

# Edimbrujo: Definiendo de un modelo didáctico para la enseñanza de la Inteligencia Artificial en Juegos

Federico Amigone      Pablo Kogan      Gastón Michelan  
Jorge Rodriguez

{fede.amigone, pablo.kogan,gaston.michelan,j.rodrig }@fi.uncoma.edu.ar

*Grupo de Investigación en Lenguajes e Inteligencia Artificial*  
Departamento de Teoría de la Computación - Facultad de Informática  
UNIVERSIDAD NACIONAL DEL COMAHUE

## Resumen

Comprender el funcionamiento de los sistemas inteligentes resulta esencial para el desenvolvimiento y la comprensión de un mundo donde los agentes inteligentes comienzan a ganar presencia y relevancia en múltiples procesos.

Por tratarse de un área emergente en materia de avances y estratégica respecto a sus implicancias futuras, resulta cada vez más necesario considerar su incorporación en los planes de estudio.

En este contexto se plantea la necesidad de avanzar en la definición de modelos didácticos y plataformas tecnológicas que faciliten la enseñanza y el aprendizaje de la Inteligencia Artificial.

Este artículo presenta un modelo didáctico que busca favorecer la enseñanza de Inteligencia Artificial haciendo uso de una metodología lúdica y creativa, en la que los conocimientos relativos a la construcción de agentes inteligentes sean transmitidos mediante una experiencia activa y lúdica de los estudiantes.

A tal efecto se describe un ecosistema digital construido como plataforma de soporte al proceso de enseñanza y de aprendizaje de las técnicas que sustentan el razonamiento artificial en el contexto de su aplicación a los juegos.

**Palabras Clave:** ENSEÑANZA DE LAS CIENCIAS DE LA COMPUTACIÓN, AGENTES, INTELIGENCIA ARTIFICIAL.

## 1. Introducción

La aplicación de técnicas de Inteligencia Artificial se encuentra presente en muchos aspectos de la interacción de los sujetos con el medio social y económico. Conocer sobre cómo funcionan los sistemas inteligentes es importante para mejorar las posibilidades de comprender un mundo donde los agentes inteligentes están en todos lados.

Esta situación contribuye a que se considere su enseñanza tanto en el ámbito universitario como la educación obligatoria. En este sentido la Inteligencia Artificial es reconocida como área de conocimiento en documentos de recomendaciones curriculares para carreras de Informática [18, 19].

En este contexto se plantea que resulta necesario avanzar en la definición de modelos didácticos y plataformas tecnológicas para facilitar la enseñanza y el aprendizaje de la Inteligencia Artificial. Varios estudios proponen prestar atención las actividades y dispositivos que a las que las personas les resultan familiares, como los juegos y las prácticas colaborativas[5, 3].

El modelo didáctico propuesto en este trabajo parte del supuesto de que el área de conocimiento de referencia presenta grandes oportunidades para construir propuestas de enseñanza basadas en el proceso de enseñanza que incorporen el juego y la construcción colaborativa de Agentes Inteligentes como perspectivas metodológicas. Se busca desarrollar experiencias formativas que resulten motivadoras y con excelentes oportunidades para plantear dinámicas grupales que constituyan experiencias de aprendizaje exitosas en el contexto de la enseñanza de la Inteligencia Artificial [3].

El modelo propuesto emerge del binomio Inteligencia Artificial + aprendizaje. Respecto a éste último, la educación formal se caracteriza por un modelo estructurado de aprendizaje sistemático conducido por un conjunto de directivas curriculares, contenidos y metodologías dirigidas tanto a estudiantes como a docentes. Sin embargo, resulta atendible el argumento de que este no es el modo natural en que el Ser Humano aprende [7].

Cuando se consideran los mecanismos que construyen la adquisición del saber en los niños, resulta evidente la efectividad de los juegos como contexto de aprendizaje. Los niños, consistentemente, muestran señales de placer al adquirir conocimiento al experimentarlo como un juego, un excitante viaje de exploración y diversión. No existe evidencia que señale que tal relación se pierde en edades maduras [10].

