

# Specification of a Managing Agent of Emergent Serious Games for a Smart Classroom

J. Aguilar, *Member, IEEE*, J. Altamiranda, and F. Díaz

**Abstract**—This article presents the specification of a Managing Agent of emergent serious games for a smart classroom. The Managing Agent of Emergent Serious Games is a manager of emergent serious games, which seeks to adapt emergent serious games to the subject that is developed in the smart classroom, with the aim of motivating the acquisition of knowledge and skills in students during their learning processes. The introduction of emergent serious games in a smart classroom allows incorporating a new type of teaching resource in the learning dynamic, which allows enriching with new strategies the learning process.

**Index Terms**—Serious Games, Smart Classroom, Emerging Games, Smart Environment.

## I. INTRODUCCIÓN

UN juego Serio Emergente (JSE) proporciona un ambiente de motivación, entretenimiento y auto-fortalecimiento, para que los jugadores aprendan haciendo desde sus propios errores, gracias a desafíos adecuados y a una retroalimentación autónoma constante. Un JSE mezcla dos áreas de videojuegos, los Juegos Serios (JS) y los Juegos Emergente (JE). Los JS establecen que un juego tiene un propósito específico, que puede estar relacionado con el aprendizaje o con la comprensión de un tema complejo; y los JE comprenden juegos cuyos comportamientos surgen a partir de las interacciones espontáneas de los elementos vinculados al juego (los jugadores, los personajes, etc.), sin leyes explícitas.

Algunos de los trabajos relacionados a este artículo son: en [1] se enuncia que los Entornos de Aprendizaje Inteligente basados en JS, complementan las plataformas inteligentes educativas, tales como los Sistemas de Tutoría Inteligentes, para definir entornos de aprendizaje inteligentes. En [2] buscan ampliar el conocimiento y la comprensión de la práctica educativa centrada en JS, específicamente, en simulaciones por computadora para estudiar modelos de emprendimiento.

En [3] se analiza el uso de JS en el área de la salud, y particularmente, en el campo de enfermedades neurodegenerativas como la enfermedad de Alzheimer. En [4] se usan los algoritmos Beam-ACO y Ant-Q, en un JS que resuelve problemas de optimización como el de vendedor ambulante.

J. Aguilar está en el CEMISID, Universidad de Los Andes, Mérida Venezuela and Universidad EAFIT, Medellín, Colombia. (e-mail: aguilar@ula.ve).

J. Altamiranda está en el CEMISID, Facultad de Ingeniería, Universidad de Los Andes, Mérida, Venezuela. (e-mail: altamira@ula.ve).

F. Díaz está en el CEMISID, Facultad de Ingeniería, Universidad de Los Andes, Mérida, Venezuela. (e-mail: franjdv@gmail.com).

En el área de JE, uno de los principales trabajos es el libro editado por Sweeter [5], donde se establecen las bases conceptuales de los JE. Por otro lado, en [6] se presentan cinco enfoques de narrativas emergentes: crear mundos, incorporar la experiencia del jugador dentro del juego, establecer espacios de múltiples opciones de decisión en el juego, permitir la incertidumbre con situaciones inesperadas, y posibilitar la co-autoría tal que el jugador sea responsable de la historia y la dirección que toma el juego. A su vez, en [7] proponen Metrópolis, un JE que parte de la premisa que las ciudades son auto-gestionadas por decisiones tomadas en conjunto por sus habitantes (jugadores). Debido a estas decisiones emergen patrones urbanísticos. En [8] se propone una extensión a Metrópolis, introduciendo mecanismos emergentes a la dinámica del juego, para que se adapte a sus jugadores. También, en [9] combinan análisis de video y minería de datos educativos, para identificar estrategias cognitivas que surgen a través del juego. Finalmente, en [10] establecen narrativas de la experiencia personal de los jugadores, que se transforman según las interacciones entre los jugadores y el sistema, generando un mundo de comportamientos y eventos impredecibles.

Alternativamente, en [11] examinan los resultados derivados de la combinación de la inteligencia artificial con entornos virtuales, para producir Entornos Virtuales Inteligentes (EVI). En [12] presentan un estado del arte sobre juegos digitales de aprendizaje, para estudiantes de 6 a 25 años en diferentes contextos. Por otro lado, en [13] presentan un entorno de educación inteligente que reconoce los elementos presentes en un ambiente educativo, para definir métodos de enseñanza centrados en el estudiante.

Por otro lado, en [14, 15, 16] explican que un Ambiente Inteligente (AmI) en la educación debe integrar diferentes aspectos relacionados con el proceso de enseñanza, como el perfil de los estudiantes, el paradigma pedagógico utilizado, entre otras cosas, en tiempo real. En esos trabajos se caracteriza un Salón de Clases Inteligentes (SaCI), el cual es un aula inteligente cuyos componentes se auto-organizan para facilitar procesos de aprendizaje [17]. Específicamente, el SaCI es definido como un AmI, modelado usando Sistemas Multiagentes (SMA) [18], y ayuda a la formación de los estudiantes, a través de actividades introducidas por sus componentes, algunos de los cuales pueden estar en la nube [15, 16]. SaCI define dichas actividades, acorde con el contenido que se está impartiendo en él, y con el perfil de los estudiantes. Los componentes del SaCI son modelados como agentes [14].

En general, en la revisión de la literatura se consiguieron sistemas inteligentes de gestión de juegos, pero ninguno basado en agentes. Además, se limitan a ver a los JS como objetos de

aprendizaje en repositorios, con imposibilidad de adaptarlos al entorno. Lo más cercano a nuestra propuesta es el sistema clasificador gestor de JS propuesto en [18]. La principal contribución de este artículo es que propone un sistema gestor de JSEs basado en agentes, que permite la adaptación de los mismos a los entornos de aprendizajes.

