

22

Fecha de presentación: octubre, 2021

Fecha de aceptación: diciembre, 2021

Fecha de publicación: enero, 2022

AMBIENTES HÍBRIDOS

DE APRENDIZAJE PARA EL DESARROLLO DE ASIGNATURAS MEDIANTE UN ENFOQUE CONSTRUCTIVISTA

HYBRID LEARNING ENVIRONMENTS FOR SUBJECT DEVELOPMENT USING A CONSTRUCTIVIST APPROACH

Hermes Yesser Pantoja Carhuavilca¹

E-mail: hpantojac@unmsm.edu.pe

ORCID: <https://orcid.org/0000-0002-2926-0936>

Rosmeri Agustina Mayta Huatuco¹

E-mail: rmaytah@unmsm.edu.pe

ORCID: <https://orcid.org/0000-0001-6130-8795>

Luis Miguel Núñez Ramírez¹

E-mail: lnuñezr@unmsm.edu.pe

ORCID: <https://orcid.org/0000-0002-6888-4039>

Oswaldo José Rojas Lazo¹

E-mail: orojasl@unmsm.edu.pe

ORCID: <https://orcid.org/0000-0003-4085-7327>

Ezzard Omar Álvarez Díaz¹

E-mail: ealvarezd@unmsm.edu.pe

ORCID: <https://orcid.org/0000-0001-7313-0533>

¹ Universidad Nacional Mayor de San Marcos. Perú.

Cita sugerida (APA, séptima edición)

Pantoja Carhuavilca, H Y., Mayta Huatuco, R. A., Núñez Ramírez, L. M., Rojas Lazo, O. J., & Álvarez Díaz, E. O. (2022). Ambientes híbridos de aprendizaje para el desarrollo de asignaturas mediante un enfoque constructivista. *Revista Universidad y Sociedad*, 14(1), 221-231.

RESUMEN

El año 2021 se ha visto amenazado por la pandemia y se ha detenido el avance de la educación con el cierre de las instituciones educativas, la pérdida del aprendizaje en los estudiantes y el aumento de la deserción escolar. Para ello las universidades e instituciones educativas se han refugiado en los ambientes híbridos de aprendizaje para el desarrollo de asignaturas. Los ambientes híbridos de aprendizaje combinan instrucción cara a cara con instrucción mediada por las tecnologías de información y la comunicación. El ambiente híbrido de aprendizaje virtual constituye una herramienta de apoyo al docente, con la facilidad de procesar cualquier magnitud de información que tendría que llevar en varias hojas para poder realizar su clase, guardar toda su información con facilidad y podrá abrirlo en cualquier lugar. El uso del ámbito híbrido permite el uso de la enseñanza no presencial, sin reemplazar el apoyo al desarrollo de las actividades presenciales. Por ello, el presente trabajo pretende apoyar al docente en el desarrollo de asignaturas mediante un enfoque constructivista a través del ambiente híbrido de aprendizaje, por medio de la modelación de las metodologías Rational Unified Process (RUP) y Extreme Programming (XP).

Palabras clave: Ambiente híbrido de aprendizaje RUP, XP.

ABSTRACT

The year 2021 has been threatened by the pandemic and the advancement of education has been halted with the closure of educational institutions, loss of learning in students and increased dropout rates. Universities and educational institutions have taken refuge in hybrid learning environments for subject development. Hybrid learning environments combine face-to-face instruction with ICT-mediated instruction. The hybrid virtual learning environment constitutes a support tool for the teacher, with the ease of processing any amount of information that he/she would have to carry on several sheets of paper in order to conduct his/her class, save all his/her information with ease and be able to open it anywhere. The use of the hybrid environment allows the use of non-face-to-face teaching, without replacing the support to the development of face-to-face activities. Therefore, this paper aims to support the teacher in the development of subjects through a constructivist approach using the hybrid learning environment, by modeling the Rational Unified Process (RUP) and Extreme Programming (XP) methodologies.

Keywords: RUP, XP, hybrid learning environment.

INTRODUCCIÓN

Los sistemas e-learning aparecieron en el siglo IX con el objetivo de facilitar el acceso a la educación a todos aquellos que por diversas razones no podían acceder a las clases presenciales (O'Neill, et al., 2011). El e-learning es un nuevo concepto de educación a distancia en el que se integra el uso de las Tecnologías de Información y la Comunicación (TIC) y otros elementos didácticos para el aprendizaje y la enseñanza (Pressman, 1998).

Entre las ventajas de los sistemas e-learning (Planella, et al., 2006):

- Se eliminan las barreras espaciales: la formación se acerca al estudiante y lo libera de viajes con el consiguiente ahorro de tiempo y dinero.
- Se eliminan las barreras temporales: el horario de estudio es totalmente flexible salvo en los casos donde se realiza una videoconferencia en tiempo real.
- Menor costo para los alumnos, ya que suele ser más barata la tele-formación que la formación presencial.
- Familiarización con las TIC: al realizar un curso online los estudiantes se ponen al día en el uso de las TIC tan extendidas hoy en día.
- Acceso permanente a toda la información relativa al curso.
- Facilidad de comunicación entre los estudiantes y los profesores gracias a las herramientas de comunicación.
- Posibilidad de realizar trabajo en grupos.
- Posibilidad de evaluación continua a los alumnos
- Fácil elaboración y actualización de recursos.

