

Universidad de las Ciencias Informáticas
Facultad 4



**Trabajo de Diploma para optar por el título de Ingeniero en
Ciencias Informáticas**

**Módulo de reportes para la Plataforma
Educativa XAUCE ZERA**

Autora: Beatriz Vargas Más

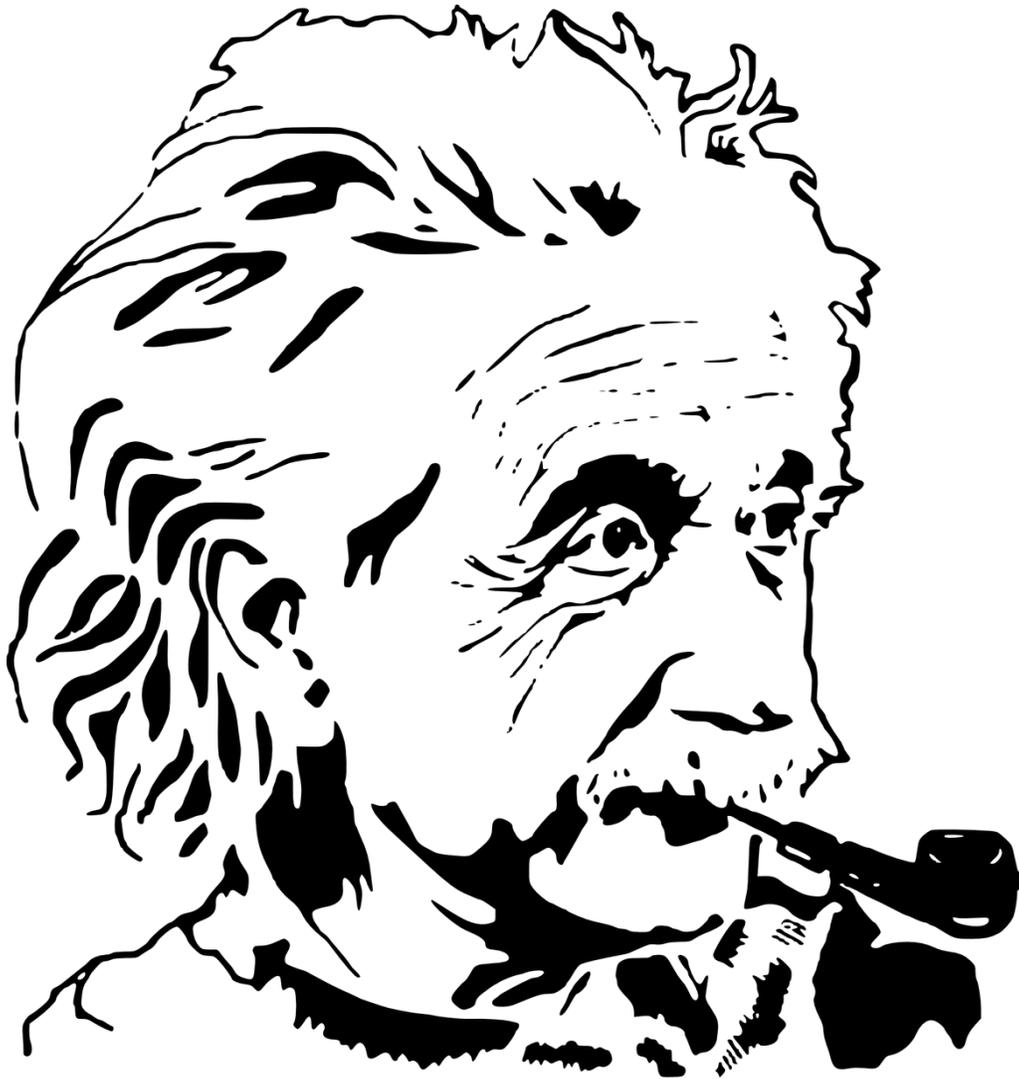
Tutores: Ing. Miguel Medina Ramírez

Ing. Lisset Salazar Gómez

Co-tutor: Ing. Alex Acosta Montejo

La Habana, Junio de 2019

“Año 61 de la Revolución”



“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”

Albert Einstein

DECLARACIÓN DE AUTORÍA

Declaro ser la única autora del presente Trabajo de Diploma y reconozco al Centro de Tecnologías para la Formación (FORTES) de la Facultad 4 de la Universidad de las Ciencias Informáticas, los derechos patrimoniales del mismo, con carácter exclusivo, para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Beatriz Vargas Más

Miguel Medina Ramírez

Firma del Autor

Firma del Tutor

Lisset Salazar Gómez

Alex Acosta Montejo

Firma del Tutor

Firma del Co-tutor

DEDICATORIA

“A mi madre Carmen Emilia Más Cantero”

AGRADECIMIENTOS

“A mis padres por haber hecho de mi la persona que soy, en especial a mi madre por todo el apoyo y por todo el sacrificio que hace día a día, por ser la mejor madre del mundo.”

“A mis hermanos Elier, Karla, Ernesto y Alexis por su incondicional amor y cariño.”

“A mi compañero de Elián por ser estar en todos los momentos difíciles y tener tanta paciencia.”

“A todos mis amistades de la universidad, a los que están y a los que no, gracias por ser hermanos para mí.”

“A mis tutores Miguel Medina Ramírez y Lisset Salazar Gómez por la ayuda brindada y por contribuir con mi formación como profesional y en especial a mi co-tutor Alex Acosta Montejo por la ayuda, la paciencia y dedicación.”

“A todo aquel que contribuyó con mi formación y que de una manera u otra me ha ayudado a lograr este sueño.”

A todos muchas gracias.

Resumen

Los avances tecnológicos actuales han abierto posibilidades de innovación en el ámbito educativo. Estos permiten potenciar el ejercicio de la enseñanza, el aprendizaje de calidad y el desarrollo profesional de los docentes a través de las plataformas educativas. Este tipo de plataforma genera un gran cúmulo de información, por lo que se ha hecho necesario contar con un espacio que permita gestionar toda la información contenida en base de datos. Al realizar un análisis de la gestión de la información que se genera en la Plataforma Educativa XAUCE ZERA de la Universidad de las Ciencias Informáticas, se obtuvo que es necesario desarrollar un módulo que sea capaz de manejar toda la información importante de la plataforma. Este módulo le facilitará el trabajo a los administradores y a los profesores, al organizarle la información que resulte más importante para cada rol. Para el desarrollo del módulo se empleó como guía la metodología AUP-UCI. Además, se utilizaron diversas tecnologías para la implementación de las funcionalidades pedidas por el cliente. En la última fase de desarrollo, se le realizaron pruebas al módulo de reportes, para asegurarse que cumple correctamente con sus funcionalidades. Como resultados de todo el minucioso trabajo realizado se obtuvo un módulo de reportes que agiliza la gestión de la información contenida en las bases de datos de la Plataforma Educativa XAUCE ZERA.

Palabras clave: Gestión, reporte, módulo de reporte, plataforma educativa.

Abstract

Current technological advances have opened up possibilities for innovation in the field of education. These make it possible to promote the exercise of teaching, quality learning and the professional development of teachers through educational platforms. This type of platform generates a large accumulation of information, so it has become necessary to have a space to manage all the information contained in the database. By carrying out an analysis of the management of the information generated in the XAUCE ZERA Educational Platform of the University of Computer Science, it was obtained that it is necessary to develop a module that is capable of handling all the important information of the platform. This module will make it easier for administrators and teachers to organize the most important information for each role. For the development of the module, the AUP-UCI methodology was used as a guide. In addition, various technologies were used to implement the functionalities requested by the client. In the last development phase, the reports module was tested to ensure that it correctly complies with its functionalities. As a result of all the meticulous work carried out, a report module was obtained that streamlines the management of the information contained in the databases of the XAUCE ZERA Educational Platform.

Keywords: Management, report, report module, educational platform.

ÍNDICE

Introducción	1
Capítulo I. Fundamentación teórica de la investigación	6
1.1 Conceptos asociados al dominio de la investigación	6
1.2 Sistemas similares	7
1.3 Herramientas, lenguajes y metodología	12
1.4 Marco de trabajo Xalix	23
Conclusiones del capítulo.....	24
Capítulo II. Análisis y diseño de la solución	25
2.1 Propuesta de solución	25
2.2 Requisitos funcionales	26
2.3 Requisitos no funcionales.....	33
2.4 Historia de usuarios	34
2.5 Arquitectura de software	36
2.6 Patrones del diseño.....	37
2.7 Modelo de datos	41
2.8 Modelo de análisis.....	43
2.9 Modelo de diseño	45
Conclusiones del capítulo.....	48
Capítulo III. Implementación y pruebas de la solución	49
3.1 Modelo de implementación	49
3.2 Pruebas de software	52
Conclusiones del capítulo.....	62
Conclusiones generales.....	63
Recomendaciones	64
Referencias bibliográficas	65
Anexos.....	70

ÍNDICE DE FIGURAS

Figura 1. Elaboración propia. Diagrama de la propuesta de solución.....	25
Figura 2. Patrón Modelo-Vista-Controlador. Fuente: (57).....	37
Figura 3. Elaboración propia. Patrón Experto en la solución.	38
Figura 4. Elaboración propia. Patrón Creador en la solución.....	38
Figura 5. Elaboración propia. Patrón Controlador en la solución.	39
Figura 6. Elaboración propia. Patrón Singleton en la solución.....	40
Figura 7. Elaboración propia. Patrón Decorador en la solución.....	40
Figura 8. Elaboración propia. Patrón Inyección de dependencias en la solución.....	41
Figura 9. Elaboración propia. Modelo de datos de Zera.	42
Figura 10. Elaboración propia. Modelo de datos de la solución.....	43
Figura 11. Elaboración propia. Diagrama de clases del análisis.....	44
Figura 12. Elaboración propia. Diagrama de colaboración para requisitos con filtro cursos.	44
Figura 13. Elaboración propia. Diagrama de clases del diseño actividades de un curso. 46	
Figura 14. Elaboración propia. Diagrama de clases del diseño usuarios registrados en la plataforma.....	47
Figura 15. Elaboración propia. Diagrama de secuencia para requisitos con filtro cursos. 48	
Figura 16. Elaboración propia. Diagrama de componentes.	50
Figura 17. Elaboración propia. Diagrama de despliegue.	50
Figura 18. Elaboración propia. Estándar de codificación en archivo JavaScript.	51
Figura 19. Elaboración propia. Estándar de codificación en archivo html.twig.....	51
Figura 20. Elaboración propia. Estándar de codificación en clase PHP. Servicios.	52
Figura 21. Elaboración propia. Estándar de codificación en clase PHP. Namespace.....	52
Figura 22. Elaboración propia. Método exportAction.	54
Figura 23. Elaboración propia. Grafo de flujo.	55
Figura 24. Elaboración propia. Vista para el administrador.	57
Figura 25. Elaboración propia. Vista para el profesor.....	58
Figura 26. Elaboración propia. No conformidades.....	61

ÍNDICE DE TABLAS

Tabla 1. Análisis de sistemas similares	9
Tabla 2. Comparación de las herramientas de reporte	14
Tabla 3. Comparación de los diseñadores de reporte.....	16
Tabla 4. Requisitos Funcionales.....	26
Tabla 5. Historia de usuario. Actividades de un curso	34
Tabla 6. Historia de usuario. Usuarios registrados en la plataforma	35
Tabla 7. Caso de prueba. Ruta independiente 1.....	56
Tabla 8. Caso de prueba. Ruta independiente 2.....	56
Tabla 9. Caso de prueba. Ruta independiente 3.....	56
Tabla 10. Caso de prueba. Actividades de un curso.....	59
Tabla 11. Caso de prueba. Usuarios registrados en la plataforma.....	60

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) han contribuido al desarrollo en diferentes esferas de la sociedad entre las que se encuentra la educación. Los avances tecnológicos abren posibilidades de innovación en el ámbito educativo, que llevan a repensar los procesos de enseñanza-aprendizaje y a llevar a cabo un proceso continuo de actualización profesional. La Pedagogía encuentra en las TIC nuevas actividades profesionales (1): análisis y evaluación de los recursos tecnológicos y su uso educativo, integración de los medios de comunicación para lograr el aprendizaje, así como, diseño de estrategias educativas para favorecer la integración de recursos tecnológicos en diferentes ambientes de aprendizaje (1).

Los cursos en línea son de los nuevos modelos de educación que han ganado auge en los últimos tiempos, los mismos potencian el ejercicio de la enseñanza, el aprendizaje de calidad y el desarrollo profesional de los docentes. Los Cursos Masivos Abiertos en Línea (MOOC, del inglés "Massive Open Online Course"), están dirigidos a un amplio número de participantes a través de Internet según el principio de educación abierta (2). Bajo las siglas TIC se puede encontrar gran cantidad de información, como: los Sistemas Gestores de Contenido (CMS, del inglés "Content Management Systems"), y sus variantes como los Sistemas de Gestión de Aprendizaje (LMS, del inglés "Learning Management Systems"), sinónimo de Entornos virtuales de aprendizaje (EVA) (3). Las plataformas educativas también son LMS, pues permiten que los profesores puedan crear cursos sobre distintos temas y gestionar sus elementos. En la actualidad, estas se usan para complementar las técnicas docentes presenciales (4).

Las plataformas educativas son una herramienta física, virtual o una combinación de ambas, que brinda la capacidad de interactuar con uno o varios usuarios con fines pedagógicos. Se considera, además, que contribuye en la evolución del proceso de enseñanza-aprendizaje, complementando o presentando alternativas a las prácticas de educación tradicional (5). Este tipo de sistema genera un gran cúmulo de información, que puede ser administrada para lograr una buena toma de decisiones a partir del análisis de la información contenida en las bases de datos.

Cuba presta especial atención al desarrollo económico social del país y al bienestar de la población. En tal sentido, se ha involucrado en la creación de sistemas informáticos fomentando los avances de la informática y del proceso de informatización de la sociedad,

lo cual ha contribuido con el apoyo y desarrollo del proceso enseñanza- aprendizaje (6). El país cuenta con varias universidades que promueven el desarrollo de soluciones libres, entre las que se encuentra la Universidad de la Ciencias Informáticas, líderes en la migración del país a tecnologías de software libre y código abierto (7).

Desde sus inicios, la Universidad de las Ciencias Informáticas (UCI) está encargada de la formación y superación de profesionales de la informática, aportando al país personal capacitado que apoye el proceso de desarrollo tecnológico. En la UCI existen varios centros de desarrollo de software entre los que se encuentra el Centro de Tecnologías para la Formación (FORTES) perteneciente a la Facultad 4, el cual se encarga de ofrecer servicios y productos informáticos para la educación. Uno de los productos del centro es la Plataforma Educativa XAUCE ZERA 2.0, la cual apoya el proceso de enseñanza- aprendizaje y proporciona a los profesores y aprendices un sistema integrado en línea único, robusto, seguro y fácil de usar para crear ambientes de aprendizaje personalizados. La plataforma permite ser empleada tanto en clases pequeñas, así como, soportar las necesidades de grandes organizaciones debido a su flexibilidad y escalabilidad. También aporta a la enseñanza y el aprendizaje con tecnología educativa innovadora que ayuda a los profesores a adaptarse a las nuevas normas y personalizar el aprendizaje ofreciendo experiencias nuevas y emocionantes de aprendizaje digital y garantizando que todos los estudiantes tengan la oportunidad de desarrollar todo su potencial (8).