En este sentido, el artículo presenta la especificación del Agente de Juegos Serios Emergentes (AJSE) para el SaCI. El AJSE permite incorporar en el SaCI los beneficios que ofrecen los JS, tales como [19]: a) lograr un aprendizaje más afectivo-cognitivo, b) enfrentar los problemas a partir de simulaciones, c) hacer lectura interpretativa de la realidad desde un espacio-tiempo de innovación en el que es posible cometer errores y al mismo tiempo aprender de ellos. Para ello, el AJSE permite las siguientes emergencias en un juego [7, 20, 21]: de estrategias, de secuencias, y de propiedades.

## II. ASPECTOS TEÓRICOS

### A. Salón de Clase Inteligente (SaCI)

SaCI es una propuesta presentada en [14], para caracterizar un aula inteligente mediante el paradigma de SMA. Por otro lado, SaCI utiliza un *middleware* reflexivo autónomo para entornos de aprendizaje inteligente, llamado AmICL, propuesto en [14, 16, 22, 23, 24].

En particular, el SaCI es un AmI que plantea un modelo centrado en los estudiantes, para ayudar en los procesos de aprendizaje a través de dispositivos y programas que colaboran y apoyan su auto enseñanza. Para ello, las aulas inteligentes poseen diferentes tipos de elementos, tanto de *hardware* (tales como cámaras, pizarras, tabletas inteligentes, entre otros), como de *software* (Sistemas Tutoriales Inteligentes (ITS), Entornos Virtuales de Aprendizaje (VLE), entre otros) modelados como agentes.

AmICL explota la nube de aprendizaje (*C-Learning*), para proporcionar servicios en la nube a los agentes presentes en el SaCI [16], para adaptar y optimizar el proceso de enseñanza-aprendizaje. Según [15], la arquitectura AmICL está compuesta por las siguientes capas:

- 1) **Capa Física (PL):** comprende los componentes de *hardware* y *software* presentes en el SaCI.
- 2) **Capa de Gestión del SMA (MMAL):** Esta capa es una adaptación del estándar FIPA [16], que gestiona los agentes en el SMA.
- 3) **Capa de Gestión de Servicios (SML):** permite la administración de servicios educativos (locales y en la nube).
- 4) **Capa de Gestión AmICL Lógica (ILL):** caracteriza todas las aplicaciones y personas presentes en el SaCI. En esta capa existen diferentes tipos de agentes, por cada aplicación/persona: SPA (Agente de Perfil de Estudiante), TA (Agente Tutor), entre otros. En esta capa estará el agente de nuestra propuesta, el AJSE.
- 5) **Capa de Gestión AmICL Física (IPL):** representa los elementos físicos en el SaCI (gemelos digitales). Así, cada dispositivo físico se caracteriza por un agente, llamado Agente Dispositivo.

### B. Juegos Serios Emergentes

Los JS están diseñados específicamente para lograr un objetivo principal distinto al de la pura diversión [19], ya que no se usan para entretener al jugador. Esta característica es un valor agregado, que se utiliza para lograr aumentar la calidad en el proceso de aprendizaje. Actualmente, están recibiendo un interés creciente en la educación, siendo utilizados con éxito en otras áreas como: salud, políticas públicas, planificación urbana, entre otras [25, 26].

Por otro lado, un JE, de acuerdo a [7, 20], aparece cuando un conjunto relativamente simple de reglas que conducen a estrategias de juego complejas, considerando lo que haga el jugador y el entorno para adecuarse. Según [5, 6 10, 20, 21], las formas de emergencia en un juego se pueden expresar por la aparición de:

- 1) **Estrategias:** se generan nuevas tácticas y logísticas en el juego.
- 2) **Secuencias:** se crean nuevos escenarios, tramas o temáticas en los juegos.
- 3) **Finales:** cambia el final de juego.
- 4) **Propiedades:** cambian las características en los objetos que los componen.
- 5) **Utilidad:** cambia el uso del JSE, es decir, si se utilizará para explicar conceptos, narrar hechos históricos, etc.

Actualmente, los motores de juego no permiten los diferentes tipos de emergencia.

### C. Motor de Juegos Serios Emergentes (MJSE)

Los “motores de juegos” permiten realizar videojuegos, usando *ibrería* o *framework* ya diseñados, validados y probados. En [4, 9, 27] se ha propuesto la siguiente arquitectura de un MJSE (ver Figura 1):



Fig. 1. Arquitectura de un motor de juegos serios emergentes.

- 1) **Núcleo del Motor de Videojuego:** es el elemento central del MJSE, ya que tiene los seis submotores base de un videojuego: submotor de gráficos (SG), submotor físico (SF), submotor de sonido (SS), submotor de interacción (SI), submotor de video (SV) y submotor de renderización (SR) (ver [21, 28] para detalles).
- 2) **Los Subsistema de Emergencia del Videojuego (SEV) y de Adaptación del Videojuego (SAV),** son las capas de la arquitectura MJSE que permiten hacer emerger un JSE, y están compuestos por los siguientes submotores:
  - a) **Submotor de Inteligencia Artificial (SIA):** se encarga de introducir comportamientos inteligentes en los personajes y componentes del JSE.
  - b) **Submotor de Trama Emergente (STE):** permite adaptar las narrativas de los videojuegos al contexto deseado. Este módulo es crucial para el AJSE, porque

lo usa para adaptar un JSE al tema que se está dando en el SaCI. El STE posee las siguientes sub-capas:

- i. Gestor de Materia (GM): detecta el curso dictado en el SaCI, y a partir de allí, el tipo de trama que se debe aplicar en la clase, por ejemplo: física, química, historia, u otras. Para ello, usa la siguiente información: universidad, carrera, pensum, semestre, asignatura, tema, entre otros.
- ii. Gestor de Videojuegos (GV): detecta el tipo de videojuego (*edugame*, *advergame*, etc.) pertinente para ese momento, usando los agentes Sistema Recomendador de Objeto de Aprendizaje (SROA) y Repositorio de Objeto de Aprendizaje (ROA). Si no lo consigue, llama al Módulo de Generación de JSE para generar el videojuego usando la información de GM.
- iii. Módulo de Generación de JSE (MGJSE): es el responsable del ensamblaje de un nuevo JSE. En un trabajo previo, se ha definido el MGJSE basado en el algoritmo de Optimización por Colonias de Hormiga (ACO, por sus siglas en inglés, ver [21, 28] para más detalles).

