. [IEEE].
- [S43] Wang, A., Øfsdahl, T. and Mørch-Storstein, O. 2008. "An Evaluation of a Mobile Game Concept for Lectures". Proc. of the 21st Conference on Software Engineering Education and Training, (CSEET '08), pp. 197-204. [IEEE].
- [S44] Wang, T. and Zhu, Q. 2009. "A Software Engineering Education Game in a 3-D Online Virtual Environment". First International Workshop on Technology and Computer Science, (ETCS '09), pp. 708-710. [IEEE].
- [S45] Rusu, A., Russell, R. and Robinson, J. 2010. "Learning software engineering basic concepts using a five-phase game". Proc. of the Conference on Frontiers in Education, (FIE '10), pp. S2D1-S2D6. [IEEE].
- [S46] Phelps, A., Egert, C. and Bierre, K. 2005. "MUPPETS: Multi-User Programming Pedagogy for Enhancing Traditional Study: An Environment for both Upper and Lower Division Students". Proc. of the 35th Annual Conference on Frontiers in Education, (FIE '05), pp. S2H-8. [IEEE].
- [S47] Chau, B., Robinson, A., Pace, J., Nash, R. and Sung, K. 2014. "Corrupted: A game to teach programming concepts". Computer, Vol. 47, pp. 100-103. [IEEE].
- [S48] Nunes-Lino, J.E., Paludo, M., Vinicius, F., Reinehr, S. and Malucelli, A. 2015.

- “Project management game 2D (PMG-2D): A serious game to assist software project managers training”. Proc. of the Conference on Frontiers in Education Conference, (FIE '14), pp.1-8. [IEEE].
- [S49] Oliviera-Chaves, R., von Wangenheim, C., Costa-Furtado, J., Bezerra-Oliviera, S., Santos, A. and Favero, E. 2015. “Experimental Evaluation of a Serious Game for Teaching Software Process Modeling”. Transactions on education, Vol. 59, pp. 289-296. [IEEE].
- [S50] Ramírez-Rosales, S., Vázquez-Reyes, S., Villa-Cisneros, J. and De León-Sigg, M. 2016. “A Serious Game to Promote Object Oriented Programming and Software Engineering Basic Concepts Learning”. Proc. of the 4th International Conference in Software Engineering Research and Innovation, (CONISOFT '16), pp. 97-103. [IEEE].
- [S51] Andrews, J. 2013. “Killer App: A Eurogame about Software Quality”. Proc. of the 26th International Conference on Software Engineering Education and Training, (CSEE&T '13), pp. 319-323. [IEEE].
- [S52] Rong, G., Zhang, H. and Shao, D. 2013. “Applying competitive bidding games in software process education”. Proc. of the 26th International Conference on Software Engineering Education and Training, (CSEE&T '13), pp. 129-138. [IEEE].
- [S53] Scharlau, B. 2013. “Games for teaching software development”. Proc. of the 16th International Conference on Computer Games, (CGAMES '13), pp. 1-7. [IEEE].
- [S54] Rusu, A., Russell, R. and Cocco, R. 2011. “Simulating the software engineering interview process using a decision-based serious computer game”. Proc. of the 16th International Conference on Computer Games, (CGAMES 2011), pp. 235-239. [SC].
- [S55] Caulfield, C., Veal, D. and Maj, S. 2011. “Teaching software engineering project management-A novel approach for software engineering programs”. Modern Applied Science, Vol. 5, No. 5, pp. 87-104. [SC].
- [S56] Knauss, E., Schneider, K. and Stapel, K. 2008. “A game for taking requirements engineering more seriously”. Third International Workshop on Multimedia and Enjoyable Requirements Engineering, (MERE '08), pp. 22-26. [SC].
- [S57] Zapata, C. 2009. “Teaching Software Development by means of a classroom game: the software development game”. Developments in Business Simulation and Experiential Learning, Vol. 36, pp. 156-164. [SC].
- [S58] Zapata, C. and Awad-Aubad, G. 2007. “Requirements Game: Teaching Software Project Management”. Clei Electronic Journal, Vol. 10, No.1, Paper 3. [SC].
- [S59] Oh Navarro, E. and van der Hoek, A. 2009. “Multi-site evaluation of SimSE”. Proc. of the 40th ACM Technical Aymposium on Computer Science Education, (SIGCSE '09), pp. 326-330. [SC].
- [S60] Ernst, M. 2005. “The groupthink specification exercise”. Proc. of the 27th International Conference on Software Engineering, (ICSE '05), pp. 617-618.

- [SC].
- [S61] Chung, C., Matsuoka, A., Yang, Y., Rubi, J. and Chechik, M. 2016. "Serious Games for NP-hard Problems: Challenges and Insights". Proc. of the 5th International Workshop on Games and Software Engineering, (GAS '16), pp. 29-32. [SC].
- [S62] Miljanovic, M. and Bradbury, J. 2016. "Robot ON!: A Serious Game for Improving Programming Comprehension". Proc. of the 5th International Workshop on Games and Software Engineering, (GAS '16), pp. 33-36. [SC].
- [S63] Letra, P., Ramada-Paiva, A. and Flores, N. 2015. "Game design techniques for software engineering management education". Proc. of the 18th International Conference on Computational Science and Engineering, (CSE '15), pp. 192-199. [IEEE].
- [S64] Eagle, M. and Barnes, T. 2009. "Experimental evaluation of an educational game for improved learning in introductory computing". Proc. of the 40th ACM Technical Symposium on Computer Science Education, (SIGCSE '09), pp. 321-325. [ACM].
- [S65] O'Kelly, J. and Gibson, P. 2006. "RoboCode and problem-based learning: a non-prescriptive approach to teaching programming". Proc. of the 11th Annual Conference on Innovation and technology in Computer Science Education, (SIGCSE '06), pp. 217-221. [ACM].
- [S66] Lee, M. and Ko, A. 2011. "Personifying programming tool feedback improves novice programmers' learning". Proc. of the 7th International Workshop on Computing Education Research, (ICER '11), pp. 109-116. [ACM].
- [S67] Shaw, K. and Dermoudy, J. 2005. "Engendering an Empathy for Software Engineering". Proc. of the 7th Australasian Conference on Computing Education, (ACE '05), Vol. 42, pp. 135-144. [ACM].
- [S68] Sheth, S., Bell, J. and Kaiser, G. 2011. "HALO (Highly Addictive, socialLly Optimized) Software Engineering". Proc. of the 1st International Workshop on Games and Software Engineering (GAS' 11), pp. 29-32. [ACM].
- [S69] Henry, T. and LaFrance J. 2006. "Integrating Role-Play into Software Engineering Courses". Journal of Computing Sciences in Colleges, Vol.2, No. 2, pp. 32-38. [ACM].
- [S70] Tillmann, N., de Halleux, J., Xie, T. and Bishop, J. 2014. "Code Hunt: Gamifying Teaching and Learning of Computer Science at Scale". Proc. of the 1st Learning @ scale conference, (L@S '14), pp. 221-22. [ACM].
- [S71] Heikkilä, V., Paasivaara, M. and Lassenius, C. 2016. "Teaching university students Kanban with a collaborative board game". Proc. of the 38th International Conference on Software Engineering Companion, (ICSE '16), pp. 471-480. [ACM].
-

Apéndice D – Mapeo de Estudios Primarios

ID	Sub-Preguntas									Evaluación de la calidad					Calidad promedio	Porcentaje de calidad
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	a	b	c	d	e		
S01	a.4	e	a	a	a	b	f	d	c	3	3	3	3	3	3	100%
S02	b.1	a	c	c	a	a	a	a	a	3	3	3	3	3	3	100%
S03	b.1	d	c	c	a	b	a	b	c	3	3	3	3	3	3	100%
S04	b.2	e	c	a	c	b	d	c	e	3	3	3	3	3	3	100%
S05	b.1	a	c	c	a	a	a	b	a	3	3	3	3	2	2.8	93.3%
S06	a.3	c	a	c	e	a	d	d	e	2	2	3	3	3	2.6	86.7%
S07	b.2	c	c	c	c	a	d	b	a	3	3	3	3	3	3	100%
S08	a.2	a	c	c	a	a	c	b	a	3	3	3	3	3	3	100%
S09	a.4	c	a	b	e	b	f	c	e	3	3	3	3	2	2.8	93.3%
S10	b.1	a	c	b	a	a	c	a	a	2	3	3	3	3	2.8	93.3%
S11	a.4	d	c	b	a	b	f	b	c	3	3	3	3	3	3	100%
S12	b.1	d	c	c	a	b	f	a	c	3	3	3	3	3	3	100%
S13	a.3	b	b	a	a	a	f	d	e	3	3	3	3	3	3	100%
S14	b.2	a	c	b	b	a	e	c	a	3	3	3	3	3	3	100%
S15	b.1	d	c	c	a	a	f	a	e	3	3	3	3	3	3	100%
S16	a.4	a	c	c	b	a	e	b	a	3	3	3	3	3	3	100%
S17	a.4	e	c	a	c	a	d	e	a	3	3	3	3	3	3	100%
S18	b.1	c	c	c	a	a	c	c	a	3	3	3	3	3	3	100%
S19	a.3	e	c	a	a	a	f	d	a	3	3	3	3	3	3	100%
S20	a.3	e	b	a	a	b	e	c	d	3	3	3	3	3	3	100%
S21	a.4	d	c	a	a	b	e	b	c	2	2	3	3	2	2.4	80%
S22	b.2	d	c	a	a	b	e	b	e	2	3	2	3	2	2.4	80%
S23	b.2	c	c	c	a	a	f	b	e	2	3	2	3	2	2.4	80%
S24	b.1	a	c	c	a	a	e	d	a	3	2	3	3	2	2.6	86.7%
S25	a.1	e	a	a	f	b	e	d	e	3	2	3	3	2	2.6	86.7%
S26	b.1	a	c	c	a	a	c	d	e	3	3	2	3	1	2.4	80%
S27	b.1	a	c	b	a	a	c	d	e	3	3	2	3	2	2.6	86.7%
S28	b.1	c	c	c	a	a	c	d	e	3	3	3	3	3	3	100%
S29	b.1	a	c	c	a	a	a	d	e	2	3	2	3	3	2.6	86.7%
S30	b.1	a	c	c	a	a	a	d	e	2	3	2	3	2	2.4	80%
S31	b.1	a	c	c	a	a	c	e	e	2	3	2	3	2	2.4	80%
S32	b.1	a	c	c	a	a	c	c	a	2	2	2	3	3	2.4	80%
S33	b.1	c	c	b	d	a	e	b	c	3	2	3	3	1	2.4	80%
S34	b.1	a	c	c	a	a	c	d	c	3	2	3	2	3	2.6	86.7%
S35	b.1	a	c	c	a	a	a	c	a	3	3	3	3	2	2.8	93.3%