Por otro lado, uno de los aspectos más complejos que enfrentan los educadores consiste en abordar efectivamente la experiencia de enseñar con un grupo heterogéneo y diverso de estudiantes. Cada educando comienza el proceso de aprendizaje con niveles muy diversos de motivación, habilidades y demás construcciones subjetivas y propias respecto de la significación del contenido disciplinar. En general, los modelos teóricos convergen en una conclusión relevante: la motivación, mas que ningún otro factor, es el más efectivo predictor de éxito académico [3].

Este hecho basal del aprendizaje constitu-

ye, desde el punto de vista del aprendizaje lúdico, una oportunidad: más allá de la étnia, extracción social o económica o cualquier otro background social, las personas entienden y disfrutan el lenguaje del juego [2].

En contraposición con el concepto formal de aprendizaje, resulta necesario mencionar lo que podría ser descripto como un proceso más informal. La educación informal refiere a experiencias de la vida real, donde las aptitudes individuales, valores y habilidades son naturalmente adquiridas a través de la experiencia diaria [15].

Los procesos conductores de la educación informal le dan a los estudiantes la oportunidad de abordar un proceso de aprendizaje mediante la participación proactiva y la experimentación de sistemas flexibles de aprendizaje [4].

Este proceso informal tiende a desarrollar más favorablemente las habilidades personales de los estudiantes que los rígidos modelos formales, como por ejemplo, liderazgo, resiliencia, disciplina, responsabilidad, trabajo en equipo, planeamiento, organización, etc. En consecuencia, el proceso informal se percibe como mucho más estimulante que los modelos rígidos.

Los modelos lúdicos e informales están cambiando la manera de pensar la construcción del conocimiento y los procesos de aprendizaje. Esta propuesta parte de esta coyuntura e intenta proponer un modelo lúdico para la enseñanza de la Inteligencia Artificial.

Este artículo presenta el diseño de una plataforma para la enseñanza de las Ciencias de la Computación, en particular, en el contexto del área de conocimiento Inteligencia Artificial: Edimbrujo.

El desafío requiere desarrollar dos productos que actúan en conjunto para favorecer el proceso de enseñanza y de aprendizaje de conceptos y prácticas relacionados a la Inteligencia Artificial.

El modelo didáctico parte de la capacidad con que cuentan los ambientes lúdicos para mejorar las posibilidades de aprender con-

ceptos y prácticas sobre Inteligencia Artificial [1, 5]. Partiendo de esa premisa, se pone en la centralidad la construcción de una vivencia didáctica haciendo uso de un framework tecnológico que permite a los estudiantes construir y disfrutar de una experiencia de laboratorio orientada al desarrollo de un artefacto computacional haciendo uso de técnicas de Inteligencia Artificial[3].

La mejora de la motivación y retención del interés por parte del sujeto de aprendizaje contribuye a aumentar el rendimiento académico y sus habilidades personales como resultado del aprendizaje a través del juego[10].

El desafío no es menor, ya que la conformación de un ecosistema lúdico en conjunción con técnicas de Inteligencia Artificial requiere articular una arquitectura heterogénea tanto en sus componentes fundantes como en la tecnología utilizada.

Respecto del framework tecnológico sobre el cual opera el modelo, es importante notar que pese a la evidente complejidad que puede representar convocar al uso tecnologías heterogéneas, el motor emocional que tracciona el proceso de aprendizaje en el contexto de la construcción de juegos de competencia, contribuye a consolidar una experiencia exitosa. Es decir, el modelo didáctico actúa como el propulsor lúdico capaz de vehiculizar el proceso a través de una curva de aprendizaje leve. Este resultado se explica también gracias a la abstracción que propone el framework: se oculta la complejidad del andamiaje que resuelve la arquitectura y se propone como única complejidad la construcción del agente. Por otro lado, ofrece un protocolo de interoperatividad entre el agente que construye el alumno y el resto de la plataforma. Dicho protocolo es de características simples, abierto y consolidado y su abordaje didáctico ya ha sido probado en las experiencias en las que intervino Hornero [8].