El SEV, cuando invoca al STE es para hacer emerger la primera versión del JSE, según los objetivos que se deben cumplir con él. El SAV, cuando invoca al STE permite ir adaptando el JSE a las características requeridas en el SaCI.

### III. AGENTE DE JUEGOS SERIOS EMERGENTES EN UN SAcI

En este trabajo, proponemos el AJSE para SaCI, el cual es incluido en la arquitectura AmICL. El AJSE interactúa con los agentes del SaCI: Sistema Académico (SA), ROA, SROA, VLE, entre otros. El AJSE se encarga de adaptar los JSEs al tema que se esté impartiendo en el SaCI, tomando en cuenta al jugador y al entorno.

#### A. Contexto del AJSE

El AJSE se encarga de gestionar las interacciones y conversaciones de los agentes del SaCI con los JSEs disponibles. Además, el AJSE, una vez recolectada la información del entorno del SaCI (perfil de los estudiantes, objetivos del actual proceso de aprendizaje, etc.), adapta el JSE al SaCI, compaginando la información del tema que se está dando en clases con la trama del juego. Básicamente, el AJSE es un agente de la capa lógica del AmICL, que interactúa con los siguientes componentes del SaCI, los cuales son instanciados como agentes: Usuarios de SaCI (los profesores y estudiantes); Dispositivos: es el *hardware* presente en un SaCI, para interactuar con el AJSE; Proyectores: es el *hardware* de despliegue del JSE; VLE: representa la plataforma de aprendizaje usada en el SaCI (*moodle*, *sakai*, entre otros); ROA: almacena los recursos de aprendizaje (RA) posibles a usar durante el proceso de aprendizaje (JSE, sonido, videos, libros, revistas, entre otros); SROA: revisa el tema de lo que se está explicando en el SaCI, para recomendar RA que estén en un ROA; SA: registra la actuación de los estudiantes en sus procesos de aprendizaje, tales como sus notas académicas, su perfil educativo, etc.; MJSE: el AJSE lo usa, conjuntamente con los datos del VLE sobre el tema del curso, para adecuar el JSE (las tramas, sus propiedades, entre otras cosas).

#### B. Especificación del AJSE

El AJSE debe incorporarse a SaCI a través de AmICL. Particularmente, el AJSE está en la capa ILL del *middleware* reflexivo. Para la especificación del agente AJSE se usan los modelos que la metodología de MASINA propone [29, 30]:

1. Modelo de Agente del AJSE: este agente permite gestionar los JSE, para adaptarlos a las necesidades del tema tratado en el SaCI, tal que el usuario tenga una experiencia de aprendizaje adaptada a sus necesidades a través del SaCI. Sus objetivos específicos son:
  - a) Presentar el JSE adaptado a los perfiles de usuario presentes en el SaCI. De este objetivo específico se derivan los siguientes sub-objetivos: i) Detectar el tipo de JSE que se necesita en un tema determinado, basado en el perfil de los usuarios; ii) Optimizar la interacción con los dispositivos físicos del SaCI en donde se van a visualizar.
  - b) Presentar el JSE adaptado al tema educativo impartido en un momento dado en un SaCI: i) Adecuar el JSE al contexto actual del SaCI; ii) Ofrecer a los usuarios una experiencia en un tema, tal que el JSE potencie el proceso de aprendizaje.

En la Tabla I presentamos un ejemplo de la descripción detallada de uno de los objetivos específicos.

TABLA I  
OBJETIVO DEL AJSE

<b>Objetivo:</b> desplegar el JSE en los dispositivos presentes en el SaCI
<b>Tipo:</b> gestionar el JSE.
<b>Parámetros de entrada:</b> solicitud al AJSE, por medio de los agentes de la capa ILL (identificador de usuario, perfil, identificador de JSE, identificador de dispositivos y tipo del JSE).
<b>Parámetros de salida:</b> despliegue de un JSE en un dispositivo disponible en un SaCI.
<b>Condición de activación:</b> recepción de solicitud de JSE.
<b>Condición de finalización:</b> reproducción de JSE.
<b>Condición de éxito:</b> se reproducen los JSEs adaptados al tema de clases del SaCI para el usuario.
<b>Condición de fracaso:</b> no se reproducen los JSEs si: los dispositivos no son los correctos, la información del usuario no es la adecuada o existen errores de comunicación entre los agentes.
<b>Lenguaje de representación:</b> lenguaje máquina.
<b>Descripción:</b> reproduce los JSEs adaptados al tema de un perfil del usuario de un SaCI, en diferentes tipos de dispositivos de JSE.

2. Modelo de tareas del agente AJSE: las tareas que este agente realiza, dependen de los servicios que debe brindar según sus objetivos. En general, tiene un grupo de tareas como AJSE, que realiza secuencialmente:
  - a) Recolectar datos del VLE sobre el tema.
  - b) Identificar requerimiento en el SaCI.
  - c) Determinar el JSE.
  - d) Adecuar el JSE.
  - e) Incorporar el JSE en el SaCI.
  - f) Ejecutar el JSE.

Además de esas tareas, el AJSE requiere realizar otras tareas. En ese sentido, algunas de esas tareas específicas que realiza el AJSE en el SaCI se muestran en la Tabla II.

3. Modelo de Inteligencia del AJSE: este agente debe tener un mecanismo de aprendizaje para responder rápidamente a situaciones que se le han presentado anteriormente, pero

también para adaptarse a los contextos y dominios que se les vayan presentando. A continuación, se presenta el modelo de inteligencia de este agente (Tablas III y IV).

TABLA II  
TAREAS ESPECÍFICAS DEL AJSE EN EL SaCI

Tareas
1. Realizar consulta al AS sobre datos de perfiles de los usuarios.
2. Realizar consulta al VLE sobre el tema en que se está dando.
3. Revisar los JS que están almacenados en ROA.
4. Usar las recomendaciones de SROA.
5. Activar a MGJSE, según el tema del SaCI.
6. Activar STE, según el tema del SaCI.
7. Recolectar subtramas de JSE, y guardar en ROA.
8. Utilizar dispositivos para proyectar el JSE en el SaCI.