Modelos de E-Learning

Podemos considerar por sus características dos modelos de e-learning (figura 1):

- E-learning sincrónico: es el proceso de aprendizaje en el cual, el instructor y todos los alumnos estarán al mismo tiempo y se comunicarán directamente, pero no es una presencia física en la misma localización. La forma más común de e-learning sincrónico es la llamada de conferencia por teléfono. Todos los interlocutores en línea al mismo tiempo (Sandars, et al., 2008).
- E-learning asincrónico: En este modelo la comunicación entre la gente no ocurre simultáneamente. Algunos ejemplos asincrónicos incluyen la toma de un curso a ritmo individual, al intercambiar mensajes de correo electrónico por un mentor, y el envío de mensajes a un grupo de discusión. Algunos lo llaman aprendizaje distribuido. Esta clase de e-learning que recibe más atención debido a su costo más bajo de

desarrollo, de componentes reutilizables, y de conveniencia al estudiante.



Figura 1. Procesos en e-learning.

El desarrollo del e-learning es quizás el fenómeno más emblemático de los medios tecnológicos que han intervenido en la Educación. A pesar de este fenómeno se habla ahora del b-learning (blended learning) como el modelo más adecuado, ya que conjuga sistemas presenciales seguidos o acompañados de soporte digital (figura 2) (Álvarez-Gómez, et al., 2020; Estupiñán, et al., 2020).

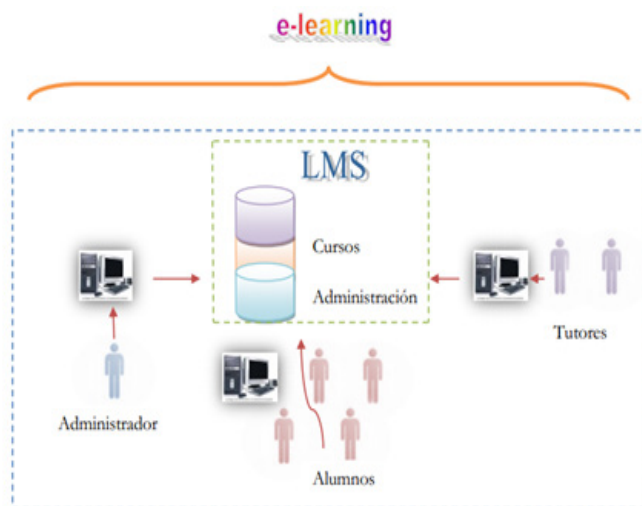


Figura 2. E-learning.

Sistema B-Learning

B-learning es una abreviatura de Blended Learning, término inglés que en términos de enseñanza virtual se traduce como "Formación Combinada" ó "Enseñanza Mixta". Se trata de una modalidad semipresencial de estudios que incluye tanto formación no presencial (cursos on-line, conocidos como e-learning) como formación presencial (George-Walker, et al. 2010).

Este sistema se presenta como una experiencia diferente para el alumno mediante un diseño pedagógico que combina la formación presencial con el ambiente virtual. En su dimensión pedagógica contempla la integración de recursos tecnológicos en busca de resultados formativos aplicables a necesidades de aprendizaje individualizadas.

Diferencias entre educación a distancia tradicional, e-learning y b-learning (tabla 1).

Tabla 1. Educación a distancia, e-learning, b-learning.

| | Educación a Distancia | E-Learning | B-Learning |
|-------------------------|---|--|--|
| Interactividad | El aprendizaje no es interactivo entre alumno y profesor. | El aprendizaje es interactivo entre alumno y profesor. | El curso es interactivo, dinámico y más flexible para el aprendizaje de materias que requieren la enseñanza presencial |
| Herramientas | El alumno aprende por sí solo mediante libros y dispone de un profesor para dudas | Aprovecha todos los recursos que ofrecen la informática e Internet para proporcionar al alumno una gran cantidad de herramientas didácticas como Chat, e-mail, videoconferencias, etc. | Combina las prestaciones de e-learning y la educación tradicional presencial. |
| Material de Apoyo | Los libros, CDs multimedia, guías de cursos. | Hipertexto, hipermedia, links externos, documentos digitales, artículos, noticias, información de foros. | Material de apoyo en educación virtual y el material de los laboratorios. |
| | Educación a Distancia | E-Learning | B-Learning |
| Conocimientos del Tutor | Conocimientos de la materia que imparte y de pedagogía para elaboración de las guías del curso | Conocimientos de la materia que imparte y de pedagogía para la elaboración de los cursos. Además, los tutores deben ser capacitados en la aplicación de TICs en el proceso de enseñanza/aprendizaje. | Son necesarios los mismos conocimientos que en la educación virtual, además se requiere el dominio y manejo del grupo para las clases presenciales. |
| Motivación del alumno | La oportunidad para estudiar al no disponer del tiempo suficiente para adquirir los conocimientos necesarios para lograr metas profesionales. | Las estrategias que usa el tutor para impartir el curso por Internet. | El sentirse complementado con la interrelación personal con profesores y compañeros. |
| | | Además de la flexibilidad para tomar los cursos. | |
| Entorno de formación | No exige ningún entorno. | El entorno virtual a través de la plataforma | El entorno virtual a través de la plataforma LMS y el entorno presencial a través de la infraestructura de aulas y laboratorios de la institución. |
| | | LMS (Learning Management System). | |
| Calidad de Formación | No garantiza la calidad de la formación, ya que es autónoma y depende de la voluntad de cada estudiante. | Depende de las estrategias y pedagogía del tutor para impartir los cursos y motivar a los alumnos. | De las tres modalidades es la que mejor complementa el aprendizaje del alumno, ya que permite desarrollar habilidades en materias que exigen un conocimiento práctico. |
| Costos | El costo de la matrícula y de material de apoyo. | Además de los costos de inscripción a los cursos virtuales, se suma el costo de conexión a Internet. | Inscripción al curso y costo de conexión a Internet, además el gasto de transporte. |

Las tecnologías de información y comunicación (TICS) en el sistema educativo

La irrupción de las Tecnologías de la Información y Comunicación (TIC) en el ámbito educativo le será de gran ayuda al alumno, permitirá nuevos métodos de aprendizaje y enseñanza tales como la Plataforma Virtual Educativa y que toma importancia en el proceso educativo en la actualidad.