Este documento está estructurado de la siguiente manera. En la Sección 2, se presentan trabajos que pueden ser relevantes para esta

propuesta. El modelo didáctico y la plataforma que lo soporta se describen en detalle en Sección 3, en la Sección 4 se expone la experiencia desarrollada y los resultados obtenidos. Finalmente, en la Sección 5 se presentan las conclusiones construidas y líneas de trabajo futuro.

## 2. Trabajos relacionados

Desde que Alan Turing sugirió que los juegos podían ser automáticamente jugados por máquinas usando algoritmos lógicos, estos se ubicaron como lugar para experimentar distintos conceptos de la Inteligencia Artificial [23]. Diversos juegos han emergido como opciones potencialmente válidas para estimular la motivación, favorecer la espontaneidad y aportar la flexibilidad e interacción necesarias para un proceso de aprendizaje efectivo.

Los ejemplos más representativos en educación son juegos de tableros, por ejemplo:

- Backgammon, utilizados para enseñar métodos de exploración [14].
- Juego de Damas, utilizado para desarrollar búsqueda técnicas basadas en la resolución de problemas [20].
- Ta-Te-Ti, utilizado para la poda mínima y máxima alfa-beta [13].
- Rompecabezas, utilizado para búsquedas basadas en estado [11].

Esos ejemplos no agotan en lo más mínimo el amplio abanico de experiencias que se han realizado. Un buen ejemplo lo constituye The Open Racing Car Simulator, una plataforma de código abierto y altamente portable, ha sido utilizada tradicionalmente como un juego de autos para enseñar principios mecánicos en la Universidad de Illinois [6].

La Universidad Nacional de Maynooth ha organizado ligas de competencia de Bots, con el objetivo de enseñar lenguajes de programación [16]. En ellas, los estudiantes son

desafiados a diseñar agentes inteligentes, llamados Bots, para competir entre sí intentando imitar conductas humanas.

Con el objetivo de utilizar sistemas de Inteligencia Artificial como plataformas de promoción de la educación y la investigación en el campo, se han registrado muchas competencias nacionales e internacionales. Por ejemplo, la Universidad de Standford ha utilizado en su conferencia AAAI (Association for the Advancement of AI) un framework llamado General Game Playing durante las competiciones de verano[9].

### 3. Propuesta

Edimbrujo es una plataforma sobre la que es posible programar Agentes Inteligentes que interactúen en un ambiente lúdico creativo, cuyo comportamiento se encuentra restringido al Mundo o Historia asociado a Edimbrujo. Sobre ella opera el Modelo Didáctico desarrollado en este trabajo que se basa en la colaboración grupal para la competencia entre pares cuya finalidad persigue la enseñanza de técnicas de Inteligencia Artificial utilizando la programación de Agentes Inteligentes[1].

En esta sección se describen el modelo didáctico, los componentes arquitecturales que posibilitan el funcionamiento de Edimbrujo, las herramientas de diseño que permiten construir los escenarios y personajes y finalmente nos centraremos en el objetivo primario de la propuesta que consiste en la experiencia formativa de desarrollar Agentes Inteligentes.

#### 3.1. La Historia

Edimbrujo es tanto un juego como la plataforma tecnológica que lo soporta. Ahora bien, todo juego tiene una historia asociada. En la historia o relato, se define lo que en el argot lúdico suele denominarse Mundo o Historia, esto es, la contextualización de los personajes y su misión en la historia del juego[12].