A partir de una encuesta de satisfacción realizada por el Centro Nacional de Educación a Distancia (CENED) sobre el funcionamiento de la Plataforma Educativa XAUCE ZERA a los profesores que más hacen uso de esta, se identificaron algunos problemas existentes que provocan el aumento de tiempo y esfuerzo en el análisis de los datos, dificultando para ellos la toma de decisiones. Los aspectos fundamentales identificados que afectan el trabajo de los profesores son:

- No controlan las acciones realizadas por los estudiantes sobre los contenidos y las actividades de los cursos, por lo que no se les puede evaluar correctamente.
- No saben cuál es el tema más interesante de un curso para los estudiantes, lo que trae consigo que no se pueda lograr una mayor motivación de estos.
- No pueden identificar los ejercicios donde los estudiantes presentan mayor dificultad y, por tanto, los contenidos que requieren mayor tiempo de estudio para los estudiantes.

- No conocen si en su curso están matriculados todos sus estudiantes del grupo, lo que conlleva que el profesor no tenga un control total del mismo.
- No les brinda información del uso de las wikis¹ del curso por los estudiantes, por lo que se desconoce la interacción de estos ya sea para agregar o consultar contenido.

Por otro lado, los administradores después de realizar un análisis minucioso de la plataforma identificaron los siguientes problemas que les dificulta su trabajo:

- No controlan las acciones de los usuarios en el sistema, por lo cual no se registran las trazas de los mismos.
- No pueden identificar los cursos que presentan un mayor número de recursos, lo que implica que puede sobrecargar la plataforma y ralentizarla².
- No se conoce a cuáles cursos se les han realizado copias de seguridad, teniendo como desventaja que pueda haber alguno sin respaldo y en caso de existir algún fallo se podría perder su información.

Las limitaciones mencionadas anteriormente, conllevan a que no exista una mejor gestión de la información en la Plataforma Educativa XAUCE ZERA, ya que en la misma se genera un gran volumen de datos que se encuentran dispersos. Esto trae consigo un aumento en tiempo y esfuerzo del análisis de la información tanto para los profesores como para los administradores, dificultando la toma de decisiones para un mejor uso de la plataforma. Además, esta no cuenta con un espacio que permita organizar la información sobre los cursos y la interacción de los estudiantes con estos.

A partir de la situación problemática descrita anteriormente se plantea como **problema a resolver**: ¿Cómo contribuir con la gestión de la información generada en la Plataforma Educativa XAUCE ZERA para el apoyo a la toma de decisiones?

Este trabajo tiene como **objeto de estudio** la gestión de información en plataformas educativas.

Para dar solución al problema a resolver se define como **objetivo general**: desarrollar un módulo que permita la gestión de la información generada en la Plataforma Educativa

¹ *Wiki* es una mezcla o combinación de ideas de varias personas sobre un determinado tema, ya sean profesores o estudiantes (70).

² Hacer más lenta una actividad o un proceso.

XAUCE ZERA para el apoyo a la toma de decisiones, enmarcado en el **campo de acción** de la gestión de información de la Plataforma Educativa XAUCE ZERA.

Para lograr el cumplimiento del objetivo general se proponen los siguientes **objetivos específicos**:

1. Caracterizar plataformas educativas para determinar cómo se gestiona la información en ellas.
2. Definir las tecnologías, herramientas y metodología de desarrollo de software que serán utilizadas para el cumplimiento del objetivo general.
3. Desarrollar la propuesta de solución en correspondencia con las fases de metodología seleccionada.
4. Ejecutar las pruebas de software a las funcionalidades implementadas de la solución para comprobar su correcto funcionamiento.

Métodos teóricos:

- **Analítico-Sintético:** Este método permitió realizar el estudio teórico a través del análisis de documentos, libros, artículos y otras fuentes bibliográficas de diferentes autores.
- **Histórico-lógico:** Se utilizó para para evidenciar el avance que ha tenido la utilización de las Tecnologías de la Información y las Comunicaciones en la educación hasta la actualidad mediante el desarrollo de plataformas vinculadas al proceso enseñanza-aprendizaje.

Métodos empíricos:

- **Consulta bibliográfica:** Se pone de manifiesto al revisar la documentación sobre los motores de reportes y sus procesos administrativos.
- **Entrevista:** Se utilizó este método con el fin de precisar el problema a resolver, así como valorar la propuesta que se presenta, identificando de esta forma los requisitos con que puede contar la Plataforma Educativa XAUCE ZERA.

El presente trabajo de diploma estará estructurado en III capítulos:

Capítulo I: Fundamentación teórica de la investigación.

En este capítulo se fundamenta el marco teórico de la investigación en el que se hace un análisis valorativo de algunas de las herramientas para la generación de reportes. Se caracteriza la metodología de desarrollo, las herramientas, tecnologías y lenguaje de

programación utilizados en el proceso de desarrollo de la propuesta de solución para la Plataforma Educativa XAUCE ZERA.

Capítulo II: Análisis y diseño de la solución.

En este capítulo se realiza el análisis y diseño de la propuesta de solución para la Plataforma Educativa XAUCE ZERA. Se definen los requisitos funcionales y no funcionales necesarios para dar cumplimiento al objetivo general de la investigación.

Capítulo III: Implementación y pruebas de la solución.

En este capítulo se muestra las actividades de implementación y prueba de la propuesta solución. Además, se aplican métodos y técnicas de prueba relacionados con la implementación y el cumplimiento con los requisitos de software definidos.

Capítulo I. Fundamentación teórica de la investigación

Este capítulo está dirigido a plantear todos los elementos teóricos que sustentan el objetivo de la investigación. Se realiza un estudio de las herramientas existentes para generar reportes, así como, de soluciones similares en plataformas educativas. También se presentan las diferentes tecnologías y lenguajes, así como la metodología que se emplea en el proyecto y sirve de apoyo para dar cumplimiento a los objetivos, las tareas propuestas y de esta forma dar solución al problema planteado.

1.1 Conceptos asociados al dominio de la investigación

Para la mejor comprensión del objetivo del presente trabajo de diploma es necesario conocer los siguientes conceptos:

Plataforma educativa

Existen varios conceptos de plataformas educativas, dependiendo de cada autor, por ello a continuación se mencionan dos de ellas:

Una plataforma educativa es un software de aplicación web que permite administrar, gestionar e impartir cursos en línea. Se le conoce también como: sistema de gestión de aprendizaje o sistemas integrados para educación distribuida (SIED) (9).

Las plataformas educativas son una herramienta física, virtual o una combinación de ambas, que brinda la capacidad de interactuar con uno o varios usuarios con fines pedagógicos. Se considera, además, que contribuye en la evolución del proceso de enseñanza-aprendizaje, complementando o presentando alternativas a las prácticas de educación tradicional (5).

Existen varias plataformas educativas que se clasifican según el tipo de funciones específicas con las que cuentan. Todas tienen un único objetivo el cual es diseñar, crear y gestionar cursos en línea en donde el profesor interactúe con el alumno. Dentro de sus clasificaciones se tienen las de software libre, que son gratuitas para el público, pero no tiene todos los privilegios. También, las comerciales, que cuentan con todos los privilegios debido a que pagan para poder utilizarla. Por último, se encuentran las de desarrollo propio, las cuales se crean según las necesidades de la institución que la implemente (10). En estos tipos de plataformas educativas se genera y maneja mucha información que debe ser gestionada.

Gestión de información

La gestión de información en plataformas educativas tiene como propósito promover la eficacia organizacional al elevar las potencialidades de la organización para cumplir con las demandas del ambiente interno y externo en condiciones tanto dinámicas como estables (11). Es todo lo que tiene que ver con obtener la información correcta, en la forma adecuada, para la persona indicada, al costo correcto, en el momento oportuno, en el lugar indicado, para tomar la acción precisa (12). Para una mejor gestión de la información de un sistema se realizan reportes, que permiten obtener un mayor entendimiento de los datos.

Reporte

Un reporte es un informe o una noticia que puede ser impreso, digital o audiovisual, que pretende transmitir una información (13). En el ámbito de la informática, los reportes son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios. El reporte, de esta forma, confiere una mayor utilidad a los datos. Los reportes tienen diversos niveles de complejidad, desde una lista o enumeración hasta gráficos mucho más desarrollados. En el caso de los informes impresos, el texto suele ir acompañado por gráficos, diagramas, tablas de contenido y notas al pie de página (14).

Generador de Reportes

Los generadores de reportes son herramientas complementarias de los sistemas para la gestión de información. Por medio de estos se realizan consultas a la base de datos del sistema para obtener información en forma de reporte. Las consultas que se realizan son un lenguaje transparente al usuario. Los reportes o informes que se generan permiten visualizar la información de manera estructurada y resumida. En el uso de los generadores de reportes, siempre se tienen en cuenta los sistemas gestores de bases de datos soportados, y la forma en que se acoplaría a los sistemas y aplicaciones que lo usan. Los generadores de reportes están compuestos por dos elementos básicos, un diseñador o editor de informes y un motor de generación (15).

1.2 Sistemas similares

Para un mejor desarrollo del módulo es necesario conocer cómo se gestiona la información en las plataformas educativas. Con el objetivo de lograr un mejor entendimiento, se investigaron cuatro de las plataformas más utilizadas: Dokeos, Sakai, Moodle y Blackboard.

Dokeos

Dokeos inicialmente era una versión modificada de Claroline³, pero en la actualidad es una distribución independiente. Es una aplicación libre, aunque algunas de las herramientas que puede incluir no lo son, esto condiciona que existan distribuciones libres y propietarias. Actualmente, se distribuye en cuatro versiones (Libre, Educación, Pro Y Medical) dependiendo de las herramientas o el soporte que incluye. Las funcionalidades varían según la versión elegida, y en su versión gratuita, brinda las siguientes funcionalidades: crear plantillas de contenido, gestionar tutoriales, crear y organizar encuestas, visualizar informes acerca del progreso de los alumnos, entre otras. El estudiante puede llegar a tener una sobrecarga de información y la gran cantidad de herramientas que posee dificulta su aprendizaje y creación de cursos por parte de los docentes, debiendo mejorar su herramienta de creación de contenidos (16) (17). Reúne e integra todos los componentes necesarios para permitir la gestión, administración, comunicación, evaluación y seguimiento de las actividades de enseñanza-aprendizaje en el espacio virtual (18).

Moodle

Moodle (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular) es un entorno modular de código fuente abierto (software libre). Permite crear 8 tipos diferentes de extensiones: módulos, bloques, filtros, tareas, tipos de cursos, tipos de preguntas, tipos de informes, plugins de autenticación y de inscripción. Los tipos de informes que se generan son capaces de hacerse de manera automática. Algunos de los informes son: evolución del alumno, uso de un curso o de estado del sitio, además de esto brinda la posibilidad de crear nuevos tipos de informes a partir de toda la información que almacena su base de datos. Dispone de las herramientas necesarias para la evaluación y autoevaluación de los estudiantes a través de la realización de tareas, cuestionarios, talleres y foros (19) (17). La instalación de la plataforma es bastante compleja, al igual que trabajar con alguna de sus funciones. Trabaja con las bases de datos: MySQL, PostgreSQL, Oracle, SQLite y MSSQL.

Sakai

El proyecto de Sakai es un entorno modular de código fuente abierto (software libre) bajo la licencia de comunidad educativa, cuyo objetivo es integrar diversas funcionalidades del e-learning en un portal académico, crear un entorno de colaboración y aprendizaje para la

³ Claroline es una plataforma de aprendizaje que permite a los formadores construir cursos online y gestionar las actividades de aprendizaje y colaboración en la *web* (8).

educación superior, que pueda competir con sus equivalentes comerciales Blackboard / WebCT y que mejore otras iniciativas de código abierto como Moodle. La aplicación ha sido desarrollada en Java, lo cual permite su uso en múltiples sistemas operativos. Sakai divide sus funcionalidades en cuatro categorías de herramientas: generales de colaboración, enseñanza-aprendizaje, administrativas y, por último, de portafolios, este posee diversas funcionalidades entre la que se encuentran: generar, mostrar y exportar informes de la actividad del sitio y los usuarios (20) (17). La documentación de la plataforma Sakai es poca y está bastante desorganizada y dispersa. Pocas empresas le brindan soporte. Trabaja solo con las bases de datos: MySQL y Oracle. Es complicada la integración de Sakai con otras plataformas (21).

Blackboard

Blackboard Academic Suite fue desarrollada originalmente en la Universidad de Cornell en el año de 1997. Actualmente Blackboard tiene una alianza estratégica con Microsoft. Blackboard es una plataforma con licencia comercial. El contrato es anual renovable y flexible. Existen varios niveles de licenciamiento dependiendo del paquete que se quiera adquirir. Fue diseñada para instituciones dedicadas a la enseñanza y el aprendizaje. Proporciona la funcionalidad necesaria para administrar correctamente programas de educación a distancia o a través de la web. Blackboard es multilenguaje, es decir que permite a las instituciones utilizar varios idiomas en el mismo sistema. Su arquitectura de bases de datos relacionales soporta el rápido crecimiento de las necesidades de sus clientes (22).

Análisis de sistemas similares

A continuación, se establece una comparación entre las plataformas educativas con módulos de reportes similares a la Plataforma Educativa XAUCE ZERA. Para realizar esta valoración se tendrán en cuenta los siguientes aspectos: código abierto, tipo de herramientas que utilizan, principales funcionalidades y, por último, tipos de informes que realizan.

Tabla 1. Análisis de sistemas similares

Plataformas	Libre	Herramientas	Funcionalidades	Tipos de informes
Dokeos	Sí	Libres y privativas	<ul style="list-style-type: none"> • Crear plantillas de contenido, con explicación de diagramas, vídeo y flash. 	<ul style="list-style-type: none"> • Permite visualizar informes acerca de los resultados de los ejercicios que hacen los

			<ul style="list-style-type: none"> • Crear test: de respuesta múltiple y preguntas abiertas. • Gestionar tutoriales. • Interacción: grupos, chat, foros. • Crear y organizar encuestas. • Visualizar informes acerca del progreso de los alumnos. • Extender la comunidad a través de libros de notas, wikis. 	<p>alumnos y el progreso que van acumulando, así como, exportarlos a Microsoft Excel.</p>
Moodle	Sí	Libres	<p>Sus funcionalidades se encuentran divididas en módulos:</p> <ul style="list-style-type: none"> • Módulos de actividades: la puesta en práctica de los conocimientos adquiridos por el estudiante durante el curso, así como, las evaluaciones y calificaciones de los alumnos. • Módulo de comunicación: foros y chats. • Módulo de contenido de materiales: editor de texto HTML, etiquetas, recursos, lecciones, glosarios. 	<ul style="list-style-type: none"> • Los profesores tienen informes que indican el número de veces, hora, fecha, frecuencia y dirección IP de los accesos de los alumnos a los contenidos y módulos de la asignatura (23). • Se pueden generar informes de las encuestas los cuales incluyen gráficos. Los datos pueden descargarse con formato de hoja de cálculo Excel o como archivo de texto. • Permite realizar informes de curso, participación, estado de la wiki, bitácoras, glosario de términos, evaluaciones, comentarios, respaldos, foro, actividad, cambios

				de configuración, lista de eventos, estadísticas de examen, instancias de preguntas y desglose de competencias.
Sakai	Sí	Libres	<p>Sus funcionalidades se dividen en cuatro categorías:</p> <ul style="list-style-type: none"> • Herramientas generales de colaboración. • Herramientas de enseñanza y aprendizaje. • Herramientas administrativas. • Herramientas de portafolios 	<ul style="list-style-type: none"> • Permite generar reportes de calificaciones, estadísticas y acerca del uso de los recursos por parte de los participantes (23).
Blackboard	No	Privativas	<p>Sus funcionalidades se dividen en cuatro categorías:</p> <ul style="list-style-type: none"> • Herramientas pedagógicas. • Herramientas de evaluación. • Herramientas de autor. • Herramientas administrativas. 	<ul style="list-style-type: none"> • Permite realizar un informe con las estadísticas de la actividad por alumno y cursos. • Informes de resultados de exámenes. • Genera un reporte de actividad docente.