S36	b.1	a	c	c	a	a	g	b	a	3	3	3	3	3	3	3	100%
S37	a.4	e	c	a	a	a	d	b	e	2	3	2	3	2			80%
S38	a.4	e	c	a	c	a	d	c	e	2	2	3	3	2			80%
S39	b.3	e	c	a	c	a	d	e	e	2	2	3	3	2			80%
S40	a.4	c	c	b	b	a	e	b	e	3	3	3	3	3			100%
S41	a.4	e	c	a	c	b	d	e	d	3	3	3	3	2			93.3%
S42	b.2	c	c	c	a	a	c	d	e	3	3	3	3	1			86.7%
S43	a.3	a	c	b	a	a	a	a	a	2	2	2	3	3			80%
S44	a.3	c	c	a	e	a	e	c	e	2	2	2	3	3			80%
S45	a.4	c	c	c	a	a	c	c	a	3	1	3	3	2			80%
S46	b.1	a	c	a	a	a	d	e	e	3	2	3	3	1			80%
S47	b.1	a	c	c	a	a	a	d	c	3	2	2	2	3			80%
S48	b.2	d	a	c	a	b	g	b	e	3	2	3	3	3			93.3%
S49	a.3	a	c	c	a	a	f	b	a	3	2	3	3	3			93.3%
S50	b.1	b	c	c	a	a	f	a	a	3	2	3	3	3			93.3%
S51	a.4	c	c	b	b	a	d	a	a	3	3	3	3	3			100%
S52	a.3	c	c	b	f	a	e	b	e	3	2	2	3	2			80%
S53	a.4	a	c	a	b	a	e	a	a	3	3	2	3	1			80%
S54	a.3	c	c	c	a	a	f	b	e	3	3	3	3	1			86.7%
S55	b.2	c	b	c	d	a	f	d	e	2	2	3	3	2			80%
S56	a.3	b	a	c	a	b	f	b	c	2	3	2	3	2			80%
S57	b.2	a	c	b	b	a	d	b	e	3	3	3	3	3			100%
S58	b.2	a	c	b	b	a	e	d	e	3	3	3	3	2			93.3%
S59	a.4	c	c	c	a	a	d	b	e	3	3	3	3	3			100%
S60	a.3	e	c	a	a	a	f	b	a	1	3	3	3	3			86.7%
S61	b.1	a	c	c	a	a	g	b	e	3	3	2	3	2			86.7%
S62	b.1	a	c	c	a	b	g	a	c	3	3	3	3	3			100%
S63	b.2	c	c	a	a	a	g	b	a	3	3	3	3	3			100%
S64	b.1	d	c	c	a	a	a	b	a	1	3	3	3	3			86.7%
S65	b.1	d	c	c	a	a	a	b	a	2	2	3	3	2			80%
S66	b.1	b	a	c	a	b	f	d	c	3	3	3	3	3			100%
S67	b.2	c	c	c	a	a	f	c	a	3	3	3	3	3			100%
S68	a.2	b	b	a	a	b	e	c	e	2	2	3	3	2			80%
S69	a.4	e	c	a	c	a	d	e	e	2	2	3	3	2			80%
S70	b.1	b	c	c	a	a	f	a	a	3	3	3	3	3			100%
S71	a.4	b	c	a	b	a	e	b	a	3	3	3	3	3			100%

Apéndice E – ISO/IEC 29110 Deployment Package

El *Deployment Package* o paquete de despliegue de la ISO/IEC 29110, es un conjunto de artefactos creado para facilitar la adopción de la norma ISO/IEC 29110 a las Muy Pequeñas Entidades de desarrollo de software (en inglés, *Very Small Entities*) (IEEE, 2008). Contiene un modelo general para organizar las actividades de los procesos de administración e implementación de software, los cuales se realizan en tres fases: a) desarrollo de prueba de concepto (Pre-producción), b) implementación (Producción) y, c) entrega (Post-producción) (véase Figura E-1). Durante la implementación, el proceso es altamente iterativo en ciclos para diseñar-desarrollar-probar-evaluar (en inglés, *design-develop-test-assess*).

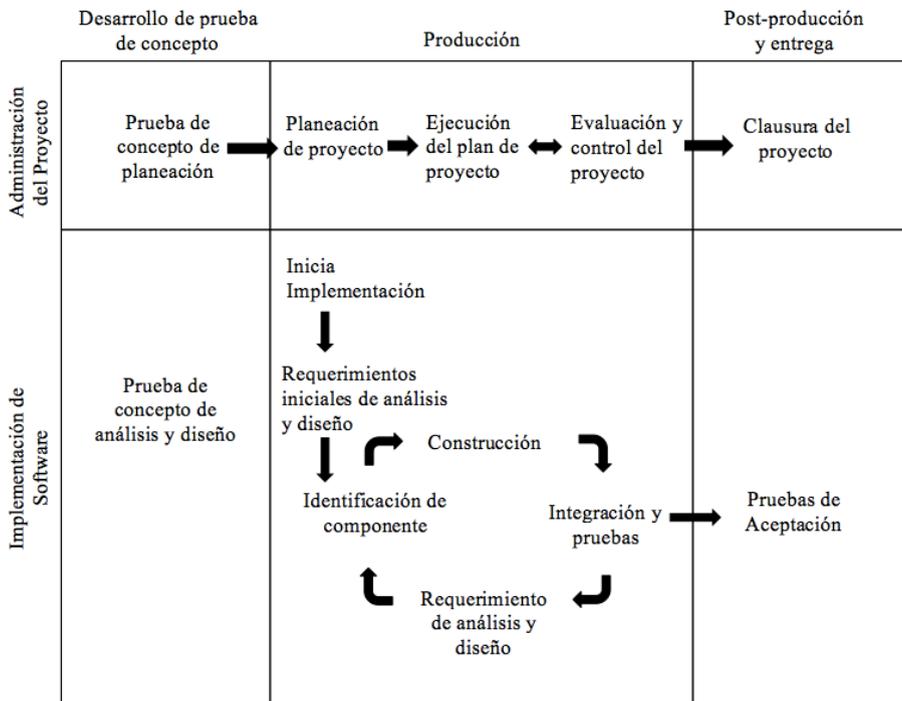


Figura E-1. Arquitectura de ISO/IEC 29110 Deployment Package

Apéndice F – Motores de videojuego

1. Unity

La primera versión de este motor de videojuego fue lanzada en 2005 por la compañía “Unity Technologies”, y la versión más reciente, Unity 5, fue presentada en el año 2017. Algunas de sus ventajas y desventajas principales se muestran a continuación.

Ventajas:

- Interfaz fácil de comprender.
- Por medio de la licencia gratuita de Unity, es posible portar el videojuego en Android, iOS, Windows, Linux y Mac OS.
- Brinda la posibilidad de desarrollar videojuegos para consolas como Xbox 360, Xbox One, Wii U, PlayStation 4, Nintendo Switch, entre otras.
- Permite usar elementos gráficos o “*assets*”, creados en diversos programas de modelado como: 3ds Max, Maya, Blender, Softimage, Cinema 4D, Adobe Photoshop, Adobe Fireworks, etc.
- Cuenta con una amplia biblioteca de *assets*.

Desventajas:

- Tiene funciones de modelado básicas, por lo que se requiere del uso de programas de modelado externos.
- La mayoría de los *assets* de su biblioteca tienen costo.
- El costo de la licencia para desarrollar videojuegos de consolas no es tan accesible.