La incorporación de las TIC en el sistema educativo ha producido una interacción entre los actores del proceso educativo (Figura 3). Entre los actores tenemos:

- Educación: Es la encargada de definir el proceso, el rol, las acciones a través de una pedagogía, didáctica, currículo y evaluación en el proceso de enseñanza – aprendizaje a través de las TIC.

- Informática: Posibilita la organización de los ambientes de aprendizaje con el apoyo de las tecnologías.
- Comunicación: Constructoras de relaciones e interacciones, ambientes de aprendizaje, medios, lenguaje y procesos comunicativos.
- Cognición: Que vela por el aprendizaje y la recreación del mundo a través de las “operaciones mentales” de los actores del proceso educativo.

Ambientes de aprendizaje: Establecen el escenario en el que interactúan los sujetos, a través de las relaciones pedagógicas, y las conexiones articuladas y posibilitadas por Tecnologías de Información y Comunicación (TIC) (Kendall & Kendall, 2005).

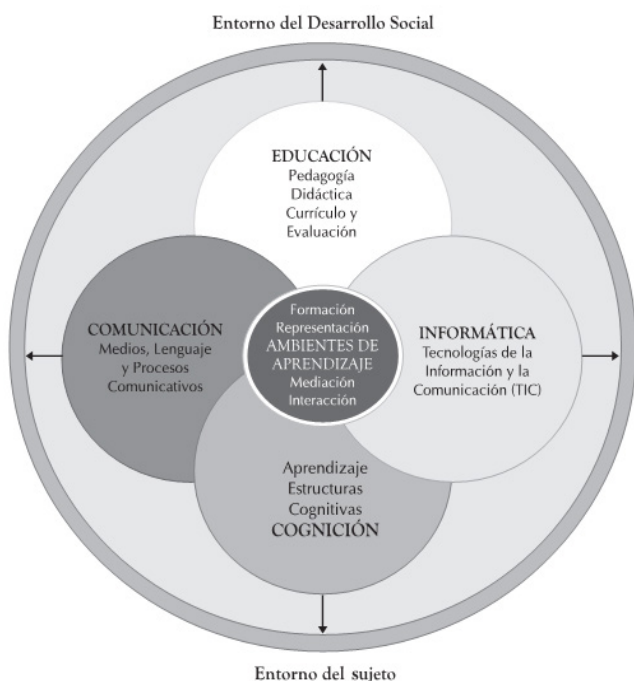


Figura 3. Relación: Actores del Ámbito Educativo.

Ambiente de aprendizaje híbrido

Es una plataforma educativa web que integra un conjunto de herramientas para la enseñanza-aprendizaje en línea, permite una enseñanza no presencial (e-learning) y/o una enseñanza mixta (b-learning), donde se combina la enseñanza por internet con experiencias en la clase presencial.

El principal objetivo de la plataforma virtual es la creación y gestión de espacios de enseñanza y aprendizaje en Internet, donde los profesores y los estudiantes se puedan interaccionar durante el proceso de enseñanza y aprendizaje (Silva, et al., 2007).

Las plataformas educativas son lugares donde se realiza un conjunto de procesos de enseñanza y aprendizaje dirigidos a la adquisición de una o varias competencias.

Así los espacios de aprendizaje pueden ser:

- Las aulas de un centro educativo, en la enseñanza presencial.
- Los sitios en Internet, en la enseñanza no presencial, virtual o e-learning.
- La combinación de las aulas del centro y los sitios en Internet, enseñanza mixta o b-learning.

Un Ambiente Híbrido de Aprendizaje, que sirva de apoyo al docente en el desarrollo de asignaturas usa un enfoque constructivista. El constructivismo es todo proceso de enseñanza-aprendizaje basado (1) en la reflexión de las personas, de manera que construyen su entendimiento de la realidad en base a las nuevas experiencias y el conocimiento previo. Para entender verdaderamente el material, los estudiantes deben descubrir ellos mismos los principios básicos. Este proceso constructivista conlleva a:

- El descubrimiento (investigación y experimentación).
- La manipulación de material concreto.
- La interacción social.
- La estimulación de todos los sentidos.

El Ambiente Híbrido de Aprendizaje Virtual le servirá porque podrá enseñar a los alumnos de manera presencial como siempre, así como también de forma virtual. Permitirá que al final de cada clase los estudiantes tengan acceso a los recursos proporcionados por el docente; con ello facilitará publicar horarios de tutorías, seminarios, exámenes, pero además podrá comunicarse virtualmente e intercambiar opiniones acerca del tema a tratar permitiéndole a los alumnos poder pensar, razonar e intercambiar conclusiones a la que él mismo ha llegado mediante su búsqueda del nuevo conocimiento basado en las pautas que le brindará el docente como base de su aprendizaje (Khogali, et al., 2011).

El ambiente híbrido de aprendizaje permitirá alcanzar las siguientes ventajas:

- El alumno podrá elegir libremente cuándo estudiar, y se le será más fácil porque podrá acceder al curso desde cualquier lugar que tenga acceso a internet.
- El alumno podrá contar con su tutor y solo hace falta estar conectado a internet para hacerle las preguntas pertinentes de caso, sin tener la necesidad de ir a un salón de clase.

Para el análisis y desarrollo del estudio se tiene como objetivo general: apoyar al docente en el desarrollo de asignaturas y usar el enfoque constructivista a través del ambiente híbrido de aprendizaje (Attwell, 2007).