La plataforma Dokeos no constituye una vía de solución para el desarrollo del módulo de reportes por contener herramientas privativas, aunque es una aplicación libre. Por otro lado, Moodle y Sakai trabajan únicamente con los datos que se generan en ellas. Además, en el caso de Sakai solo genera los que tiene preestablecidos y, al matricular alumnos en un curso, se debe conocer el nombre del usuario porque no ofrece una lista de los usuarios existentes. Blackboard es una plataforma comercial con licencia y herramientas privativas por lo que no se puede usar para la solución. A pesar de que estas no se pueden integrar y no cumplen con todas las necesidades que presenta la Plataforma Educativa XAUCE ZERA, cuentan con características muy útiles que puede servir como guía a la solución que se quiere desarrollar, como por ejemplo la manera en que se usan los foros, el glosario de

términos y las actividades de un curso, la realización de las copias de seguridad y en el diseño utilizan para la visualización de los reportes.

1.3 Herramientas, lenguajes y metodología

Para el desarrollo de la solución se utilizan herramientas, lenguajes y la metodología que permitirá guiar el proceso de desarrollo.

1.3.1 Servidor Dinámico de Reportes

El Servidor Dinámico de Reportes (SDR) es un software que permite de manera eficiente la gestión, compilación, publicación y exportación de reportes. El principio del SDR es brindar una capa de servicios que facilita su integración con cualquier otro sistema, web o de escritorio, con independencia de sus plataformas de desarrollo. A través de este los clientes pueden generar reportes en una gran cantidad de formatos sobre múltiples fuentes de datos. Estos reportes se mantienen almacenados en el servidor, logrando con ello mejores tiempos de respuesta y acceso al historial de los reportes (24). Algunas de las características y funcionalidades son:

- Permite la generación de reportes en múltiples formatos: PDF, XML, HTML, CSV, XLS, XLSX, RTF, JPG, ODT, PPTX, DOCX, ODS y XML4SWF, a partir de diferentes fuentes de datos.
- Permite la generación de subreportes.
- Visualiza las vistas previas de los reportes.
- Posibilita la selección de estilos de estructura para un grupo de reportes.
- No limita el tamaño del diseño del reporte para la exportación de reportes.
- Constituye un sistema multiplataforma.
- Utiliza un gestor de base de datos, que puede ser PostgreSQL, SQLite, MySQL o SQL, para persistir la información del servidor y los datos de los reportes.
- Permite la programación de tareas para la exportación automática de reportes en un momento específico, y su envío por correo electrónico o por FTP.

Se emplea este servidor ya que el proyecto ha trabajado anteriormente con este y lo tiene definido como una de sus herramientas.

1.3.2 Herramientas para la generación de reportes

En la actualidad, existen herramientas que viabilizan el proceso de generación de reportes. Estas herramientas son capaces de construir reportes dinámicos de una forma rápida y sencilla (25).

JasperReports

JasperReports es un motor de creación de informes en código abierto. Está escrito completamente en Java y es capaz de utilizar los datos procedentes de cualquier tipo de fuente de dato. Permite exportar en una variedad de formatos de documentos incluyendo HTML, PDF, Excel, OpenOffice y Word. JasperReports se usa comúnmente con iReport (26).

Características y funciones:

- Servidor de informes y análisis de gestión centralizada.
- Simplificación y mejora del rendimiento en la generación de informes y la entrega a los usuarios.
- Entrega automática de los informes en varios formatos de salida.
- Arquitectura de datos flexible que apoya los informes basados en fuentes de datos relacionales y no relacionales.
- Sub-informes reutilizables y almacenados en el repositorio.

Crystal Reports

Crystal Reports facilita la creación e integración de reportes con datos provenientes de múltiples fuentes de datos. Utiliza una tecnología de elaboración de reportes permitiendo el diseño y generación de informes a partir de datos almacenados en una base de datos. Se distingue por la extensión .RPT (report) en el que se almacena la definición de los informes. Presenta licencia privativa (27).

Características y funciones:

- Administrar y diseñar los reportes.
- Exportar los reportes.
- Controlar la fuente de datos.
- Crear una nueva conexión.

Dynamic JasperReports

Es una herramienta desarrollada sobre plataforma JAVA que mantiene las principales características del JasperReports ya que trabaja con él reduciendo su complejidad para reportes simples y de mediana complejidad ayudando a los desarrolladores a ahorrar tiempo cuando los diseñan. Permite crear diseños de informes dinámicos y no necesita de un diseñador de informes visual. Puede crear rápidamente informes y preparar los documentos que se pueden visualizar, imprimir o exportar en muchos formatos, tales como: PDF, Excel, Word, HTML y XLS y otros. Un diseño es creado por un código simple de Java

puro. Permite definir en tiempo de ejecución las columnas, controlando su posicionamiento, ancho, título. Soporta gran variedad de orígenes de datos, gráficos, imágenes y sub-reportes, que pueden ser creados dinámicamente (28) (29).

Grato Report

Una herramienta desarrollada en el año 2008 por el proyecto Plataforma Inteligente para la Predicción de Actividad Biológica de Compuestos Orgánicos (alasGRATO) de la Facultad 6, en la Universidad de las Ciencias Informáticas, para la generación de reportes referentes a la Química. Fue desarrollado sobre el lenguaje JAVA usando la biblioteca para la generación de reportes Dynamic JasperReports como base para su funcionamiento (30).

GEReport v1.0

GEReport fue creada en La Universidad de Cienfuegos, es una herramienta capaz de construir reportes dinámicos que cumplan con los estándares mundiales de estos tipos de aplicaciones, y cuya utilización no se restringiese debido a las licencias de software, se ejecutase sobre un entorno web, se minimizaran las dependencias tecnológicas y fuese multiplataforma. La arquitectura que utiliza es cliente-servidor (25).

Selección del generador de reportes

Son varios los generadores de reportes que existen, para seleccionar el que se utilizará hay que tener en cuenta las necesidades de los usuarios. Para realizar esta valoración se tendrán en cuenta los siguientes aspectos: tipo de software, tipo de plataforma en la que se trabaja, formatos en los que permite exportar y las bases de datos con las que puede trabajar. La siguiente tabla muestra una comparación entre las diferentes herramientas para generar reportes.

Tabla 2. Comparación de las herramientas de reporte

Generadores	Libre	Plataforma	Formatos	Fuente de datos
JasperReports	Sí	Escritorio, Web	HTML, PDF, Excel, CSV, ODT, RTF, XLS	postgresql, mysql, oracle, mssql, odbc, interbase, informix, xml, csv
CrystalReports	No	Escritorio, Web	RTF, PDF, Excel, HTML, TIFF	postgresql, mysql, oracle, mssql, odbc, interbase
Dynamic Reports	Sí	Escritorio	PDF, Excel, Word, HTML, OpenOffice	postgresql, mysql
Grato Report	Sí	Escritorio	PDF, HTML, XLS	postgresql, mysql,
GEReport	Sí	Web	HTML, PDF	PostgreSQL

El generador más idóneo es el JasperReport, debido a que es una herramienta con altas prestaciones en la generación de reportes, enfocada al entorno de escritorio y web, bajo licencia GPL lo cual facilita su adquisición y contribuye a la soberanía tecnológica aspirada por la Universidad y el país. Además, permite exportar los archivos a múltiples formatos y varias conexiones a bases de datos. JasperReports se usa comúnmente con iReport.

1.3.3 Diseñador de reportes

El diseñador de reportes permite crear proyectos, conexiones a bases de datos, definir usuarios y grupos de usuarios, así como asegurar el sistema de reportes. Permite la creación de reportes, hojas de datos, gráficas y tablas dinámicas. Puede distribuir sus reportes a través de archivos físicos de reportes los cuales pueden ser abiertos por los usuarios finales con el Visor de Reportes Gratuito o a través del Servicio Web de Reportes donde los usuarios pueden abrir cualquier reporte con un navegador de Internet (31).

iReport v5.6

iReport es un diseñador de reportes de código abierto con licencia GPL (Licencia Pública General) para JasperReports y JasperReports Server. Es un programa escrito en las secuencias de comandos de código de Java, que se distribuye sin costo y por tiempo ilimitado. Permite crear diseños muy sofisticados que contienen gráficos, imágenes, subreportes y tablas de contingencia. Además, está implementado sobre la plataforma NetBeans. Con esta herramienta visual de diseño es posible crear nuevos reportes fácilmente usando asistentes de configuración y plantillas, con la opción de poder tener una pre-visualización de la exportación de los reportes en PDF, HTML, XLS, CSV, TXT y XML (26).

Características importantes de iReport:

- 100% escrito en Java y además de ser código abierto y gratuito.
- Maneja el 98% de las etiquetas de JasperReport.
- Permite diseñar con sus propias herramientas: rectángulos, líneas, elipses, campos de texto, cartas y subreportes.

Generador Dinámico de Reporte

El Generador Dinámico de Reportes (GDR) fue creado en la Universidad de las Ciencias Informáticas. Es una aplicación web para la gestión de la información de cualquier empresa o institución, mediante el cual muestra el cumplimiento de las metas establecidas y con ello contribuir a la toma de decisiones. Fue desarrollado con el lenguaje de programación PHP, en conjunto con el marco de trabajo Symfony. Posibilita diseñar reportes tabulares (con

gráficos incluidos), tabla pivote y cruzada desde los gestores de Bases de Datos: SQL Server, SQLite, Oracle, MySQL y PostgreSQL. Dicha herramienta permite a los usuarios abstraerse de los conocimientos relacionados con los gestores de bases de datos y generar reportes en varios formatos con gran variedad de opciones en su diseño, marcando una diferencia entre los reportes tradicionales y los reportes dinámicos. Mediante esta herramienta los reportes pueden ser exportados a formatos de salida tales como: html, pdf, xls y csv (15).

Selección del diseñador de reportes

Se estableció una comparación entre los diseñadores generadores de reportes presentados. Para realizar esta valoración se tendrán en cuenta los siguientes aspectos: tipo de plataforma, compatibilidad con la herramienta de generación de reportes seleccionada, tipo de software, lenguaje en que se desarrolló; para poder identificar la compatibilidad con el Servidor Dinámico de Reportes y, por último, los formatos en los que se pueden exportar los reportes. La siguiente tabla muestra una comparación entre las diferentes herramientas para diseñar reportes.

Tabla 3. Comparación de los diseñadores de reporte

Diseñadores	Multiplataforma	Servicio	Compatibilidad con JasperReport	Licencia	Lenguaje en que está desarrollado	Fuente de salida
iReport	Sí	Escritorio	Sí	Libre	Java	pdf, xml, html, xhtml csv, xls, rtf, txt, docx, odt, pptx, docx, ods, xlsx
GDR	Sí	Web	Sí	Libre	PHP	html, pdf, xls, csv

El diseñador más idóneo es el iReport, debido a que es una herramienta con altas prestaciones en la generación de reportes y está enfocada al entorno de escritorio. Además, está implementado en Java al igual que el Servidor Dinámico de Reportes, que es el que se emplea en el proyecto.

1.3.4 Metodología de desarrollo de software

La metodología de desarrollo a emplearse en los proyectos productivos de la Universidad de las Ciencias Informáticas (UCI) se basa en una variación de la metodología “Proceso

Unificado Ágil” (AUP por sus siglas en inglés) en unión con el modelo CMMI-DEV⁴ v 1.3 (Modelo de Madurez de Capacidades Integrado), conocido como: AUP-UCI. Esta metodología permite entregas en períodos cortos de tiempo, además, de una constante interacción entre el cliente y el grupo de desarrollo (32).

Fases AUP-UCI:

1. Inicio: se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no.
2. Ejecución: se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto.
3. Cierre: en esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

Ciclo de vida:

Se decide para el ciclo de vida de los proyectos de la UCI tener 7 disciplinas. Estas son: modelado de negocio, requisitos, análisis y diseño, implementación, pruebas internas, pruebas de liberación y pruebas de aceptación (32).

En AUP-UCI a partir de que el modelado de negocio propone tres variantes a utilizar en los proyectos (Casos de Uso del Negocio, Descripción de Procesos de Negocio, Modelo Conceptual) y existen tres formas de encapsular los requisitos (Casos de Uso del Sistema, Historias de Usuario, Descripción de Requisitos por Proceso), surgen cuatro escenarios para modelar el sistema en los proyectos (32).

Para el desarrollo de este trabajo se seleccionó el escenario No 4, ya que el cliente siempre está acompañando al equipo de desarrollo. Este escenario se recomienda para proyectos no muy extensos y es el mismo con el que se trabaja en el proyecto. Propone los siguientes

⁴ CMMI es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software. Es un enfoque de mejora de procesos que provee a las organizaciones de los elementos esenciales para un proceso efectivo.

artefactos: especificación de requisitos de software, historias de usuario, diagrama de clases con descripción de las clases, diagrama de paquetes y diseño de casos de pruebas.

1.3.5 Herramientas de programación

Entorno de desarrollo integrado

Un entorno de desarrollo integrado (IDE, del inglés Integrated Development Environment) es una herramienta que permite el desarrollo de aplicaciones con características modulares. Entre sus funcionalidades permite escribir, depurar, compilar y ejecutar programas, además soporta otros lenguajes de programación. La mayoría de los IDEs tienen auto-completado inteligente de código (33).

Un IDE debe tener las siguientes características:

- Multiplataforma.
- Soporte para diversos lenguajes de programación.
- Integración con Sistemas de Control de Versiones.
- Reconocimiento de Sintaxis.
- Extensiones y Componentes para el IDE.
- Integración con Framework populares.
- Depurador.
- Importar y Exportar proyectos.

Netbeans 8.2

NetBeans es un entorno de desarrollo integrado (IDE) que permite editar programas en java, compilarlos, ejecutarlos, depurarlos y construir rápidamente el interfaz gráfico de una aplicación eligiendo los componentes de una paleta, entre otros. También, es una plataforma de ejecución de aplicaciones, es decir, facilita la escritura de aplicaciones Java, proporcionando una serie de servicios comunes, que a su vez están disponibles a través del IDE. Destaca el código fuente de manera sintáctica y semántica, permite refactorizar fácilmente el código, con una gama de herramientas útiles y poderosas. Además, está escrito en Java lo que brinda la posibilidad de ser instalado en todos los sistemas operativos compatibles con Java, es decir, Windows, Linux y Mac OSX. Proporciona editores, asistentes y plantillas que permiten crear aplicaciones en Java, PHP y otros lenguajes de programación (34). Es de fuente abierta, es decir, se proporciona el código fuente del entorno para que se pueda modificar de acuerdo a ciertos parámetros de licencia (35).

