

## Una revisión sistemática de Juegos Serios para la enseñanza de Métodos Ágiles

Guillermo Rodríguez<sup>1,3</sup> Matías Glessi<sup>2</sup> Alfredo Teyseyre<sup>2,3</sup>

<sup>1</sup>Universidad Argentina de la Empresa – Instituto de Tecnología (UADE - INTEC), CABA, Argentina

<sup>2</sup>Universidad Nacional del Centro de la Provincia de Buenos Aires, Tandil, Argentina

<sup>3</sup>Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

guirodriguez@uade.edu.ar, matiasglessi@gmail.com

alfredo.teyseyre@isistan.unicen.edu.ar

**Resumen.** Los juegos serios han surgido con el objetivo de mejorar el compromiso y el desempeño de los usuarios jugadores, ya que realizan una misión exacta, integrándose con la mecánica del juego, creando así una misión muy atractiva. Se ha investigado con mucho interés el uso de juegos serios en Ingeniería de Software para aumentar el compromiso de desarrolladores. Por otro lado, el uso de juegos serios resulta de gran interés para capacitar y entrenar a futuros profesionales para experimentar situaciones a las que podrían enfrentar en el desarrollo de software. El objetivo de este trabajo es presentar un relevamiento sistemático de la literatura en el campo de los juegos serios enfocados en la enseñanza de los hoy tan usados Métodos Ágiles, y evaluar diferentes métodos y procedimientos utilizados para evaluar juegos serios. La revisión sigue un procedimiento predefinido que implica buscar artículos en bases de datos digitales conocidas. La revisión sistemática identificó los principales métodos seguidos para evaluar juegos serios, los dominios de aplicación en que se realizaron las evaluaciones, las principales características consideradas para evaluar la efectividad educativa de los juegos serios y los procedimientos seguidos para las evaluaciones. Los resultados son útiles para los investigadores y practicantes dispuestos a evaluar juegos serios en el área de enseñanza de Métodos Ágiles.

**Palabras Claves:** Juegos serios, Relevamiento sistemático, Educación en Ingeniería de Software, Métodos Ágiles, Framework de evaluación.

### 1 Introducción

Dentro de las nuevas tecnologías, una de las cuáles ha crecido indiscutiblemente en los últimos tiempos ha sido la industria de los juegos digitales. Desde los primeros juegos para plataformas de los años 80s, pasando por la explosión de la PC en los años 90s y los juegos para smartphones de los últimos años, los juegos digitales han mejorado increíblemente, siendo ahora reconocida como una industria multimillonaria. Sin embargo, emergiendo dentro de este paradigma, surgió una vertiente de juegos que no tienen como único objetivo la diversión per se, sino también otras implicancias. El aprendizaje basado en juegos tiene el potencial de mover a los entrena-

dores y estudiantes a un nuevo enfoque de enseñanza/aprendizaje donde el juego guía a los estudiantes a descubrir y experimentar su adquisición de conocimiento. Los juegos aplicados en este contexto, para educar o entrenar a los usuarios se les llama juegos serios [1, 2].

Por un lado, la educación en Ingeniería de Software ha cobrado considerable atención por las universidades para formar estudiantes capaces de desarrollar software conforme la demanda de la industria [3], y por otro lado, la academia se encuentra incorporando juegos serios como estrategias de aprendizaje en sus planes de estudios [4]. El uso de técnicas, elementos y dinámicas propias de los juegos y el ocio en actividades no recreativas es un concepto denominado ludificación. Este concepto es utilizado en muchas áreas con la intención de potenciar la motivación, así como de reforzar la conducta para solucionar un problema, mejorar la productividad, obtener un objetivo, activar el aprendizaje y evaluar a individuos concretos [5].

Esta tendencia busca facilitar la consecución de ciertos objetivos relacionados con la formación, fidelización, cohesión social, creatividad, etc. Ha cambiado la forma en que los estudiantes aprenden, corriendo del eje a los modelos de educación tradicional [6]. La educación a través de dispositivos (en su mayoría móviles) permite que los usuarios no están limitados a un lugar y a un tiempo para aprender, logrando flexibilidad para adaptarse a sus rutinas, así como también brindando experiencias más auténticas [7]. A su vez, existe un crecimiento exponencial de los recursos digitales disponibles, y permite poner en práctica de forma relativamente sencilla nuevos modelos pedagógicos.

El objetivo principal de este trabajo es identificar los temas de investigación de los estudios relacionados con los juegos serios en la enseñanza de Ingeniería de Software y caracterizar los juegos serios actuales disponibles para la educación de Ingeniería de Software, particularmente en Metodologías Ágiles. El resto del artículo está organizado de la siguiente manera: la Sección 2 analiza los trabajos relacionados. La Sección 3 ilustra la metodología utilizada para llevar a cabo la revisión sistemática de la literatura. Finalmente, la Sección 4 presenta conclusiones y proporciona ideas para posibles trabajos futuros.

## **2 Trabajos Relacionados**

Inicialmente, se ha realizado una búsqueda de estudios secundarios en bases de datos académicas para saber caracterizar el estado del arte de los juegos serios en educación en Ingeniería de Software e identificar líneas de trabajo que podrían ser exploradas para enriquecer la investigación en el área.