Algunos videojuegos desarrollados con Unity son: “Pokémon Go”, un videojuego que consiste en buscar y capturar personajes de la saga Pokémon localizados en el mundo real (Niantic, Inc., 2016); “Wasteland 2”, un videojuego de rol post apocalíptico donde el jugador debe combatir con otros personajes para sobrevivir (inXile entertainment, 2018) y; “Firewatch”, un videojuego donde el jugador debe prevenir incendios (Campo Santo, 2018).

2. Unreal Engine

Este motor de videojuego fue lanzado por primera vez en 1998 por la compañía “Epic Games”. Su versión más reciente Unreal Engine 4, fue lanzada a inicios del 2018. Algunas de sus ventajas y desventajas principales se presentan a continuación.

Ventajas:

- Iluminación gráfica avanzada.
- Permite manejar hasta 1 millón de partículas por escena.
- Con una licencia gratuita, es posible crear videojuegos para las plataformas: Windows, Mac OS, Linux, Android y iOS.
- El costo de las licencias para desarrollar videojuegos de consolas es accesible.

Desventajas:

- Se requiere tener cierta experiencia en el desarrollo de videojuegos para usar este motor.
- En caso de que se desee comercializar un videojuego desarrollado con este motor, una parte de las regalías deben destinarse a la compañía del motor.

Algunos videojuegos que han sido desarrollados con Unreal Engine son: “American’s Army”, que permite interactuar en una misma misión militar hasta 32 jugadores (Department of Defense, 2002); “Batman: Arkham Asylum”, basado en el personaje de historietas “Batman” (Rocksteady Studios, 2009) y; “BioShock”, un videojuego en que el jugador debe luchar contra seres humanos mutantes y robots (2K Boston , 2009).

3. CryENGINE

Este motor de videojuego fue lanzado por primera vez en el 2002 por la compañía “Crytek”. La versión más reciente CryEngine V, fue lanzada en el 2016. Algunas de las ventajas y desventajas principales de este motor se presenten a continuación.

Ventajas:

- Iluminación gráfica avanzada.
- Efectos visuales de alta calidad a bajo costo de rendimiento.
- En caso de ser comercializado un videojuego que fue desarrollado con este motor, no se debe hacer pago de un porcentaje de las regalías a la empresa.
- Permite desarrollar videojuegos para las consolas de PlayStation 4 y Xbox One, así como para computadoras con Windows.

Desventajas:

- Se requiere tener cierta experiencia en el desarrollo de videojuegos para usar este motor.
- Es necesario pagar una licencia mensual para hacer uso del motor.

Algunos videojuegos que han sido desarrollados con CryENGINE son: “Evolve”, un videojuego de ciencia ficción (Take-Two Interactive Software, Inc, 2017); “Star Citizen”, un videojuego de simulación espacial (Cloud Imperium Games, 2018) y; “Ryse: Son of Rome”, un videojuego ambientado en la época del emperador romano Nerón, en el que el jugador interactúa como un general romano que debe luchar con personajes históricos (CryTek, 2018).

Apéndice G – Duolingo

Duolingo© es una aplicación móvil para el aprendizaje de 80 idiomas diferentes, que fue lanzada en el 2012 y cuenta con más de 150 millones de estudiantes en todo el mundo (Duolingo, 2013).

Posee un diseño basado en gamificación que combina incentivos de recompensa puntual con instrucción implícita (DeKeyser, 2008), aprendizaje de dominio (Block *et al.*, 1971) y explicaciones (Fahy, 2004).

Emplea los efectos de aprendizaje espaciado y progresivo, los cuales han sido ampliamente estudiados en el aprendizaje de segundas lenguas (Atkinson, 1972; Bloom y Shuell, 1981; Cepeda *et al.*, 2006; Pavlik Jr y Anderson, 2008).

El efecto de aprendizaje espaciado, consiste en distribuir el tiempo de estudio en periodos cortos. Este fenómeno fue estudiado por primera vez por Ebbinghaus en 1885, en su “teoría de la curva del olvido”, en la que expuso que es posible memorizar un mayor número de enunciados por medio del efecto espaciado, que con la práctica de “abarrotar”, que consiste en distribuir el tiempo de estudio en periodos largos y extenuantes.

Por otra parte, el efecto de aprendizaje progresivo fue estudiado por Melton, en 1970, quien observó que las personas aprenden mejor si el tiempo de las prácticas aumenta gradualmente.

De este modo, Duolingo implementa el efecto de aprendizaje espaciado y progresivo por medio de la regresión de media vida (en inglés, *Half-Life Regression HLR*), combinado con teoría de psicolingüística y técnicas de aprendizaje automático (en inglés, *machine learning*) (Settles y Meeder, 2016).

Apéndice H – Cuestionario para la evaluación de la motivación

Instrucciones: de la pregunta 1 a la 36, usando la escala que aparece a continuación, indique qué tan cierta considera cada afirmación relacionada con el juego serio computacional que usó para aprender sobre las Alfas del núcleo del estándar Essence.

Escala de Evaluación

- 1- Totalmente en desacuerdo
- 2- En desacuerdo
- 3- Ni de acuerdo ni en desacuerdo
- 4- De acuerdo
- 5- Totalmente de acuerdo

Afirmación	Evaluación				
	1	2	3	4	5
1. Cuando abrí por primera vez el juego serio computacional tuve la impresión de que sería fácil para mí.					
2. Había algo interesante en el juego serio computacional que llamó mi atención.					
3. El juego serio computacional fue más difícil de entender de lo que me hubiera gustado.					
4. Después usar el juego serio computacional, me sentí seguro de saber que había aprendido sobre las Alfas del núcleo de Essence.					
5. Usar el juego serio computacional para aprender sobre las Alfas del núcleo de Essence, me dio una sensación satisfactoria de haber obtenido un logro.					
6. Para mí está claro que el juego serio computacional estaba relacionado con el aprendizaje de las Alfas del núcleo de Essence.					
7. Había tanto contenido en el juego serio computacional que me fue difícil recordar elementos importantes.					

8. Este tipo de juegos serios computacionales captan mi atención.					
9. El juego serio computacional me mostró que era importante aprender sobre las Alfas del núcleo de Essence.					
10. Completar todos los módulos del juego serio computacional fue muy importante para mí.					
11. La calidad del juego serio computacional me ayudó a mantener la atención.					
12. El contenido educativo del juego serio computacional era tan abstracto que era difícil mantener la atención.					
13. Mientras usé el juego serio computacional estaba seguro de que podía aprender el contenido educativo.					
14. Disfruté tanto usando el juego serio computacional que me gustaría aprender más sobre las Alfas del núcleo de Essence.					
15. El contenido del juego serio computacional no era atractivo en lo absoluto.					
16. El contenido del juego serio computacional es relevante para mis intereses.					
17. La organización del contenido educativo del juego serio computacional contribuyó a mantener mi atención.					
18. El juego serio computacional incluyó explicaciones sobre cómo usar las Alfas del núcleo de Essence.					
19. Las explicaciones del juego serio computacional fueron demasiado difíciles.					
20. El juego serio computacional avivó mi curiosidad con respecto a las Alfas del núcleo de Essence.					
21. Realmente disfruté con el juego serio computacional.					
22. La cantidad de información en el juego serio computacional me pareció aburrida.					
23. El juego serio computacional me dio la impresión de que merecían utilizarse.					
24. El juego serio computacional mantuvo mi atención.					
25. Después de usar el juego serio computacional estaba seguro de que podía aprobar un examen sobre las Alfas del núcleo de Essence.					

26. Usar el juego serio computacional no era relevante para mis necesidades, porque ya tenía conocimiento sobre las Alfás del núcleo de Essence.					
27. El juego serio computacional me hizo sentir recompensado por mi esfuerzo.					
28. El contenido educativo del juego serio computacional llamó mi atención.					
29. El juego serio computacional no fue aburrido.					
30. Puedo relacionar el contenido educativo del juego serio computacional con cosas que he visto o hecho.					
31. Eran tantos los módulos del juego serio computacional que resultó ser molesto.					
32. Me sentí bien usando el juego serio computacional.					
33. El contenido educativo del juego serio computacional me será útil.					
34. No pude entender algunos módulos del juego serio computacional.					
35. La buena organización en de los módulos en el juego serio computacional me hizo sentir que podía aprender el contenido educativo.					
36. Fue un placer usar un juego serio computacional tan bien diseñado.					

Apéndice I – Cuestionario para la evaluación diagnóstica del aprendizaje (Pre-prueba)

Grupos: Control y experimental.
Fase del experimento: Pre-prueba.
Descripción: Este cuestionario fue empleado antes de la experimentación en ambos grupos, con el objetivo evaluar el conocimiento previo que poseían los participantes respecto a las Alfas del núcleo de Essence.
Instrucciones: De la pregunta 1 a la 12, indique con una sola “X” la opción que responde a cada pregunta.