En el caso de Edimbrujo la Historia podría resumirse así:

*Con el debacle del Imperio Romano las ciudades más lejanas quedan totalmente abandonadas, algunas esconden antiguos tesoros. Entre ellas, la más famosa es Edimbrujo ubicada en el fin del mundo donde no hay más tierra y solo existe el mar.*

*Se trata de una ciudad maldita, una ciudad habitada por todo tipo de seres crapulentos, no obstante los antiguos tesoros ocultos llaman la atención de aventureros con sed de fama y riquezas. Se rumorea que una vez al año, durante el 31 de octubre, la ciudad cobra vida y las más grandes riquezas afloran.*

#### 3.2. El Modelo Didáctico

El modelo didáctico propuesto se basa fundamentalmente en un proceso lúdico, creativo y grupal basado en competencias entre pares sobre la plataforma Edimbrujo.

A tal efecto, se organiza inicialmente una instancia de definición de equipos que busca avanzar en la construcción de una identidad social que fortalezca el sentido colectivo y la identificación con el par: el grupo debe identificarse con un nombre y ciertas estrategias de supervivencia.

Luego se propone una instancia creativa en el marco de Edimbrujo que ubique en el centro el aspecto constructivo de la experiencia. Esta es una oportunidad para que el grupo desarrolle una dinámica creativa y desestructurada, situación que en líneas generales los posiciona del lado opuesto al rol que suelen tener en su interacción con los juegos: de jugadores pasan a ser diseñadores. Así que se estimula la producción creativa de personajes, nombres, intereses, objetivos, peligros y dificultades. Como salida de este proceso es deseable obtener en forma documentada las siguientes producciones:

- Nuevas Reglas
- Nuevos personajes, héroes y villanos.
- Nuevas misiones.

- Nuevos Diseños de escenarios.
- Aportes al relato y ampliación de la historia.

En cualquier caso, el modelo alienta instancias en las que cada estudiante pueda valorar su participación como co-creador de una historia, a través de la generación de aportes del mundo Edimbrujo.

Luego de la participación creativa, los grupos deberán implementar Agentes Inteligentes que insuflen vida a los nuevos personajes diseñados, haciendo uso de técnicas de Inteligencia Artificial y respetando las reglas del mundo Edimbrujo.

El proceso de construcción de Agentes Inteligentes cuenta con una serie de características que favorecen el la eficacia formativa de la competencia[22, 5]:

- El desarrollo es independiente del lenguaje de programación y el sistema operativo.
- El juego es divertido para los jugadores humanos.
- El código de las competencias de años anteriores es de fuente abierta.
- Las reglas de la competencia son claras y estables.
- Es factible para los estudiantes desarrollar una solución.
- Los docentes se constituyen como grupos de soporte y asesoramiento.

Una vez concretada la implementación de las estrategias y técnicas en los Agentes Inteligentes y fundamentados los criterios y decisiones por parte de los grupos desarrolladores, tiene lugar La Competencia. A tal efecto existe un Reglamento de Edimbrujo, según el cual:

- Debe propiciarse el mejor clima competitivo posible.

- Puede disponerse de una instancia de precalentamiento: se realiza un test de competencia durante 5 minutos. Se inicia el motor Edimbrujo y cada grupo arranca 1 Agente de cada personaje.
- Sobre La Competencia en sí: Se inicia el motor Edimbrujo y cada grupo arranca 1 agente de cada personaje. Los equipos no pueden tocar sus computadoras durante la competencia con excepción de que algún Agente se cuelgue. Los docentes actúan como árbitros analizando el log del servidor y con facultad de multar descontando vida o descalificando equipos si detectan acciones fraudulentas. A los 5 minutos termina la competencia y gana el equipo que logre acumular más vida.

### 3.3. El framework

Además de la Historia, el modelo requiere de un entorno de trabajo que lo viabilice en términos constructivos. Se requiere por tanto la definición y construcción de un andamiaje tecnológico que actúe como framework de trabajo. El diseño del entorno de trabajo no debe ser materia de investigación o elaboración para los estudiantes, ya que el núcleo del significativo disciplinar queda circunscripto a la aplicación de técnicas de Inteligencia Artificial más que a las cuestiones de interoperatividad arquitectural.