Como resultado, encontramos varios estudios secundarios, como el mapeo sistemático realizado por Heredia et al. que caracteriza el estado del arte de la educación en procesos de software [8], el relevamiento sistemático de Kosa et al. que clasificó los diferentes usos de los juegos en el campo de la educación en Ingeniería de Software [9] y la revisión de Petri et al. que identifica los métodos para evaluar los juegos serios para la educación en Ingeniería de Software [10]. En esta línea, otro trabajo ha propuesto una revisión de literatura para identificar juegos serios para la educación de estándares de procesos de software [11]. Dos trabajos estaban directamente relaciona-

dos con el campo de la educación de procesos de software [8, 11]. En [11], los autores se centraron en el área específica de la educación sobre estándares de procesos de software. Concretamente, identificaron tres juegos serios para enseñar el ISO / IEC 12207, el ISO / IEC 29110 e ISO / IEC 15504 y concluyeron que los juegos serios tienen potencial como herramientas de apoyo para la educación de estándares de procesos de software, aunque, se necesitan más resultados de investigación y experimentos para observar todo el potencial de los juegos serios como recursos de aprendizaje para la enseñanza en este campo. Por otro lado, en [8], los autores analizaron la literatura para aclarar las características generales de la educación en procesos de software y las iniciativas de capacitación en el campo.

En el estudio realizado por Rabail et al., los autores intentaron llenar este vacío llevando a cabo un relevamiento sistemático de la literatura enfocado específicamente solo en la evaluación de juegos educativos e identificación de las dimensiones claves para evaluar el aprendizaje basado en juegos [12]. Siguiendo esta línea, Pedreira et al. realizaron un mapeo sistemático del campo de la gamificación en ingeniería de software en un intento de caracterizar el estado del arte de este campo identificando nichos y oportunidades para futuras investigaciones. Como resultado del mapeo sistemático, encontraron 29 estudios primarios, publicados entre enero 2011 y junio de 2014. La mayoría de ellos se centran en el desarrollo de software y, en menor medida, los requerimientos, gestión de proyectos y otras áreas de soporte. En general, los trabajos relevados consideran una ludificación muy simple, y pocos proporcionan evidencia empírica del impacto de la ludificación [13]. Calderon et al. elaboraron una revisión sistemática de la literatura para resumir el estado actual del arte de los diferentes métodos y procedimientos utilizados para evaluar juegos serios. La revisión de la literatura identificó los principales métodos seguidos para evaluar juegos serios, los dominios de aplicación para evaluar la efectividad educativa de los juegos serios, los procedimientos seguidos para las evaluaciones y el tamaño de la población que participó en las evaluaciones [14]. Ahmadi et al. realizaron un mapeamiento sistemático para evaluar el uso de los juegos serios en Ingeniería de Software, dejando de lado aquellos prototipos netamente académicos. Luego de catalogar y clasificar los diferentes estudios primarios, los autores detectaron que la aceptación del juego serio en Ingeniería de Software está siendo más gradualmente mejor en comparación con otras áreas como, ventas, educación o aplicaciones móviles [15].

En este contexto, no existe un estudio secundario que caracterice el estado del arte de juegos serios utilizados en la enseñanza de Ingeniería de Software, principalmente la enseñanza de Métodos Ágiles. Por esa razón, en este trabajo, hemos realizado una revisión sistemática para caracterizar el estado del arte de los juegos serios para la educación de Métodos Ágiles y, por lo tanto, identificar los juegos actuales en esta área de investigación.

### **3 Método**

Este trabajo tiene como objetivo analizar los estudios actuales relacionados con el campo de los juegos serios para enseñanza en Ingeniería de Software. Para obtener el

estado del arte de este alcance se realizó un relevamiento sistemático de la literatura basado en el método científico propuesto por Kitchenham et al. [16, 17]. De acuerdo con Kitchenham y Charters [16], el relevamiento sistemático de la literatura es una metodología efectiva y recomendada para investigar estudios empíricos con el fin de analizar una pregunta de investigación, una vez que la evidencia más confiable proviene de agregar estudios empíricos sobre un tema en particular. La Figura 1 ilustra el proceso definido llevado a cabo.

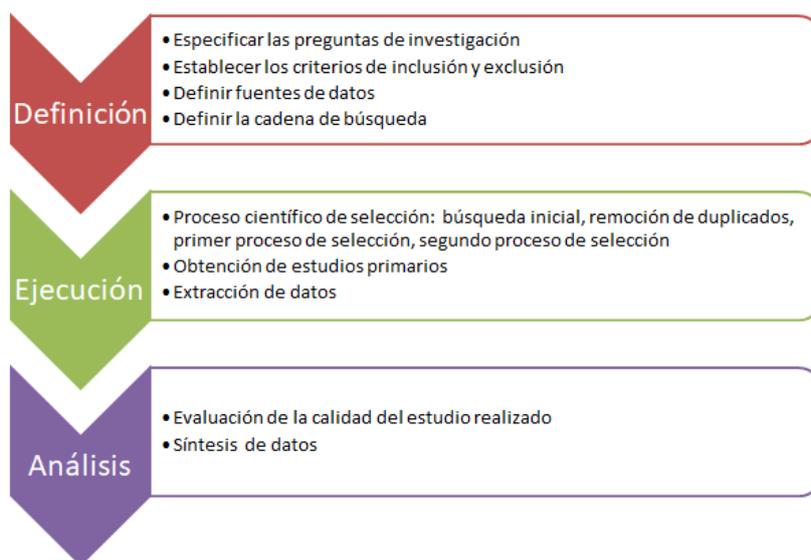


Fig. 1. Proceso definido del relevamiento sistemático.