Se plantean los objetivos específicos:

- Identificar las componentes esenciales de las asignaturas que deben ser controladas por el ambiente híbrido de aprendizaje.
- Analizar los recursos informáticos que utilizan actualmente los docentes en el desarrollo de asignaturas.
- Dotar al docente de información que facilite la preparación de materiales de cualquier asignatura y algún tema complementario a la misma.
- Diseñar un ambiente híbrido de aprendizaje que apoye al docente en el desarrollo de asignaturas con enfoque constructivista.

MATERIALES Y MÉTODOS

El desarrollo del estudio se sustenta sobre las bases de la investigación descriptiva y explicativa de acuerdo a la finalidad de la Investigación. En esta sección se analizarán las metodologías Rational Unified Process (RUP) y Extreme Programming (XP). La metodología seleccionada dependerá del tipo de proyecto y será la que más se ajuste a los requerimientos del proyecto y en la que se basará el siguiente trabajo de investigación (Rising & Janoff, 2000).

RUP (Rational Unified Process): Proceso Racional Unificado es una metodología tradicional que se caracteriza por:

- Ser operativo e incremental.
- Estar centrada en la arquitectura.
- Manejar el uso de requerimientos, metas bien definidas.
- Verifica la calidad del software.

La metodología RUP es un proceso de Ingeniería de Software planteado por Kruchten cuyo objetivo es producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecido. Cubre el ciclo de vida de desarrollo de software (otras consideraciones sobre RUP en el Anexo B, Rational Unified Process (RUP): Resumen de Etapas)

Una particularidad de esta metodología es que, en cada ciclo de iteración, se exige el uso de artefactos (documentos), al constituir, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software (Ramírez, 2002; Darwish, 2011).

Extreme Programming (XP): es una metodología ágil bien estructurada, se enfoca en la comunicación con el cliente y el trabajo en equipo. La metodología XP, cuyas siglas en inglés significan "Extreme Programming" (Programación Extrema) fue desarrollada por Kent Beck a finales de los años 90s, la cual es una disciplina de desarrollo que está basada en valores de simplicidad, comunicación y retroalimentación (Sohaib, et al., 2019). Trabaja al unir a todo el equipo en presencia de prácticas sencillas con suficiente retroalimentación para ajustar las prácticas a su situación única. XP recomienda aumentar la velocidad en el desarrollo del producto (algunas características sobre XP en el Anexo C, Extreme Programming (XP): Algunas Características)

Según los valores promovidos por XP se fundamentan en los siguientes doce prácticas (Wellington, 2005):

- Planificación Incremental.
- Testing.
- Programación en parejas.
- Refactorización.
- Diseño simple
- Propiedad colectiva del código.
- Integración continua.
- Cliente en el equipo
- Release pequeñas.
- Semanas de 40 horas.
- Estándares de codificación.
- Uso de Metáforas.

Comparación de las Metodologías: RUP y XP (tabla 2).

La metodología RUP presenta una carga excesiva de documentación a diferencia de XP que utiliza poca documentación. Sin embargo, RUP y XP ambas utilizan iteraciones en su desarrollo (Solís, 2000).

Tabla 2. Xp vs RUP.

| | Extreme Programming (XP) | Racional Unified Process (RUP) |
|------------------------------------|---|--|
| Tamaño de los equipos | Proyectos cortos y equipos más pequeños. | Proyectos y equipos grandes, en cuanto al tamaño y duración. |
| Carga de trabajo | XP es un proceso ligero. En el desarrollo de un proyecto con XP es más importante la entrega del software que necesita al cliente. | RUP es un proceso pesado. RUP define en cada momento del ciclo de vida del proyecto. |
| Relación con el cliente | Fuerte comunicación con el cliente. Feedback del cliente. | Se presentarán al cliente los artefactos del final de una fase y solo después de que el cliente acepte los artefactos generados se pasará a la siguiente fase. |
| Desarrollo | Las iteraciones de XP tienen por lo general una iteración menor que en RUP, puesto que la carga a llevar por los programadores a parte del desarrollo del propio software es menor. | Proceso iterativo. RUP genera también releases basados en los artefactos después de cada fase, pero en su caso no solo se limitan sólo al código, sino que las releases viene acompañada de todo lo que traería el producto final, es decir, notas de la versión, instrucciones de instalación, ayuda de uso, etc. |
| Código Fuente | XP es el único que presenta la compartición de código. | RUP opta por la compartición del código. |
| Conocimiento de la Arquitectura | Programación en parejas. Discusiones que llevan a mejores estructuras y algoritmos y que este proceso aumente la calidad de software. | En RUP se intentará reducir la complejidad del software a producir a través de una planificación intensiva. |
| Evaluación del estado del proyecto | XP define esos componentes pequeños para hacer un seguimiento de las mismas, pero la tarea del reporting recae solo en los jefes del proyecto. | RUP por su parte, es tan grande y complejo en ese sentido como en el resto, por lo que manejar el volumen de información que puede generar requiere mucho tiempo. |
| Punto Flacos | Pobre en cuanto a documentación debido a que XP es un proceso muy orientado a la implementación. | Mucha documentación y en ocasiones para el desarrollo de software por medio de equipos pequeños RUP es muy grande y prácticamente inalcanzable. |

RESULTADOS Y DISCUSIÓN

Metodología DX

Fases de la metodología dX

DX Inception

1. Los casos de uso principales son escritos en fichas por un representante del cliente con ayuda de los desarrolladores.
2. Se crean prototipos sencillos a partir de los casos de uso.
3. Los prototipos son utilizados para medir la velocidad de desarrollo del equipo y determinar el nivel de alcance y detalle de los casos de uso. Esta información se emplea para empezar con la agenda del proyecto y afinar las especificaciones de los casos de uso.
4. Los prototipos también se emplean para explorar algunas arquitecturas de sistemas.

En el hito que marca el fin de la fase dX Inception se debe tener una buena idea de los casos de uso principales, la agenda del proyecto y los primeros planteamientos de la arquitectura.