TABLA III  
MECANISMO DE APRENDIZAJE

<b>Nombre:</b> JSE adaptados al tema.
<b>Tipo:</b> supervisado.
<b>Técnica de representación:</b> basada en reglas de Sistemas de Clasificación Difusos, en ACO y en Algoritmos Culturales, para ensamblar un conjunto de subtramas adecuados a un tema dado en el SaCI.
<b>Fuente de aprendizaje:</b> JSE ensamblados en base al tema del SaCI y las características de usuarios para las que son adecuadas.
<b>Mecanismo de actualización:</b> retroalimentación

TABLA IV  
MECANISMO DE RAZONAMIENTO

<b>Fuente de información:</b> reglas que representan cómo se debe utilizar el JSE, según el tema, la actividad y los usuarios.
<b>Tipo de inferencia:</b> motor de reglas.
<b>Lenguaje de representación de conocimiento:</b> sistema clasificador difuso
<b>Relación utilidad tarea-objetivo:</b> resultados académicos satisfactorios de acuerdo el JSE usadas en el proceso de aprendizaje que ofrece el SaCI.
<b>Estrategias de búsqueda de soluciones:</b> deductivo.

4. Conversaciones del AJSE en un SaCI: algunas de las conversaciones predeterminadas donde interviene el agente AJSE en un SaCI son:
- Solicitud de recomendación de JSE al SROA,
  - Diseño de un JSE.
  - Ejecución de un JSE

Las Tablas V y VI describen en detalle una conversación: Ejecución de JSE con el tema ya adaptado.

TABLA V  
CONVERSACIÓN: EJECUCIÓN DE JSE CON EL TEMA YA ADAPTADO

<b>Objetivos:</b> proyección de JSE
<b>Agentes participantes:</b> AJSE, dispositivo, ROA, SA, Usuario (tutor y estudiantes), VLE y SROA.
<b>Iniciador:</b> usuario autorizado del sistema
<b>Actos de habla:</b> solicitar recomendación de SROA, consultar perfil usuario en VLE, buscar JSE en ROA, consultar dispositivos de proyección en un SaCI, desplegar JSE, adecuar JSE según interacciones usando el MGJSE.
<b>Pre-condición:</b> existir una solicitud de JSE, existir la comunicación entre los agentes involucrados, existir dispositivo de proyección de JSE.
<b>Condición de terminación:</b> proyección o detección de error de JSE.
<b>Descripción:</b> mediante esta conversación, el agente solicitante inicia la conversación que permite al AJSE interactuar con los actores involucrados en el servicio, para lograr saber que JSE va a ser utilizado, en el dispositivo de salida y en el tema indicado por el agente solicitante del servicio.

TABLA VI  
ESQUEMA DE COORDINACIÓN DE LA CONVERSACIÓN DE LA TABLA IX

<b>Objetivo:</b> coordinar la comunicación entre los agentes implicados en la conversación
<b>Tipo:</b> por defecto
<b>Coordinación por defecto:</b> negociación entre AJSE y entre el agente dispositivo, VLE, SROA, ROA y SA.
<b>Mecanismo de comunicación</b>
<b>Tipo:</b> directa
<b>Técnica:</b> envío de mensajes
<b>Meta-lenguaje:</b> ACL
<b>Ontologías:</b> ontología del dominio de JSE, ontología general de un SaCI

#### IV. CASO DE ESTUDIO

##### A. Escenario Experimental

En este caso de estudio, el AJSE emplea el juego *Ajedrez Apple versión 3.13 Inc. 2003-2015* para adaptarlo al contexto, según sugerencias del agente VLE, el cual provee la información sobre el tema y los estudiantes que están en el SaCI. El AJSE adapta las características de un JSE, por ejemplo: establece los escenarios compatibles con el contenido del curso, define las tramas a usar, escoge el formato en los dispositivos de despliegue, entre otras cosas.

Se parte del supuesto que en el SaCI se está impartiendo el curso "Computación Ajedrecística", para estudiantes de primer y segundo año de la carrera de Ingeniería de Sistemas. En dicho curso, al estudiante se le presentan los conceptos de base sobre "Ajedrez Computacional", que consiste en aprender a jugar ajedrez y variantes de ajedrez, como por ejemplo el "Pasapiezas", revisar sus reglas, para realizar programas basados en estos tipo de juegos, así como analizar las herramientas tecnológicas usadas por la computadora para vencer a un humano.

Durante el curso, el profesor explica varios ejemplos de jaque mate en ajedrez y pasapiezas, usando la pizarra inteligente, y apoyándose en el AJSE, que va ejecutando el juego de *Ajedrez Apple v3.13*, para ir desplegando los ejemplos de esas jugadas. A continuación, mostramos el comportamiento del SaCI en una sesión de este curso, haciendo hincapié en el comportamiento del AGJSE:

- 1) Agente Tutor: activa el inicio de la sección de clase en el VLE, indicando cuales son los estudiantes presentes de forma presencial o virtual.
- 2) Agente VLE: establece los temas a tratar, la duración del mismo, la planificación del trabajo, los OAs a usar, entre otras cosas (ver Tabla VII).
- 3) El Agente SA: busca información sobre los estudiantes en el curso y genera un promedio de sus datos (ver tabla VIII).
- 4) Agente SROA: revisa el tema: jaque mate y sugiere varios OAs, entre estos, el videojuego "*Ajedrez Apple v3.13*"

TABLA VII  
INFORMACIÓN DEL VLE

Datos Básicos	Planificación
a. <b>Materia:</b> Ajedrez	a. <b>Presentación:</b> 10 minutos.
b. <b>Modulo:</b> Jaque Mate (JM)	b. <b>Explicación del Juego:</b> 10 minutos.
c. <b>Tema:</b> JM del Loco	c. <b>Pregunta:</b> 5 minutos.
d. <b>Clase:</b> 30 minutos	d. <b>Finalización:</b> 5 minutos.
e. <b>OAs:</b> JSE	

