Recibido 14 Jul 2021 *ReCIBE, Año 10 No. 1, Mayo 2021*

Aceptado 02 Sep 2021

Diseño de un prototipo para configuración de entornos virtuales de aprendizaje basados en gamificación utilizando UWE

Design of a prototype for configuring virtual learning environments based on gamification using UWE

David I. Chán Chán[1]
davch987@gmail.com

Raúl A. Aguilar Vera[1]
avera@correo.uady.mx

Juan P. Ucán Pech [1*]
juan.ucan@correo.uady.mx

Julio C. Díaz Mendoza [1]
julio.diaz@correo.uady.mx

Jesús A. Santos Tejero[2]
jesus.st@valladolid.tecnm.mx

[1] *Facultad de Matemáticas,Universidad Autónoma de Yucatán,Mérida, Yucatán, México.*

[2]*Instituto Tecnológico Superior de Valladolid, Tecnológico Nacional de México, Valladolid, Yucatán, México*
*Autor para Correspondencia

RESUMEN

En este artículo se describe el diseño y evaluación del prototipo de un sistema para configuración de Entornos Virtuales de Aprendizaje basados en Gamificación. El prototipo utilizó los requisitos funcionales derivados del análisis de un modelo instruccional configurable y computable basado en elementos de gamificación. Para el desarrollo del prototipo se utilizó la metodología *UML Web Engineering* (UWE) la cual intenta cubrir el ciclo de vida completo de una aplicación Web; dicha metodología propone el desarrollo de cinco modelos. La evaluación del prototipo utilizó en un primer momento la técnica de juicio de expertos, y en un segundo momento, se evaluó la usabilidad del prototipo por parte de un conjunto de aprendices (alumnos) a quienes se les solicitó realizar una actividad bajo un esquema competitivo y contestar un instrumento que recopilaba opiniones de los alumnos con base en una escala *Likert* de cinco niveles. Las opiniones en términos generales fueron positivas (de acuerdo o en total acuerdo) respecto del esfuerzo requerido para su uso, respecto de la dinámica de interacción promovida para mantener la motivación e interés de los alumnos, de los tiempos de respuesta para enfrentar los retos, así como de las prestaciones que ofrece el entorno virtual de aprendizaje en escenarios gamificados.

PALABRAS CLAVE

Gamificación, Entorno Virtual de Aprendizaje, Informática Educativa, Software Educativo.

ABSTRACT

This paper describes the design and evaluation of a system prototype for the configuration of Virtual Learning Environments based on Gamification. The prototype used the functional requirements derived from the analysis of a configurable and computable instructional model based on gamification elements. For development of the prototype was used the UML Web Engineering (UWE) methodology which tries to cover the complete life cycle of a Web application. This methodology proposes the development of five models. The evaluation of the prototype used at first the expert judgment technique and in a second moment the usability of the prototype was evaluated by a group of learners (students) who were asked to carry out an activity under a competitive scheme and answer an instrument that collected opinions from students based on a five-level Likert scale. The opinions in general terms were positive (in agreement or in total agreement) regarding the effort required for its use, regarding the interaction dynamics promoted to maintain the motivation and interest of the students, the response times to face the challenges as well as the benefits offered by the virtual learning environment in gamified scenarios.

KEYWORDS

Gamification, Virtual Learning Environment, Educational Computing, Educational Software.

1. INTRODUCCIÓN

La Informática Educativa utiliza el conocimiento derivado de la Informática para el desarrollo de sistemas de cómputo y aplica dicho conocimiento al dominio de la educación, particularmente para asistir a los procesos de enseñanza y aprendizaje, es decir, es una disciplina orientada al desarrollo de aplicaciones educativas innovadoras, así como al estudio de sus usos, efectos y consecuencias en el proceso educativo.

Actualmente, el desarrollo de Software Educativo ha recurrido a la Ingeniería Web para el diseño y construcción de aplicaciones con alto nivel de interacción, donde debe cumplir con los principios básicos de la Ingeniera de Software y se requiere integrar aspectos técnicos y pedagógicos.

En el contexto de la educación formal, la falta de compromiso y de motivación para el aprendizaje —por parte de los aprendices— son problemas recurrentes que han sido afrontados mediante la incorporación del juego como dinámica instruccional. En las últimas décadas, ha surgido una variante a dicha dinámica instruccional denominada Gamificación, la cual ha despertado un creciente interés entre los académicos y profesionales en diversas áreas de la sociedad y con diversos propósitos (Johnson et al., 2016) (Hamari et al., 2014).

El trabajo que se describe en el presente artículo se deriva de la propuesta de generar un Modelo Instruccional Configurable y Computable Basado en Elementos de Gamificación (Aguilar et al., 2012), en particular, se presenta el diseño del prototipo del Sistema de Gestión de Entornos Virtuales de Aprendizaje (SGEA).

2. ANTECEDENTES

La Gamificación ha sido definida como el uso de elementos de diseño de juegos en contextos que no son juegos (Deterding et al., 2011); es concebida como una técnica de diseño que crea un juego, a través de un software educativo con el fin de mejorar la experiencia del aprendiz, lo que fomenta la aceptación por parte del mismo (Hamari et al., 2014). Los escenarios de aprendizaje gamificados suelen incorporar elementos de gamificación como: insignias, puntos, narración de historias, logros, niveles, barra de progreso y retroalimentación; también, se suele usar una tabla de clasificación y métodos para seguir el proceso del aprendizaje, así como un componente social donde los alumnos o participantes pueden compartir sus logros con otros participantes (Kapp et al., 2014) (Nah et al., 2014).

El Modelo Instruccional utilizado como referencia (Briceño, 2020), hace uso de cuatro elementos de gamificación: nivel de complejidad, acumulación de puntos, obtención de insignias y retroalimentación. Así mismo, el modelo incorpora tres constructos con base en los cuales es posible configurar los escenarios de aprendizaje gamificados, según convengan al docente, dichos constructos son:

(1) Reto: fue implementado mediante reactivos de respuesta estructurada, con niveles de complejidad acordes a la taxonomía de Bloom. El reto permite al alumno obtener puntos.