DX Elaboration

La mayor parte del código producido en la fase de dX Inception se descarta y se empieza con el diseño y la programación.

1. Se continúa con la elaboración de las fichas de caso de uso.

2. La cantidad de trabajo por cada caso de uso es estimada por los desarrolladores y es escrita en la ficha del caso de uso.
3. El cliente prioriza los casos de uso al considerar como principales a los de mayor riesgo.
4. Se planifican las iteraciones con una duración no mayor a una semana. El cliente selecciona las fichas de los casos de uso para desarrollar en cada iteración y la suma de la cantidad de trabajo estimado no debe superar al total planificado para la iteración. Al final de la iteración se debe registrar el factor de carga que es una razón calculada a partir del número de casos de uso desarrollados dividido entre el número de casos de uso que se estimó desarrollar.
5. Cada iteración es analizada y diseñada para estar dentro de los límites de la arquitectura del sistema identificado. El análisis y el diseño están a cargo de los desarrolladores y en las sesiones de diseño ellos pueden emplear los medios que les parezcan apropiados (por ejemplo: diagramas UML, tarjetas CRC).
6. Los modelos de diseño se deben plasmar en código por un par de desarrolladores y cuando se concluye en esta tarea el modelo puede ser descartado.
7. Las pruebas son muy importantes para dX, por lo que se desarrolla primero el código para realizar pruebas y luego el código a probar. Los casos de uso se dividen en unidades que pueden ser probadas y los desarrolladores alternan su actividad de programación con la creación de pruebas.
8. El diseño y el código sólo deben adquirir complejidad si así lo señalan los casos de uso.
9. La calidad del código es tan importante que las modificaciones son frecuentes y las pruebas aseguran que los cambios no introduzcan errores.
10. El código de los diversos módulos puede ser modificado por cualquier miembro del equipo de desarrollo.
11. La integración se realiza por lo menos con una frecuencia diaria.

DX Construction

La fase de construcción no se distingue de la fase de elaboración más que por la estabilidad de la arquitectura y la estabilidad del plan de proyecto. El equipo crea una agenda de lanzamientos a partir del factor de carga y la suma de las fichas de caso necesitadas para cada lanzamiento.

DX Transition

La fase de transición empieza después del primer lanzamiento y desde ese momento se tendrá un sistema en

función. Incluso cuando el sistema se encuentra en una etapa de modificaciones, la versión anterior puede ejecutarse en paralelo.

El equipo planificará las iteraciones y lanzamientos, buscará iteraciones realmente cortas para obtener retroalimentación del sistema ejecutándose en lapsos pequeños.

Justificación:

Al analizar estas dos metodologías en relación al presente Trabajo de investigación se optó por escoger la metodología Ágil RUP conocida como metodología dX, esta metodología dX es una metodología de desarrollo ágil que se inspira en un modelo mínimo de RUP y recoge algunas características de la metodología XP.

Esta metodología permite que el desarrollo del presente trabajo de investigación sea apto a los requerimientos de los usuarios. Este tipo de metodología se enfatiza más en la comunicación que en la documentación. Existe una comunicación más fluida entre el programador y los usuarios permite optimizar el tiempo en su desarrollo. Se minimizan los riesgos ya que el proyecto se realiza en lapsos de tiempo corto.

A continuación, se describe las fases, las iteraciones y los entregables de la metodología Ágil RUP
Fases de la Metodología Ágil RUP

1era Fase - Inicio:

Esta primera fase del proyecto comprende identificar el alcance del sistema, la captura de requerimientos, su arquitectura y la aceptación por parte de los usuarios. Esta fase, para el proyecto, tuvo una sola iteración y los artefactos generados fueron:

- Documento de Alcance del Proyecto. Este documento permite saber cuáles son los requerimientos funcionales y no funcionales, especificaciones de los principales casos de uso, con las descripciones simples.
- Documento de Estimación de Costo y Planificación. Este documento permite calcular el costo y el tiempo que dura el proyecto.
- Documento de Riesgo. Este documento permite saber si el riesgo asociado al proyecto está dentro del alcance permitido.
- Documento de Viabilidad. Este documento permite saber si el costo del proyecto y la tecnología disponible cumple con los requerimientos de los usuarios en el tiempo establecido.

2da Fase Elaboración:

En esta fase donde se comienza con el diseño, el interés principal estuvo en la especificación de los

requerimientos, análisis y diseño de la Plataforma Virtual, definición de la arquitectura de software y estrategia de pruebas. Al definir la arquitectura se utilizaron los principales requerimientos del proyecto. En el caso de las pruebas, se realizó la versión inicial de pruebas, que tiene como entrada los casos de uso seleccionados. Esta fase para el proyecto tuvo una sola iteración y los artefactos generados fueron:

- Especificación total de los casos de uso.
- Especificación de requisitos de software.
- Determinación de análisis y diseño. Se usan diagramas UML para representar los modelos de análisis y diseño.
- Especificaciones del plan de prueba que se realiza paralelamente a la codificación

Al finalizar la elaboración se tiene como resultado el documento de Especificación de Requisitos de Software completo, el documento de Análisis y el Catálogo Pruebas preliminar.

3era Fase Construcción:

En los proyectos dX, muchas veces las fases de elaboración y construcción son casi indistintas ya que de forma paralela va en evolución la arquitectura y la codificación.

Para el presente proyecto, conforme éste avance a la etapa de construcción se realiza la actualización de los documentos elaborados, en etapas previas y se concluye el Plan de Pruebas.

4ta Fase Transición:

En la mayoría de los proyectos dX, la fase de transición empieza luego del primer reléase.

Hay que mencionar que gran parte del proceso está basado en la creación de casos de usos.