TABLA VIII  
SEGUNDA PARTE DE INFORMACIÓN DEL SA

Datos Básicos		Datos Básicos SA	
a. Idioma:	Español.	a. Nro. de Estudiantes:	20 inscritos
b. Lugar:	Mérida - Venezuela	b. Promedio de Exámenes:	1er parcial: 15 – 2do parcial: 17
c. Institución:	ULA.	c. Asistencia de Estudiantes:	17 asisten
d. Carrera:	Ingeniería de Sistemas.	d. Promedio total o SP =	16 pts.
e. Pensum:	Sistemas.	e. Nro. de Clases:	10 clases de 30
f. Semestre:	Primer.	f. Porcentaje de Asistencia:	86,5 %
g. Asignatura:	Computación Ajedrecística		

- 5) Agente ROA: es usado por el agente SROA para buscar en su base de datos por OAs.
- 6) AJSE: se activa dicho agente para empezar a hacer la gestión del JSE, por intermedio del STE y del SIA. Algunos ejemplos de las tareas que el AJSE realiza son:
- a) **Identificación del requerimiento en el SaCI**: consiste en definir y caracterizar el tema que se va a tratar en el SaCI con el JSE. Para ello, AJSE revisa la Materia: “Ajedrez” y el Tema: “Jaque Mate”, determina una lista de posibles tipos de jaque mates que se pueden explicar en menos de 10 minutos, y presenta la siguiente lista:
    - i. Jaque Mate del Loco (JML): 2 jugadas, con dialogo explicativo y en 3D.
    - ii. Jaque Mate Pastor (JMP): 4 jugadas, con dialogo explicativo y en 3D.
    - iii. Pasapiezas: sin dialogo y en 3D.
    - iv. Jaque Mate Caballo (JMC): con dialogo, en 2D, y solo imágenes.
    - v. Jaque Mate Alfil (JMA): sin dialogo, en 2D y solo imágenes.
    - vi. ....
  - b) **Inicialización del JSE**: el AJSE genera un JSE según el objetivo del curso, utilizando el algoritmo ACO [28, 30]. Para ello, ACO toma los elementos de la lista anterior con más alto valor de feromona, como JML, para definir la secuencia inicial del juego para el curso actual en el SaCI (ver Figura 2 y Tabla IX). Así, según la Figura 2, el algoritmo ACO podría escoger los nodos JML y pasapiezas, con valores de feromona 0.826 y 0.804, respectivamente, como secuencial inicial del juego.

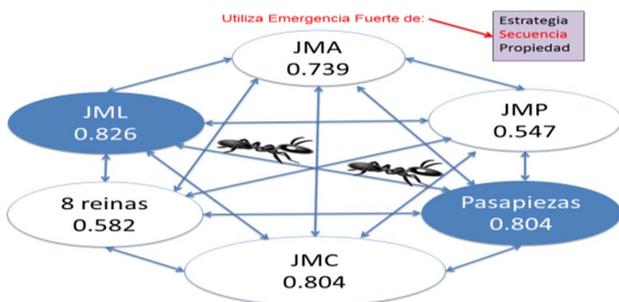


Fig. 2. Adecuación del JSE.

- c) **Ejecución del JSE**: en primera instancia se ejecuta el JML, y después pasapiezas. Ahora bien, la dinámica en

el SaCI (lo que acontece en el proceso de aprendizaje) hace que el AGJSE modifique las estrategias, la secuencia y las propiedades del JSE. Para ello, el AGJSE usa a SAV, a STE, a SIA y a las subtramas almacenadas, descompone el actual JSE, entre otras cosas.

TABLA IX  
SELECCIÓN DE JAQUE MATE DEL LOCO

Datos Básicos		Planificación	
a. Ambiente:	<u>Ubicación:</u> Mérida, <u>Cronología:</u> 21/03/2017.	a. Trama:	Diversas formas de explicar Jaque Mate del Loco con: JML o Pasapiezas.
b. Personajes:	Avatar	b. Dialogo:	“Es el mate más rápido que existe en el ajedrez y sucede en 4 jugadas, dos de las blancas y dos de las de cada bando.
c. Eventos:	<u>Movimientos:</u> Piezas <u>Interacción:</u> Turno de		.....

B. Proceso Emergente Gestionado por el AJSE del SaCI

Para el caso anterior, vamos a detallar el comportamiento del AJSE en dos situaciones, para desplegar inicialmente el Jaque Mate del Loco, y como después él hace emerger una nueva situación derivado de las interacciones en el SaCI, en el caso del Pasapiezas.

1) Situación 1: JML

En este caso, el AJSE modifica la *Estrategia* de la trama moviendo las piezas según las reglas, y el *Final* terminando el juego al dar jaque mate. En la Figura 3 se representa la conversación en el SaCI para este caso:

- a) El agente VLE requieren los servicios del agente AJSE para explicar el “JML” durante la sesión del curso que está gestionando.
- b) El agente AJSE hace emerger las tramas que permitan explicar dicha situación “JML”.
- c) Los estudiantes interactúan con el JSE, utilizando los Agentes Dispositivos que lo despliegan, tales como una pizarra, una tableta, un celular inteligente, etc.
- d) El AJSE va adecuando el JSE, en función de la interacción con el usuario. En particular, el AJSE usa los siguientes componentes de STE:

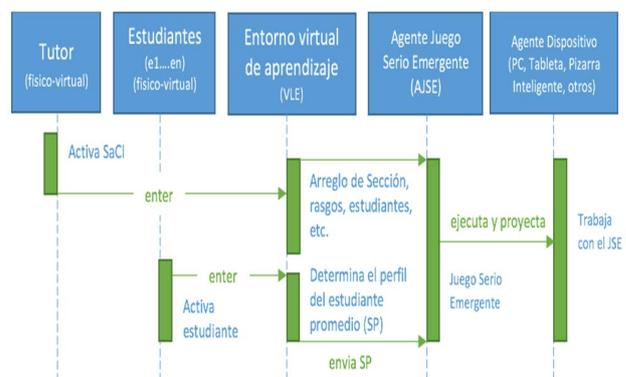


Fig. 3. Diagrama de interacción para el caso de estudio en el SaCI.