(2) Bloque: Se conforma por un conjunto de n retos, los cuales se diseñan con el mismo nivel de complejidad.

(3) Esquema: Se integra por un conjunto debloques de niveles de complejidad progresivos; dichos esquemas son configurados con dinámicas de interacción de tres tipos: individual, pareja-competitivo y pareja-colaborativo.

Los Entornos Virtuales de Aprendizaje —configurados con el modelo— (EVAs) pueden incorporar en su máxima configuración, hasta por tres esquemas con las tres dinámicas de interacción antes citadas. En Aguilar et al. (2020) se presenta la evaluación empírica realizada al modelo, con base en el primer nivel de interacción propuesto, el individual. Para la evaluación del prototipo se configuró un escenario de aprendizaje utilizando la segunda dinámica de interacción definida en el modelo: el esquema pareja-competitivo como se muestra en la Figura 1.

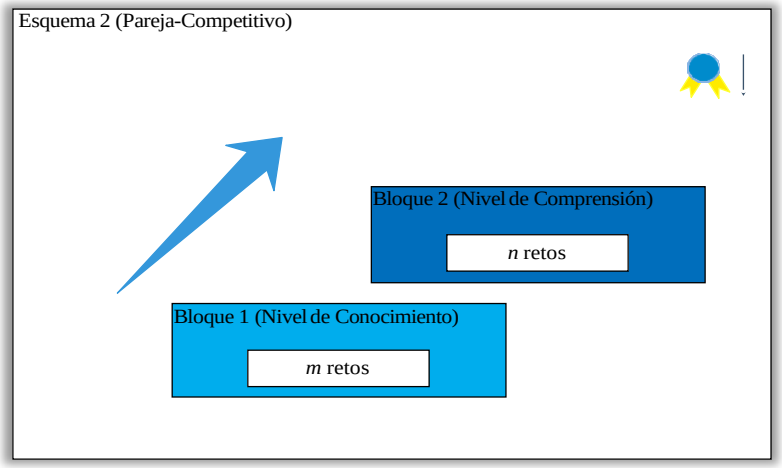


Figura 1. Esquema con la dinámica de interacción Pareja-Competitivo.
Fuente: (Briceño, 2020).

La dinámica de este esquema dos aprendices que trabajarán colaborativamente para resolver la serie de retos de los bloques de aplicación y análisis, los aprendices se comunican a través de un chat para que compartan sus ideas, con la finalidad de responder correctamente las preguntas.

3. METODOLOGÍA

La Ingeniería Web, podría decirse que a menudo suele asociarse este concepto con la construcción de sitios Web estáticos o dinámicos por igual, no obstante, la Ingeniería Web se enfoca en el modelado y diseño sistemático de aplicaciones Web con alto contenido de interacción, donde además de cumplir con los principios básicos de la Ingeniera de Software obliga a observar la usabilidad, navegabilidad, reusabilidad y mantenimiento, entre otros, de los artefactos, componentes, modelos de representación y procesos de la aplicación

UWE (*UML Web Engineering*) intenta cubrir el ciclo de vida completo de una aplicación Web especificando el proceso de desarrollo de una mejor manera (Koch et al., 2008). Dicha metodología propone la elaboración de cinco modelos que utilizan una notación estándar basada en el uso de UML (Unified Modeling Language): Diagrama de casos de uso, Diagrama de contenido, Diagrama de navegación, Diagrama de presentación y Diagrama de proceso. El modelo de casos de uso describe los requisitos de la aplicación, el modelo conceptual (modelo de contenido) que se corresponde con un modelo orientado a objetos tradicional, el modelo de interfaz que describe la presentación de objetos de la interfaz, modelo de navegación que describe la vista de navegación en el modelo conceptual y el modelo de proceso que representa las actividades que se conectan con cada clase de proceso (Soto et al., 2018).

4. PROTOTIPO PARA LA GESTIÓN DE EVAS BASADO EN GAMIFICACIÓN

En esta sección se describe el diseño y construcción del prototipo desarrollado para el Sistema de Gestión de Entornos Virtuales de Aprendizaje que implementa el Modelo Instruccional Configurable y Computable Basado en elementos de Gamificación propuesto por Briceño (2020).

4.1 Diseño del Prototipo

El diagrama de casos de uso de la aplicación SGEA modela la interacción de los actores Administrador, Maestro, Alumno con el sistema. Algunos casos de uso pueden dividirse en funciones más específicas, pero englobando el mismo concepto. En algunos casos son obligatorios otros casos de uso para poder llevar a cabo algún otro, por ejemplo, “VALIDAR USUARIO” en el cual los usuarios necesitan ingresar sus credenciales al sistema y este debe validarlos para que los usuarios puedan realizar alguna operación. En la Figura 2 se ilustra el diagrama de casos de uso para la aplicación web.

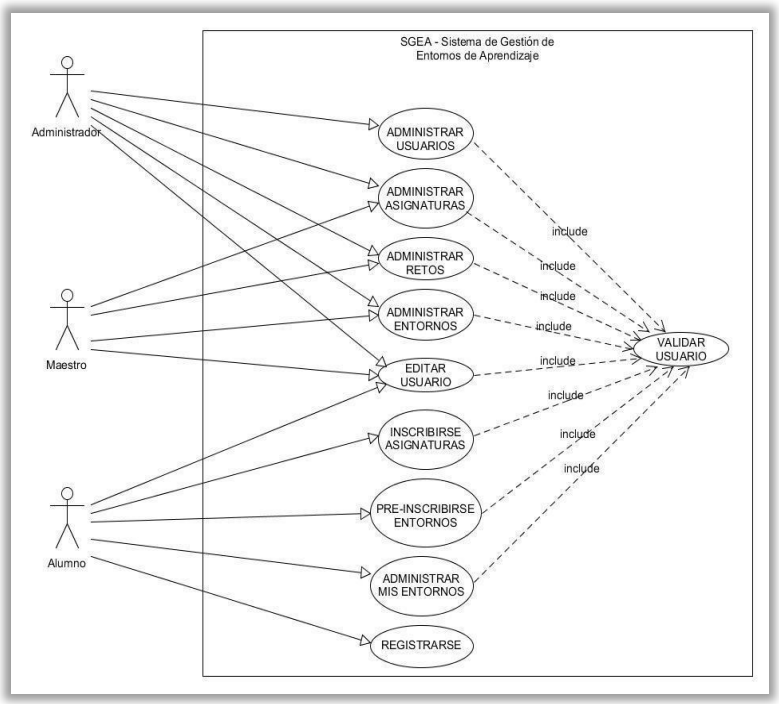


Figura 2. Diagrama de Casos de Uso.
Fuente: Elaboración propia.