### 3.1 Definición

Este trabajo tiene como objetivo caracterizar el estado del arte de los juegos serios para la educación en Ingeniería de Software, principalmente en Métodos Ágiles. De acuerdo con este propósito, realizamos un relevamiento sistemático de la literatura que aborda las siguientes preguntas de investigación (PI):

- **PI 1:** ¿Qué estudios abordan el alcance de los juegos serios para la educación de algún aspecto de los Métodos Ágiles?
- **PI 2:** ¿Qué juegos serios se centran en la educación de algún aspecto de los Métodos Ágiles?

Teniendo en cuenta nuestro objetivo y las preguntas de investigación, establecimos los criterios de inclusión y exclusión para seleccionar solo estudios relevantes:

**Criterios de Inclusión.** El estudio analizado aborda el alcance de los juegos serios para la educación en Ingeniería de Software, principalmente en Métodos Ágiles. El estudio analizado introduce un juego serio para la educación en Métodos Ágiles. El estudio analizado describe las características educativas de un juego serio para la educación en Métodos Ágiles. El estudio analizado evalúa un juego serio para la

enseñanza de Métodos Ágiles. El estudio analizado agrega valor al campo de los juegos serios para la educación en Métodos Ágiles. El estudio fue escrito en inglés.

**Criterios de Exclusión.** El estudio analizado presenta los resultados de evaluar un juego serio para la enseñanza en Ingeniería de Software, pero no muestra ninguna información sobre las características de dicho juego. El estudio analizado se enfoca solo en aspectos teóricos y filosóficos, sin introducir ningún recurso de aprendizaje relacionado con el campo de los juegos serios para la enseñanza de Métodos Ágiles. El estudio analizado solo tiene su resumen disponible y no es posible encontrar su texto completo. El estudio analizado no proporciona la información requerida en forma clara. El estudio analizado fue escrito en un idioma diferente al inglés.

**Fuente de Datos.** Las bases de datos digitales elegidas fueron: IEEE Xplore, ISI Web of Science, SpringerLink, ACM, SCOPUS y Wiley. La ventana de tiempo seleccionada fue 2014-2019.

**Cadena de Búsqueda.** Para construir la cadena de búsqueda, primero identificamos los términos de búsqueda. Por esa razón, realizamos algunas búsquedas iniciales para probar y calibrar la cadena de búsqueda. Finalmente, definimos la cadena de búsqueda como la siguiente expresión booleana definida en la Tabla 1.

**Tabla 1.** Cadena de búsqueda.

| <b>Términos Principales</b> | <b>Términos Alternativos</b>  |
|-----------------------------|---|
| Gamification                | (gamification OR gamifying OR gamify OR funware OR computer games)<br>AND   |
| Software Engineering        | (software engineering) OR (agile methods) OR (software process) OR (agility) OR (Scrum) OR (XP) OR (Extreme Programming) OR (Teaching Scrum) OR (Crystal Clear) OR (Lean) OR (Kanban) OR (RUP) OR (Dynamic Systems Development) OR (Rational Unified Process) |

### 3.2 Ejecución

**Extracción de Datos.** Durante el proceso de extracción, los estudios primarios se leyeron completamente para recopilar toda la información necesaria y garantizar que los datos sean precisos. Todos los datos recopilados se almacenaron en una hoja de cálculo. Esto permitió colocar toda la información del estudio en el mismo lugar, al mismo tiempo que se facilitó el análisis y comparación de los datos recopilados durante el proceso de síntesis. Los datos de los trabajos seleccionados se clasificaron según los siguientes criterios:

- El tema principal del estudio (abordar PI 1).
- Nombre de la herramienta o juego serio (abordar PI 2).
- Descripción de la herramienta o juego serio (abordar PI 2).
- Tipo de juego serio o herramienta (abordar PI 2).

- Los objetivos de aprendizaje del juego serio (abordar PI 2).

### 3.3 Análisis

Finalmente quedaron seleccionados en la última etapa de nuestro proceso definido 10 estudios primarios. Una vez que hemos extraído los datos de los estudios primarios, realizamos la fase de análisis, en la que se evaluó la calidad de los estudios primarios para la calidad y se sintetizaron datos en pos de dar respuesta a las preguntas de investigación previamente definidas.