NetBeans 8.2 es el IDE que va a ser utilizado debido a que el diseñador de reportes seleccionado iReport trabaja con este software integrado a él y además compila todos los lenguajes de programación que serán utilizados para el desarrollo de la solución.

Gestor de base de datos PostgreSQL v9.5

PostgreSQL es un Sistema Gestor de Base de Datos (SGBD) objeto relacional. Presenta propiedades como atomicidad, consistencia, aislamiento y durabilidad. La atomicidad asegura la realización de una operación, por lo tanto, ante un fallo del sistema no puede quedar a medias. La consistencia posibilita la ejecución de aquellas operaciones que no van a romper las reglas y directrices de integridad de la base de datos. El aislamiento asegura que una operación no pueda afectar a otras, de esta manera dos transacciones sobre la misma información no generan error. Durabilidad es la propiedad que asegura que, una vez realizada la operación, esta persistirá y no se podrá deshacer, aunque falle el sistema (36). Entre sus principales características se encuentra:

- Permite la declaración de funciones propias, así como la definición de disparadores.
- Incluye herencia entre tablas.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de estos.

PgAdmin III v1.22.0

PgAdmin III es una herramienta gráfica y multiplataforma para administrar y desarrollar bases de datos en PostgreSQL. Está diseñada para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. Ofrece una interfaz gráfica que soporta todas las características de PostgreSQL, permitiendo realizar cambios en la base de datos desde el sistema de manera sencilla e incluye un editor con resaltado de sintaxis SQL (37).

1.3.6 Lenguaje de programación

Un lenguaje de programación es un conjunto de reglas, notaciones, símbolos y/o caracteres que permiten a un programador poder expresar el procesamiento de datos y sus estructuras en la computadora. Cada lenguaje posee sus propias sintaxis. Existen varios lenguajes de programación entre los que se encuentran JavaScript, PHP y Java (38).

PHP v7.0

El acrónimo recursivo que significa PHP (Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser

incrustado en HTML⁵. Lo que distingue a PHP de algo del lado del cliente como JavaScript es que el código es ejecutado en el servidor, generando HTML, enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP.

Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno (39).

Nuevos elementos en PHP 7:

En PHP 7 se han añadido nuevos tipos de variables para definir funciones y métodos: int, float, bool y string. Es lo que en esta nueva versión de esta sintaxis recibe el nombre de declaraciones de tipo escalar. El otro tipo de declaraciones que son una novedad en PHP 7 son las declaraciones de tipo retorno. Estas declaraciones permitirán al programador tipar el retorno de una función con los valores que debe devolver, se interpondrá ese valor tras los argumentos y antes de la apertura de llaves, precedido siempre de dos puntos. Además, tiene la capacidad para implementar clases anónimas. Se pueden usar en vez de las definiciones de clases completas para objetos desechables (40).

JavaScript v1.3

JavaScript es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, débilmente tipado y dinámico. Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web que permite mejoras en la interfaz de usuario y páginas web dinámicas (41).

HTML5

HTML (Hyper Text Markup Language) es un lenguaje que sirve para escribir hipertexto, es decir, documentos de texto, presentado de forma estructurada, con enlaces (links) que conducen a otros documentos o a otras fuentes de información, que pueden estar en la máquina del usuario o en máquinas remotas de la Red.

HTML5 permite crear nuevas etiquetas, atributos y eliminar aquellas marcas que están en desuso o se utilizan inadecuadamente (42).

⁵ HTML: HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto.

CSS3

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe la manera que se muestra un documento en la pantalla. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento (43).

1.3.7 Herramienta y lenguaje de Modelado

Las Herramientas CASE (Computer Aided Software Engineering, del español Ingeniería de Software Asistida por Computadoras) son aplicaciones de apoyo al desarrollo, mantenimiento y documentación automatizados de software. Permiten aplicar la metodología de análisis y diseño orientados a objetos. Cuanto más grande es un proyecto, más importante es utilizar una herramienta CASE, ya que contribuyen a aumentar la productividad en el desarrollo de software reduciendo el coste en términos de tiempo y dinero (44).

Visual Paradigm for UML 8.0

Visual Paradigm for UML es una herramienta CASE de diseño UML implementada para ayudar al desarrollo del ciclo de vida completo de un software. Ofrece un conjunto de herramientas necesarias para la realización de las diferentes tareas del proceso, tales como: captura de requisitos, planificación y modelado de clases. Genera documentación en formato PDF, Word o HTML (45).

En la Universidad de las Ciencias Informáticas se ha estandarizado el uso del Visual Paradigm for UML para el modelado de los procesos de desarrollo de software que en ella se llevan a cabo.

Lenguaje Unificado de Modelado 2.0

Lenguaje Unificado de Modelado (UML) prescribe un conjunto de notaciones y diagramas estándares para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. UML se puede usar para modelar distintos tipos de sistemas: de software, hardware, y organizaciones del mundo real, ayudando al usuario a entender la realidad de la tecnología y la posibilidad de que reflexione antes de

invertir en proyectos que no estén seguros en su desarrollo, reduciendo el coste y el tiempo empleado en la construcción de las piezas que constituirán el modelo. La especificación de UML no define un proceso estándar, pero está pensado para ser útil en un proceso de desarrollo iterativo (46).

Los principales beneficios de UML son:

- Alta reutilización y minimización de costos.
- Crear un lenguaje de modelado utilizado tanto por humanos como por máquinas.
- Modelar sistemas utilizando conceptos orientados a objetos.

1.3.8 Framework de desarrollo

Un framework es un producto que sirve como base para la programación avanzada de aplicaciones, que aporta una serie de funciones o códigos para realizar tareas habituales. Son librerías de código que contienen procesos o rutinas ya listos para usar. Son utilizados para no tener que desarrollar tareas más básicas, pues en el framework hay implementaciones que están probadas, funcionan y no se necesitan volver a programar (47).

Bootstrap v3.3.2

Es el framework para CSS más popular que existe para el desarrollo web, posee una amplia documentación, haciéndose extensible su uso. Provee de forma fácil las mejores prácticas en el diseño de plantillas, se generan vistas muy agradables y ligeras para el cliente. Es de código abierto, con una comunidad que lo mantiene a través de gitHub (48).

Symfony v2.7.16

Symfony es un framework diseñado para optimizar el desarrollo de las aplicaciones web de acuerdo a algunas de sus características. Primeramente, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Presenta varias herramientas y clases que permiten reducir el tiempo de desarrollo de las aplicaciones web complejas. Symfony es compatible con la mayoría de los gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas Unix y Linux, como en Windows. Alguna de las características que presenta son:

- Sigue las mejores prácticas de patrones de diseño para la web.
- Independiente del sistema gestor de bases de datos.
- Utiliza programación orientada a objetos.
- Fácil de instalar y configurar en plataformas como Windows y Linux.
- Posee una potente línea de comandos que facilitan la generación de código, lo cual permite ahorrar tiempo de trabajo.

- Utiliza la arquitectura Modelo – Vista – Controlador (MVC) como la capa de abstracción de base de datos, el controlador frontal y las acciones. (Potencier)

Symfony es el framework sobre el que se desarrolla el marco de trabajo Xalix, debido a que es un conjunto de componentes independientes, que cohesionados resuelven con mayor facilidad los problemas comunes del desarrollo web.

1.3.9 Servidor web NGINX 1.14

Es un servidor HTTP de código abierto, distribuido bajo la licencia LSB y de proxy inverso de alto rendimiento, además funciona como servidor proxy para IMAP/POP3/SMTP. NGINX es muy conocido por su estabilidad, sus características, configuración simple y su bajo consumo de recursos, lo último como consecuencia de manejar requerimientos basados en eventos, a diferencia de Apache que lo hace basado en procesos y que permite asegurar un funcionamiento óptimo bajo mucha carga (49). Algunas de las principales características de NGINX son:

- Capacidad de manejar más de 10 000 conexiones simultáneas con bajo uso de memoria.
- Autenticación de acceso.
- Reescritura de URL.

1.4 Marco de trabajo Xalix

El marco de trabajo Xalix surge en el centro FORTES perteneciente a la Facultad 4 de la Universidad de las Ciencias Informáticas, con el objetivo de obtener una homogenización en la arquitectura de los distintos sistemas que se desarrollan en dicho centro. Xalix es una arquitectura de desarrollo basada en componentes que extiende del concepto de bundle de Symfony, tomándolos como plugins y creando un sistema de gestión de los mismos que incluye la instalación y desinstalación. Esta característica otorga independencia a los componentes desarrollados, lo que permite que el mantenimiento o la transformación de uno de ellos no influya en la integridad de los demás componentes (50).

Otras características que diferencian a Xalix de Symfony son:

- El appKernel, ya no almacena la referencia a los bundles de terceros debido a que esta tarea se realiza mediante el fichero bundles.ini ubicado en el directorio config de app.
- El fichero config.yml es eliminado del directorio app/config y la configuración es controlada por cada plugin.

Para el desarrollo de sus productos Xalix incluye la siguiente base de tecnologías:

- Lenguaje de programación para el servidor: PHP 7.0.x.
- Lenguaje para el cliente: HTML5.
- Gestor de base de datos: PostgreSQL 9.5.x.
- Librería de CSS: Bootstrap.
- Librería de JavaScript: jQuery v1.10.2 con jQuery UI 1.10.3.

El marco de trabajo Xalix es un completo framework de trabajo que permite hacer uso de los distintos componentes que contiene y adopta una arquitectura basada componentes donde cada elemento puede desacoplarse y evolucionar de manera independiente, lo que conforma un producto genérico una vez ensamblado y posee un mecanismo de integración para los componentes creados (50).

Conclusiones del capítulo

Después de realizado este capítulo se puede concluir que:

- Se analizaron cuatro de las plataformas educativas más utilizadas. Para este análisis se tuvo en cuenta cómo gestionan su información, identificando que lo visualizan a través de un espacio donde se generan diferentes tipos de reportes.
- Se seleccionó la metodología AUP-UCI en el escenario 4 como guía del proceso de desarrollo de software, lo que permitirá generar la documentación y artefactos sobre la misma línea del proyecto, para así minimizar errores y cambios en un futuro.
- Se seleccionaron las herramientas, lenguajes y tecnologías definidas por el proyecto como: JavaScript 1.3, PHP 7.0, HTML 5 y CSS 3 como lenguajes de programación, Visual Paradigm 8.0 para el modelado UML, PostgreSQL como sistema gestor de bases de datos, Nginx 1.14 como servidor web, SDR para la gestión de los reportes y los frameworks Bootstrap 3.3.2 y Symfony 2.7.16. Además de otras herramientas propias para la solución como: JasperReports, motor generador de reportes, el iReport 5.6 como diseñador de reportes y el IDE NetBeans 8.2.

Capítulo II. Análisis y diseño de la solución

En el presente capítulo se describe la solución propuesta utilizando los artefactos que propone la metodología AUP-UCI. Se definen los requisitos funcionales y no funcionales del sistema a desarrollar, así como los diagramas de clases del análisis y del diseño.

2.1 Propuesta de solución

Para dar solución al problema tratado en esta investigación, se propone el desarrollo de un módulo que permita la gestión de reportes para contribuir a la gestión de información generada en la Plataforma Educativa XAUCE ZERA. Su objetivo es facilitar el trabajo de los administradores y profesores, brindando la posibilidad de generar reportes de forma dinámica para los diferentes roles dentro de la plataforma y visualizar la información generada por la interacción de los usuarios con los cursos. Estas funcionalidades apoyan notablemente a los profesores y administradores en la toma de decisiones. Por un lado, los primeros conocerán el tema que les resulta más interesante a sus pupilos para así poderlos motivar, además podrán identificar los ejercicios de mayor dificultad para sus estudiantes de manera que pueda fortalecer sus debilidades. Por otro lado, los administradores podrán ver los usuarios inactivos para poderlos eliminar con una notificación previa, de esta manera optimiza el funcionamiento del sistema, también serán capaces de apreciar qué cursos cuentan con más recursos para notificar a los profesores y gestionar la eliminación de recursos innecesarios y así evitar la sobrecarga del sistema.

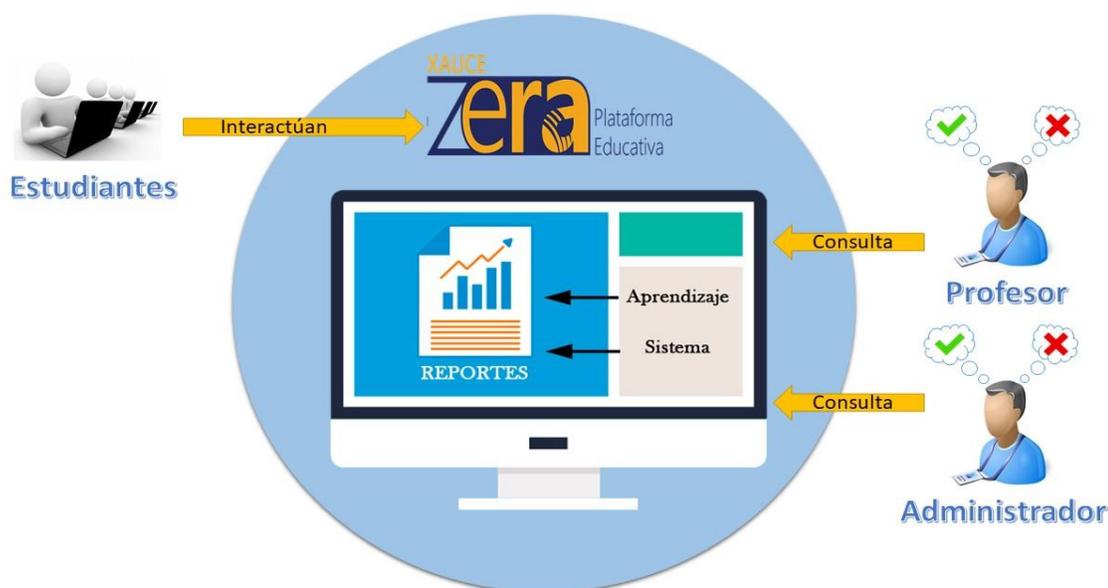


Figura 1. Elaboración propia. Diagrama de la propuesta de solución.

Los reportes que se desean generar pueden tener un formato determinado para la visualización de los datos de manera más organizada, precisa y fácil de interpretar. Las extensiones en las que se podrán exportar los reportes son: PDF, DOC y XLS. Los elementos que pueden presentar son: gráficos y tablas de contenidos.