El diagrama de contenido ilustra el modelo conceptual del dominio de la aplicación SGEA. En la Figura 3 se puede observar las diferentes clases que son necesarias para el funcionamiento de la aplicación y la interacción con los usuarios. A continuación, se describe cada una de las funciones de las clases modeladas:

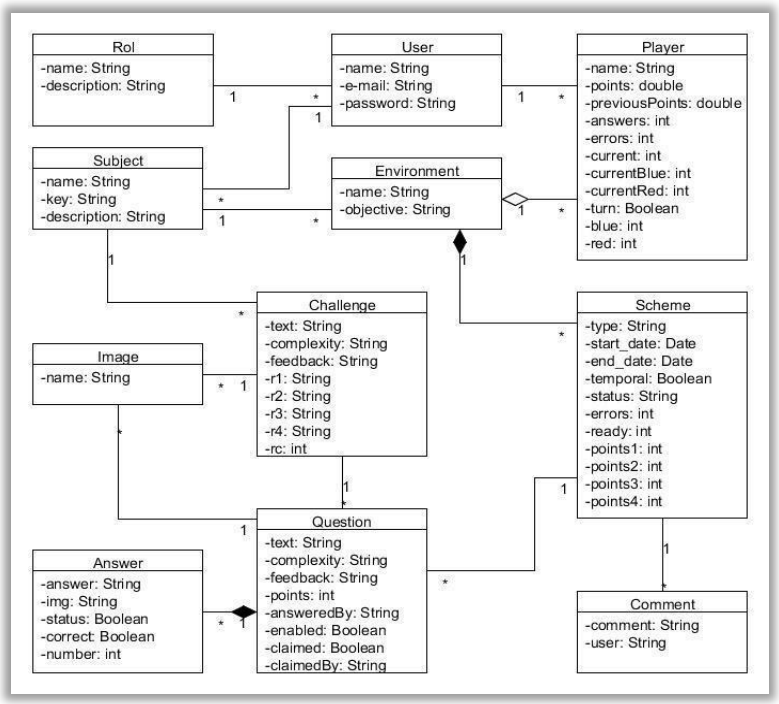


Figura 3. Diagrama de Contenido.
Fuente: Elaboración propia.

- La aplicación es utilizada por usuarios con diferentes roles por lo que la clase *User* está relacionada con la clase *Rol*. *User* contiene datos generales del usuario y las credenciales para que pueda acceder al sistema, *Rol* contiene el nombre y la descripción del rol.
- La clase *Player* modela los jugadores que tiene el alumno por cada entorno en el cual participa, en este se pueden obtener datos que van acorde a su avance durante los diferentes esquemas.
- La clase *Environment* modela los entornos que son creados por el Administrador o Maestro en los cuales los Alumnos participaran.
- La clase *Scheme* modela los diferentes esquemas disponibles para la creación de un entorno, cada esquema tiene una modalidad (Individual, Competitivo o Colaborativo), de igual manera hay esquemas que son base para la creación de otros y algunos que son temporales y tienen un comportamiento dinámico.
- La clase *Comment* modela los comentarios pertenecientes a un esquema, estos son desplegados en un chat durante la modalidad de tipo Colaborativo para que los Alumnos puedan interactuar con mensajes acerca de un reto.
- La clase *Challenge* representa los retos pertenecientes a una asignatura, estos sirven como base para la creación de retos dinámicos durante la ejecución del entorno en sus diferentes modalidades.
- La clase *Image* representa las imágenes relacionadas a un reto.
- La clase *Question* modela los retos de tipo dinámico que son utilizados durante la ejecución del entorno, estos guardan datos relevantes para la interacción de cada Alumno.
- La clase *Answer* modela las respuestas pertenecientes a cada reto.

- Cada usuario puede estar relacionado con una asignatura, esta es representada por la clase *Subject*, de igual manera los retos y entornos que son creados tienen como referencia a una asignatura, es decir, cada reto y entorno pertenece a una asignatura.

En los diagramas de navegación se diseña las diferentes vías por las cuales el usuario puede navegar en la aplicación, así como el acceso a las acciones y procesos disponibles. En la Figura 4 se presenta uno de estos diagramas y corresponde al módulo para administrar entornos. Como se puede observar en el diagrama presentado en la Figura 4, al acceder a este módulo habrá dos procesos generales *SearchEnvironment* (Buscar entorno) este consultará la base de datos y desplegará un *EnvironmentList* (Listado de entornos) y *CreateEnvironment* (Crear entorno) que dará de alta a un entorno en la aplicación.