**CodeCombat**<sup>1</sup>. En este trabajo, los autores buscan utilizar juegos para acompañar la enseñanza de conceptos de Ingeniería del Software, sobre todo en estudiantes de otras Ingenierías con poco o nulo conocimiento de los mismos. En este trabajo utilizaron el juego CodeCombat para enseñar conceptos del Paradigma Orientado a Objetos en estudiantes de carreras de Ingeniería de Negocios. Para ello, los estudiantes tomaron en paralelo el curso online con la clase tradicional. A través de una encuesta de motivación y percepción, se encontró que la mayoría de los estudiantes creen que el juego los ayudó a aprender y entender algunos conceptos de una mejor manera. A su vez, se midió el impacto en las notas tras haber utilizado el juego, contrastandolo con el curso que solo mantuvo la educación tradicional. El resultado mostró un decremento en las malas notas y un incremento en las buenas [18].

**Hard Choices**. En este juego, el autor analiza que los estudiantes una vez que se graduaron no se encuentran con un nivel de preparación del mundo real en lo que Ingeniería del Software respecta. Para esto, toma el concepto de la Deuda Técnica y afirma que los estudiantes no se encuentran expuestos a su aparición durante la resolución de diferentes proyectos. El autor utiliza el juego *Hard Choices*, el cual representa las actividades del desarrollo de software. En este juego, los alumnos compiten entre ellos, acumulando puntos a medida que avanzan, simulando el proceso de creación de software. El objetivo del juego es llegar a la meta antes que los demás competidores, lo cual brinda una cierta cantidad de puntos. En cierto momento los jugadores tienen la posibilidad de tomar un atajo, dejando puntos en el camino pero llegando antes al final. Así, tienen que tomar decisiones difíciles en cuanto a la posible pérdida de puntos, lo cual traza un paralelo con la acumulación de Deuda Técnica en el desarrollo de software [19].

Para analizar el efecto del juego en la impartición del concepto se desarrollaron pre-tests y post-tests, con preguntas multiple-choice, con una única respuesta correcta, la cual se relacionaba directamente a un concepto del juego. Por otro lado se realizaron cuestionarios de 5 puntos para medir factores cualitativos, tales como el entendimiento, la diversión y la facilidad. Los resultados de los tests arrojaron que el juego funcionó como estrategia efectiva para el aprendizaje de la Deuda Técnica. Asimismo, dados los valores del cuestionario cualitativo, los alumnos percibieron que el aprendizaje a través de un juego fue divertido y atrapante. El autor concluyó que el juego *Hard Choices* tiene una fuerte analogía con el concepto de Deuda Técnica, que

---

<sup>1</sup> <https://codecombat.com/>

junto a escenarios impredecibles, se le da a los estudiantes una herramienta que la educación tradicional no ha logrado hasta ahora proveer.

**Badges – Leaderbooks.** En este trabajo, los autores describen la experiencia de introducir 2 elementos de juego a una clase introductoria de Ingeniería del Software. El objetivo fue el de evaluar la percepción de los estudiantes en el impacto de estos elementos en motivarlos a completar el curso. Los elementos introducidos fueron *badges* (insignias) y *leaderbook* (tabla de liderazgo). Mientras las primeras premiaban a los estudiantes por logros específicos, la segunda indicaba la performance general de cada estudiante con respecto a sus pares. Estos elementos se encontraban disponibles en la página web del curso, donde los estudiantes podían acceder y tener feedback de forma regular. Para analizar la percepción se realizó una evaluación en 2 pasos. Primero, se llevaron a cabo encuestas con el objetivo de obtener feedback del impacto de las insignias y la tabla de liderazgo durante el curso. Segundo, se entrevistaron alumnos para una investigación más profunda sobre percepciones del curso y el uso de los elementos de juego mencionados [20].

Los resultados mostraron una percepción positiva al uso de insignias en el curso. Los alumnos mostraron interés y las notaron como una recompensa social, y como un objetivo más durante el curso, aparte de las notas. En cuanto a la tabla de liderazgo las evaluaciones resultaron en una percepción negativa. Sin embargo, las entrevistas mostraron que los alumnos comparan su rendimiento entre pares, y cuando éste es menor al resto de la clase, se ven motivados a mejorar, lo cual puede verse como el objetivo de la tabla, presentado de otra forma. Los autores concluyeron que el rol de la ludificación en la educación de la Ingeniería del Software debe continuar siendo explorada y evaluada. En este trabajo se encontró que su aplicación es relevante a la hora de plantear una alternativa educacional. Por otro lado, se encontró que la aplicación de la misma requiere un esfuerzo significativo por parte de los instructores para configurar y mantener los elementos durante el curso.

**LEGO Blocks.** Los autores introdujeron un enfoque para enseñar conceptos de Scrum en la práctica, simulando Sprints, mientras que incrementalmente planeaban y desarrollaban un producto armado con bloques de LEGO. El trabajo se dividió en dos partes, por un lado la simulación de Scrum en sí, y por el otro, el estudio de las reacciones y aprendizajes de los estudiantes. La simulación consta de tres equipos de Scrum que trabajan en simultáneo, con un Product Owner principal encargado de asistirlos a todos. Se busca, a través de un proyecto en particular, que los alumnos aprendan sobre procesos de Scrum y sus roles, análisis de requerimientos y colaboración con el cliente, estimación, trabajo en equipo, visualización del trabajo y progreso. En cuanto a las reacciones y aprendizajes, los alumnos mostraron compromiso y entusiasmo al aprender de forma lúdica en contraste con la enseñanza tradicional. Se observó que incluso los alumnos con mayor experiencia reportaron haber aprendido nuevos conceptos [21].