2.2 Requisitos funcionales

Los requisitos funcionales (RF) son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar ante situaciones determinadas (51). El módulo de reportes a desarrollar debe satisfacer los siguientes RF:

Tabla 4. Requisitos Funcionales

No.	Nombre	Descripción
RF1	Generar reporte de las actividades de un curso	<p>El sistema deberá permitir generar reportes de las actividades por curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Actividad • Tipo
RF2	Generar reporte del comportamiento de los estudiantes por actividad en un curso	<p>El sistema deberá permitir generar reportes del comportamiento de los estudiantes por actividad en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre • Usuario • Fecha • Nota • Escala • Tipo

RF3	Generar reporte de los cuestionarios por curso	<p>El sistema deberá permitir generar reportes de los cuestionarios y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los cuestionarios:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre del cuestionario • Cantidad de intentos • Estudiantes que lo realizaron • Fecha por intento • Evaluación por intento • Evaluación general
RF4	Generar reporte del estado de wiki de un curso	<p>El sistema debe permitir generar reportes del estado de wiki de un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Total de páginas • Participante más activo/cantidad de páginas • Página • Estado • Autor
RF5	Generar reporte de estudiantes activos por curso	<p>El sistema deberá permitir generar reportes de los estudiantes activos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Cantidad de estudiantes activos • Nombre/ Apellidos • Usuario • Correo electrónico

		<ul style="list-style-type: none"> • Grupo
RF6	Generar reporte del glosario de términos en un curso	<p>El sistema debe permitir generar reportes del glosario de términos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Total de términos • Curso • Participante más activo/cantidad de términos • Término del glosario • Estado • Autor
RF7	Generar reporte de los roles activos en un curso	<p>El sistema deberá permitir generar reportes de los roles activos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los roles activos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Rol • Tipo de rol • Cantidad de usuarios asociados
RF8	Generar reporte de los roles inactivos en un curso	<p>El sistema deberá permitir generar reportes de los roles inactivos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los roles inactivos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Rol • Nombre/ Apellidos • Cantidad de usuarios
RF9	Generar reporte del uso de foros en un curso	<p>El sistema debe permitir generar reportes del uso de foros en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p>

		<ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre del foro • Cantidad de temas • Temas • Es comentable • Fecha de expiración • Fecha de cierre • Cantidad de cometarios • Participante más activo • Cantidad de respuestas
RF10	Generar reporte de usuarios activos en un curso	<p>El sistema deberá permitir generar reportes de los usuarios activos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre/Apellidos • Rol • Correo electrónico • País • Fecha de nacimiento • Ciudad • Género
RF11	Generar reporte de los usuarios inactivos en un curso	<p>El sistema deberá permitir generar reportes de los usuarios inactivos en un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los usuarios inactivos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre/ Apellidos • Rol

		<ul style="list-style-type: none"> • Usuario • Correo electrónico • Grupo
RF12	Generar reporte de los usuarios por curso	<p>El sistema deberá permitir generar reportes de los usuarios existentes por curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los usuarios:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Nombre/ Apellidos • Rol • Fecha de inscripción • Grupo
RF13	Generar reporte de los usuarios registrados en un tiempo dado	<p>El sistema deberá permitir generar reportes de los usuarios registrados en un tiempo dado y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Fecha inicio • Fecha fin • Nombre/ Apellidos • Correo electrónico • País • Fecha de nacimiento • Ciudad • Género
RF14	Generar reporte de las acciones realizadas en la plataforma	<p>El sistema deberá permitir generar reportes de las acciones realizadas en la plataforma y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los datos según la acción:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Usuario • Fecha

		<ul style="list-style-type: none"> • Ip del cliente • Usuario agente • Operación/ Tabla
RF15	Generar reporte de las acciones de los participantes en la plataforma	<p>El sistema debe permitir generar reportes de las acciones realizadas por los usuarios participantes en la plataforma y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Usuario • Referencia • Controladora • Acción • Fecha
RF16	Generar reporte de las copias de seguridad de un curso	<p>El sistema debe permitir generar reportes de las copias de seguridad de un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Curso • Autor del respaldo • Tipo de respaldo • Fecha de creación
RF17	Generar reporte de los datos del usuario	<p>El sistema deberá permitir generar reportes de los datos del usuario y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos del perfil de usuario:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Nombre/ Apellidos • Correo electrónico • País • Fecha de nacimiento • Ciudad

		<ul style="list-style-type: none"> • Género
RF18	Generar reporte de los recursos de la plataforma	<p>El sistema deberá permitir generar reportes de los recursos de la plataforma y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Tipo • Cantidad • Tamaño • Porcentaje de almacenamiento
RF19	Generar reporte de los recursos de un curso	<p>El sistema deberá permitir generar reportes de los recursos de un curso y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Tipo • Cantidad • Tamaño • Porcentaje de almacenamiento
RF20	Generar reporte de los roles registrados	<p>El sistema deberá permitir generar reportes de los roles registrados y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos:</p> <ul style="list-style-type: none"> • Generado por • Creado (Fecha de creado) • Rol • Tipo • Fecha de creado • Cantidad de usuarios
RF21	Generar reporte de los usuarios registrados en la plataforma	<p>El sistema deberá permitir generar reportes de los usuarios registrados en la plataforma y exportarlo en los formatos (xls, doc y pdf). El reporte mostrará los siguientes datos de los usuarios registrados en la plataforma:</p> <ul style="list-style-type: none"> • Generado por

		<ul style="list-style-type: none"> • Creado (Fecha de creado) • Nombre/ Apellidos • Usuario • Correo electrónico • Fecha
--	--	---

2.3 Requisitos no funcionales

Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y normas o estándares. Además, no se aplican regularmente a rasgos o servicios individuales del sistema (51).

- **Requerimiento de seguridad**

RNF 1. Garantizar el acceso a las funcionalidades definidas para los usuarios con rol administrador y profesor.

- **Requerimiento de portabilidad:**

En la PC Cliente para poder visualizar la aplicación se debe disponer de:

RNF 2. Navegador web: Mozilla Firefox 10.x o superior, Google Chrome 20.x o superior, Opera 10.x o superior. En el caso de los dispositivos móviles deben tener android v4.2.x o superior y navegadores de móviles tales como: Safari en iOS, Firefox, Opera Mini, Opera Mobile Classic y Chrome.

- **Requerimientos de implementación:**

Para poder usar el sistema en las estaciones de trabajo se debe contar con:

RNF 3. Gestor de base de datos: PostgreSQL v9.5. x.

RNF 4. Servidor web: NGINX v1.10.3 o Superior.

RNF 5. Sistema operativo: CentOS 7.

RNF 6. PHP v7.0, CSS v3.0, HTML v5.0 y JavaScript v1.3.

- **Requerimientos de hardware:**

RNF 7. Servidor de aplicaciones NGINX debe tener como mínimo: 16 GB de RAM, 500 GB de disco duro y un microprocesador (6 x 800 GHz).

RNF 8. Servidor de base de datos PostgreSQL debe tener como mínimo: 16 GB de RAM, 100 GB de disco duro y un microprocesador (6 x 800 GHz).

RNF 9. Servidor de medias debe tener como mínimo: 16 GB de RAM, 1 TB de disco duro y un microprocesador (6 x 800 GHz).

2.4 Historia de usuarios

Una historia de usuarios (HU) describe una funcionalidad que será útil para el usuario, o comprador, de un sistema de software. Aunque normalmente las HU, asociadas a las metodologías ágiles, suelen escribirse en tarjetas (52). Son la técnica utilizada para especificar los requisitos del software. Cada historia de usuario debe ser comprensible y delimitada para que los programadores puedan implementarla en poco tiempo (53).

Se muestran las historias de usuarios de los requisitos funcionales: Generar reporte de las actividades de un curso y Generar reporte de los usuarios registrados en la plataforma; el resto de los artefactos de este tipo se encuentran en el directorio Historia de Usuarios.

Tabla 5. Historia de usuario. Actividades de un curso

Número: 1	Nombre del requisito: Generar reporte de las actividades de un curso.	
Programador: Beatriz Vargas Más	Iteración asignada: 1	
Prioridad: Media	Complejidad: Alta	
Tiempo estimado: 32 horas/hombre	Riesgo en desarrollo: N/A	
Descripción:		
1- Objetivo:		
Permitir generar el reporte de las actividades de un curso.		
2- Acciones para lograr el objetivo (precondiciones y datos):		
Para generar el reporte de las actividades en un curso se debe:		
<ul style="list-style-type: none"> - Estar autenticado en el sistema. - Tener registrado al menos un curso en el sistema. - Tener al menos una actividad asociada a un curso. 		
3- Comportamientos válidos y no válidos (flujo central y alternos):		
Es obligatorio seleccionar en el filtro un curso.		
4- Flujo de la acción a realizar:		
Cuando el usuario selecciona la opción generar reporte de las actividades de un curso, se muestre un listado con los siguientes datos:		
<ul style="list-style-type: none"> - Generado por - Creado (Fecha de creado) - Curso - Actividad 		
Observaciones:		
Prototipo de interfaz:		



Tabla 6. Historia de usuario. Usuarios registrados en la plataforma

Número: 21	Nombre del requisito: Generar reporte de los usuarios registrados en la plataforma.	
Programador: Beatriz Vargas Más	Iteración asignada: 1	
Prioridad: Media	Complejidad: Media	
Tiempo estimado: 32 horas/hombre	Riesgo en desarrollo: N/A	
Descripción:		
1- Objetivo:		
Permitir generar el reporte de los usuarios registrados en la plataforma.		
2- Acciones para lograr el objetivo (precondiciones y datos):		
Para generar el reporte de los usuarios registrados en la plataforma se debe:		
<ul style="list-style-type: none"> - Estar autenticado en el sistema. - Tener registrado al menos un usuario en el sistema. 		
3- Comportamientos válidos y no válidos (flujo central y alternos):		
4- Flujo de la acción a realizar:		
Cuando el usuario selecciona la opción generar reporte de los usuarios registrados en la plataforma, se muestre un listado con los siguientes datos:		
<ul style="list-style-type: none"> - Generado por - Creado (Fecha de creado) - Nombre/ Apellidos - Usuario - Correo electrónico - Fecha 		
Observaciones:		

Prototipo de interfaz:

zera Plataforma Educativa

Usuarios registrados en la plataforma

Generado por: Nombre Apellido
Fecha: dd/mm/aa , HH:mm AM/PM

0
Usuario(s) registrado(s)

Nombre / Apellidos	Usuario	Correo electrónico	Fecha

Usuarios registrados en la plataforma

UCI

2.5 Arquitectura de software

La arquitectura de software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema. La vista arquitectónica es una vista abstracta, aportando el más alto nivel de comprensión y la supresión o diferimiento del detalle inherente a la mayor parte de las abstracciones (54).

Patrón Modelo-Vista-Controlador

El patrón arquitectónico Model-View-Controller (MVC) divide una aplicación interactiva en tres componentes. El Modelo contiene la funcionalidad central y los datos, las Vistas despliegan la información al usuario y los Controladores manejan la entrada del usuario. Estos dos últimos componentes, en conjunto, forman la interfaz del usuario. Un mecanismo de propagación de cambio asegura la consistencia entre la interfaz del usuario y el modelo (55). El patrón MVC permite 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.

Para el desarrollo del módulo de reportes para la Plataforma Educativa Xauce Zera 2.0 deseado se utilizó Symfony 2. Este framework diseñado para optimizar el desarrollo de las aplicaciones web que utiliza la arquitectura MVC como la capa de abstracción de base de datos, el controlador frontal y las acciones. Además, sigue las mejores prácticas de patrones de diseño para la web y está escrito en PHP. La plataforma está implementada con el framework Symfony.

Modelo: administra el comportamiento y maneja los datos del dominio de aplicación, da respuesta a peticiones de información sobre el estado (usualmente formulados desde la vista), y responde a instrucciones de cambiar el estado (habitualmente desde el controlador) a la vista.

Vista: gestiona lo relacionado con mostrar la información al usuario. La vista está representada por ficheros escritos en PHP que se encargan de construir la página HTML con la que interactúa el usuario.

Controlador: interpreta las acciones del ratón y el teclado, que son los eventos lanzados por la entrada estándar del usuario, informando al modelo y/o a la vista de estas acciones para que se ejecuten los cambios según resulten apropiados (56).

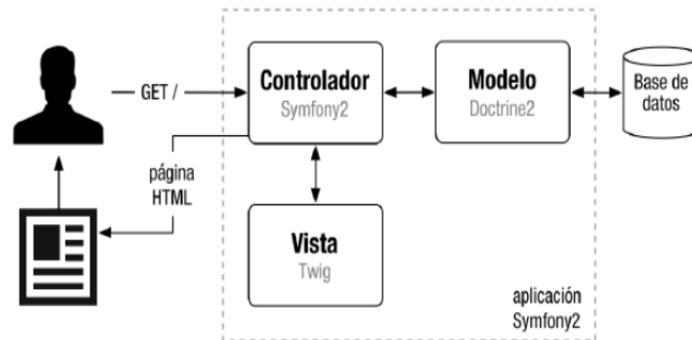


Figura 2. Patrón Modelo-Vista-Controlador. Fuente: (57).

2.6 Patrones del diseño

Un patrón de diseño describe una estructura de diseño que resuelve un problema particular del diseño dentro de un contexto específico y entre “fuerzas” que afectan la manera en la que se aplica y en la que se utiliza dicho patrón (57).

El objetivo de cada patrón de diseño es proporcionar una descripción que permita a un diseñador determinar 1) si el patrón es aplicable al trabajo en cuestión, 2) si puede volverse a usar (con lo que se ahorra tiempo de diseño) y 3) si sirve como guía para desarrollar un patrón distinto en funciones o estructura (57).

Patrones GRASP

Los patrones de principios generales para asignar responsabilidades (GRASP), describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones (58).

- **Experto en Información**

En el caso del módulo de reportes este patrón es utilizado en la capa de abstracción del modelo de datos. Con el uso del ORM Doctrine, Symfony genera automáticamente las

clases que representan las entidades del modelo de datos con todas las funcionalidades comunes. En la solución este patrón se evidencia en la clase entidad Reports.php.

```
/** Reports. ... */
class Reports
{
    /** @var integer $id ... */
    private $id;

    /** Get id ... */
    public function getId(){...}

    /** @var string $name ... */
    private $name;

    /** @return string ... */
    public function getName(){...}

    /** @param string $name ... */
    public function setName($name){...}
}
```

Figura 3. Elaboración propia. Patrón Experto en la solución.

- **Creador**

Este patrón permite crear objetos de una clase determinada, lo que es muy empleado en sistemas orientados a objetos. Se evidencia en la clase controladora DefaultController.php donde se crean las instancias de clases necesarias para el trabajo con los reportes y la base de datos.

```
public function previewAction()
{
    $request = $this->get('request_stack')->getCurrentRequest();
    $reportsManager = $this->get('fortes_reports.manager');
    $reportsServiceManager = $this->get('fortes_reports_sdr_manager');
    $id = $request->get('id');
    $report_obj = $reportsManager->findOneBy(array('id' => $id));
}
```

Figura 4. Elaboración propia. Patrón Creador en la solución.

- **Alta Cohesión**

Este patrón es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase, cada elemento del diseño debe realizar una labor única dentro del sistema. El patrón alta cohesión se evidencia en las clases que extienden de ReportsRepository.php. Ver en (Figura 13).

- **Bajo Acoplamiento**

El acoplamiento mide el grado en que una clase está conectada a otras clases, tiene conocimiento y recurre a ellas o de alguna manera depende de otra. Acoplamiento bajo significa que una clase no depende de muchas clases, permite que las clases sean más reutilizables, lo que implica mayor productividad. El patrón bajo acoplamiento se evidencia en las clases que extienden de ReportsRepository.php. Ver en (Figura 13).