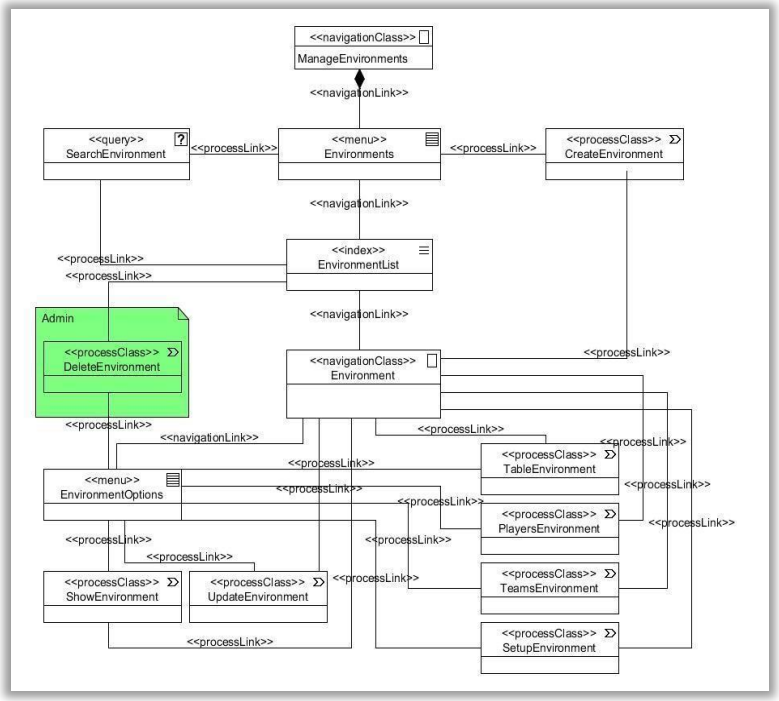


Figura 4. Diagrama de Navegación: Administrar Entornos.
Fuente: Elaboración propia.

En los diagramas de presentación se modela las representaciones de lo que son las vistas de la aplicación web, en estos modelos se puede encontrar los diferentes apartados y su contenido. En la Figura 5 se modela el módulo de entornos, a este módulo solo tienen acceso el Administrador y Maestro. Existe una representación de un menú "navBar" de la barra de navegación la cual dependiendo el rol contiene los accesos correspondientes.

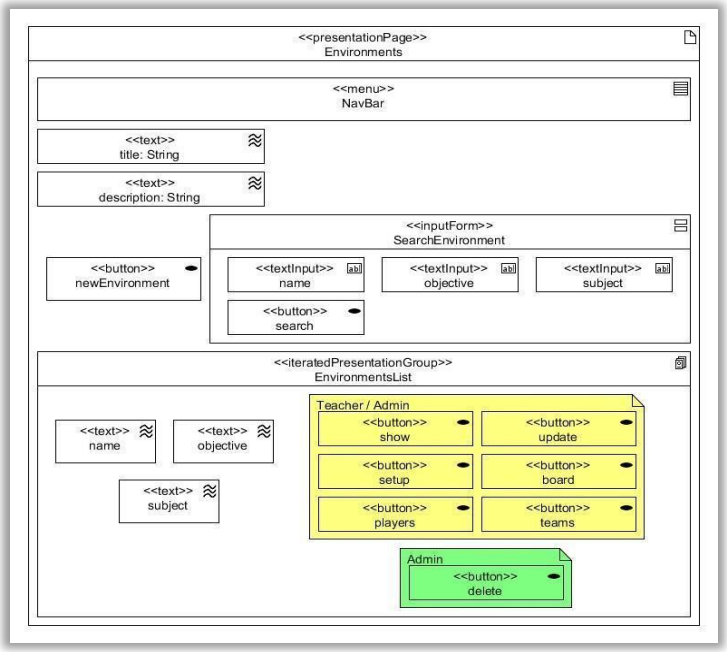


Figura 5. Diagrama de Presentación: Entornos.
Fuente: Elaboración propia.

En la Figura 6 se visualiza uno de los diagramas de procesos donde se modela la página de actualización de entornos en la cual tienen acceso el Administrador y Maestro, en ella se puede encontrar un menu "navbar" que contiene accesos a Asignaturas, Retos, Entornos y Usuarios, a este último solo tienen acceso los administradores, también, existe un desplegable que contiene las opciones Editar usuario y cerrar sesión.

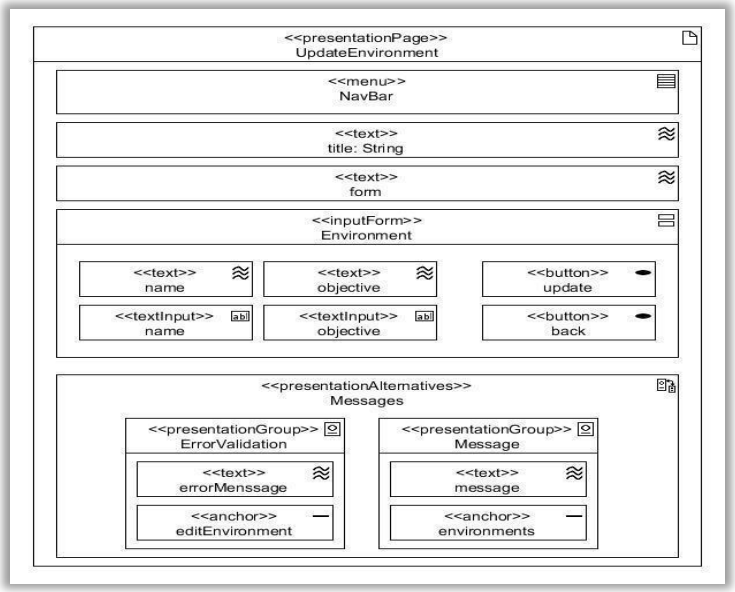


Figura 6. Diagrama de Procesos: Actualizar Entorno.
Fuente: Elaboración propia.

Finalmente, Figura 7 se presenta uno de los diagramas de actividades correspondiente a actualizar entorno del módulo de entornos. Esta actividad muestra los datos originales y toma los datos de entrada que el usuario modifica, estos datos son validados y en caso de error se muestra de nuevo el formulario de actualización, en caso de validarse correctamente el sistema actualiza el entorno, regresa a la página de entornos y muestra un mensaje de éxito.