Se evaluaron las condiciones técnicas, operativas y económicas que aseguren el cumplimiento de las metas y objetivos del presente proyecto, y así, determinaron su viabilidad.

Requerimientos Funcionales

Los requerimientos funcionales cubren las funciones y operaciones a realizar para proporcionar un sistema que operará de acuerdo a las necesidades del usuario. Al obtener un detalle de las apreciaciones de los usuarios respecto a sus requerimientos, se definen las funciones que tendrán que ser realizadas por la Plataforma Virtual a desarrollar.

Gestión de Cuentas

- Solo el administrador general del campus puede crear y eliminar cuentas, así como modificar el password asignado
- Para crear una cuenta es necesario verificar que se trate de un alumno inscrito o de un profesor.

Gestión de Alumnos

- Permitir el registro de nuevos alumnos al proporcionar los datos personales y académicos necesarios.
- Permitir la consulta de los datos de un alumno por su código.
- Permitir eliminar y modificar los datos de un alumno.
- Permitir la inscripción de alumnos matriculados en el curso.
- Permitir la modificación de calificaciones.

Gestión de Profesores

- Permitir el registro de profesores al proporcionar los datos personales y estudios realizados.
- Permitir las consultas de profesores, así como la modificación y eliminación de sus datos.

Gestión de Materias

- Es necesario controlar las materias que se imparten a partir del conocimiento de un plan de estudios.
- Poder registrar, modificar y eliminar materias del plan de estudios.

Gestión de Asignaturas

- Poder registrar uno o varios profesores a una Asignatura correspondiente al plan de estudios.
- Permitir la consulta, eliminación y modificación de las Asignaturas creadas.

Otros requisitos

1. El sistema debe ser funcional y ajustarse a los propósitos educativos para el cual fue diseñado.
2. El sistema permitirá la autenticación de los usuarios mediante nombre de usuario y contraseña.
3. Garantizar la integridad de los datos.
4. Programar periódicamente respaldos de la información de la Plataforma Virtual.
5. Disponibilidad las 24 horas del día.
6. Apoyar el desarrollo de las actividades a través de Internet.

Este módulo permitirá registrar cursos al relacionar una materia del plan de estudios con el profesor asignado a dicho curso, también permitirá la consulta, eliminación y modificación de las asignaturas creadas (figura 4, 5 y 6).

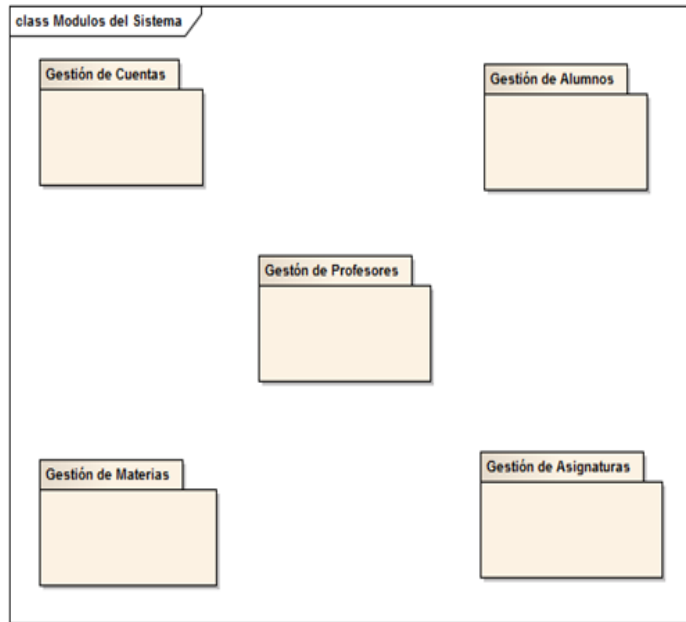


Figura 4. Módulos del sistema: Diagrama de Paquetes.

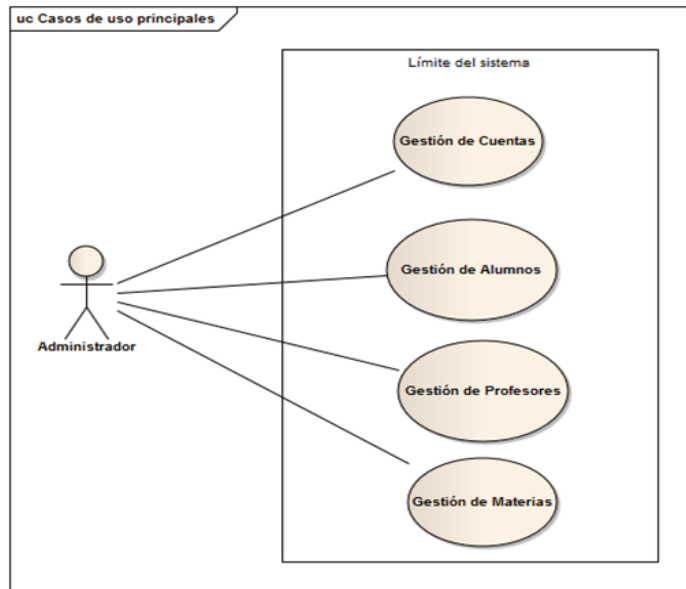


Figura 5. Caso de uso para los requisitos funcionales.