*MGJSE*: construye la trama del “JML”, invocando a los respectivos módulos indicados más abajo.

*SB*: este componente establece el guion narrativo a usar en este caso. El mismo contendrá algo como: “JML”: Es el mate

más rápido que existe en el ajedrez, y sucede en cuatro jugadas: se mueve primero el peón de caballo de rey dos pasos en las blancas, las negras mueven peón de Rey dos pasos, las blancas mueven peón de alfil un paso y el negro mueve Dama casilla 5 de Torre de Rey y Jaque Mate al Rey Blanco”.

*SE*: genera las acciones en el juego de Ajedrez Apple v3.13, utilizando los scripts respectivos, para reproducir el jaque mate (ver Figura 4).



Fig. 4. Primer movimiento de piezas blancas y negras.

## 2) Situación 2 pasapiezas:

A raíz de las dinámicas que van ocurriendo en el SaCI vinculadas al proceso de aprendizaje (por ejemplo, preguntas de los aprendices), el AJSE decide modificar las *Propiedades del juego*, agregando más cantidad de piezas, casillas, tablero, jugadores y variante de ajedrez. En particular, introduce una variante del juego de ajedrez denominada “pasapiezas”, en la que juegan cuatro jugadores en dos tableros en equipos de dos. Se aplican las reglas de ajedrez, excepto que las piezas capturadas en un tablero se pasan al compañero de al lado, quien tiene la opción de ubicarlas en su propio tablero (Ver Figura 5). En particular, el AJSE usa el *MGJSE* de STE para construir la trama para explicar el caso de Jaque Mate en la modalidad del juego de ajedrez de “pasapiezas”, haciendo lo siguiente:

- Establece el guion narrativo a usar en este caso para narrar el juego de “pasapiezas”.
- Establece el nuevo escenario, compuesto por dos tableros.
- Genera las acciones en el juego de Ajedrez Apple v3.13, utilizando los scripts respectivos, para reproducir el Jaque Mate del Loco en la modalidad de “pasapiezas”.



Fig. 5. Modalidad “Pasapiezas de ajedrez”.

## C. Comparación con Otras Propuesta

En la Tabla X se presenta un cuadro comparativo de nuestro AJSE para el SaCI, con respecto a otros trabajos que presentan a agentes inteligentes que gestionan videojuegos, o en AmI educativos que usan videojuegos en sus dinámicas. La comparación se basa en el tipo de juego que gestionan, la capacidad adaptativa del juego, el motor de juego que usa, si posee un agente que lo gestiona (adecua, modifica, etc.), para qué AmI ha sido diseñado, y qué tipo de emergencia posibilita según [7, 20]: débil (el producto del juego es emergente) o fuerte (permite el surgimiento de un nuevo juego).

En cuanto a los videojuegos, solo el nuestro está vinculado a JSE, y solo 3 del resto consideran JS [1, 31, 32]. Ahora bien, todos permiten la adaptación de los juegos, pero solo dos posibilitan la emergencia de nuevos juegos, el nuestro y [6]. Con respecto a si usan un motor de juego para esas tareas, el nuestro es el único que usa un motor para JSE, mientras que los otros usan motores de juegos clásicos [32] o de JS [1, 31]. En cuanto a quien gestiona los juegos en el ambiente, en nuestro caso es AJSE, y en el resto de trabajos proponen sistemas clasificadores [18], motores [31, 33] o scripts [6]. Solo dos han sido pensando para AmIs, el nuestro y [32]. Finalmente, el único que posibilita emergencias fuertes y débiles es el nuestro, y solo dos apuntan a algún tipo de emergencia [6, 31]. Así, nuestra propuesta es el único que usa agentes para gestionar JSE en un SaCI usando un MJSE.

Nuestra propuesta tiene varias diferencias importantes con respecto a los trabajos previos. El AJSE puede interactuar con los agentes del SaCI, y a partir de allí, definir específicamente las adecuaciones requeridas. Además, el AJSE provee un conjunto de servicios que permite la generación de nuevos JSEs o la adecuación de existentes, a los requerimientos del SaCI en un momento dado. Para ello, usa un conjunto de componentes del STE y del SIA, que permiten la adaptación de un JSE a nivel de la secuencia, estrategia, final del juego, propiedades de su componente, y utilidad del juego, según lo requerido por SaCI

## V. CONCLUSIONES

El presente artículo tiene como objetivo, proponer la especificación de un AJSE para SaCI, el cual gestiona los JSEs presentes en él, para irlos adecuando al proceso de enseñanza-aprendizaje ocurriendo en el mismo.

En el artículo se presentan las características del AJSE y su interacción con los otros agentes del SaCI (conversaciones). En particular, se definen sus objetivos, tareas, y sus relaciones con los agentes del SaCI (SA, VLE, SROA, ROA, entre otros), con el fin de proponer JSE adaptados al tema tratado en el SaCI en un momento dado. Además, se especifican sus capacidades para detectar información del entorno, determinar el JSE adecuado en un momento dado en el SaCI, y hacer emerger los comportamientos adecuados en los JSE, entre otras cosas.

Así, en este artículo se propone la arquitectura del AJSE para servir como gestor de JSEs que ayuden en los procesos de enseñanza aprendizaje en el SaCI. En este contexto, el JSE es un agente autónomo, el cual provee un conjunto de servicios para la gestión de JSEs, tal que les permitan adecuarse al entorno del SaCI. Estos servicios son realizados, a través de las tareas definidas para el AJSE.

El caso de estudio permite describe su comportamiento en SaCI. Por otro lado, la comparación hecha con trabajos previos muestra que es una tecnología innovadora, lo que motiva a continuar trabajando en temas vinculados al AJSE.