- **Controlador**

Este se basa en asignar la responsabilidad de todos los eventos realizados a una clase específica que constituye un único punto de entrada para cada evento. En del framework Symfony el patrón se evidencia en las clases que forman la capa Controller, todas las peticiones son procesadas por un solo controlador frontal. En la solución este patrón se evidencia en la clase controladora DefaultController.php.

```
public function exportAction(){  
  
    $request = $this->get('request_stack')->getCurrentRequest();  
    $reportsManager = $this->get('fortes_reports.manager');  
    $reportsServiceManager = $this->get('fortes_reports_sdr_manager');  
    $id = $request->get('id');  
    $format = $request->get('format');  
    $report_obj = $reportsManager->findOneBy(array('id' => $id));  
    if ($report_obj){  
        if (!empty($report_obj->getJrxmlFileName() )){  
            $report_model = $reportsServiceManager->getReportByJRXMLFileName($report_obj->getJrxmlFileName());  
            $jrxmlFileName = $report_model->getJrxmlFileName();  
            $filter_form = $report_model->getReportFormFilter(null, array());  
            $sdrParams = array();  
            if (!empty($filter_form)){  
                $allParams = $request->get('form_filters');  
                $sdrParams = $this->get('fortes_reports.utils')->getSDRFormatParams($allParams);  
            }  
        }  
    }  
}
```

Figura 5. Elaboración propia. Patrón Controlador en la solución.

Patrones GOF

Los patrones de la "pandilla de los cuatro" (GoF ⁶), describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos (59).

- **Singleton**

Permite el manejo de objetos únicos y que sean accesibles a otros objetos. Además, brinda el acceso controlado a una única instancia. Symfony usa por defecto este patrón al crear las instancias de los servicios. En la solución se evidencia en el servicio fortes_reports.utils, que tiene como función devolver una instancia de la clase ReportsUtils.php.

⁶ Del inglés "Gang of Four" integrada por: Erich Gamma, Richard Helm, Ralph Jonson y John Vlissides.

```

    if(!empty($filter_form)){
        $allParams = $request->get('form_filters');
        $sdrParams = $this->get('fortes_reports.utils')->getSDRFormatParams($allParams);
    }

    $report_parameters = $report_model->getReportParameters();

    // parametros para plantilla del reporte
    $sdrParams = $this->get('fortes_reports.utils')->addSDRTemplateParameters($sdrParams);

    $_sdrParams = array_merge($report_parameters, $sdrParams);

    // preview del reporte
    $preViewReport = $report_model->export($jxmlFileName, $_sdrParams, true, $format);

    return $preViewReport;
}

```

Figura 6. Elaboración propia. Patrón Singleton en la solución.

- Decorador

Permite añadir funcionalidades o responsabilidades a un objeto de forma dinámica, propone que cuando cambie el estado de un objeto sean actualizados de forma automática todos los otros objetos que dependen de este. El motor de plantillas Twig, está provisto de un mecanismo de herencia gracias al cual la decoración de plantillas se realiza de una manera flexible y versátil. Este patrón se observa en todos los archivos de tipo .html.twig del módulo y que son los que contienen el código HTML que es común para todas las páginas que heredan de estos.

A continuación, se muestra un ejemplo del empleo del patrón en la aplicación. La vista es: preview_storage_report.html.twig.

```

{% trans_default_domain "ReportsBundle" %}
<link rel="stylesheet" href="{{ asset('bundles/reports/css/reports.css') }}">

<br/><br/>

<div id="reports_storage_reources_container_id" >
    <div id="reports_storage_reources_container_inner_id" style="...">

        
        <label id="report_name">{{ report_obj.getName() }}</label></br>

    <div id="div">
        <label id="report_general">{{ 'reports.parameters.generated_by'|trans |raw }} :</label>
        <label id="report_general">{{ user.__toString() }}</label>
    </div></br>

    <div id="div" ...>

        <hr id="hr" noshade="noshade" />

        <span class="pull-right">
            {% set totalSize = totalSize|default(0) %}
            <b id="b1">{{ 'Total: ' ~ container.get('fortes_learning.resource.utils').formatBytes(totalSize, 2) |raw }}</b>
        </span>

        <table id="reports_storage_reources_table_id" class="table table-striped table-bordered table-hover dataTable no-foot">
            <thead>
                <tr id="thTable">
                    <th id="thTable">{{ 'reports.generals.type'|trans |raw }}</th>

```

Figura 7. Elaboración propia. Patrón Decorador en la solución.

- **Inyección de dependencias**

En la solución se utiliza en varias clases, ejemplo de ello es la clase ReportsManager, donde se hace uso del objeto entitymanager (em), mediante el cual Doctrine realiza la manipulación de la información (buscar, insertar, modificar y eliminar registros en las tablas) sin que el programador tenga que escribir sentencias SQL.

```
1  /** GET a report by criteria. ... */
2  public function findOneBy($params)
3  {
4      return $this->em->getRepository( 'entityName: ReportsBundle:Reports')->findOneBy($params);
5  }
6
7  /** GET all available reports. ... */
8  public function findAll(){
9      return $this->em->getRepository( 'entityName: ReportsBundle:Reports')->findAll();
10 }
11
12 /** GET all available reports by type. ... */
13 public function getAvailableReports($type = null){
14     return $this->em->getRepository( 'entityName: ReportsBundle:Reports')->getAvailableReports($type);
15 }
```

Figura 8. Elaboración propia. Patrón Inyección de dependencias en la solución.

2.7 Modelo de datos

El modelado de datos permite definir todos los objetos de datos que se procesan dentro del sistema, la relación entre ellos y otro tipo de información que sea pertinente para las relaciones. El diagrama entidad-relación (DER) aborda dichos aspectos y representa todos los datos que se introducen, almacenan, transforman y generan dentro de una aplicación (57).

El modelo que se muestra ofrece una descripción sobre la representación de los datos utilizados para realizar los reportes de la solución. Este modelo contiene características propias de estos objetos y la forma en que están relacionados.

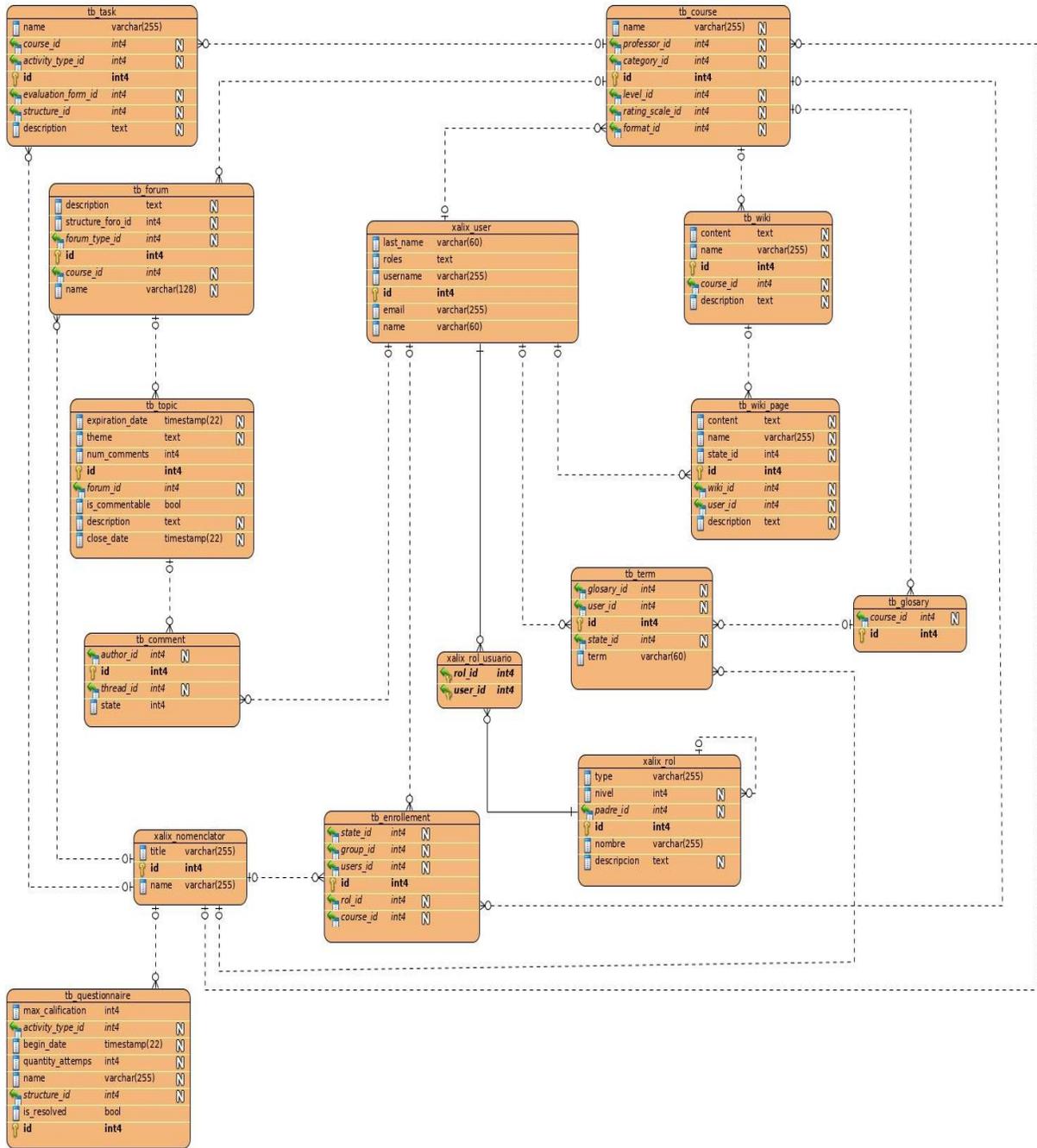


Figura 9. Elaboración propia. Modelo de datos de Zera.

El siguiente modelo de datos es el utilizado en la solución. Este modelo está compuesto por una sola tabla, porque al hacer uso de una herramienta externa para trabajar con los reportes, lo único que se guarda en la tabla es el reporte completo.

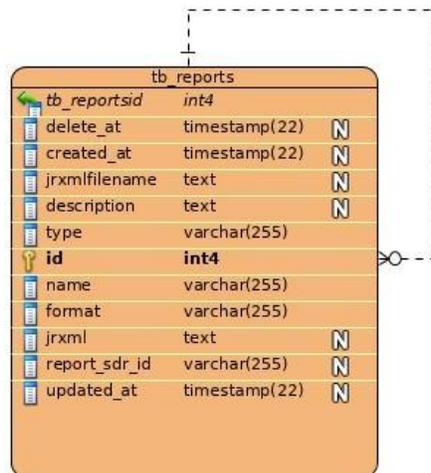


Figura 10. Elaboración propia. Modelo de datos de la solución.

Descripción de las tablas de la base de datos

tb_reports		
Descripción: En esta tabla se agrupa la información correspondiente a los reportes		
Atributo	Tipo	Descripción
tb_reportsid	int4	Etiqueta única que identifica al reporte
Id	int4	Etiqueta única que identifica al reporte
type	varchar(255)	Identifica el tipo de reporte
update_at	timestamp(22)	Almacena la fecha de subida del reporte
description	text	Contiene la descripción asociada al reporte
report_sdr_id	varchar(255)	Represente el id del reporte en el SDR
jrxmlfilename	text	Guarda el nombre del archivo jrxml
created_at	timestamp(22)	Almacena la fecha de creación del reporte
jrxml	text	Identifica si el reporte se encuentra diseñado en un archivo jrxml
delete_at	timestamp(22)	Almacena la fecha de eliminación del reporte
format	varchar(255)	Etiqueta para especificar el formato del reporte
name	varchar(255)	Representa el nombre el reporte

2.8 Modelo de análisis

El modelo de análisis es una abstracción, o generalización, del diseño. Un modelo de análisis puede verse como un primer corte de un modelo de diseño (58). El objetivo del modelo de análisis es traducir los requisitos a una especificación que describe cómo implementar el sistema (60).

2.8.1 Diagrama de clases del análisis

Una clase de análisis representa una abstracción de una o varias clases y/o subsistemas del diseño del sistema. Las clases del análisis siempre encajan en uno de los tres estereotipos básicos: clase de interfaz, de control y de entidad (60). Los diagramas de clases del análisis (DCA) son utilizados para comprender de forma general la estructura del sistema y sirve como entrada para la etapa de diseño.

La siguiente figura muestra el diagrama de clases del análisis correspondiente a los requisitos funcionales.

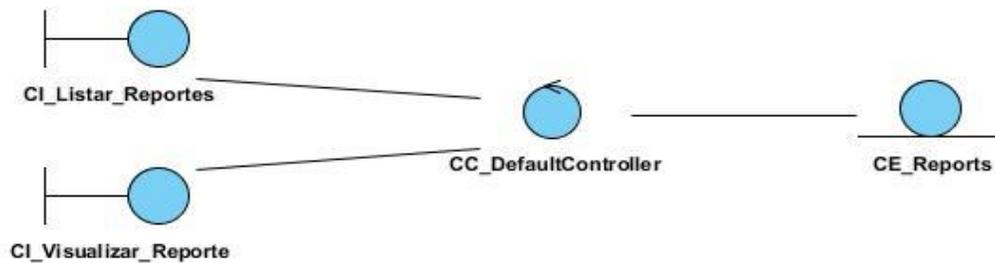


Figura 11. Elaboración propia. Diagrama de clases del análisis.

2.8.2 Diagrama de colaboración del análisis

Los diagramas de colaboración muestran las interacciones entre objetos creando enlaces entre ellos y añadiendo mensajes a esos enlaces. El nombre de un mensaje debería denotar el propósito del objeto que invoca en la interacción con el objeto invocado. Son una representación más concreta y detallada que los diagramas de clases del análisis, aunque también representan la parte estática del sistema (60).

La figura 12 representa el diagrama de colaboración del análisis (DC) perteneciente a los requisitos funcionales que deben seleccionar en el filtro un curso. El resto de los artefactos de este tipo se encuentran en el directorio Diagramas de Colaboración del Análisis.

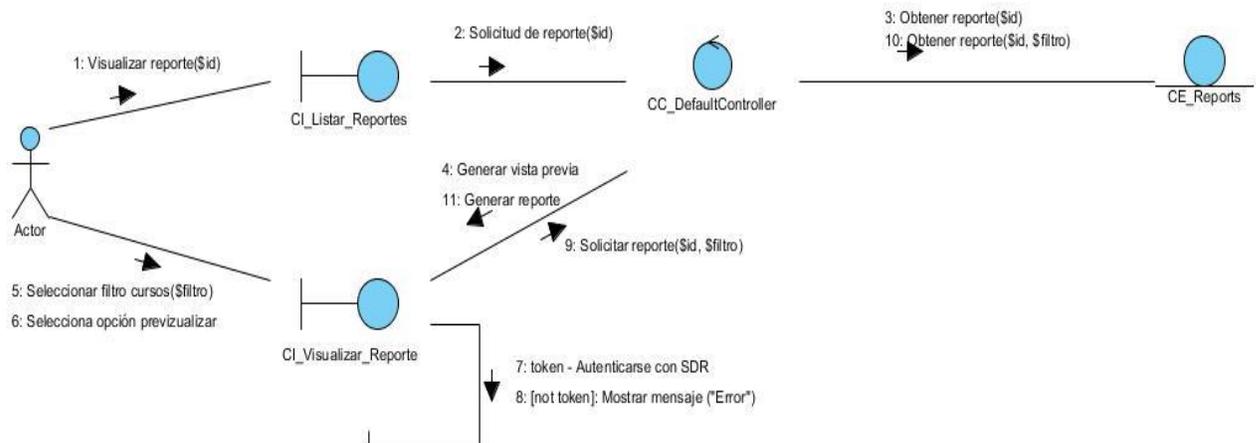


Figura 12. Elaboración propia. Diagrama de colaboración para requisitos con filtro cursos.