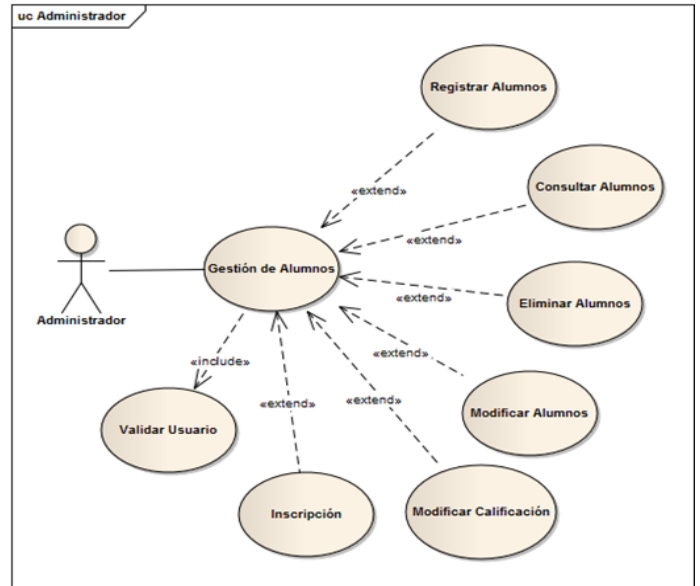


Figura 6. Especificación de un caso de uso.

Confiabilidad del Sistema

El sistema es confiable en cuanto a seguridad ya que la información que se maneja por parte de los usuarios es de carácter confidencial y se mantendrá en reserva; sólo será accedida por las personas adecuadas. Se trabaja con validación de usuarios, por ende, sólo aquellos con los permisos adecuados pueden actualizar o visualizar los datos.

Facilidad de Uso

A pesar de su versatilidad, el ambiente híbrido de aprendizaje es fácil de usar. No es necesaria programación para crear contenidos atractivos. La interfaz utilizada (tanto para el alumno como para el profesor) es bastante intuitiva y fácil se aprende su uso (Figura 7).

Arquitectura de Servidores

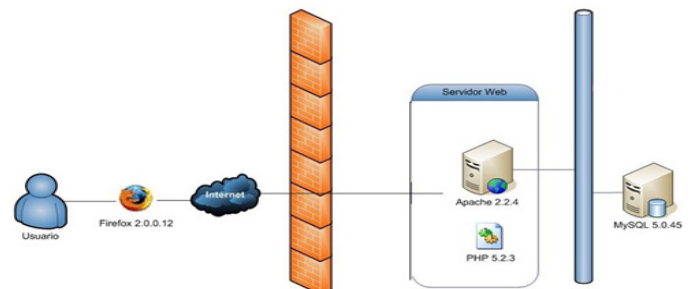


Figura 7. Arquitectura del servidor.

Arquitectura de la solución

El primer paso dentro del desarrollo de la Plataforma Virtual fue el de definir una arquitectura que permitiera garantizar la interoperabilidad del producto, para ello los servicios web debían de cumplir los siguientes requisitos:

- Los servicios web deben ser accesibles desde cualquier sistema de conexión, tanto actual como futura, y deben de poder ser invocados independientemente del lenguaje utilizado para ello (interoperabilidad).
- La estructura de los servicios web debe de desarrollarse de tal manera que, aunque se realicen cambios dentro de core de Moodle, sea necesario realizar pocas o ninguna modificación de API.
- Las funciones que conforman la API deben ser ampliables para favorecer las contribuciones.

El sistema web debe adaptarse al sistema de privilegios de Moodle (capabilities) para garantizar la seguridad. De acuerdo a estas series de requisitos, los servicios web de Moodle 2.0 están divididos en tres capas fundamentales:

1. Conectores. Hasta el momento se puede conectar con la plataforma a través de 4 protocolos, estos son: REST, SOAP, XML-RPC y AMF(Flash)
2. Externallib. Esta capa está formada por un conjunto de ficheros denominados externallib.php los cuales se encuentran expandidos por todo el árbol de directores de Moodle. Dichos ficheros son llamados desde los conectores y en ellos se encuentran todas las funciones que se ofrecen en la API de los servicios web. Resume todas las funcionalidades de Moodle para ofrecerles al exterior, permite gestionar aspectos relacionados con los usuarios de la Plataforma, por ejemplo, chequear los permisos de los usuarios.
3. Núcleo de Moodle. La capa de núcleo de Moodle está formada por todas las librerías que contienen funciones que puedan interesar dentro de la capa de los externallib, es decir, funciones relaciones con los usuarios, los cursos, los grupos, etc. Esta capa ha sido mejorada en Moodle 2.0 porque muchas de estas funciones imprimían mensajes de error en pantalla cuando había algún problema, por lo que se han reescrito parte de estas funciones del núcleo para que en caso de error devuelvan excepciones (hasta ahora Moodle no poseía una API y gracias a estos cambios es que se genera una).

Diagrama de clases de diseño (Figura 8)

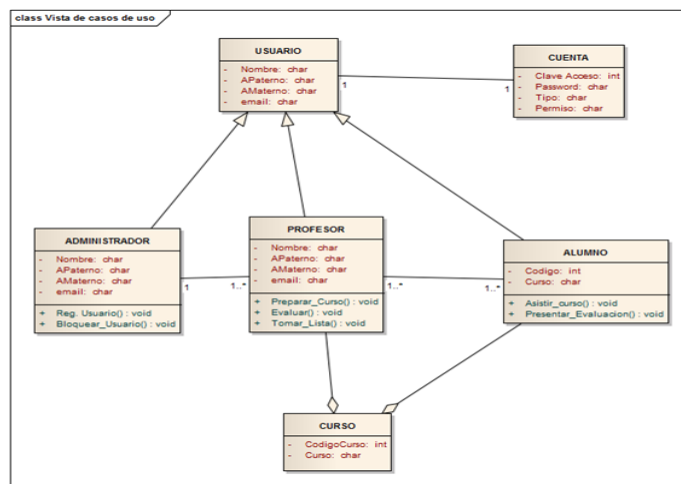


Figura 8. Diagrama de Clases de Diseño.

Diseño del ambiente híbrido de aprendizaje

En el siguiente diagrama se muestra el mapa web que conforma la aplicación desarrollada.