**SCRUM-X.** Los autores han presentado la ejecución de un juego para la enseñanza de Scrum, con el objetivo de comparar la educación basada en simulaciones y juegos contra la educación tradicional. Scrum-X es un juego creado en una hoja de cálculo, que simula la aplicación de Scrum. Involucra el proceso completo con sus respectivas fases, como Sprint Planning, ejecución de la iteración, Sprint Review y Sprint Retro-

spective. Para ello, el juego muestra los diferentes pasos a seguir, ayudado de información teórica, con información a completar y pop-ups que llevan al usuario a tomar decisiones. Se observó que este tipo de enseñanza puede tener un impacto positivo en la motivación para aprender del estudiante, mientras que provee una experiencia de aprendizaje atractiva y disfrutable. El autor concluyó además, que es necesario tener un mínimo de conocimiento de Scrum para jugarlo, o al menos proveer más información teórica dentro de la misma aplicación [22].

**SCRUMIA.** Los autores presentan un juego hecho con papel y lápiz con el objetivo de reforzar y enseñar a alumnos la aplicación de SCRUM en los programas de Ciencias de la Computación, complementando sus clases teóricas. El juego surge como una alternativa de bajo presupuesto para complementar la forma tradicional de educación en el aula. Este puede ser aplicado durante una típica clase universitaria con una duración aproximada de 60 minutos. Se trabaja sobre la planificación y la ejecución de un Sprint de un proyecto prototipo, donde los productos incrementales son creados con papel, como aviones, botes y sombreros. La ejecución de SCRUMIA consta de 7 partes: explicación inicial (del Proyecto e Introducción a Scrum), estimación en Story Points, Sprint Planning, ejecución del Sprint ( Reunión de Kick-Off, Ejecución 1, Daily Meeting 1, Ejecucion 2, Daily Meeting 2, Ejecución 3, Daily Meeting 3), Sprint review, Release, Conclusiones [23].

El autor concluyó que la utilización de una herramienta lúdica genera una contribución positiva para la motivación a estudiar, dándoles confianza en el proceso y satisfacción por ser un contenido aplicable en su futuro profesional. Por otro lado, encontraron que la explicación y/o la secuencia del juego debería ser simplificada. Además, si bien los alumnos observaron que el juego tenía un ritmo adecuado, lo consideraron no lo suficientemente desafiante.

**SCRUMI.** Los autores han presentado el desarrollo y validación de un juego serio de mesa, en un formato electrónico, llamado SCRUMI, con el objetivo de enseñar conceptos inherentes a Scrum. Según los autores, el uso de la ludificación en el campo educacional representa una herramienta valiosa para fomentar los intereses de los estudiantes frente a las metodologías tradicionales, haciendo el aprendizaje más interesante. El juego propuesto busca ayudar a los estudiantes a entender contenidos relacionados a la disciplina del Project Management así como también, la divulgación del uso de juegos serios con la intención de aumentar su utilización en educación. SCRUMI es un juego de preguntas y respuestas, pensado para introducir conceptos claves de Scrum. Consta de 5 fases: Preparación (presentación de conceptos desde la formación de un equipo hasta la definición de un Product Backlog inicial), Análisis (actividades relacionadas al Sprint Planning), Ejecución (todos los conceptos relacionados a Sprints, haciendo foco en el desarrollo del product), Monitoreo y Control (referente a Daily Meeting y Sprint Review) y Cierre (contempla Review Meeting y Sprint Retrospective). En SCRUMI, cada casillero se estipula en resolver en 1 minuto y consiste en un total de 44 minutos; quien logra completarlo en el tiempo estipulado gana el juego, por el contrario, quien falla más de 10 preguntas o tarda más del 30% del tiempo estimado, pierde. Se observó que la mayoría de los estudiantes se encontraron motivados a enfrentar los problemas del juego, buscando resolver los desafíos propuestos. Solo muy pocos demostraron sentirse no desafiados o indiferentes, y/o en

desacuerdo con algunas preguntas. Los autores concluyeron que el uso de juegos y sus mecánicas podría ser más explotado en la enseñanza de Project Management y otras disciplinas, ya que aportan una experiencia diferente y disfrutable, comparada con los contenidos vistos en el aula tradicional [24].

**Minecraft.**<sup>2</sup> Los autores presentan la enseñanza de Scrum de forma práctica con ludificación, utilizando el juego Minecraft. Se trabajó en clases universitarias con aproximadamente 110 alumnos en 2 períodos distintos. Previamente los alumnos recibieron, como marco teórico, la siguiente información: introducción general al Project Management y su organización, presentaciones externas de tres Project Managers, un estudio de los propios estudiantes sobre obstáculos encontrados en proyectos de la vida real en empresas, una presentación de un enfoque general de gestión ágil de proyectos, una presentación de Scrum y un video de su aplicación en empresas [25].