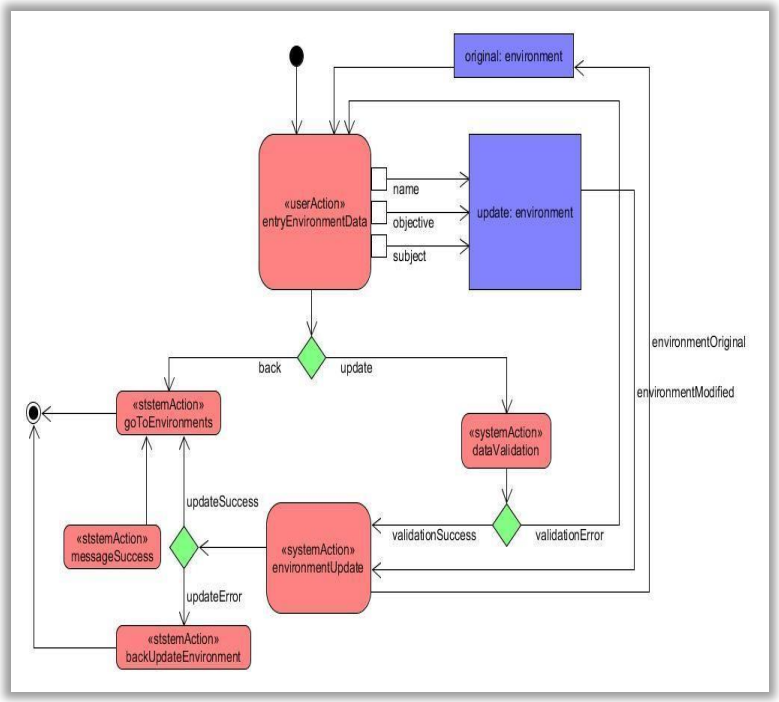


Figura 6. Diagrama de Actividades: Actualizar Entorno.
Fuente: Elaboración propia.

4.2 Construccion del Prototipo

Entre las tecnologías implementadas para el desarrollo del Sistema de Gestión de Entornos de Aprendizaje se tienen del lado del cliente el framework de php "Laravel" el cual tiene una estructura que facilita el uso del modelo MVC (McCool, 2012), que permite hacer aplicaciones modulares y de manera rápida. Entre algunas de las características se tienen:

- El uso de plantillas *Blade* para la creación de sus vistas.
- El uso de *Eloquent*, que es el ORM que nos sirve para manejar de una manera más ágil las operaciones de nuestra base de datos.
- *Laravel* proporciona un sistema de organización y gestión de rutas que nos permite controlar de manera exhaustiva las rutas de nuestro sistema.
- El uso de *Middlewares*, que son una especie de controladores que se ejecutan antes y después de una petición al servidor, lo que nos permite insertar múltiples controles, validaciones o procesos en estos puntos del flujo de la aplicación.

Por el lado del servidor se tiene "Apache2" que permite que la aplicación se encuentre alojada en un servidor virtual. Para la gestión de la base de datos se hace uso de "Mysql", es un sistema de gestión de bases de datos relacionales de código abierto con un modelo cliente-servidor. Como se mencionó anteriormente, el *framework* Laravel implementa el modelo vista controlador (MVC), donde MVC es un paradigma que divide

las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo.

A partir del uso de *frameworks* basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores (González y Romero, 2012).

En la Figura 8, presenta la vista de un usuario administrador para la gestión de entornos, dicha vista fue construida con las tecnologías mencionadas anteriormente.

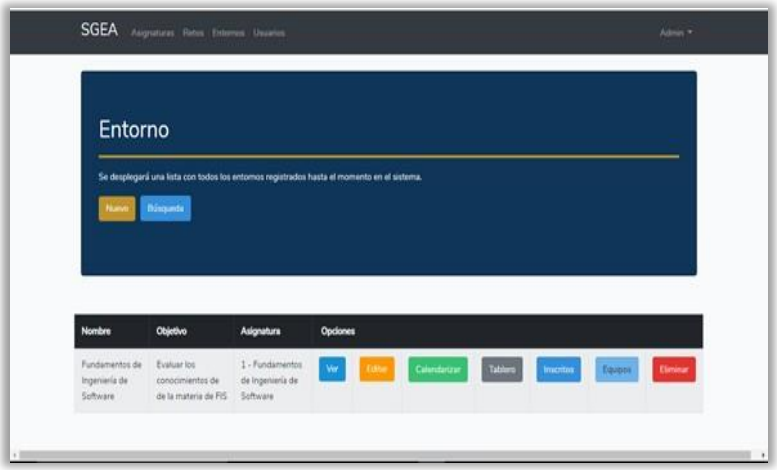


Figura 8. Vista del Entorno.
Fuente: Elaboración propia.

5. EVALUACIÓN DEL PROTOTIPO

El desarrollo de un sistema software, tiene como objetivo satisfacer un conjunto de necesidades acordadas con un cliente; en el caso del software educativo, dichas necesidades están vinculadas con los procesos de enseñanza y aprendizaje. No obstante, ¿Cómo puede saber un Ingeniero de Software si el producto construido se corresponde exactamente con lo que el cliente deseaba?, y por otro lado, ¿Cómo puede estar seguro de que el sistema va a funcionar correctamente? Dichas preguntas se corresponden con los procesos de validación y verificación del software (Juristo, 2004).

5.1 Validación

El proceso de validación del Sistema de Gestión de Entornos Virtuales de Aprendizaje se realizó desde dos perspectivas, en primera instancia, la de los clientes, y en segunda, desde la de los usuarios.

En el caso de los clientes, siendo el sistema parte de un proyecto de investigación, quienes fungieron como expertos, fueron los tres integrantes del Cuerpo Académico de Ingeniería de Software para la Educación (con registro PRODEP: UADY-CA-121) en conjunto con la proponente del proyecto de investigación. Para dicha evaluación se utilizó como método el recorrido cognitivo, en éste los clientes, en conjunto con el Ingeniero de Software, exploran todas las funcionalidades del sistema, utilizando como referencia el conjunto de funcionalidades acordadas; el prototipo fue aceptado para su utilización y experimentación con el modelo instruccional.

Una vez aceptada la primera versión del prototipo, éste —conjuntamente con el modelo instruccional— fue sometido a una evaluación empírica en la que participaron tanto los maestros o investigadores —en el papel de profesores— como un conjunto de alumnos —en su rol de aprendices. Dicha evaluación tenía como propósito corroborar que la funcionalidad educativa del sistema —orientada a los alumnos— fuese percibida correctamente.