1. ¿Qué es Essence?
- | | |
|--------------------------|-------------|
| <input type="checkbox"/> | Un método |
| <input type="checkbox"/> | Un estándar |
| <input type="checkbox"/> | Un proceso |
| <input type="checkbox"/> | Un patrón |
2. ¿Qué permite evaluar Essence?
- | | |
|--------------------------|---|
| <input type="checkbox"/> | El alcance y el tiempo de un proyecto de software |
| <input type="checkbox"/> | El tiempo y el costo de un proyecto de software |
| <input type="checkbox"/> | El progreso y el alcance de un proyecto de software |
| <input type="checkbox"/> | La salud y el progreso de un proyecto de software |
3. ¿Cuáles son las áreas de interés de Essence?
- | | |
|--------------------------|------------------------------|
| <input type="checkbox"/> | Software, hardware y proceso |
| <input type="checkbox"/> | Cliente, solución y esfuerzo |
| <input type="checkbox"/> | Plan, código y pruebas |
| <input type="checkbox"/> | Equipo, plan y software |
4. ¿Qué elementos componen el núcleo de Essence?
- | | |
|--------------------------|---|
| <input type="checkbox"/> | Roles, prácticas y responsabilidades |
| <input type="checkbox"/> | Métodos, prácticas y alfas |
| <input type="checkbox"/> | Alfas, espacios de actividad y competencias |
| <input type="checkbox"/> | Roles, alfas y prácticas |
5. ¿Qué es un Alfa?
- | | |
|--------------------------|--|
| <input type="checkbox"/> | Es un estado esencial en un proyecto de software |
| <input type="checkbox"/> | Es una fase esencial en un proyecto de software |
| <input type="checkbox"/> | Es un elemento esencial en un proyecto de software |
| <input type="checkbox"/> | Es un periodo esencial en un proyecto de software |

6. ¿A qué Alfa pertenecen los estados: identificada, solución necesaria, valor establecido, viable, abordada y beneficio percibido?

- Solución
- Forma de Trabajar
- Oportunidad
- Prueba

7. ¿A qué Alfa pertenecen los estados: reconocidos, representados, implicados, de acuerdo, satisfechos para el despliegue y satisfechos en uso?

- Equipo
- Usuarios
- Involucrados
- Clientes

8. ¿A qué Alfa pertenecen los estados: concebidos, acotados, coherentes, aceptables, abordados y cumplidos?

- Requerimientos
- Códigos
- Costos
- Métodos

9. ¿A qué Alfa pertenecen los estados: arquitectura seleccionada, demostrable, utilizable, listo, operacional y retirado?

- Arquitectura
- Código
- Diseño
- Sistema de Software

10. ¿A qué Alfa pertenecen los estados: iniciado, preparado, comenzado, bajo control, concluido y cerrado?

- Código
- Sistema de Software
- Trabajo
- Esfuerzo

11. ¿A qué Alfa pertenecen los estados: sembrado, formado, colaborando, ejecutando y retirado?

- Equipo
- Trabajo
- Sistema de Software
- Método

12. ¿A qué Alfa pertenecen los estados: principios establecidos, base establecida, en uso, en marcha, trabajando bien y retirada?

- | | |
|--------------------------|---------------------|
| <input type="checkbox"/> | Oportunidad |
| <input type="checkbox"/> | Sistema de Software |
| <input type="checkbox"/> | Arquitectura |
| <input type="checkbox"/> | Forma de Trabajar |

Apéndice J – Cuestionario para la evaluación final del aprendizaje (Post-prueba)

Grupos: Control y experimental.
Fase del experimento: Post-prueba.
Descripción: Este cuestionario fue empleado después de la experimentación en ambos grupos, con el objetivo evaluar el conocimiento adquirido por los participantes respecto a las Alfas del núcleo del estándar Essence.
Instrucciones: De la pregunta 1 a la 12, indique con una sola “X” la opción que responde a cada pregunta.

1. ¿Qué es Essence?
- | | |
|--------------------------|-----------------------|
| <input type="checkbox"/> | Un método del OMG |
| <input type="checkbox"/> | Un estándar del OMG |
| <input type="checkbox"/> | Un método de la ISO |
| <input type="checkbox"/> | Un estándar de la ISO |
2. ¿Qué se puede evaluar con Essence?
- | | |
|--------------------------|---|
| <input type="checkbox"/> | El tiempo y el costo de un proyecto de software |
| <input type="checkbox"/> | El progreso y el alcance de un proyecto de software |
| <input type="checkbox"/> | El alcance y el tiempo de un proyecto de software |
| <input type="checkbox"/> | La salud y el progreso de un proyecto de software |
3. ¿Cuáles son las áreas de interés de Essence?
- | | |
|--------------------------|------------------------------|
| <input type="checkbox"/> | Plan, código y pruebas |
| <input type="checkbox"/> | Equipo, plan y software |
| <input type="checkbox"/> | Cliente, solución y esfuerzo |
| <input type="checkbox"/> | Software, hardware y proceso |
4. ¿De qué está compuesto el núcleo de Essence?
- | | |
|--------------------------|---|
| <input type="checkbox"/> | Roles, alfas y prácticas |
| <input type="checkbox"/> | Roles, prácticas y responsabilidades |
| <input type="checkbox"/> | Alfas, espacios de actividad y competencias |
| <input type="checkbox"/> | Métodos, prácticas y alfas |
5. ¿Qué es un Alfa y qué contiene?
- | | |
|--------------------------|---|
| <input type="checkbox"/> | Es un periodo esencial en un proyecto de software, que contiene estados consecutivos. |
| <input type="checkbox"/> | Es una fase esencial en un proyecto de software, que contiene estados secuenciales. |

6. ¿ Cuáles son los estados del Alfa Oportunidad?

- Es un elemento esencial en un proyecto de software, que contiene estados secuenciales.
- Es un módulo esencial en un proyecto de software, que contiene estados consecutivos.
- Iniciada, confiable, factible, abordada, llevada a cabo y completada.
- Visualizada, factible, viable, iniciada, abordada y percibida.
- Identificada, solución necesaria, valor establecido, viable, abordada y beneficio percibido.
- Objetivos identificados, factible, prácticas seleccionadas, en curso, documentada y cerrada.

7. ¿ Cuáles son los estados del Alfa Involucrados?

- Seleccionados, de acuerdo, comprometidos, colaborando, satisfechos para la ejecución y satisfechos con la entrega.
- Identificados, reconocidos, colaborando, satisfechos para el desarrollo, satisfechos para la construcción y satisfechos con el producto.
- Reconocidos, representados, implicados, de acuerdo, satisfechos para el despliegue y satisfechos en uso.
- Establecidos, identificados, involucrados, colaborando, desarrollando y satisfechos.

8. ¿ Cuáles son los estados del Alfa Requerimientos?

- Concebidos, acotados, coherentes, aceptables, abordados y cumplidos.
- Establecidos, viables, formalizados, desarrollados, verificados y aceptados.
- Identificados, coherentes, acotados, dirigidos,

construidos y entregados.

9. ¿Cuáles son los estados del Alfa Sistema de Software ?

- Acordados, fiables, delimitados, formalizados, desarrollados y liberados.
- Especificado, analizado, modelado, codificado, integrado y entregado.
- Concebido, arquitectura establecida, desarrollado, liberado, probado y cerrado.
- Acordado, viable, establecido, construido, usable y retirado.
- Arquitectura seleccionada, demostrable, utilizable, listo, operacional y retirado.

10. ¿Cuáles son los estados del Alfa Trabajo?

- Iniciado, preparado, ejecutado, realizado, documentado y concluido.
- Preparado, iniciado, bajo control, terminado, documentado y cerrado.
- Iniciado, preparado, comenzado, bajo control, concluido y cerrado.
- Planificado, iniciado, elaborado, monitoreando, terminado y documentado

11. ¿Cuáles son los estados del alfa Equipo?

- Sembrado, formado, colaborando, ejecutando y retirado.
- Concebido, preparado, trabajando, colaborando y concluido.
- Formado, integrado, trabajando, realizando y terminado.
- Seleccionado, formado, ejecutado, documentado y cerrado.

12. ¿Cuáles son los estados del alfa Forma de Trabajar?

- Metas establecidas, prácticas seleccionadas, en curso, ejecutando bien, documentando y concluida.
- Objetivos establecidos, base establecida, en uso, desempeñando, bajo control y completada.
- Metodología establecida, prácticas seleccionadas, ejecución correcta, avanzando, documentada y cerrada.
- Principios establecidos, base establecida, en uso, en marcha, trabajando bien y retirada.

Apéndice K – Cuestionario para identificar las características de los participantes en el experimento

Grupos: Control y experimental.
Fase del experimento: Pre-prueba.
Descripción: Este cuestionario fue empleado antes de la experimentación en ambos grupos, con el objetivo de identificar las características de los participantes.
Instrucciones: De la pregunta 1 a la 10, indique con una sola “X” la opción que responde a cada pregunta.

Información Personal

1. ¿Cuál es su género?

- Femenino.
 Masculino.

2. ¿Cuál es su nacionalidad?

- Mexicana.
 Otra: _____

3. ¿Cuál es su edad?

- Menos de 24 años.
 De 24-30 años.
 De 31-40 años.
 De 41-50 años.
 Más de 50 años.

4. ¿Cuál es su estado civil?