2.9 Modelo de diseño

Es el conjunto de diagramas que describen el diseño lógico. Comprende los diagramas de clases software, diagramas de interacción, diagramas de paquetes, entre otros (58).

El modelo de diseño proporciona detalles sobre arquitectura del software, estructuras de datos, interfaces y componentes que se necesitan para implementar el sistema. Es el trabajo principal que se produce durante el diseño del software. Los elementos del modelo de diseño usan muchos de los diagramas UML que se utilizaron en el modelo del análisis (57).

2.9.1 Diagrama de clases del diseño

Los diagramas de clases del diseño se encuentran conectados a la realización de un caso de uso o una historia de usuario. Son los encargados de mostrar las clases participantes, subsistemas y sus relaciones, así como, los atributos y operaciones correspondientes a cada clase (60).

Los diagramas de clases del diseño (DCD) que se muestran pertenecen a los requisitos funcionales generar reporte de las actividades de un curso y generar reporte de los usuarios registrados en la plataforma, respectivamente. El resto de los artefactos de este tipo se encuentran en el directorio Diagramas de Clases del Diseño.

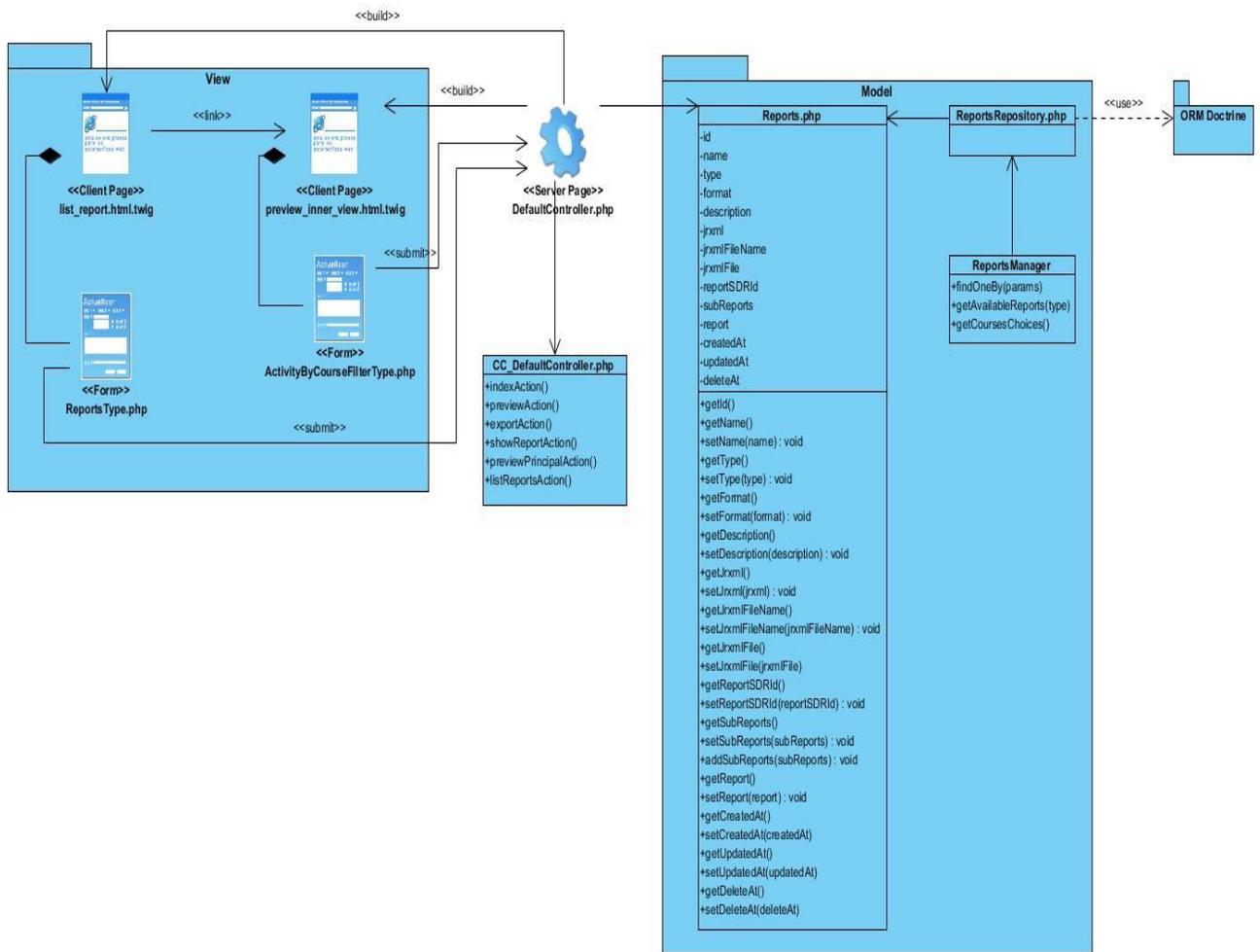


Figura 13. Elaboración propia. Diagrama de clases del diseño actividades de un curso.

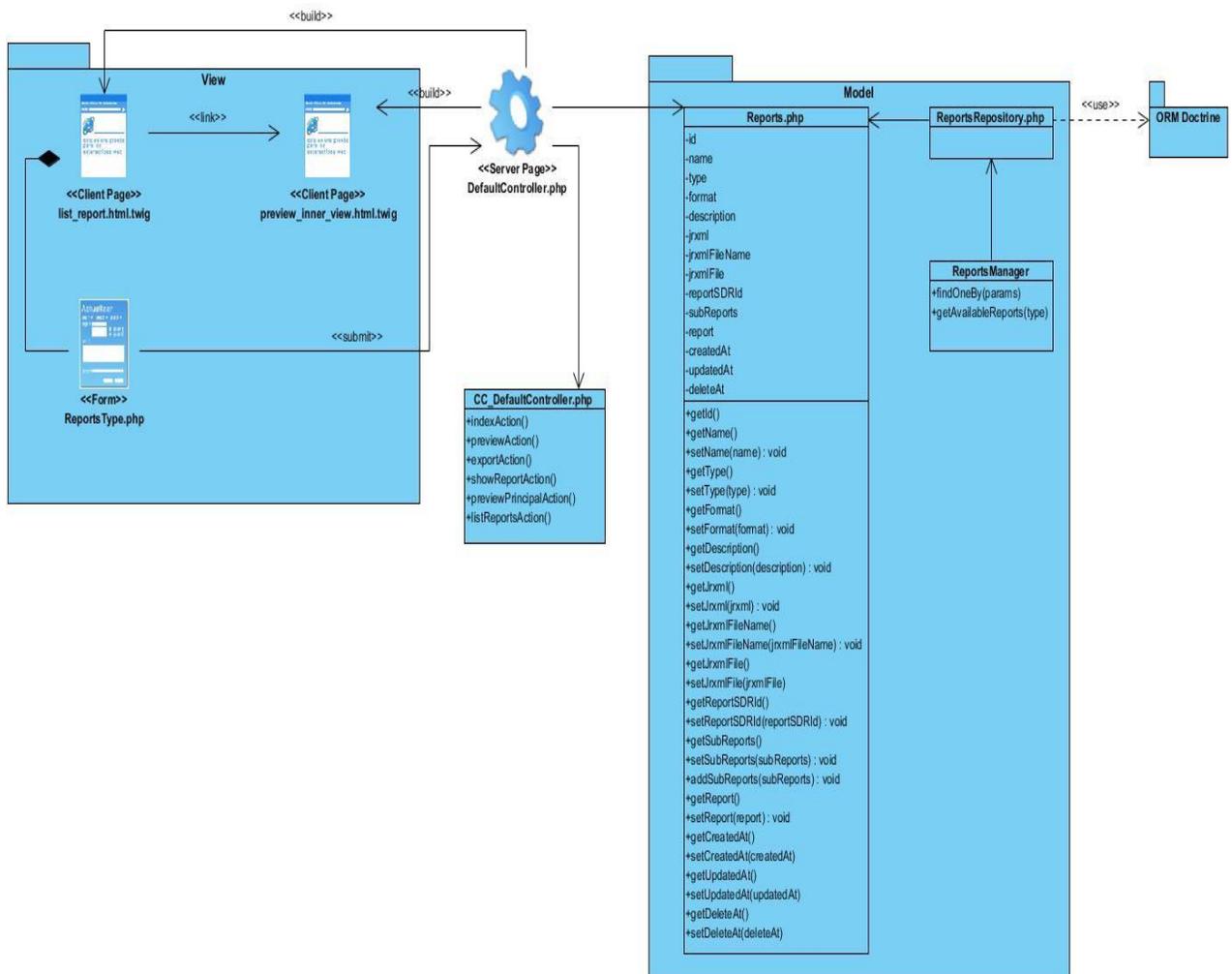


Figura 14. Elaboración propia. Diagrama de clases del diseño usuarios registrados en la plataforma.

2.9.2 Diagrama de secuencia del diseño

Los diagramas de secuencia muestran las interacciones entre objetos mediante transferencia de mensajes entre objetos o subsistemas. El nombre del mensaje debería indicar una operación del objeto que recibe la invocación o de una interfaz que el objeto proporciona (60). Un diagrama de secuencia se usa para mostrar las comunicaciones dinámicas entre objetos durante la ejecución de una tarea (57).

Se presenta el diagrama de secuencia del diseño (DSD) perteneciente a los requisitos funcionales que deben seleccionar en el filtro un curso. El resto de los artefactos de este tipo se encuentran en el directorio Diagramas de Secuencia del Diseño.

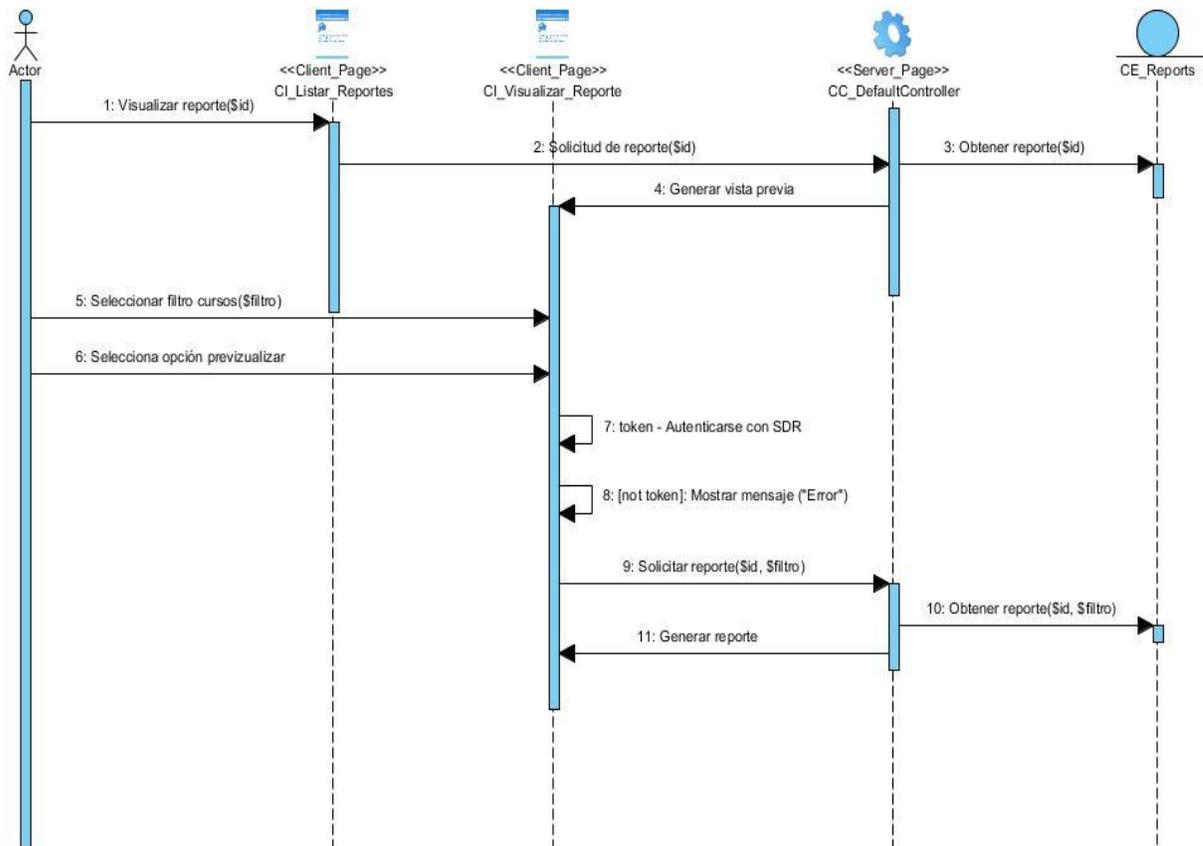


Figura 15. Elaboración propia. Diagrama de secuencia para requisitos con filtro cursos.

Conclusiones del capítulo

Después de realizado este capítulo se puede concluir que:

- Fue descrita la propuesta de solución determinando realizar un módulo a la Plataforma Educativa XAUCE ZERA para gestionar la información mediante reportes.
- En la especificación de requisitos fueron identificados 21 requisitos funcionales y 9 no funcionales.
- Se confeccionaron los artefactos definidos en la etapa de análisis y diseño de la metodología AUP-UCI, permitiendo definir las bases necesarias para la implementación del módulo y posibilitando que se obtuvieran de una forma concreta y detallada las relaciones que se establecen entre las entidades del sistema.
- Se definió el patrón arquitectónico MVC y los patrones de diseño a utilizar. El uso de estos patrones permitirá garantizar la reutilización de funcionalidades.

Capítulo III. Implementación y pruebas de la solución

En el presente capítulo se describen los elementos requeridos para desarrollar el módulo de reportes para la Plataforma Educativa XAUCE ZERA a partir de los resultados obtenidos de las etapas de análisis y diseño, dando inicio a la fase de implementación de la propuesta de solución. Se presenta el diagrama de componentes, diagrama de despliegue, diagrama de paquetes y se definen los estándares de codificación utilizados para el desarrollo del módulo de reportes. También, se describen y realizan las pruebas de software a aplicar al módulo de reportes implementado, haciendo uso de los Diseños de Casos de Prueba.