Con los resultados positivos de la evaluación empírica antes citada, los investigadores solicitaron ligeras modificaciones vinculadas con las funcionalidades de interacción del sistema y se procedió a una última evaluación orientada a la usabilidad del sistema por parte de los aprendices. De acuerdo con Holzinger (2005) la usabilidad puede ser definida como la facilidad de uso y la aceptabilidad de un sistema para una clase particular de usuarios que realizan tareas específicas en un entorno específico; en nuestro caso, el tipo particular de usuarios fueron los aprendices. Entre los métodos más comunes para las pruebas de usabilidad se encuentran los cuestionarios, en nuestro caso, se diseñó un instrumento que recopilaba la opinión de los alumnos con base en una escala *Likert* de cinco niveles; la Figura 9 presenta el instrumento utilizado.

Instrucciones: Con base en las siguientes aseveraciones, marca con una X la casilla que se corresponda con el grado en el que estas de acuerdo o en desacuerdo con éstas.		**TA** Total Acuerdo **AC** de ACuerdo **NA** No Aplica **DA** en DesAcuerdo **TD** Total Desacuerdo				
		TA	**AC**	**NA**	**DA**	**TD**
01	El esfuerzo requerido para aprender a utilizar el entorno virtual ha sido mínimo.					
02	La dinámica de interacción (competitiva) generada por el entorno virtual me resulto motivante durante la sesión de ejercitación.					
03	Los tiempos de respuesta del entorno virtual durante mi sesión, me permitieron ejercitarme sin generar desesperación.					
04	El número de retos generados por el entorno durante la sesión fue apropiado para mantener mi atención.					
05	La dinámica de interacción (competitiva) generada por el entorno virtual mantuvo mi interés durante la sesión de ejercitación.					
06	Las prestaciones (funcionalidades) disponibles en el entorno me permitieron realizar las actividades propias de la dinámica competitiva.					

Figura 9. Instrumento de evaluación de la usabilidad del Prototipo.
Fuente: Elaboración propia.

Como se indicó en la sección de antecedentes, se diseñó un escenario de aprendizaje basado en el segundo esquema del modelo instruccional (competitivo), y se administró en una sesión de aprendizaje a 22 alumnos de la carrera de Ingeniería de Software que habían cursado la asignatura denominada Fundamentos de Ingeniería de Software.

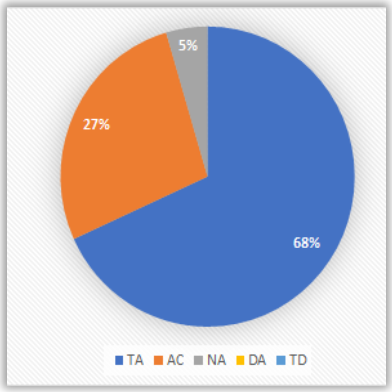
Las opiniones recogidas con el instrumento se ilustran y analizan a continuación:

- *Ítem 1. El esfuerzo requerido para aprender a utilizar el entorno virtual ha sido mínimo.*

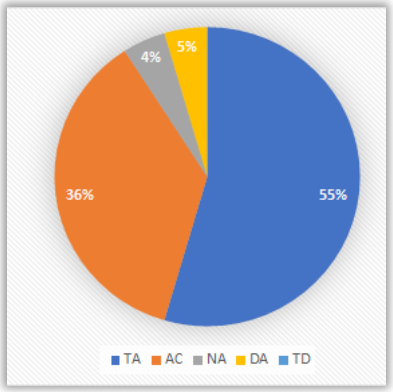
El primer ítem tuvo como objetivo que los alumnos evaluaran el esfuerzo que tuvieron que ejercer durante el experimento para aprender a utilizar la aplicación. Los resultados indican que los alumnos en un 95% coinciden en haber realizado un esfuerzo mínimo al aprender a utilizar la aplicación; el 5% restante indicó la opción no aplica (ver Figura 10A).

- *Ítem 2. La dinámica de interacción (competitiva) generada por el entorno virtual me resultó motivante durante la sesión de ejercitación.*

En este ítem los alumnos calificaron la dinámica competitiva, en específico, se deseaba saber si fue motivante. Los resultados indicaron que el 91% indico que la dinámica competitiva fue motivante y una minoría del 5% reporto estar en desacuerdo con esta aseveración (ver Figura 10B).



(A) Esfuerzo



(B) Motivación

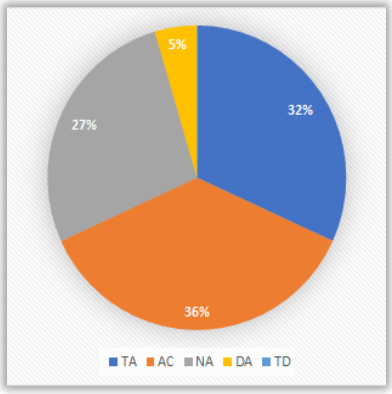
Figura 10. Gráfico de pastel para las opiniones de los Items 1 y 2.
Fuente: Elaboración propia.

- *Ítem 3. El esfuerzo Los tiempos de respuesta del entorno virtual durante mi sesión, me permitieron ejercitarme sin generar desesperación.*

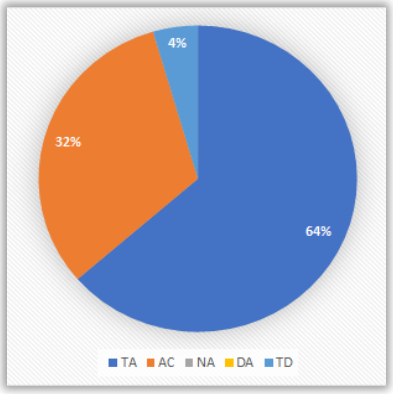
Los alumnos calificaron los tiempos de respuesta al interactuar con el entorno virtual, este aspecto resulta importante para una adecuada interacción, ya que es conveniente que los alumnos no se desesperen por los tiempos de respuesta, esto permite que el alumno no pierda la concentración en su proceso de aprendizaje o de ejercitación. Las opiniones de los alumnos fueron las siguientes: 68% estuvo de acuerdo o en total acuerdo, 27% respondió no aplica, y solamente el 5% restante indicó estar en desacuerdo (ver Figura 11A).