El producto fue definido por un profesor, en su rol de Product Owner: una versión real a escala de forma virtual 3D en Minecraft del campus universitario, para 1700 estudiantes, con edificios secundarios, incluyendo aulas, laboratorios y otros salones. La aplicación del proyecto se realizó de forma presencial y a distancia, pudiendo los alumnos continuar el desarrollo desde sus hogares, y comunicándose a través del sistema Moodle<sup>3</sup>, el cual fue creado con un chat, foro y un repositorio por cada equipo. Luego de una evaluación posterior, se concluyó que, si bien algunos alumnos se quejaron de la carga de trabajo, el proyecto fue realizable y disfrutable. Solo 2 personas de los 110 alumnos hubieran preferido un enfoque más tradicionalista (proyecto real de desarrollo de software). Además, en una reunión retrospectiva final, los alumnos explícitamente mencionaron que la distribución de trabajo y la comunicación funcionaron de forma correcta. Se encontraron sorprendidos por el trabajo final y con cuanto puede ser logrado entre 50-60 personas, utilizando una visión ágil del Project Management y una correcta división de tareas.

**Trello-based game.** Los autores han propuesto un aprendizaje basado en juegos para Scrum, utilizando Trello<sup>4</sup>. Si bien no es una herramienta de Scrum, sus características permiten que sea usada para el desarrollo de proyectos. Realizando esta actividad, los alumnos inicialmente aprenden el enfoque ágil de Scrum a través de Trello, para luego ser divididos en equipos y crear un plan de acción para un juego en dicha aplicación, dado un caso de estudio. Una vez hecho esto, los equipos presentan sus tableros de Trello en una competición, para obtener recompensas y feedback por el trabajo realizado. Finalmente, utilizan su conocimiento y experiencia obtenida para el desarrollo de un proyecto real. Esta metodología de aprendizaje de Scrum se aplicó durante 4 años a dos cohortes diferentes. Luego de finalizar cada período, se evaluó a los alumnos en sus conocimientos de Scrum. Al notar significativas mejoras en los resultados obtenidos en comparación a años anteriores, los autores concluyeron que el aprendizaje de forma lúdica e interactiva resulta ser más exitoso en contraste con la educación tradicional, considerando el aprendizaje de Scrum como aspecto principal,

---

<sup>2</sup> <https://www.minecraft.net/es-es/>

<sup>3</sup> <https://moodle.org/>

<sup>4</sup> <https://trello.com/>

pero también otras habilidades “blandas” como presentación, comunicación y trabajo en equipo [26].

**Scrum gamificado.** En este trabajo, los autores han propuesto un sistema gamificado denominado Scrum Master, con el propósito que el jugador realice dicho rol. El sistema se implementó utilizando Unity y fue exportado para la plataforma de dispositivos móviles Android. El sistema guía al usuario, explicándole los conceptos básicos de Scrum, sus roles, artefactos y eventos. El personaje explicará las funciones que debe realizar para desarrollar satisfactoriamente un proyecto de software, seleccionando otros personajes para formar a su equipo y posteriormente asignando actividades a cada uno. Para su validación, se realizaron pruebas con 16 estudiantes de un curso universitario de Ingeniería de Software, los cuales recibieron previamente capacitación sobre el proceso Scrum. Se evaluaron conceptos de jugabilidad y usabilidad. Para evaluar la usabilidad, se realizó un cuestionario sobre diferentes aspectos, buscando obtener datos descriptivos del sistema. En cuanto a la jugabilidad, se evaluaron diferentes facetas:

- Jugabilidad intrínseca: Esta faceta es inherente a la naturaleza del sistema en sí y cómo se presenta al jugador. Está relacionada con el diseño de juego. Se analizan las reglas, metas, objetivos, ritmo y otras mecánicas del juego.
- Jugabilidad mecánica: Está relacionada con la calidad del sistema como software. Se asocia con el motor del juego, la fluidez de las escenas, iluminación, sonido, música, gráficos animados, etc.
- Jugabilidad artística: Esta faceta se relaciona con la calidad de la representación artística y estética de los elementos del sistema (gráficos visuales, efectos de sonido, música, historia y narración).
- Jugabilidad interactiva: Se asocia con la interacción del jugador y el desarrollo de la interfaz de usuario. Incluye el diálogo de interacción y los controles del juego.
- Jugabilidad intrapersonal: Se refiere a la perspectiva individual, las percepciones y los sentimientos que el sistema produce en cada jugador. Los resultados muestran que el sistema gamificado fue evaluado positivamente en su jugabilidad intrínseca, mecánica, artística e interactiva. Sin embargo, el sistema requiere mejoras en la jugabilidad interpersonal.

La evaluación general del sistema fue positiva por parte de los jugadores. Se concluyó que la implementación de un sistema gamificado que apoye el aprendizaje del proceso ágil Scrum, es un primer acercamiento a una solución de la problemática de aplicar estrategias diversas que fomenten el reforzamiento de conceptos de desarrollo de software. Los resultados mostraron que el sistema gamificado favorece el aprendizaje de conceptos de Scrum a partir de una buena usabilidad y jugabilidad evaluada [27].