Sin embargo, sí es necesario introducirlo y presentarlo para su uso. Los componentes principales del framework son los siguientes:

- Arquitectura.
- Diseño de los escenarios y desarrollo de sprites.
- Servidor REST.
- Servidor Socket.io.
- Cliente Web y Móvil.

- Agentes Autónomos en python, java y php.
- Infraestructura.

Resulta fundamental que en éste punto quienes guían el proceso de enseñanza atiendan al sentido de uso del framework: cada ítem señalado posee un peso teórico y práctico específico cuya complejidad no se desea abordar en el ámbito de la experiencia formativa más que para fines prácticos.

Los conceptos, protocolos y componentes arquitecturales de software se presentan a los estudiantes con un criterio pragmático evitando describir aspectos teóricos relacionados a la construcción del framework. Esto es, con un nivel operativo suficiente que posibilite los objetivos programáticos del modelo pero con la menor implicancia técnica posible a efectos de no contaminar de contenidos el núcleo del signifiante disciplinar, que como dijimos, debemos mantener circunscriptos a las técnicas de Inteligencia Artificial.

### 3.4. Componentes de la Plataforma Edimbrujo

La plataforma Edimbrujo posee una arquitectura cliente servidor que hace uso de determinados componentes y protocolos. La figura 1 muestra la arquitectura.

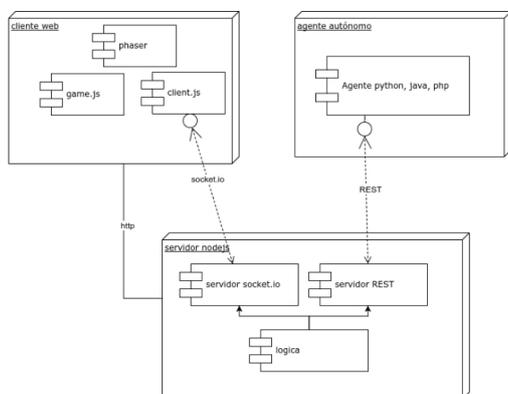


Figura 1: Arquitectura Edimbrujo

#### 3.4.1. Motor Edimbrujo

Para el armado del componente arquitectural que hace las veces de servidor o motor de Edimbrujo, se utiliza el lenguaje JavaScript sobre NodeJs. El motor agrupa la lógica del juego, el servidor socket.io encargado de la comunicación con clientes web y el servidor REST que se ocupa de la comunicación con agentes autónomos. Para soportar la distribución al servidor NodeJs se adosaron la siguientes librerías:

- Socket.io : para permitir la comunicación bidireccional en tiempo real basada en eventos a través de la creación de web sockets con los clientes web.
- Express: un framework que provee un robusto set de características web. En este caso utilizada para linkear el index.html, librerías externas de estilos css y funcionalidades en JavaScript. También es el encargado de publicar los servicios REST.
- HTTP: para el uso del protocolo HTTP.
- System-sleep: para demorar, detener o “hacer dormir” procesos por un tiempo determinado.
- Lógica de aplicación: creada por los estudiantes de la materia con el fin de proveer los métodos para mantener el ambiente y las reglas del juego.

#### 3.4.2. Clientes Edimbrujo

Edimbrujo es accesible a usuarios finales, es decir jugadores, través de un cliente web. Pero también resulta necesario un mecanismo de conexión entre los Agentes Autónomos y el motor Edimbrujo. Para ello se utilizan los servicios web REST publicados por el motor y que son consumidos por los procesos Agentes Autónomos desarrollados en php, python y/o java.

### 3.4.3. Escenarios Edimbrujo

Para la creación del mundo se utiliza el programa Tiled Map Editor, un editor libre y de código abierto que permite diseñar el mundo a través de capas de objetos, colocando directamente imágenes sobre un lienzo y luego exportando el resultado final como un archivo CSV o JSON, el cual será utilizado en Phaser para representar el mundo[21].