- *Ítem 4. El número de retos generados por el entorno durante la sesión fue apropiado para mantener mi atención.*

Las opiniones respecto del número de retos generados por el entorno en un 96% fueron positivas (ver Figura 11B). Una sobrecarga de retos puede ocasionar que los alumnos pierdan el enfoque o se frustren y el interés en responder adecuadamente vaya decayendo. El prototipo desarrollado con tecnología web mantiene reglas durante la construcción del entorno, el número de retos va decreciendo por cada bloque, lo cual limita al profesor a mantener un número considerable de retos, a mayor complejidad menor número de retos.



(A) Tiempos de respuesta



(B) # de retos

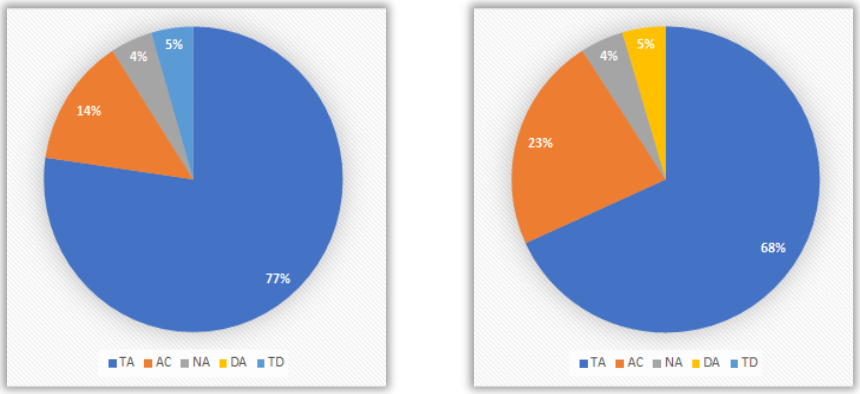
Figura 11. Gráfico de pastel para las opiniones de los Items 3 y 4.
Fuente: Elaboración propia.

- *Ítem 5. La dinámica de interacción (competitiva) generada por el entorno virtual mantuvo mi interés durante la sesión de ejercitación.*

La dinámica de interacción fue calificada por los alumnos en un 91% como adecuada para mantener el interés en la actividad y solo un 5% estuvo en desacuerdo en dicha aseveración (ver Figura 12A). Es importante destacar que ésta dinámica es la más compleja de los tres esquemas propuestos por el modelo; en esta dos alumnos compiten por obtener un mejor puntaje al ir respondiendo cada uno de los retos que se les van presentando. Los datos indican que la dinámica competitiva es adecuada y mantiene el interés de los alumnos.

- *Ítem 6. Las prestaciones (funcionalidades) disponibles en el entorno me permitieron realizar las actividades propias de la dinámica competitiva*

Con este ítem los alumnos calificaron el que las prestaciones disponibles en el entorno fueran útiles para realizar las actividades correspondientes en una dinámica competitiva. Los resultados dela encuesta indican que el 91% de los aprendices estuvo de acuerdo o en total acuerdo y solamente el 5% estuvo en desacuerdo con dicha aseveración (ver figura 12B).



(A) Interés (B) Prestaciones

Figura 12. Gráfico de pastel para las opiniones de los Items 5 y 6.
Fuente: Elaboración propia.

5.2 Verificación

En este apartado se documentan los resultados obtenidos de la prueba de rendimiento a la que fue sometido el sistema utilizando la herramienta JMeter; dicha herramienta es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el rendimiento de una variedad de servicios, con énfasis en aplicaciones web, en las cuales se requiere el acceso de muchos usuarios simultáneamente.

Uno de los módulos verificados fue el módulo individual y la funcionalidad verificada corresponde a la dinámica del juego individual al seleccionar una respuesta, lo cual significa verificar si es correcta o no y asignar los puntajes al usuario.

Tabla 1. Resultados de JMeter sobre el módulo individual

Etiqueta: Respuesta									
# Usuarios	❑	**Min**	**Max**	**D. E.**	**% Error**	**Rend**	**Kb/sec**	**Sent KB/sec**	**Media de Bytes**
1	19371	19371	19371	0	0.00%	0.05162	0.05	0.02	933
50	17140	4951	45933	7334.9	0.00%	0.6248	0.57	0.24	933
100	18980	5925	42183	7326.43	0.00%	1.32378	1.21	0.51	932.6
1000	18480	2621	59374	7969.8	0.10%	12.47427	11.38	4.79	933.9

Fuente: Elaboración propia.

En la Tabla 1 se muestran los resultados obtenidos de la capacidad del sistema para soportar estas situaciones; se observa que para 1 usuario el resultado fue un 0.00% de error al obtener los requerimientos de la función, en cuestión del tiempo requerido para la obtención de respuesta se tiene como mínimo (Min) y máximo (Max) de 19371 milisegundos. Por lo tanto, es una función estable y rápida con un usuario.

Para 50 usuarios el margen de error de los requerimientos sigue siendo un 0.00%, en este caso el mínimo es de 4951 milisegundos y un máximo de 45933, la media es de 17140 lo cual está por debajo del medio minuto y cuenta con una desviación estándar de 7334 milisegundos. Estos resultados demuestran que para un salón de clases numeroso el tiempo de respuesta es muy rápido aun siendo que utilicen la función al mismo tiempo.

En el caso de la prueba con 100 usuarios el margen de error no sufre cambios, sin embargo, los tiempos de carga son de un mínimo de 5925 y un máximo de 42183, con una desviación estándar de 7326.43 milisegundos con respecto a la media de 18980, lo cual es menor a medio minuto. Para esta prueba los tiempos de respuesta resultan rápidos, lo cual permite al usuario no frustrarse por la espera.

Por último, en la prueba con una cantidad de 1000 usuarios el tiempo de respuesta también es suficientemente rápido. Para estas pruebas el porcentaje de error se mantiene nulo en los casos menores o iguales a 100 usuarios, sin embargo, con 1000 usuarios se eleva un mínimo el porcentaje de error en los recursos, estos resultados indican que la aplicación web puede soportar una gran cantidad de usuarios a la vez respondiendo una pregunta en el entorno individual.