- Soltero.
 Casado.
 Divorciado.
 Viudo.

5. ¿Cuál es su último grado académico?

- Estudiando la Preparatoria o Licenciatura.
- Licenciatura.
- Maestría.
- Doctorado.

6. ¿Cuál es su experiencia laboral¹ en el área de Tecnologías de la Información?

- Menos de 3 años.
- De 3-5 años.
- Más de 5 años.

7. ¿Qué rol² desempeña en su trabajo actual?

- Analista.
- Arquitecto.
- Programador.
- Gerente de proyecto.
- Líder técnico.
- Otro: _____

Experiencia en el Uso de Aplicaciones Móviles

8. ¿Con qué frecuencia usa aplicaciones móviles?

- Nunca.
- Raramente.
- A veces.
- Frecuentemente.
- Siempre

9. ¿Con qué frecuencia usa aplicaciones móviles educativas?

- Nunca.
- Raramente.
- A veces.
- Frecuentemente.
- Siempre

¹ Basado en los años de experiencia que se consideran para las categorías de: aprendiz, novato (junior), intermedio y experto (senior).

² Software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile, ISO/IEC TR 29110-5-1-2, ISO/IEC, 2011.

Apéndice L – Resultados de la evaluación diagnóstica del aprendizaje (Pre-prueba)

Grupos: Control y experimental.

Fase del experimento: Pre-prueba.

La Tabla L-1, muestra las respuestas correctas a las preguntas del cuestionario diseñado para la evaluación diagnóstica del aprendizaje (véase Apéndice I).

Tabla L-1. Respuestas correctas del cuestionario para la evaluación diagnóstica del aprendizaje

Pregunta	Respuesta correcta
1. ¿Qué es Essence?	Un estándar.
2. ¿Qué permite evaluar Essence?	La salud y el progreso de un proyecto de software.
3. ¿Cuáles son las áreas de interés de Essence?	Cliente, solución y esfuerzo.
4. ¿Qué elementos componen el núcleo de Essence?	Alfas, espacios de actividad y competencias.
5. ¿Qué es un Alfa?	Es un elemento esencial en un proyecto de software.
6. ¿A qué Alfa pertenecen los estados: identificada, solución necesaria, valor establecido, viable, abordada y beneficio percibido?	Oportunidad.
7. ¿A qué Alfa pertenecen los estados: reconocidos, representados, implicados, de acuerdo, satisfechos para el despliegue y satisfechos en uso?	Involucrados.
8. ¿A qué Alfa pertenecen los estados: concebidos, acotados, coherentes, aceptables, abordados y cumplidos?	Requerimientos.
9. ¿A qué Alfa pertenecen los estados: arquitectura seleccionada, demostrable, utilizable, listo, operacional y retirado?	Sistema de Software.
10. ¿A qué Alfa pertenecen los estados: iniciado, preparado, comenzado, bajo control, concluido y	Trabajo.

cerrado?

11. ¿A qué Alfa pertenecen los estados: sembrado, formado, colaborando, ejecutando y retirado? Equipo.

12. ¿A qué Alfa pertenecen los estados: principios establecidos, base establecida, en uso, en marcha, trabajando bien y retirada? Forma de Trabajar.

La calificación de los cuestionarios respondidos por los participantes de ambos grupos (control y experimental), se asignó de acuerdo al número de respuestas correctas, véase Tabla L-2.

Tabla L-2. Calificaciones de acuerdo al número de respuestas correctas

No. Respuestas correctas	Calificación
0	0
1	8
2	17
3	25
4	33
5	42
6	50
7	58
8	67
9	75
10	83
11	92
12	100

De este modo fue que se obtuvieron las calificaciones de los participantes del grupo de control (véase Tabla L-3) y los del grupo experimental (véase Tabla L-4), en la Pre-prueba.

Tabla L-3. Calificaciones obtenidas por los participantes del grupo de control en la Pre-prueba

No. Partic.	Pregunta												Calificación		
	1	2	3	4	5	6	7	8	9	10	11	12			
1	1	1	1	1	0	0	0	0	0	0	0	0	0	33	
2	1	1	1	0	0	0	0	0	0	0	0	0	0	25	
3	1	1	1	0	0	1	0	0	0	0	0	0	0	33	
4	1	1	1	1	0	0	0	1	0	0	0	0	0	42	
5	1	0	1	0	0	0	1	0	0	0	0	0	0	25	
6	1	1	0	0	0	0	0	0	0	1	0	0	0	25	
7	1	1	0	0	0	0	0	0	0	0	0	0	0	17	
8	0	1	0	1	0	0	0	1	0	0	0	0	0	25	
9	0	0	1	0	1	0	0	0	0	0	0	1	0	25	
10	1	0	0	0	0	0	0	0	0	0	0	0	0	8	
11	1	1	1	1	0	0	0	0	0	0	0	0	0	33	
12	1	0	0	0	0	0	0	0	0	0	0	0	0	8	
13	0	1	1	0	0	1	0	0	0	0	0	0	0	25	
14	1	1	1	1	0	0	0	0	0	0	0	0	0	33	
15	1	0	1	0	0	0	0	0	0	0	0	0	0	17	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	1	1	0	0	0	0	0	0	0	0	0	0	0	17	
18	1	1	1	1	1	0	0	0	0	1	0	0	0	50	
19	1	0	1	0	0	0	0	0	0	0	0	0	0	17	
20	1	1	0	0	0	0	0	0	0	0	1	0	0	25	
21	1	1	1	1	0	0	0	0	0	0	0	0	0	33	
22	1	0	1	0	0	0	1	0	0	0	0	0	0	25	
23	0	1	0	0	0	0	0	0	0	0	0	0	0	8	
24	1	0	1	0	0	0	0	0	0	0	0	0	0	17	
25	1	1	1	1	1	0	0	0	0	0	0	0	0	42	
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
27	1	0	1	0	0	0	0	0	0	0	0	0	0	17	
28	1	0	0	0	0	0	0	0	0	0	0	0	0	8	
29	1	1	0	0	0	0	0	0	0	0	0	0	0	17	
30	1	1	1	1	0	0	0	0	0	0	0	0	0	33	
														Promedio	22.77

Notación empleada en la evaluación de las respuestas:

0 – Respuesta incorrecta

1 – Respuesta correcta

Tabla L-4. Calificaciones obtenidas por los participantes del grupo experimental en la Pre-prueba

No. Partic.	Pregunta												Calificación	
	1	2	3	4	5	6	7	8	9	10	11	12		
1	1	1	1	0	0	0	0	0	0	0	0	0	25	
2	1	0	0	0	0	0	0	0	0	0	0	0	8	
3	1	0	1	0	0	0	0	0	0	0	0	0	17	
4	1	1	1	1	0	0	0	0	0	0	0	0	33	
5	1	1	1	0	0	0	0	0	0	0	0	0	25	
6	1	1	1	1	0	0	0	1	0	0	0	0	42	
7	1	1	0	0	0	0	0	0	0	0	0	0	17	
8	1	1	0	0	0	0	0	0	0	0	0	0	17	
9	1	1	1	0	0	0	0	0	0	0	0	0	25	
10	1	1	1	0	0	0	0	0	0	0	0	0	25	
11	1	1	0	0	0	0	0	0	0	0	0	0	17	
12	1	1	1	1	0	0	0	0	0	0	0	0	33	
13	1	1	1	1	0	0	0	0	0	0	0	0	33	
14	1	0	0	0	0	0	0	0	0	0	0	0	8	
15	1	0	1	0	0	0	0	0	0	0	0	0	17	
16	1	1	1	1	0	0	0	0	0	0	1	0	42	
17	1	1	1	0	0	0	0	0	0	0	0	0	25	
18	1	1	0	0	0	0	0	0	0	0	0	0	17	
19	1	1	1	1	0	0	0	0	0	0	0	0	33	
20	1	1	1	1	0	0	0	0	0	0	0	0	33	
21	1	1	1	1	0	0	0	0	0	0	0	0	33	
22	0	0	0	0	0	0	0	0	0	0	0	0	0	
23	1	1	1	0	0	0	0	0	0	0	0	0	25	
24	1	0	0	0	0	0	0	0	0	0	0	0	8	
25	1	1	1	0	0	0	0	0	0	0	0	0	25	
26	1	0	0	0	0	0	0	0	0	0	0	0	8	
27	1	0	0	0	0	0	0	0	0	0	0	0	8	
28	1	0	1	0	0	0	0	0	0	0	0	0	17	
29	1	1	1	0	0	0	0	0	0	0	0	0	25	
30	1	1	0	0	0	0	0	0	0	0	0	0	17	
													Promedio	21.93

Notación empleada en la evaluación de las respuestas:

0 – Respuesta incorrecta

1 – Respuesta correcta

Apéndice M – Resultados de la evaluación final del aprendizaje (Post-prueba)

Grupos: Control y experimental.

Fase del experimento: Post-prueba.

La Tabla M-1, muestra las respuestas correctas a las preguntas del cuestionario diseñado para la evaluación final del aprendizaje (véase Apéndice J).