Phase es un framework open source, escrito íntegramente en Javascript, que se utiliza para construir juegos móviles o de escritorio en HTML5. Este framework provee una serie de herramientas que aceleran el desarrollo y ayudan a manejar las tareas genéricas necesarias para completar los juegos, como lo son la creación y manejo de escenarios, controles, movimientos y animaciones[17].

Para diseñar el mundo se utiliza una imagen compuesta por un conjunto de imágenes, todas del mismo tamaño. El tamaño de las partes se mide en patrones, los cuales se definirán del mismo tamaño que las imágenes del conjunto, de manera que el archivo que contenga el resultado final contendrá un arreglo del mismo tamaño del mundo, donde cada elemento indicará qué imagen del conjunto hay en esa posición. Cada imagen puede utilizarse para armar combinaciones simples o complejas del escenario.



Figura 2: Mundo Edimbrujo

Para completar la creación del escenario, es necesario diseñar sprites o avatares de los personajes. Naturalmente el diseño que se logre debe tener su correspondiente correlato

con el tipo de acciones que cada personaje podrá ejecutar. El diseño debe relacionarse tanto con la historia del juego como con el rol que desempeña cada personaje.



Figura 3: Personajes Edimbrujo

### 3.4.4. Servicios REST

Se utiliza un pool de servicios REST como canal agnóstico de intercambio de datos entre el cliente autónomo y el motor Edimbrujo. Este protocolo de comunicación se basa en el desarrollo del gestor de torneo de programación HORNERO [8]. Esto permite una interfaz muy débilmente acoplada entre los agentes y el motor, cualidad importante a la hora de desarrollar Agentes en cualquier lenguaje. Entre los varios métodos publicados se encuentran:

- inicio: Este método recibe por GET los parámetros necesarios para comenzar a jugar. Por otra parte, se envía el Rol del jugador, la clase del personaje seleccionado y el nombre del equipo al que pertenece.
- mover: Éste método recibe por GET una serie de parámetros entre los cuales destaca el parámetro token, que es un identificador recibido por el agente al iniciarse y el parámetro número que representa la dirección del movimiento que se quiere realizar.
- atacar: Si un Agente perteneciente a otro equipo se encuentra en el rango del Agente que ataca, la fuerza del agente que ataca es descontada a la vida del Agente enemigo. Si como resultado a este golpe la vida del Agente enemigo es 0 o menos, este se reubica.

### 3.4.5. Comunicación con el cliente

WebSockets es una tecnología que se utiliza para crear canales de comunicación bidireccional en aplicaciones cliente-servidor, permitiendo así el intercambio de información entre estas dos entidades a través de un único socket TCP.

La tecnología WebSocket es un estándar de HTML5 que se puede utilizar en la mayoría de los servidores web, entre ellos Apache y Node.js. El protocolo WebSocket se encuentra definido en el documento RFC 6455.

La implementación de WebSockets varía dependiendo del lenguaje de programación que se utiliza, principalmente en el servidor, sin embargo la estructura de la aplicación es la misma. El objetivo es crear un WebSocket en el cliente y utilizarlo para enviar datos al motor Edimbrujo. La estructura de los datos se define a partir de un lenguaje de interfaz, generalmente se utiliza JSON debido a que en el cliente los WebSockets se implementan mediante Javascript.

En el servidor se deben implementar las funciones que procesan la información enviada desde el cliente a través del WebSocket. Como el canal de comunicación es bidireccional el servidor podría enviarle información al cliente sin la necesidad de que exista un request por parte de este último, esta característica define a WebSocket como una herramienta muy importante para implementar juegos multiplayer.

En Edimbrujo los WebSockets se implementan mediante la librería Javascript socket.io en un ambiente basado en Node.js. Esta librería posee las funciones necesarias para la transferencia de datos entre el cliente y el servidor en tiempo real.