6. CONCLUSIONES

El prototipo del Sistema para Configuración de Entornos Virtuales de Aprendizaje basados en Gamificación descrito en este artículo, permitió validar el modelo instruccional propuesto; en particular, se configuró un entorno virtual de aprendizaje con el esquema competitivo; el proceso de verificación arrojó opiniones positivas por parte de los aprendices, y en cuanto al proceso de validación, los tiempos de respuesta en la prueba de rendimiento realizada fueron adecuados. En cuanto al uso de UWE como metodología para el diseño del prototipo, se pudo observar que UWE ofrece los modelos adecuados para la construcción de una aplicación de este tipo, y por consiguiente, considera las características propias que describen una aplicación web, entre estas características están el contenido, estructura de navegación y presentación. También, se pudo observar que entre los modelos que ofrece UWE considera las etapas de requisitos, análisis, diseño y desarrollo, es decir, en el modelado incorpora las etapas del ciclo de vida de un software que sin duda ayudaron a la realización de este trabajo.

AGRADECIMIENTOS

Agradecemos el apoyo brindado por la Facultad de Matemáticas (UADY), así como los alumnos de la Licenciatura en Ingeniería de Software de la UADY que participaron en la prueba piloto.

REFERENCIAS

Aguilar, R., Briceño, E., Díaz, J., y Ucán, J. (2020). Modelo instruccional configurable y computable basado en elementos de gamificación: un estudio de caso sobre la educación en Ingeniería de Software. *Revista Apertura.* 12(2), pp. 20-35.

Briceño, E. (2020). *Modelo Instruccional Configurable y Computable basado en Elementos de Gamificación*. Tesis de Maestría, Universidad Autónoma de Yucatán.

Deterding, S.; Dixon, D.; Khaled, R. & Nacke, L. (2011). *From game design elements to gamefulness defining gamification.* In Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments, (pp. 9-15), ACM, New York, NY, USA, pp. 9-15.

González, Y. D., y Romero, Y. F. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática*, 11(1), pp. 47-57.

Hamari, J.; Koivisto, J., & Sarsa, H. (2014). *Does gamification work? - A literature review of empirical studies on gamification.* In Proceedings of the Annual Hawaii International Conference on System Sciences pp. 3025-3034.

Holzinger, H. (2005). Usability Engineerig: Methods for Sotware Developers. *Communications of the ACM*, Vol. 48, No. 1

Johnson, D., Deterding, S., Kuhn, K., Staneva, A., Stoyanov, S., & Hides, L.(2016). Gamification for health and wellbeing : A systematic review of the literature. *Internet Interventions*, 6, pp. 89–106.

Juristo, N. Moreno, A. y Vegas, S. (2004). *Técnicas de Evaluación de Software*. Universidad Politécnica de Madrid.

Kapp, L. M.; Blair, L. & Mesch, R. (2014). *The gamification of learning and instruction.* San Francisco, CA: Wiley.

Koch, N., Knapp, A., Zhang, G., and Baumeister, H. (2008). Uml-Based Web Engineering An Approach Based on Standards. *Web Engineering: Modelling and Implementing Web Applications*. Chapter: 7, Springer Berlin/Heidelberg, pp.157-191.

McCool, S. (2012). *Laravel Starter.* Packt Publishing Ltd.

Nah, F. F., Zeng, Q., & Telaprolu, V. R. (2014). *Gamification of Education : A Review of Literature.* In International conference on hci in business, Springer, pp. 401–409.

Soto, C., Ucán, J. y Aguilar R. (2018). Diseñando Entornos Virtuales de Aprendizaje para el Análisis de la Colaboración con UWE. *Abstraction and Application.* Vol. 20. Pp. 55-67.

NOTAS BIOGRÁFICAS

David Israel Chán Chán es Ingeniero de Software egresado de la Facultad de Matemáticas. Comprometido con el desarrollo de software e investigación de tecnologías para la resolución de problemas del cliente. Actualmente labora como Desarrollador sr de plataformas web en una empresa importante de la península de Yucatán.

Raúl Antonio Aguilar Vera obtuvo el grado de Doctor por la Universidad Politécnica de Madrid, España (Mención de Doctor Europeo) y el de Master en Ingeniería de Software por la misma Institución, posee también el grado de Maestro en Educación Superior por la Universidad Autónoma de Yucatán. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es coordinador de la Sección Académica de Ingeniería de Software de la Academia Mexicana de Computación (AMEXCOMP). Su trabajo de investigación incluye las áreas de Ingeniería de Software e Informática Educativa.

Juan Pablo Ucán Pech es Doctor en Sistemas Computacionales por la Universidad del Sur, campus Mérida, México. Maestro en Sistemas Computacionales con especialidad en Ingeniería de Software por el Instituto Tecnológico de Mérida, México. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es miembro de la Academia Mexicana de Computación (AMEXCOMP). Su trabajo de investigación se centra en temas relacionados con la Ingeniería de Software, Ingeniería Web e Informática Educativa.

Julio Cesar Díaz Mendoza es Ingeniero Industrial en Producción por el Instituto Tecnológico de Mérida (ITM). Maestro en Tecnologías de Información, por la Universidad Interamericana para el Desarrollo (UNID). Especialista en Docencia de la Universidad Autónoma de Yucatán. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, e imparte asignaturas en las carreras de Ingeniería de Software y Ciencias de la Computación. Su interés se enfoca al área de Ingeniería de Software Educativo

Jesús Antonio Santos Tejero es Ingeniero en Sistemas Computacionales, Maestro en Tecnologías de Información, Profesor-Investigador Asociado C en el Tecnológico Nacional de México campus Valladolid, experiencia de 18 años en docencia. Perfil deseable Prodep, responsable técnico de proyectos financiados por CONACYT de 2018 a 2021, miembro de las Redes de Investigadores Internacionales INNOVADER y REDILAT, desarrollador de Software, líder de proyectos de T.I. con estándar P.M.I., evaluador de proyectos de investigación para publicaciones, actualmente cursando Doctorado en Investigación y Docencia por la UNISANT Valladolid.

Esta obra está bajo una licencia de Creative Commons
Atribución-NoComercial-CompartirIgual 4.0 Internacional.