Los próximos trabajos deberán evaluar, por una parte, si la clase es más emocionante, entretenida y dinámica, al usar el JSE en el SaCI y, por la otra, qué habilidades el estudiante puede desarrollar y fortalecer usando JSE. A la par, se deberán desarrollar específicas métricas que permitan medir la interacción del AJSE en el SaCI, y en particular, su impacto en la dinámica de aprendizaje en el SaCI.

TABLA X  
COMPARACIÓN CON OTROS TRABAJOS RECIENTES

Crterios	[1]	[6]	[18]	[31]	[32]	[33]	Presente Trabajo
<b>Tipos de Videojuego</b>	JS y Videojuegos Comerciales	Narrativa Emergente en Videojuego	Videojuegos, Juegos Combinatorios, Simulación y Teoría de Juegos Clasificadores	JS	JS	Juego de Roles (RPG)	JSE
<b>Capacidad de adaptarse del juego</b>	Se adapta usando DBN para ello	Se adapta con expansión de narrativa emergente	hacen la adaptación a diversos requisitos del juego	Máquina de aprendizaje para adaptarse a diferentes jugadores	NA	Se adapta usando un sistema de eventos	Se adapta al tema del SaCI usando un MGJSE
<b>Motores para JSE</b>	Motor de juego multiplataforma Unity	NA.	Motor Simbólico basado en Conocimiento (MSC)	Motor Adaptable	Motor de JS	Motor de Eventos	SAV, SEV, STE, MGJSE
<b>Gestión de los Videojuegos</b>	Sistemas de tutoría y tecnología narrativa inteligentes	Mediante Scripts.	Sistemas Clasificadores	Vía Motores	NA	Vía el Motor	AJSE
<b>Ambiente Inteligente (AmI)</b>	Entorno de Aprendizaje Inteligente	NA	NA	Framework	AmI	NA	SaCI
<b>Tipos de Emergencia</b>	NA	Incertidumbre, coherencia, espacio de posibilidad, co-autoría y agencia	NA	Ocurre la emergencia en la adaptación de contenido	NA	NA	Estrategia, secuencia, propiedad, final, modelo de negocio y utilidad

Otros trabajos futuros están vinculados al diseño y desarrollo de los diferentes componentes del STE, en particular de sus módulos MGJSE, SB, SE y GE, que permiten el surgimiento de subtramas, y la emergencia de secuencias de subtramas, guiados por objetivos precisos. Los actuales trabajos en desarrollo se orientan a estos temas [28]. Finalmente, próximos trabajos extenderán el caso de estudio, considerando lo aprendido por el AJSE al proveer sus servicios en diferentes cursos (por ejemplo, en un curso de inteligencia artificial, en otro de diseño de videojuegos, y en el curso descrito en este trabajo), de tal manera que pueda reusar lo aprendido en cada uno para adecuar los otros.

#### REFERENCIAS

- [1] J. Leste, E. Ha, S. Lee, B. Mott, J. Rowe, and J. Sabourin, "Serious Games Get Smart: Intelligent Game-Based Learning Environments," *AI Magazine*, vol. 34, no. 4, pp. 31–49, 2014.
- [2] J. Fox, L. Pittaway and I. Uzeugbunam "Simulations in Entrepreneurship Education: Serious Games and Learning Through Play", *First Published: Entrepreneurship Education and Pedagogy*, vol. 1, no 1, pp. 61–89, 2018.
- [3] G. Ben-Sadoun, V. Manera, J. Alvarez, G. Sacco and P. Robert, "Recommendations for the Design of Serious Games in Neurodegenerative Diseases," *Frontiers in Aging Neuroscience*, vol. 10, no 13, pp. 1-7, 2018.
- [4] T. Tregel, P. Müller, S. Göbel and R. Steinmetz, "Where's Pikachu: Route Optimization in Location-Based Games", In *Proc. 10th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*, 2018.
- [5] P Sweetser, *Emergence in Games*, Charles River Media, 2008
- [6] S. Chauvin, G. Levieux, J. Donnart and S. Natkin. "An Out-of-Character Approach to Emergent Game Narratives", In *Proc 9th International Conference on the Foundations of Digital Games*, 2014, pp. 1-4.
- [7] J. Aguilar, J. Cardozo, C. González and B. Rengifo, "Una aproximación a los Juegos Emergentes, Metrópolis, Simulador de Ciudades Autogestionada," In *Proc. XXXVII Conferencia Latinoamericana de Informática*, 2013, pp. 38- 46.
- [8] J. Aguilar J. Altamiranda, and D. Chávez, "Extensiones a Metrópolis para una Emergencia Fuerte," *Revista Venezolana de Computación*, vol. 3, no. 2, pp. 38-46, 2016.
- [9] J. Asbell-Clarke, E. Rowe, E. Sylvan. "Assessment design for emergent game-based learning". In *Proc. Extended Abstracts on Human Factors in Computing Systems*, 2013, pp. 679-684.
- [10] P. Cardoso, M. Carvalhais, "Breaking the Game: The traversal of the emergent narrative in video games", *Journal of Science and Technology of the Arts*, vol. 5, no. 1, 2013.
- [11] L. Martinez, S. Marrufo, G. Licea, R. Juarez-Ramirez, and L. Aguilar, "Using a Mobile Platform for Teaching and Learning Object Oriented Programming," *IEEE Latin America Transactions*, vol. 16, no. 6, pp. 1825-1830, June 2018.
- [12] B. Douglas, E. Emily and S. Stephen, "Digital Games, Design, and Learning: A Systematic Review and Meta-Analysis," *Review of Educational Research*, vol. 86, no. 1, pp. 79–122, 2016.
- [13] J. Hernández, E. Benítez and C. Mezura, "Ambientes inteligentes en contextos educativos: modelo y arquitectura," *Research in Computing Science*, vol. 77, pp. 55-65, 2014.
- [14] P. Valdiviezo, J. Cordero, J. Aguilar and M. Sanchez, "Conceptual Design of a Smart Classroom Based on Multiagent Systems," in *Proc. Int. Conf. Artificial Intelligence*, 2015, pp. 471-477.
- [15] M. Sánchez, J. Aguilar, P. Valdiviezo, and J. Cordero, "A Smart Learning Environment based on Cloud Learning," *International Journal of Advanced Information Science and Technology (IJAIIST)*, vol. 39, no. 39, pp. 39-52, 2015.
- [16] M. Sánchez, J. Cordero, J. Aguilar, and P. Valdiviezo, "Basic features of a Reflective Middleware for Intelligent Learning Environment in the Cloud (IECL)," In *Proc. Asia-Pacific Conf. on Computer Aided System Engineering (APCASE)*, 2015.
- [17] J. Aguilar, F. Hidrobo, A. Ríos and M. Cerrada, *Sistemas Multiagentes y sus Aplicaciones en Automatización Industrial*, Segunda Edición, Universidad de Los Andes, Mérida, Venezuela, 2013.
- [18] K. Shfi and H. Abbass, "A Survey of Learning Classifier Systems in Games," *IEEE Computational intelligence magazine*, pp. 42–55, 2017.
- [19] S. Cuz-Lara, B. Fernández and C. Vaz de Carvalho, "Enfoques Innovadore en Juegos Serios," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 1, no. 1, pp. 19-21, 2013.
- [20] J. Agilar, *Introducción a los Sistemas Emergentes*, Talleres Gráficos, Universidad de Los Andes, Venezuela, 2014.