## 3.5. El Agente Autónomo

Llegamos así al componente más valioso en términos didácticos para el modelo propuesto sobre Edimbrujo. El Agente debe ser desarrollado por los grupos en PHP, Python o Java y debe contar con dos partes: Comunicación

REST con Edimbrujo y Estrategia

- Comunicación REST con Edimbrujo
- Estrategia

La Estrategia es la dimensión del Agente donde se aplican las técnicas de Inteligencia Artificial, razón por la cual, debe comportar la mayor atención y esfuerzos de generación. A tal efecto, cada grupo debe defender los criterios y técnicas que utilizaron.

En general, el Agente desarrollado decide que acción realizar a partir del estado actual generando cada uno de los posibles estados próximos y eligiendo aquella acción que lo lleve al mejor estado. Para el caso de Edimbrujo, este estado se determina a partir de la distancia entre la próxima posición del jugador y un bono y/o enemigo más cercano y la vida actual del jugador. Este procedimiento se basa en el algoritmo de búsqueda local Hill Climbing. Al producirse un evento, tal como el de encontrar un enemigo o un premio, se dispara el análisis de reglas y las consideraciones de supervivencia del Agente.

## 4. Experiencia

La primer experiencia con la plataforma Edimbrujo se realizó en el marco de la cursada de la materia optativa Inteligencia Artificial en Juegos, de la Licenciatura en Ciencias de la Computación de la Facultad de Informática de la Universidad Nacional del Comahue durante el segundo cuatrimestre del año 2017.

El enfoque propuesto hizo uso de Edimbrujo para instrumentar una instancia de competencia grupal en la que las estrategias de desarrollo del juego fueran conducidas por la ejecución de Agentes programados por los grupos de estudiantes.

La experiencia es evaluada positivamente ya que los objetivos perseguidos se cumplieron satisfactoriamente, es decir los estudiantes lograron construir aprendizajes sobre técnicas de Inteligencia Artificial en Jue-

gos. Esto quedó experimentalmente constatado en el momento de la competencia, instancia en la que la programación de técnicas IA logró modelar comportamientos esperados en los agentes y se logró así una experiencia de aprendizaje lúdica de gran impacto.

## 5. Conclusiones y trabajos futuros

La perspectiva didáctica apoyada principalmente en el aprendizaje colaborativo y la construcción de artefactos computacionales aporta modelos teóricos para para la estructuración de las experiencias educativas potenciando el desarrollo de instancias de diálogo enfocando a los grupos sobre una problemática común.

La estructuración de la propuesta a partir del desarrollo de agentes, además de ser una opción didáctica, se constituye en contenido a aprender. La participación activa en la construcción de conocimientos, en particular los vinculados al trabajo colaborativo y la gestión de proyectos, en el contexto de un proceso participativo integran el cúmulo de conocimientos relevantes a la formación profesional. De este modo se busca mejorar la formación de profesionales capaces de desarrollar proyectos, investigar, evaluar alternativas y resolver problemas en forma conjunta.

A partir de revisar los resultados surgidos de la experiencia de enseñanza y aprendizaje concretada, se considera que se trata de una estrategia efectiva para favorecer la construcción de conocimiento sobre Inteligencia Artificial en Juegos y acerca de Inteligencia Artificial en general.

Un aspecto a considerar en la concreción de este tipo de experiencias educativas está relacionado con lograr que la competencia sea suficientemente independiente del lenguaje de programación[22]. Edimbrujo permite la interacción con Agentes Autónomos desarrollados en php, phyton y java, como trabajo futuro está previsto la ampliar el con-

junto de lenguajes con intención de hacer posible este tipo de aprendizajes a más estudiantes, no sólo en el ámbito universitario, además en la educación secundaria.