Tabla M-1. Respuestas correctas del cuestionario para la evaluación final del aprendizaje

Pregunta	Respuesta correcta
1. ¿Qué es Essence?	Un estándar del OMG.
2. ¿Qué se puede evaluar con Essence?	La salud y el progreso de un proyecto de software.
3. ¿Cuáles son las áreas de interés de Essence?	Cliente, solución y esfuerzo.
4. ¿De qué está compuesto el núcleo de Essence?	Alfas, espacios de actividad y competencias.
5. ¿Qué es un Alfa y qué contiene?	Es un elemento esencial en un proyecto de software, que contiene estados secuenciales.
6. ¿Cuáles son los estados del Alfa Oportunidad?	Identificada, solución necesaria, valor establecido, viable, abordada y beneficio percibido.
7. ¿Cuáles son los estados del Alfa Involucrados?	Reconocidos, representados, implicados, de acuerdo, satisfechos para el despliegue y satisfechos en uso.
8. ¿Cuáles son los estados del Alfa Requerimientos?	Concebidos, acotados, coherentes, aceptables, abordados y cumplidos.
9. ¿Cuáles son los estados del Alfa Sistema de Software ?	Arquitectura seleccionada, demostrable, utilizable, listo, operacional y retirado.
10. ¿Cuáles son los estados del Alfa Trabajo?	Iniciado, preparado, comenzado, bajo control, concluido y cerrado.
11. ¿Cuáles son los estados del alfa Equipo?	Sembrado, formado, colaborando, ejecutando y retirado.
12. ¿Cuáles son los estados del alfa Forma de Trabajar?	Principios establecidos, base establecida, en uso, en marcha, trabajando bien y retirada.

La calificación de los cuestionarios respondidos por los participantes de ambos grupos (control y experimental), se asignó de acuerdo al número de respuestas correctas de la Tabla L-2 (véase Apéndice L). De este modo se obtuvieron las calificaciones de los participantes del grupo de control (véase Tabla M-2) y los del grupo experimental (véase Tabla M-3) en la Post-prueba.

Tabla M-2. Calificaciones obtenidas por los participantes del grupo de control en la Post-prueba

No. Partic.	Pregunta												Calificación
	1	2	3	4	5	6	7	8	9	10	11	12	
1	1	1	1	1	1	1	1	0	1	1	0	1	83
2	1	1	1	1	1	1	0	1	0	1	1	0	75
3	1	1	1	1	1	1	0	1	0	1	1	1	83
4	1	1	1	1	1	1	1	1	0	1	1	1	92
5	1	1	1	1	1	1	0	0	1	0	1	1	75
6	1	1	1	1	1	0	1	1	0	1	1	0	75
7	1	1	1	1	1	0	1	0	1	0	0	0	58
8	1	1	1	1	1	0	1	1	0	1	0	1	75
9	1	1	1	1	1	1	0	0	1	0	1	1	75
10	1	1	1	1	1	0	0	0	1	0	1	0	58
11	1	1	1	1	1	1	0	1	0	1	1	1	83
12	1	1	1	1	1	0	1	0	1	0	0	0	58
13	1	1	1	1	1	0	1	1	0	0	1	0	67
14	1	1	1	1	1	1	0	0	1	1	0	1	75
15	1	1	1	1	0	1	1	1	0	0	0	1	67
16	1	1	1	1	0	1	0	0	1	0	0	0	50
17	1	1	1	1	1	0	1	0	0	0	1	0	58
18	1	1	1	1	1	1	1	1	1	1	1	1	100
19	1	1	1	1	0	1	0	1	0	1	0	1	67
20	1	1	1	1	1	0	1	0	1	0	1	0	67
21	1	1	1	1	1	1	1	0	0	1	0	1	75
22	1	1	1	1	1	0	0	1	0	1	0	1	67
23	1	1	1	1	1	0	1	0	0	0	1	0	58
24	1	1	1	1	1	1	0	0	1	0	0	1	67
25	1	1	1	1	1	1	0	1	1	1	1	1	92
26	1	1	1	1	0	0	1	0	0	0	1	0	50
27	1	1	1	1	0	1	0	1	1	0	0	1	67

28	1	1	1	1	0	0	1	0	1	0	1	0	58
29	1	1	1	1	1	1	0	1	0	0	0	1	67
30	1	1	1	1	1	0	1	1	1	0	1	1	83
	Promedio												70.83

Notación empleada en la evaluación de las respuestas:

0 – Respuesta incorrecta

1 – Respuesta correcta

Tabla M-3. Calificaciones obtenidas por los participantes del grupo experimental en la Post-prueba

No. Partic.	Pregunta												Calificación
	1	2	3	4	5	6	7	8	9	10	11	12	
1	1	1	1	1	1	0	1	0	1	1	1	1	83
2	1	1	1	1	1	0	1	1	0	0	1	0	67
3	1	1	1	1	1	1	0	0	1	1	0	1	75
4	1	1	1	1	1	1	1	1	1	0	1	1	92
5	1	1	1	1	1	1	1	1	0	1	1	0	83
6	1	1	1	1	1	1	1	1	1	1	1	1	100
7	1	1	1	1	1	0	1	1	0	1	0	1	75
8	1	1	1	1	0	1	1	0	1	1	1	1	83
9	1	1	1	1	1	1	1	1	0	0	1	1	83
10	1	1	1	1	1	1	1	0	1	1	1	0	83
11	1	1	1	1	1	0	1	1	0	1	0	1	75
12	1	1	1	1	1	1	1	1	0	1	1	1	92
13	1	1	1	1	1	1	1	1	1	1	0	1	92
14	1	1	1	1	1	0	0	0	1	0	1	1	67
15	1	1	1	1	1	0	1	1	0	1	0	1	75
16	1	1	1	1	1	1	1	1	1	1	1	1	100
17	1	1	1	1	1	0	1	1	1	0	1	1	83
18	1	1	1	1	1	1	1	0	0	1	0	1	75
19	1	1	1	1	1	1	0	1	1	1	1	1	92
20	1	1	1	1	1	0	1	1	1	1	1	1	92
21	1	1	1	1	1	1	1	1	1	0	1	1	92
22	1	1	1	1	1	0	0	1	0	0	1	0	58
23	1	1	1	1	1	1	1	0	1	1	0	1	83
24	1	1	1	1	0	1	0	1	0	0	1	1	67
25	1	1	1	1	1	0	1	1	1	1	0	1	83

26	1	1	1	1	1	0	1	0	1	0	1	0	67
27	1	1	1	1	0	1	0	1	0	0	1	1	67
28	1	1	1	1	1	1	0	1	0	1	0	1	75
29	1	1	1	1	1	0	1	0	1	1	1	1	83
30	1	1	1	1	1	1	1	1	0	0	0	1	75
Promedio												80.57	

Notación empleada en la evaluación de las respuestas:

0 – Respuesta incorrecta

1 – Respuesta correcta

Apéndice N – Resultados de la evaluación de la motivación (Post-prueba)

Grupos: Control y experimental.

Fase del experimento: Post-prueba.

La Tabla N-1 muestra los resultados de la evaluación de la motivación en el grupo de control. Para facilitar el análisis, se identificaron con un color diferente cada una de las afirmaciones evaluadas de acuerdo con el factor del modelo ARCS³ al que pertenecen. Esta información permitió calcular los promedios de la Tabla N-2.

Similarmente, la Tabla N-3 presenta los resultados de la evaluación de la motivación en el grupo de experimental, cuyos promedios pueden apreciarse en la Tabla N-4.

³ J. M. Keller, *Motivational Design for Learning and Performance: The ARCS Model Approach*. New York, NY, USA: Springer, 2010.