- [21] J. Aguilar, J. Altamiranda and F. Díaz, "Design of a Serious Emerging Games Engine Based on the optimization Algorithm of Ant Colony," *DYNA*, vol. 85, no 206, pp. 311-320, 2018.
- [22] L. Chamba and J. Aguilar, "Design of an Augmented Reality Component from the Theory of Agents for Smart Classrooms," *IEEE Latinoamérica Transactions*, vol.14, no. 8, pp. 3826-3837, 2016.
- [23] J. Aguilar, J. Cordero, L. Barba, M Sanchez, P. Valdiviezo, and L. Chamba, "Learning Analytics Tasks as Services in Smart Classroom," *Universal Access in the Information Society Journal*, Springer, vol 17, no. 4, pp. 693-709, 2018.
- [24] J. Aguilar, J. Cordero, O. Buendia, "Specification of the Autonomic Cycles of Learning Analytic Tasks for a Smart Classroom", *Journal of Educational Computing Research*, vol 56 no. 6, pp. 866-891, 2018
- [25] A. De Gloria, F. Bellotti, and R. Berta, "Serious Games for education and training," *International Journal of Serious Games*, vol. 1, no. 1, 2014.
- [26] T. Rasim, A. Langil, S. Munir, and Y. Rosmansyah, "A survey on adaptive engine technology for serious games," *Proceedings International Seminar on Mathematics, Science, and Computer Science Education*, vol. 1708, no. 1, pp. 050003-1–050003-9, 2016.
- [27] J. Aguilar, J. Altamiranda, F. Díaz, and D. Mosquera "Motor de Juego Serios en ARMAGAc-co". *Revista UNET*, vol. 28, pp. 100-110, 2016.
- [28] J. Aguilar, J. Altamiranda, F. Díaz, J. Gutiérrez, A. Pinto, "Sistema Adaptativo de Tramas para Juegos Serios Emergentes basado en el Algoritmo de Optimización de Colonia de Hormigas" In *Proc. XLV Conf. Latinoamericana de Informática*, 2019.
- [29] J. Aguilar, I. Besembel, M. Cerrada, F. Hidrobo and F. Narciso, "Una Metodología para el Modelado de Sistemas de Ingeniería Orientado a Agentes," *Revista Iberoamericana de Inteligencia Artificial*, vol.12, no. 38, pp. 39-60, 2008.
- [30] J. Aguilar, F. Hidrobo, M. Cerrada, "A Methodology to Specify Multiagent Systems", *Lecture Notes in Artificial Intelligence*, vol. 4496, pp. 92-101, 2007
- [31] I. Torres, "Intelligent Tutor for Programming System Using Multiple Intelligences," *IEEE Latin America Transactions*, vol. 16, no. 2, pp. 634-638, 2018.
- [32] A. Ali, A. Shoten, S. Göbel, and A. Amrich, "Playful Interactions and Serious Games," *Journal of Ambient Intelligence and Smart Environments*, IOS Press, vol. 6, no. 3, pp. 259-262, 2014.
- [33] H. Agustín, & M. Garde, "Desarrollo de un Motor de Eventos para Videojuegos," In *Proc Red de Interconexión de los Recursos Informáticos (RedIRIS)*, 2016, pp. 1-36.



**Díaz, Francisco.** Graduado de Ingeniero de Computación en 2004 en la Universidad Valle del Momboy (UVM), M. Sc. en Informática Aplicada del Instituto Superior José Antonio Echeverría (ISPJAE) en 2010, Tutor de la Comunidad de Aprendizaje del Masterdrez, en la Universidad Politécnica Territorial de Mérida Kleber Ramírez (UPTMKR) desde 2014, estudiante del Doctorado en Ciencias de la Computación en la ULA desde 2017.



**Aguilar, Jose.** Graduado en Ingeniería de Sistemas en 1987 en la ULA, M. Sc. en Ciencias de la Computación en 1991 (Université Paul Sabatier-Francia), Doctorado en Ciencia de la Computación en 1995 (Université Rene Descartes-Francia). Postdoctorado en Ciencias de Computación en la Universidad de Houston en 2000 y en el Laboratoire d'Automatique et Analyses de Systèmes-CNRS, Toulouse-France en 2011. Profesor Titular de la ULA. Ha publicado más de 500 artículos científicos en revistas científicas, libros y actas de congresos internacionales.



**Altamiranda, Junior.** Graduado de Ingeniero de Sistemas de la ULA en 2002. M. Sc. en Computación ULA en 2006. En el 2007 realizó una Estadía de Investigación en el Master 2 de Bioinformática en la Universidad de Rennes I en Francia. Formo parte del Plan II: Programa de Generación de Relevo de la ULA del 2008 al 2012. PhD en Ciencias Aplicadas, ULA en 2012. Profesor Asistente en el Departamento de Computación de la ULA