Se pretende que el presente trabajo sea utilizado como punto de partida para:

- Incorporar otras metodologías de autoevaluación.
- Profundizar el análisis de gestión.
- Incorporar criptografía en la gestión de claves, al esquematizar la seguridad.
- Desarrollar nuevas páginas para el ingreso de comentarios del alumno (Figura 9).

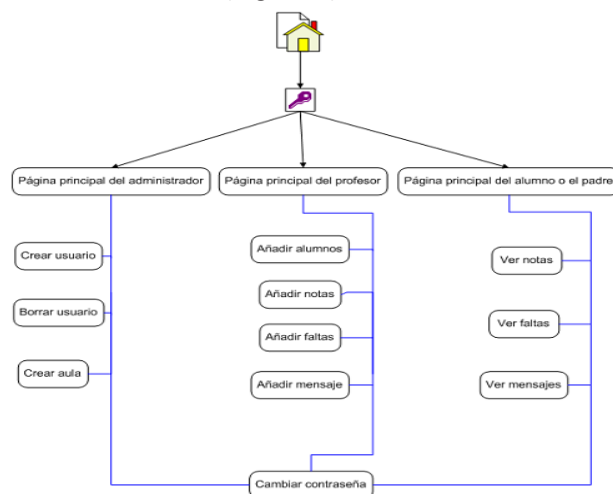


Figura 9. Diseño de la Plataforma.

CONCLUSIONES

Se concluye del desarrollo del estudio que:

Del análisis desarrollado, surge de manera clara, una diferencia clave entre el b-learning y la enseñanza presencial tradicional: "la flexibilidad".

Este sistema desarrollado presenta una experiencia diferente para el alumno mediante un diseño pedagógico que combina la formación presencial con las virtualidades de un diseño de medida. En su dimensión pedagógica contempla la integración de recursos tecnológicos en busca de resultados formativos aplicables a necesidades de aprendizaje individualizadas.

En la actualidad, el paradigma orientado a objetos es cada vez más utilizado para construir software, pues sus ideas se basan en la concepción moderna relacionada con los avances de la producción industrial.

Con respecto a las tecnologías y software utilizados se ha podido comprobar sus potenciales beneficios con respecto al soporte multiplataforma, indispensables por su eficiencia y rapidez al momento de desarrollar la aplicación con Php y MySql entorno Windows.

REFERENCIAS BIBLIOGRÁFICAS

- Álvarez-Gómez, G., Viteri-Moya, J., Estupiñán-Ricardo, J., & Viteri-Sánchez, C. B. (2020). Evaluating Strategies of Continuing Education for Academics Supported in the Pedagogical Model and Based on Plithogenic Sets. *Neutrosophic Sets and Systems*, 37, 16-23.
- Attwell, G. (2007). Entornos de aprendizaje personal: el futuro del eLearning. *Documentos de aprendizaje electrónico*, 2 (1), 1-8.
- Darwish, N. R. (2011). Improving the Quality of Applying eXtreme Programming (XP) Approach. *International Journal of Computer Science and Information Security*, 9(11), 1-16.
- Estupiñán Ricardo, J., Domínguez Menéndez, J. J., & Maldonado Manzano, R. L. (2020). Integración universitaria, reto actual en el siglo XXI. *Revista Conrado*, 16(S 1), 51-58.
- George-Walker, L. D., & Keeffe, M. (2010). Self-determined blended learning: a case study of blended learning design. *Higher Education Research & Development*, 29(1), 1-13.
- Kendall, K., & Kendall, J. E. (2005). *Análisis y Diseño de Sistemas*. Pearson Education.
- Khogali, S. E. O., Davies, D. A., Donnan, P. T., Gray, A., Harden, R. M., McDonald, J., ... & Yu, N. (2011). Integration of e-learning resources into a medical school curriculum. *Medical teacher*, 33(4), 311-318.
- O'Neill, E., Stevens, N. T., Clarke, E., Cox, P., O'Malley, B., & Humphreys, H. (2011). Use of e-learning to enhance medical students' understanding and knowledge of healthcare-associated infection prevention and control. *Journal of Hospital Infection*, 79(4), 368-370.
- Planella, J., Rodríguez, I., & Warschaver, M. (2006). Aspectos sociales y culturales del e-learning. Desarrollo organizativo del e-learning. Editorial Eureca Media S.L.
- Pressman, R. (1998). *Ingeniería del Software: Un enfoque Práctico*. Mc Graw-Hill.
- Ramírez Reyes, G. (2002). *Metodología para Auditoría Informática en Entidades Públicas*. (Tesis de Maestría), Universidad Nacional de Ingeniería.
- Rising, L., & Janoff, N. S. (2000). The Scrum software development process for small teams. *IEEE software*, 17(4), 26-32.
- Sandars, J., Homer, M., Pell, G., & Croker, T. (2008). Web 2.0 and social software: the medical student way of e-learning. *Med Teach*.
- Silva, M., Barrera, A., Arroyave, J., & Pineda, J. (2007). Un método para la trazabilidad de requisitos en el Proceso Unificado de Desarrollo. *Revista EIA*, 8, 69-82.
- Sohaib, O., Solanki, H., Dhaliwa, N., Hussain, W., & Asif, M. (2019). Integrating design thinking into extreme programming. *Journal of Ambient Intelligence and Humanized Computing*, 10(6), 2485-2492.
- Solís, V. (2000). *Modelos de Optimización Matemática para el Sistema de Soporte de Decisiones en el Planeamiento a Largo Plazo en una Mina a Tajo Abierto*. (Tesis de Maestría) Universidad Nacional de Ingeniería.
- Wellington, C. A. (2005). Managing a project course using extreme programming. (Ponencia). *IEEE Frontiers in Education 35th Annual Conference*. Indianapolis, USA.