## Referencias

- [1] C. C. Ada, T. M. Sanguino, S. Alacocer, A. Borrego, A. Isidro, A. Palanco, and J. Rodríguez. Classroom to mobile robots competition arena: An experience on artificial intelligence teaching.
- [2] J. A. Azriel, M. J. Erthal, and E. Starr. Answers, questions, and deceptions: What is the role of games in business education? *Journal of Education for Business*, 81(1):9–13, 2005.
- [3] J. Carpio Cañada, T. Mateo Sanguino, J. Merelo Guervós, and V. Rivas Santos. Open classroom: enhancing student achievement on artificial intelligence through an international online competition. *Journal of Computer Assisted Learning*, 31(1):14–31, 2015.
- [4] B. Chen and T. Bryer. Investigating instructional strategies for using social media in formal and informal learning. *The International Review of Research in Open and Distributed Learning*, 13(1):87–104, 2012.
- [5] F. Chesani, A. Galassi, P. Mello, and G. Trisolini. A game-based competition as instrument for teaching artificial intelligence. In *Conference of the Italian Association for Artificial Intelligence*, pages 72–84. Springer, 2017.
- [6] B. D. Coller and D. J. Shernoff. Video game-based education in mechanical engineering: A look at student engagement. *International Journal of Engineering Education*, 25(2):308, 2009.
- [7] C. Z. Dib. Formal, non-formal and informal education: concepts/applicability.

- In *AIP conference proceedings*, volume 173, pages 300–315. AIP, 1988.
- [8] C. C. Fracchia, P. Kogan, and S. Amaro. Competir + Motivar + Hornero = aprender programación. *Revista Iberoamericana de Educación en Tecnología y Tecnología en Educación*, 18:19–29, 2016.
- [9] M. Genesereth, N. Love, and B. Pell. General game playing: Overview of the aaai competition. *AI magazine*, 26(2):62, 2005.
- [10] D. Kirkland and F. O’Riordan. Games as an engaging teaching and learning technique: Learning or playing. In *1st Irish Conference on Engaging Pedagogies, Dublin, Ireland*, 2008.
- [11] Z. Markov, I. Russell, T. Neller, and N. Zlatareva. Pedagogical possibilities for the n-puzzle problem. In *Frontiers in Education Conference, 36th Annual*, pages 1–6. IEEE, 2006.
- [12] K. L. McClarty, A. Orr, P. M. Frey, R. P. Dolan, V. Vassileva, and A. McVay. A literature review of gaming in education. *Gaming in education*, pages 1–35, 2012.
- [13] D. Michulke and S. Schiffel. Distance features for general game playing. In *Proceedings of the IJCAI Workshop on General Intelligence in Game-Playing Agents (GIGA), Barcelona*, pages 7–14, 2011.
- [14] D. G. Moursund. Brief introduction to educational implications of artificial intelligence, 2006.
- [15] M. Novosadova, G. Selen, A. Piskunowicz, S. Mousa, S. Suoheimo, T. Radinja, and P. Reuter. The impact of non formal education on young people and society. *Non formal education book*, pages 1–58, 2007.
- [16] J. O’Kelly and J. P. Gibson. Robocode & problem-based learning: a non-prescriptive approach to teaching programming. In *ACM SIGCSE Bulletin*, volume 38, pages 217–221. ACM, 2006.
- [17] Phaser. <https://phaser.io/>.
- [18] Red UNCI. Documento de recomendaciones curriculares de la redunci 2014-2015, 2015.
- [19] M. Sahami, A. Danyluk, S. Fincher, K. Fisher, D. Grossman, E. Hawthorne, R. Katz, R. LeBlanc, D. Reed, S. Roach, et al. Computer science curricula 2013: Curriculum guidelines for undergraduate degree programs in computer science. *Association for Computing Machinery (ACM)-IEEE Computer Society*, 2013.
- [20] N. R. Sturtevant. An analysis of ict in multi-player games. In *International Conference on Computers and Games*, pages 37–49. Springer, 2008.
- [21] Tiled Map Editor. Home Page. <http://www.mapeditor.org/>.
- [22] J. Togelius. How to run a successful game-based ai competition. *IEEE Transactions on Computational Intelligence and AI in Games*, 8(1):95–100, 2016.
- [23] A. Turing. Computing intelligence and machinery. *Mind*, 59(2236), 1950.