De los 10 trabajos seleccionados, se puede observar que solo 1 ha sido publicado en un journal internacional (Journal of Systems and Software), mientras que el resto de los trabajos se encuentran publicados en diferentes conferencias de alcance internacional. Dos de esos trabajos de conferencia han tenido su espacio en la conferencia más relevante de Ingeniería de Software, denominada ICSE (International Conference

on Software Engineering). La Tabla 2 detalla los lugares donde fueron publicados estos 10 trabajos.

**Tabla 2.** Resumen de los trabajos seleccionados.

| Estudio Primario | Año de Publicación | Lugar de Publicación  | Tipo de Publicación |
|------------------|--------------------|---|---------------------|
| [18]             | 2018               | IEEE World Engineering Education Conference                     | Conferencia         |
| [19]             | 2014               | IEEE International Conference on Technology for Education       | Conferencia         |
| [20]             | 2017               | IEEE Conference on Software Engineering Education and Training  | Conferencia         |
| [21]             | <b>2014</b>        | <b>International Conference on Software Engineering</b>         | <b>Conferencia</b>  |
| [22]             | 2016               | International Conference on Education, Training and Informatics | Conferencia         |
| [23]             | <b>2014</b>        | <b>Journal of Systems and Software</b>                          | <b>Journal</b>      |
| [24]             | <b>2017</b>        | <b>International Conference on Software Engineering</b>         | <b>Conferencia</b>  |
| [25]             | 2017               | IEEE Global Engineering Education Conference                    | Conferencia         |
| [26]             | 2019               | IEEE Conference on Games  | Conferencia         |
| [27]             | 2019               | Iberian Conference on Information Systems and Technologies      | Conferencia         |

Cabe destacar, que en un trabajo previo se presentó el desarrollo de un juego serio para plataformas móviles que permite enseñar Scrum de una manera divertida y motivadora, no solamente para estudiantes de carreras afines de Software sino para desarrolladores que ya estén insertos en la industria [3]. Si bien este trabajo se encuentra publicado, ha sido excluido por los criterios de exclusión de nuestro proceso definido dado que está escrito en el idioma español.

La Figura 2 ilustra la distribución de los 10 estudios primarios seleccionados según su año de publicación en la ventana de tiempo 2014-2019. Se puede apreciar que la mayor cantidad de publicaciones se realizaron en 2014 y 2017, y la tendencia parece crecer en los años venideros.

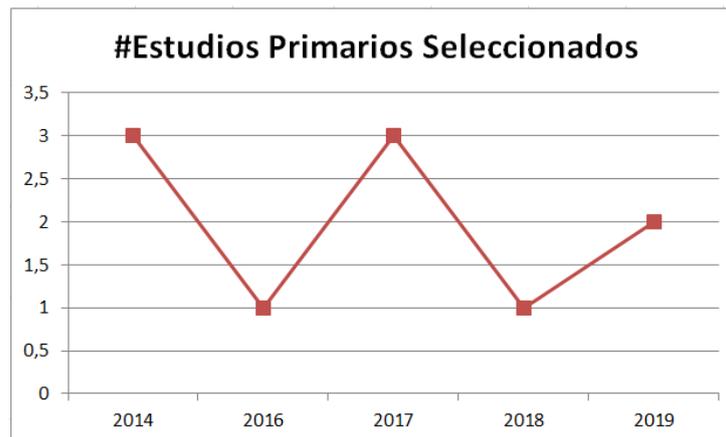


Fig. 2. Distribución de los estudios primarios según su año de publicación.

#### 4 Conclusiones

En este trabajo, presentamos una revisión sistemática de la literatura relacionada con los juegos serios en la educación de Ingeniería de Software, donde seleccionamos 10 artículos de 36 encontrados en 6 bases de datos digitales hasta diciembre de 2019. Organizamos y categorizamos la información obtenida para dar una respuesta a nuestras preguntas de investigación, que tienen como objetivo proporcionar una visión general de la investigación principal en este campo, así como también, identificar los juegos serios actuales enfocados en la enseñanza de Métodos Ágiles.

En este estudio secundario, los artículos sobre juegos serios han sido clasificados, ofreciendo así una línea base inicial para futuras investigaciones dentro del campo de juegos serios para la educación en Ingeniería de Software. Los resultados muestran que los investigadores están más interesados en abordar el diseño y uso de juegos serios para la educación de Ingeniería de Software que el diseño de modelos de simulación para soportar el diseño de juegos serios para la educación de Ingeniería de Software.

En consecuencia, 10 juegos serios para la enseñanza de Métodos Ágiles fueron identificados. La mayoría de estos juegos serios fueron diseñados como juegos que utilizan técnicas de simulación computacional para ofrecer a los estudiantes un entorno de aprendizaje libre de riesgos donde pueden aprender los conceptos y prácticas dentro del framework de los Métodos Ágiles, principalmente Scrum.

En comparación con otras áreas de investigación relacionadas, como juegos para el gerenciamento de proyectos o educación informática, podemos observar que no se han publicado suficientes trabajos en el campo de la educación en Ingeniería de Software.

Por lo tanto, proponemos complementar esta revisión buscando estudios actuales de la literatura gris, es decir, el cuerpo diverso y heterogéneo de material disponible y no sujeto a procesos académicos tradicionales de revisión por pares. Más aún, planificamos estudiar más en profundidad los juegos serios identificados para explorar todo el potencial en la enseñanza de Métodos Ágiles.