Tabla N-1. Resultados de la evaluación de la motivación en el grupo de control

P	Afirmación																																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
1	4	4	1	4	3	4	1	4	4	5	3	2	5	4	2	4	4	5	1	5	4	2	4	4	5	1	4	4	2	4	2	4	4	4	1	5	4
2	3	3	2	4	5	5	2	5	4	4	4	2	4	4	2	4	4	5	2	3	3	3	4	5	4	2	3	3	3	4	3	4	4	1	5	3	
3	4	4	2	4	4	3	2	4	4	3	4	2	4	5	1	4	5	4	2	4	4	1	3	3	4	2	4	4	1	4	2	5	4	2	4	4	
4	4	3	3	5	4	4	3	4	5	5	5	4	3	4	3	4	4	4	3	4	4	2	4	5	4	1	4	5	2	5	2	4	4	2	4	3	
5	4	4	2	5	3	4	2	3	3	5	4	3	4	3	2	3	3	4	2	5	3	2	4	4	4	2	3	4	2	4	1	3	4	3	3	3	
6	4	4	1	4	4	4	1	4	4	3	3	2	5	4	3	4	4	4	2	3	4	3	4	4	3	1	3	4	2	4	2	4	3	2	4	4	
7	4	3	2	4	3	5	2	3	4	4	4	3	4	3	2	4	4	4	2	4	4	2	4	5	4	2	4	3	3	5	3	3	4	1	5	3	
8	5	3	1	5	4	4	1	4	5	5	3	2	4	3	3	4	5	4	2	5	3	2	4	4	3	2	5	4	2	3	2	4	4	2	5	4	
9	5	4	2	4	4	4	2	3	4	5	3	1	4	4	1	3	4	3	2	4	4	2	5	4	3	1	3	4	2	5	2	4	5	1	5	3	
10	4	4	2	3	3	3	3	3	4	5	4	2	3	4	2	4	5	4	3	4	3	2	4	4	4	2	4	4	3	4	2	4	4	3	4	4	
11	3	5	1	4	4	4	1	4	5	4	3	1	4	3	2	3	3	4	2	3	4	3	4	5	3	1	5	3	3	4	1	4	4	3	4	3	
12	4	4	1	4	3	4	1	3	4	4	4	2	4	4	3	4	4	4	2	5	3	3	5	3	3	2	4	5	2	4	1	3	4	2	3	3	
13	4	4	3	4	4	4	2	3	5	5	5	2	4	4	1	3	4	4	3	3	3	2	5	4	3	1	3	5	2	5	1	3	4	2	4	3	
14	4	3	1	4	4	3	1	4	4	5	3	3	4	3	3	3	5	4	1	3	3	3	4	4	4	2	4	3	3	4	1	4	4	1	4	4	
15	3	4	2	3	5	4	2	3	4	4	4	2	3	4	3	4	5	4	3	4	4	3	4	4	4	1	3	4	2	4	3	5	4	2	5	4	
16	3	5	3	3	4	5	1	4	5	4	3	2	4	5	2	3	3	4	3	3	4	3	4	5	4	1	4	4	2	4	1	4	4	1	4	4	
17	4	4	3	3	4	4	3	4	4	4	4	2	3	4	3	4	4	4	3	3	3	2	3	4	3	3	3	4	2	3	2	4	3	3	4	3	
18	3	5	2	3	4	4	1	3	4	4	4	4	4	3	2	4	4	5	2	3	5	2	5	3	4	2	4	4	3	4	2	3	3	1	5	4	
19	4	4	1	3	4	4	2	4	5	4	4	2	3	4	2	3	5	4	3	3	4	1	4	4	4	1	3	3	3	4	2	4	4	2	4	3	

20	4	4	2	3	3	3	2	4	4	5	5	2	3	3	2	4	4	4	3	4	5	2	5	4	3	2	4	4	2	4	3	4	4	1	5	4
21	3	5	2	3	4	4	1	5	4	4	4	1	5	4	1	4	4	4	2	4	4	2	5	3	4	2	4	4	2	4	1	4	4	2	4	4
22	5	5	3	4	3	5	2	4	4	4	4	2	4	3	2	4	5	4	3	3	5	3	3	4	3	2	4	3	2	3	2	3	3	3	3	3
23	4	4	1	3	4	3	2	3	4	4	3	3	4	4	1	3	4	3	3	4	3	1	5	4	4	1	3	5	1	4	2	3	3	1	5	4
24	5	4	2	4	4	4	3	4	4	5	5	2	3	3	1	3	4	5	3	3	3	2	4	5	4	2	4	4	3	4	3	4	4	2	4	4
25	4	3	2	3	3	5	1	3	5	4	3	1	5	3	3	3	3	4	2	4	4	3	3	4	3	1	4	4	1	4	2	4	4	1	5	4
26	5	5	1	4	3	4	2	3	5	4	4	1	4	4	1	3	4	4	3	3	3	2	4	4	3	2	3	3	2	4	3	3	4	2	4	3
27	3	5	3	3	4	3	2	4	4	5	5	2	3	4	1	4	4	4	3	4	4	1	4	3	4	2	5	4	1	4	1	4	4	3	3	3
28	3	5	1	4	3	4	1	4	3	4	3	1	5	5	3	3	4	4	2	4	3	2	3	4	3	2	3	3	1	3	3	3	3	3	3	3
29	4	4	2	4	4	4	3	4	4	5	4	2	3	5	1	4	5	4	3	4	4	3	4	5	3	1	4	3	1	4	3	4	4	3	4	4
30	4	4	2	4	3	4	1	3	5	4	4	2	4	3	2	4	4	4	2	5	3	2	5	4	4	2	3	4	2	4	2	3	4	2	4	4

Notación:

P – Participante

I – Invertir valor de evaluación

		Factor de motivación			
		Atención	Relevancia	Confianza	Satisfacción
Color					

Tabla N-2. Promedios de los factores de motivación en el grupo de control

No. Partic.	Factores			
	Atención	Relevancia	Confianza	Satisfacción
1	4.00	4.33	4.78	3.83
2	3.67	4.22	4.11	3.67
3	4.25	3.67	4.00	4.33
4	3.92	4.44	3.67	3.83
5	3.92	3.89	3.89	3.00
6	3.67	3.89	4.22	3.83
7	3.58	4.22	4.22	3.33
8	3.92	4.11	4.44	3.83
9	4.00	4.33	4.22	3.67
10	3.92	4.00	3.44	3.67
11	3.83	4.11	3.89	3.83
12	3.92	4.11	4.00	3.33
13	4.17	4.44	3.67	3.33
14	3.50	3.89	4.44	3.67
15	3.75	4.11	3.67	4.17
16	3.92	4.22	3.78	4.17
17	3.83	3.56	3.22	3.50
18	3.58	4.11	4.11	3.83
19	3.92	4.11	3.78	3.67
20	4.00	4.11	3.78	3.83
21	4.33	4.11	4.00	4.00
22	3.92	3.78	3.56	3.50
23	4.08	3.78	4.11	3.50
24	4.00	4.11	3.78	3.67
25	3.67	4.11	4.22	3.67
26	3.92	4.00	4.00	3.17
27	4.42	4.00	3.22	4.00
28	3.92	3.44	3.89	3.33
29	4.08	4.22	3.44	4.17
30	4.00	4.22	4.11	3.17
Promedio	3.92	4.06	3.92	3.68
Desv. Est.	0.21	0.24	0.36	0.33
Promedio		3.90		

Tabla N-3. Resultados de la evaluación de la motivación en el grupo experimental

P	Afirmación																																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
1	5	5	2	5	5	5	2	5	4	5	4	1	5	5	2	4	3	4	1	4	5	1	5	3	4	2	5	5	1	5	3	5	4	1	5	5
2	4	5	1	4	4	4	1	4	5	5	5	2	4	3	1	3	5	5	2	4	3	1	5	4	5	1	5	5	3	4	2	4	4	2	4	4
3	5	5	3	5	5	5	3	5	3	4	4	1	3	4	2	4	4	4	3	5	3	2	4	5	5	1	5	5	1	5	1	4	4	1	3	4
4	3	4	1	3	4	3	2	5	4	4	4	3	4	5	1	4	5	5	1	3	4	1	4	4	3	1	3	4	2	4	1	5	3	3	3	4
5	4	3	1	4	4	5	1	5	5	4	5	2	3	3	1	5	5	5	2	4	5	1	5	5	3	1	3	5	1	5	2	5	5	1	3	5
6	5	5	1	3	5	5	1	3	4	4	3	1	5	4	1	5	5	5	3	5	5	2	5	5	4	2	4	5	1	5	1	3	5	1	5	4
7	3	4	1	4	4	4	1	4	5	5	5	3	5	3	1	5	5	5	1	3	4	3	5	4	4	1	4	4	2	5	1	3	5	1	5	3
8	4	5	1	4	5	5	1	3	4	5	5	1	4	5	1	4	5	4	3	4	4	2	3	5	3	1	4	3	2	4	3	4	4	3	5	4
9	4	5	3	4	5	3	1	4	5	5	4	1	4	4	2	5	5	5	1	5	5	1	5	5	4	1	5	5	1	5	1	3	5	1	5	5
10	4	3	1	5	4	5	2	4	3	5	4	3	5	4	3	4	3	4	1	5	5	2	4	3	3	1	3	3	3	4	1	4	3	1	4	5
11	5	4	1	3	5	3	1	4	3	3	4	2	5	5	2	3	4	5	3	5	5	1	4	4	4	1	5	5	1	3	2	3	5	2	5	4
12	4	5	1	4	4	4	1	4	3	3	4	1	4	5	1	4	5	4	1	4	5	1	5	5	3	1	3	4	2	5	2	4	4	1	3	4
13	5	5	3	4	5	3	3	3	4	3	3	1	5	4	2	3	3	5	2	4	5	2	5	3	4	1	4	4	2	4	2	5	5	3	4	4
14	3	4	1	5	3	2	1	3	5	4	4	1	3	4	1	3	4	4	1	4	5	1	4	5	5	1	4	5	1	4	1	5	5	1	4	3
15	4	5	3	4	4	5	1	3	5	3	3	1	4	5	1	5	4	5	3	3	4	1	4	3	4	1	5	4	1	5	1	4	5	1	4	3
16	3	5	3	5	3	5	1	5	4	4	4	1	3	3	1	4	4	5	3	5	5	1	5	4	3	1	3	5	2	3	1	4	5	1	3	5
17	5	5	1	4	5	3	1	5	3	3	4	1	5	4	1	5	3	4	1	4	5	3	3	4	3	2	3	5	3	3	3	5	4	3	3	4
18	3	4	2	3	3	3	1	3	3	3	5	3	4	4	3	4	5	5	1	5	5	1	5	4	5	3	5	5	1	5	1	5	3	1	5	5
19	4	5	3	4	4	5	1	5	4	3	5	1	3	3	1	3	5	5	1	4	5	2	3	5	5	1	5	4	2	4	3	3	3	1	3	3