## Referencias

1. Abt, C. (2002) *Serious Games*. University Press of America, Lanhan.
2. Zyda, M. (2005) From visual simulation to virtual reality to games. *Computer* 38, 25–32.
3. Rodríguez, G. H., Borromeo, N., & Gonzalez, P. (2019) Un enfoque de Serious Gaming para la enseñanza de Scrum. In I Simposio Argentino de Educación en Informática (SAEI 2019)-JAIIO 48 (Salta).
4. Scott, E., Rodríguez, G., Soria, A., Campo, M. (2014) Are learning styles useful indicators to discover how students use Scrum for the first time?
5. Deterding, S. (2011) *Gamification: Toward a Definition*.
6. Zamzuri, N.H., Kassim, E.S., Shahrom, M. (2010). The role of cognitive styles in investigating E-learning usability, in: international conference on e-education, e-business, e-management, and e-learning.
7. Tortorella, R. Graf, S. (2015) Considering learning styles and context-awareness for mobile adaptive learning.
8. Heredia, A., Colomo-Palacios, R., Amescua-Seco, A. (2015) A sytematic mapping study on software process education. In: *Proceedings of the International Workshop on Software Process Education, Training and Professionalism*, Gothenburg, Sweden.
9. Kosa, M., Yilmaz, M., O'Connor, R., Clarke, P. (2016) Software engineering education and games: a systematic literature review. *J. Univers. Comput. Sci.* 22(12), 1558–1574.
10. Petri, G., von Wangenheim, C.G. (2017) How games for computing education are evaluated? A systematic literature review. *Comput. Educ.* 107, 68–90.
11. Calderón, A., Ruiz, M., O'Connor, R.V. (2018) A multivocal literature review on serious games for software process standards education. *Comput. Stand. Interfaces* 57, 36–48.
12. Tahir, R., & Wang, A. I. (2017) State of the art in game based learning: Dimensions for evaluating educational games. In *European Conference on Games Based Learning* (pp. 641-650). Academic Conferences International Limited.
13. Pedreira, O., García, F., Brisaboa, N., & Piattini, M. (2015) Gamification in software engineering—A systematic mapping. *Information and Software Technology*, 57, 157-168.
14. Calderón, A., & Ruiz, M. (2015) A systematic literature review on serious games evaluation: An application to software project management. *Computers & Education*, 87, 396–422.
15. Ahmadi, M., Trescak, T., Rad, B. B., & Anisi, Y. (2016) *Serious Games in Software Engineering Processes, a Methodical Planning*. *Indian Journal of Science and Technology*.
16. Kitchenham, B., Charters, S. (2007) *Guidelines for Performing Systematic Literature Reviews in Software Engineering*. Keele University and Durham University Joint Report.
17. MacDonell, S., Shepperd, M., Kitchenham, B., Mendes, E. (2010) How reliable are systematic reviews in empirical software engineering? *IEEE Trans. Softw. Eng.* 36(5), 676–687.
18. Alatrística-Salas, H., & Nunez-Del-Prado, M. (2018) Teaching Software Engineering Through Computer Games. In *2018 IEEE World Engineering Education Conference (EDUNINE)* (pp. 1-4). IEEE.

19. Ganesh, L. (2014). Board game as a tool to teach software engineering concept--Technical debt. In 2014 IEEE Sixth International Conference on Technology for Education (pp. 44-47). IEEE.
20. de Almeida Souza, M. R., Constantino, K. F., Veado, L. F., & Figueiredo, E. M. L. (2017). Gamification in software engineering education: An empirical study. *Conference on Software Engineering Education and Training (CSEE&T)* (pp. 276-284). IEEE.
21. Paasivaara, M., Heikkilä, V., Lassenius, C., & Toivola, T. (2014). Teaching students scrum using LEGO blocks. In *Companion Proceedings of the 36th International Conference on Software Engineering* (pp. 382-391).
22. Lee, W. L. (2016). SCRUM-X: An interactive and experiential learning platform for teaching scrum.
23. Von Wangenheim, C. G., Savi, R., & Borgatto, A. F. (2013). SCRUMIA: An educational game for teaching SCRUM in computing courses. *Journal of Systems and Software*, 86(10), 2675-2687.
24. De Souza, A. D., Seabra, R. D., Ribeiro, J. M., & Rodrigues, L. E. D. S. (2017). SCRUMI: a board serious virtual game for teaching the SCRUM framework. *International Conference on Software Engineering Companion (ICSE-C)* (pp. 319-321). IEEE.
25. Schäfer, U. (2017). Training scrum with gamification: Lessons learned after two teaching periods. *Global Engineering Education Conference (EDUCON)* (pp. 754-761). IEEE.
26. Naik, N., & Jenkins, P. (2019). Relax, It's a Game: Utilising Gamification in Learning Agile Scrum Software Development. *IEEE Conference on Games (CoG)* (pp. 1-4).
27. Angarita, L. B., & Hernández, J. A. G. (2019). Sistema gamificado para el aprendizaje del proceso de desarrollo Scrum. *Iberian Conference on Information Systems and Technologies (CISTI)*, Coimbra, Portugal.