20	5	5	3	5	5	5	3	4	4	4	4	1	3	3	1	5	3	5	3	5	4	1	5	5	4	2	4	5	1	4	1	4	4	1	4	4
21	3	4	1	5	3	3	1	4	5	5	3	1	3	3	3	4	4	5	1	3	4	1	5	4	5	1	5	3	1	5	2	3	5	3	3	5
22	4	3	3	4	4	4	3	3	4	4	3	3	3	3	3	5	3	4	3	3	3	3	3	3	4	3	4	3	3	3	4	3	3	4	5	
23	4	5	1	3	5	5	1	5	5	5	4	1	4	4	1	3	5	3	1	5	4	1	5	5	5	1	4	4	1	3	1	5	3	1	3	5
24	5	5	1	5	5	5	3	3	3	4	4	1	3	4	2	3	5	5	1	5	5	1	4	5	5	1	4	4	2	5	2	5	5	1	4	4
25	3	5	3	3	3	4	1	5	5	3	4	2	5	4	2	5	4	4	1	4	3	2	3	4	3	1	3	4	1	4	1	4	5	1	5	5
26	4	5	1	3	4	5	1	5	4	3	5	2	5	5	2	3	5	5	2	5	5	1	5	5	4	2	5	5	1	5	1	5	4	1	3	4
27	3	3	3	3	3	5	3	3	3	3	4	2	4	5	2	4	5	5	3	3	4	1	3	5	5	1	5	4	2	4	3	3	5	3	3	3
28	4	4	1	3	4	3	1	5	3	4	5	1	5	5	1	3	4	3	1	5	5	1	3	3	3	1	5	5	1	4	1	5	3	1	3	3
29	5	5	1	4	5	4	1	5	3	5	5	1	5	5	2	5	3	4	1	5	5	2	4	3	4	1	3	3	2	5	3	3	5	2	4	4
30	4	5	1	5	4	5	1	5	4	5	5	1	4	5	1	5	5	4	1	4	5	1	5	5	4	1	5	5	1	5	1	5	5	1	5	5

Notación:

P – Participante

I – Invertir valor de evaluación

		Factor de motivación			
		<i>Atención</i>	<i>Relevancia</i>	<i>Confianza</i>	<i>Satisfacción</i>
Color					

Tabla N-4. Promedios de los factores de motivación en el grupo experimental

No. Partic.	Factores			
	Atención	Relevancia	Confianza	Satisfacción
1	4.25	4.44	4.67	5.00
2	4.42	4.44	4.33	3.83
3	4.67	4.22	3.89	4.17
4	4.25	4.00	3.67	4.17
5	4.58	4.89	4.00	4.17
6	4.58	4.67	4.44	4.17
7	4.08	4.89	4.56	3.50
8	4.25	4.22	4.00	4.33
9	4.75	4.78	4.33	4.50
10	3.58	4.11	4.44	4.17
11	4.33	3.78	4.33	4.50
12	4.50	4.11	4.22	4.17
13	3.83	4.11	3.89	4.50
14	4.50	4.00	4.44	4.00
15	4.17	4.67	4.00	4.17
16	4.67	4.44	3.67	3.83
17	4.08	3.56	4.22	4.33
18	4.33	3.78	4.33	4.50
19	4.50	3.89	4.11	3.83
20	4.67	4.44	3.89	4.00
21	3.92	4.67	4.11	3.83
22	3.00	3.67	3.44	3.83
23	4.83	4.11	4.33	4.50
24	4.42	4.33	4.44	4.50
25	4.33	4.22	4.11	3.67
26	4.83	4.22	4.22	4.67
27	3.92	4.11	3.33	3.83
28	4.67	3.44	4.22	4.50
29	4.08	4.44	4.56	4.17
30	4.92	4.78	4.67	4.83
Promedio	4.33	4.25	4.16	4.21
Desv. Est.	0.41	0.39	0.34	0.35
Promedio		4.24		

Apéndice Ñ – Características de los participantes en el experimento

Grupos: Control y experimental.

Fase del experimento: Pre-prueba.

Al inicio del experimento se les solicitó a los profesionales de software que respondieran el cuestionario presentado en el Apéndice K, con el objetivo de conocer las características de los participantes. Los resultados obtenidos se muestran en la Tabla Ñ-1.

Tabla Ñ-1. Características de los participantes en el experimento

No.	Característica	Clasificación	Grupos				Ambos grupos	
			Control		Experimental		No. Pers.	%
			No. Pers.	%	No. Pers.	%		
1	Género	Femenino	3	10%	4	13.3%	7	11.7%
		Masculino	27	90%	26	86.7%	53	88.3%
2	País de nacimiento	México	26	86.7%	25	83.3%	51	85%
		India	3	10%	4	13.3%	7	11.7%
		Brasil	1	3.3%	1	3.3%	2	3.3%
		Menos de 24 años	1	3.3%	1	3.3%	2	3.3%
3	Edad	De 24 a 30 años	25	83.3%	24	80%	49	81.7%
		De 31 a 40 años	3	10%	4	13.3%	7	11.7%
		De 41 a 50 años	1	3.3%	1	3.3%	2	3.3%
		Más de 50 años	0	0%	0	0%	0	0%
		Soltero	20	66.7%	19	63.3%	39	65%
4	Estado civil	Casado	8	26.7%	8	26.7%	16	26.7%
		Divorciado	2	6.7%	2	6.7%	4	6.7%
		Viudo	0	0%	1	3.3%	1	1.7%

		Aún estudia.	0	0%	0	0%	0	0%
5	Formación académica	Licenciatura	26	86.7%	27	90%	53	88.3%
		Maestría	4	13.3%	3	10%	7	11.7%
		Doctorado	0	0%	0	0%	0	0%
		Menos de 3 años	1	3.3%	1	3.3%	2	3.3%
6	Experiencia laboral	De 3 a 5 años	11	36.7%	12	40%	23	38.3%
		Más de 5 años	18	60%	17	56.7%	35	58.3%
		Analista	2	6.7%	1	3.3%	3	5%
7	Puesto de trabajo actual	Arquitecto	0	0%	0	0%	0	0%
		Programador	25	83.3%	26	86.7%	51	85%
		Gerente de proyecto	1	3.3%	1	3.3%	2	3.3%
		Líder técnico	2	6.7%	2	6.7%	4	6.7%
		Otro	0	0%	0	0%	0	0%
8	Frecuencia de uso de aplicaciones móviles	Nunca	0	0%	0	0%	0	0%
		Raramente	0	0%	1	3.3%	1	1.7%
		A veces	6	20%	8	26.7%	14	23.3%
		Frecuentemente	17	56.7%	15	50%	32	53.3%
		Siempre	7	23.3%	6	20%	13	21.7%
9	Frecuencia de uso de aplicaciones móviles educativas	Nunca	1	3.3%	1	3.3%	2	3.3%
		Raramente	5	16.7%	7	23.3%	12	20%
		A veces	17	56.7%	16	53.3%	33	55%
		Frecuentemente	6	20%	5	16.7%	11	18.3%
		Siempre	1	3.3%	1	3.3%	2	3.3%

Apéndice O – Acrónimos

ACM	<i>ACM Digital Library</i>
AMC	Academia Mexicana de Ciencias
AR	<i>Action-Research</i>
ARCS	<i>Attention, Relevance, Confidence and Satisfaction</i>
<i>B-Learning</i>	<i>Blended Learning</i>
BDs	Bases de Datos
CONACYT	Consejo Nacional de Ciencia y Tecnología
CONISOFT	<i>International Conference in Software Engineering Research and Innovation</i>
<i>E-Learning</i>	<i>Electronic Learning</i>
FEDER	Fondo Europeo de Desarrollo Regional
FODA	Fortalezas, Oportunidades, Debilidades y Amenazas
GC	Grupo de Control
GE	Grupo Experimental
IMI	<i>Intrinsic Motivation Inventory</i>
IS	Ingeniería de Software
ISO	<i>International Organization for Standardization</i>
KSA	<i>Knowledge, Skills and Abilities</i>
<i>M-Learning</i>	<i>Mobile Learning</i>
MINECO	Ministerio de Economía y Competitividad de España
OE	Objetivos Específicos
OMG	<i>Object Management Group</i>
OP	Objetivo Principal
PAEP	Programa de Apoyo a Estudiantes de Posgrado
RUP	<i>Rational Unified Process</i>
SC	<i>Scopus</i>
SD	<i>Science Direct</i>

SMS	<i>Systematic Mapping of Studies</i>
SP	<i>Springer</i>
TAR	<i>Technical Action-Research</i>
TecNM	Tecnológico Nacional de México
UNAM	Universidad Nacional Autónoma de México
VARK	<i>Visual, Auditory, Reading/Writing and Kinesthetic</i>
VSEs	<i>Very Small Entities</i>
XP	<i>eXtreme Programming</i>