3.1 Modelo de implementación

La implementación de un software se encarga de construir los componentes de un sistema basándose en los resultados de la disciplina Análisis y Diseño, específicamente en los artefactos generados. Entre los componentes se pueden encontrar: datos, archivos, ejecutables, código fuente y los directorios. El objetivo de esta disciplina es transformar su modelo(s) en código ejecutable y realizar un nivel básico de las pruebas, en particular, la unidad de pruebas (32). Además, se utilizan los estándares de codificación para llevar a cabo las implementaciones haciendo uso de las historias de usuarios.

Un Modelo de Implementación describe cómo los elementos del modelo de diseño, se implementan en términos de componentes, ficheros de código fuente, ejecutables, entre otras. También describe cómo se organizan los componentes de acuerdo a los mecanismos de estructuración disponibles en el entorno de implementación y lenguaje o lenguajes de implementación empleados, y cómo dependen los componentes unos de otros (60). Dentro del modelo de implementación se encuentran los diagramas de componentes.

3.1.1 Diagrama de componentes

Un diagrama de componentes representa la separación de un sistema de software en componentes físicos y muestra las dependencias entre estos, con un nivel más alto de abstracción que un diagrama de clases. Los componentes físicos incluyen: archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables y paquetes (51). Los diagramas de componentes prevalecen en el campo de la arquitectura de software, pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema (60).

Se muestra el diagrama de componentes correspondiente al módulo de reportes para la Plataforma Educativa XAUCE ZERA:

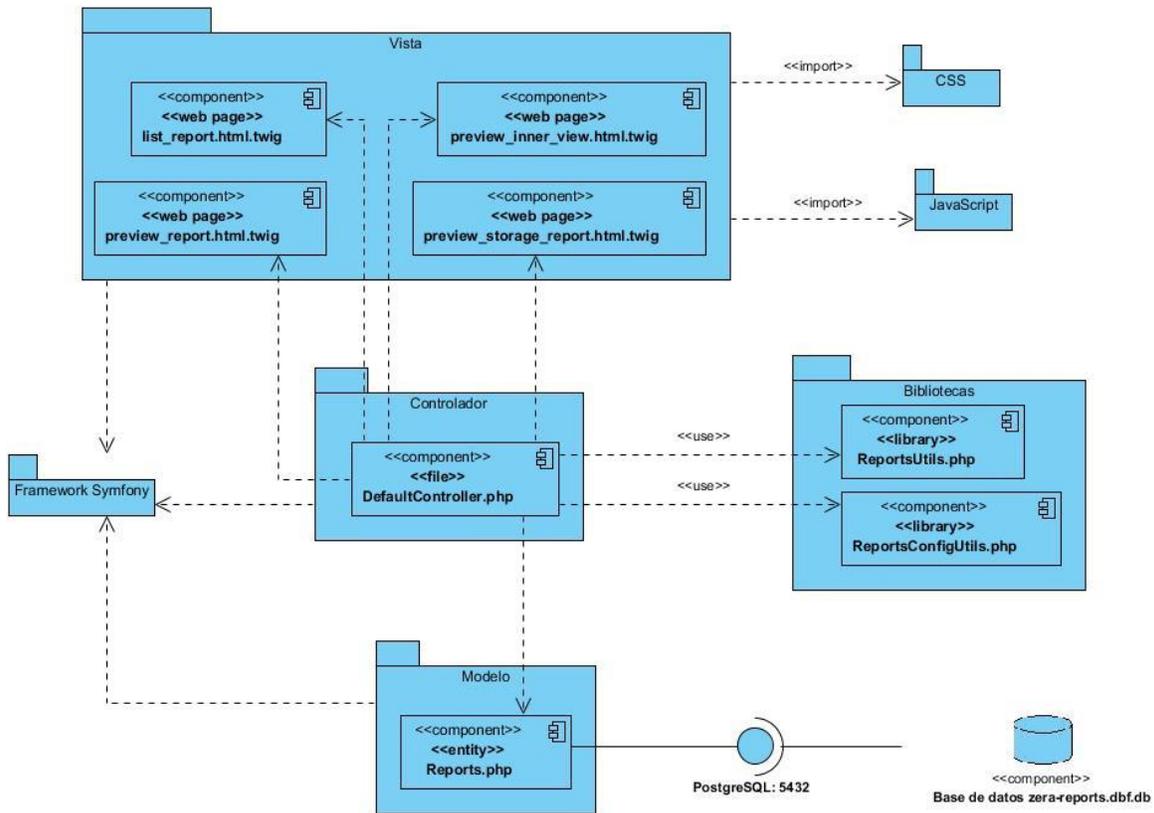


Figura 16. Elaboración propia. Diagrama de componentes.

3.1.2 Diagrama de despliegue

Un diagrama de despliegue es un artefacto generado en la disciplina Análisis y Diseño que muestra cómo se configuran las instancias de los componentes y los procesos para la ejecución run-time en las instancias de los nodos de proceso (58). Se utilizan como parte del diseño de la arquitectura y se representan en forma de descriptor. De este modo, las funciones principales de un sistema se representan en el contexto del sistema que las contendrá (57).

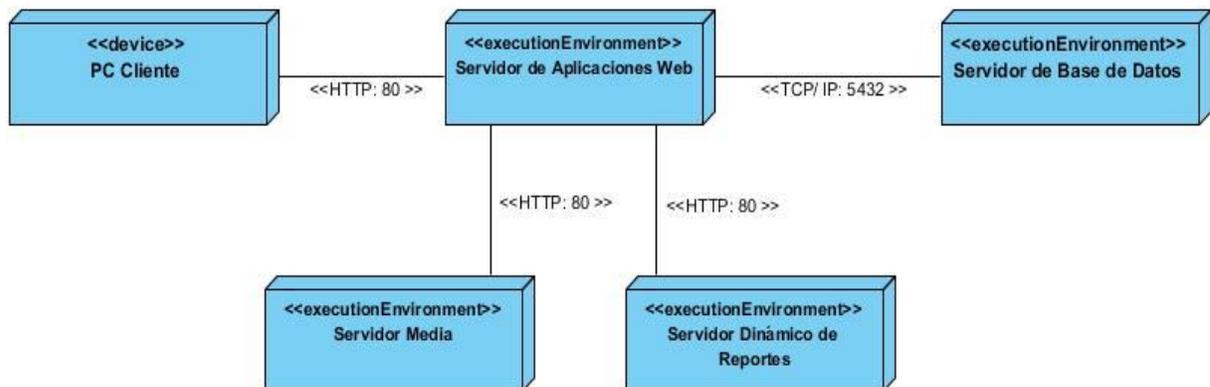


Figura 17. Elaboración propia. Diagrama de despliegue.

3.1.3 Estándares de codificación

Un estándar de codificación comprende los aspectos de la generación de código. Al inicio de un desarrollo de software, es necesario establecer un estándar de codificación para asegurar que los programadores del proyecto trabajen de forma coordinada, mantengan un código de calidad y puedan identificar las variables, las funciones o métodos (61).

- Se utiliza la notación camelCase en el archivo JavaScript para los nombres de variables y funciones. Todos los nombres comienzan con una letra. Se utiliza la palabra reservada "var" para declarar variables.

```
/* ***** FILTROS ***** */
var select_selector = '#search_filters';
var prefix = 'filter';
var url = '';

$(document).ready(function(){
    filterReport();
    exportReport();

    $(select_selector).bind('change', function () {
        if ($(this).val() != '-1') {
            addFilter($(this).val());
        }

        // instanciando los select del formulario
        $('#reports_filters_form_id').find('select').select2({theme: 'bootstrap',
            containerCss: {height: '40px !important;'}});
    });
    $('#selected_filters > div').each(function( elem ){
        var block_name = $(this).attr('id').substr(prefix.length);
        $(select_selector+ " > option[value^=" + block_name + "']").attr('disabled','disabled');
    });
});
```

Figura 18. Elaboración propia. Estándar de codificación en archivo JavaScript.

- El nombre de los atributos de las etiquetas debe estar escrito en minúscula.

```
{% trans_default_domain "ReportsBundle" %}

<div id="report_preview_container">
<div id="actions_export_container_id" class="row-fluid hide" data-report_id="{{ report_id }}">
<div id="report_export_id" class="btn-group pull-right" role="group" aria-label="Basic example">
<button type="button" class="btn btn-primary" data-export="{{ path('fortes_reports_export', {'id': report_id, 'format': 'xls' }) }}">
    xls
</button>
<button type="button" class="btn btn-primary" data-export="{{ path('fortes_reports_export', {'id': report_id, 'format': 'pdf' }) }}">
    pdf
</button>
<button type="button" class="btn btn-primary" data-export="{{ path('fortes_reports_export', {'id': report_id, 'format': 'docx' }) }}">
    doc
</button>
</div>
</div>
```

Figura 19. Elaboración propia. Estándar de codificación en archivo html.twig.

- Para definir los servicios en las clases PHP se deben cumplir los siguientes indicadores:

- Los nombres de los servicios contienen grupos separados por puntos.
- El alias de Inyección de Dependencias del bundle es el primer grupo.
- Se utiliza minúsculas para los nombres de servicios y sus parámetros.
- Un nombre de grupo utiliza la notación guión bajo.

```
$request = $this->get('request_stack')->getCurrentRequest();
$reportsManager = $this->get('fortes_reports.manager');
$reportsServiceManager = $this->get('fortes_reports_sdr_manager');
$id = $request->get('id');
```

Figura 20. Elaboración propia. Estándar de codificación en clase PHP. Servicios.

- Se utiliza namespaces (ruta de donde se encuentra cada clase en el componente) para todas las clases.

```
namespace FORTES\ReportsBundle\Controller;
```

Figura 21. Elaboración propia. Estándar de codificación en clase PHP. Namespace.

3.2 Pruebas de software

Las pruebas de software implican ejecutar una implementación del software con datos de prueba. Se examinan las salidas del software y su entorno operacional para comprobar que funciona tal y como se requiere. Son una técnica dinámica de verificación y validación (51). El objetivo de las pruebas de software es descubrir diferentes tipos de errores, que pueden ser corregidos con el mínimo tiempo y esfuerzo. Estas comienzan con la prueba de unidad, luego avanza hacia la prueba de integración y culmina con las pruebas de validación y sistema (57).

3.2.1 Estrategia de pruebas

Una estrategia de prueba de software proporciona una guía que describe los pasos que deben realizarse como parte de la prueba, cuándo se planean y se llevan a cabo dichos pasos, y cuánto esfuerzo, tiempo y recursos se requerirán. Por tanto, cualquier estrategia de prueba debe incorporar la planificación de la prueba, el diseño de casos de prueba, la ejecución de la prueba y la recolección y evaluación de los resultados (57).

Según la metodología AUP-UCI existen tres disciplinas de pruebas:

- Pruebas internas.
- Pruebas de liberación.
- Pruebas de aceptación.

Pruebas internas

En esta disciplina se verifica el resultado de la implementación probando cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones

finales a ser liberadas (32). A continuación, se detallan las pruebas llevadas a cabo durante esta disciplina, con el objetivo de asegurar la calidad del resultado de la implementación.

➤ **Pruebas unitarias:**

A medida que se implementan los componentes, se diseñan y efectúan pruebas unitarias para cada uno. Las pruebas unitarias que se crean deben implementarse con el uso de una estructura que permita automatizarlas, de modo que puedan ejecutarse en repetidas veces y con facilidad (57). Para realizar este tipo de pruebas se puede hacer uso del método de caja blanca.

El método de caja blanca del software se basa en el examen cercano de los detalles de procedimiento. Las rutas lógicas a través del software y las colaboraciones entre componentes se ponen a prueba al revisar conjuntos específicos de condiciones y/o bucles (57).

La ruta básica es la técnica de prueba del método de caja blanca que se utilizó, porque permite obtener una medida de la complejidad de un diseño procedimental, y utilizar esta medida como guía para la definición de una serie de caminos básicos de ejecución, diseñando casos de prueba que garanticen que cada camino se ejecuta al menos una vez (57). Esta técnica le fue aplicada al método `exportAction` perteneciente `DefaultController`. Se tuvo en cuenta esta función ya que todos los requisitos tenían como condición que se pueda exportar el reporte obtenido. Para iniciar la técnica se enumeraron todos los nodos definidos en cada porción del código.

```

public function exportAction() {
    $request = $this->get("request_stack")->getCurrentRequest(); //1
    $reportsManager = $this->get("fortes_reports.manager"); //1
    $reportsServiceManager = $this->get("fortes_reports_sdr_manager");//1
    $id = $request->get("id"); //1
    $format = $request->get("format"); //1
    $report_obj = $reportsManager->findOneBy(array("id" => $id)); //1

    if($report_obj && !empty($report_obj->getJrxmlFileName())){ //2
        $report_model = $reportsServiceManager-
>getReportByJRXMLFileName($report_obj->getJrxmlFileName()); //3
        $jrxmlFileName = $report_model->getJrxmlFileName(); //3
        $filter_form = $report_model->getReportFormFilter(null, array());//3
        $sdrParams = array(); //3
        if(!empty($filter_form)){ //4
            $allParams = $request->get("form_filters"); //5
            $sdrParams = $this->get("fortes_reports.utils")-
>getSDRFormatParams($allParams); //5
        }
        else{
            $report_parameters = $report_model->getReportParameters(); //6
            // parametros para plantilla del reporte
            $sdrParams = $this->get("fortes_reports.utils")->addSDRTemplateParameters(
            $sdrParams); //6
            $sdrParams = array_merge($report_parameters, $sdrParams); //6
        }
        // preview del reporte
        $preViewReport = $report_model->export($jrxmlFileName, $sdrParams, true,
        $format); //7
        return $preViewReport; //8
    }
    return $this->redirect(
        $this->generateUrl("home_page")
    ); //9
}

```

Figura 22. Elaboración propia. Método exportAction.

1. **Confeccionar el grafo de flujo:** utilizando el código anterior se realizó la representación del grafo de flujo, el cual describe un flujo de control lógico y está compuesto por los siguientes elementos (57):
 - **Nodos de gráfica de flujo:** Son círculos que representan una o más instrucciones procedimentales.
 - **Aristas o enlaces:** Son flechas que representan el flujo de control y son análogas a las flechas del diagrama de flujo.
 - **Regiones:** Son las áreas delimitadas por aristas y nodos.

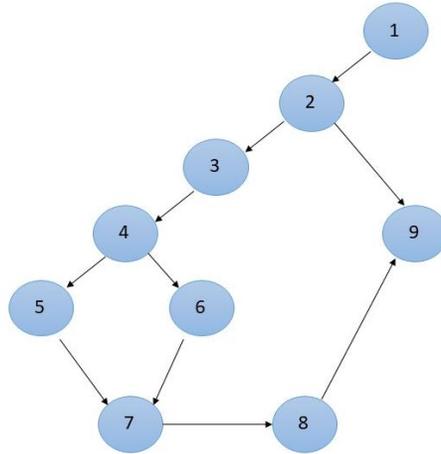


Figura 23. Elaboración propia. Grafo de flujo.

2. **Calcular la complejidad ciclomática:** proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado define el número de caminos independientes del conjunto básico de un programa, y proporciona un límite superior para el número de pruebas que deben aplicarse para asegurar que todas las instrucciones se hayan ejecutado al menos una vez (57).

Cálculo de la complejidad ciclomática:

$V(G) = 3$, $V(G)$ es la cantidad de regiones.

$V(G) = A - N + 2$, donde A es el número de aristas del grafo y N es el número de nodos.

$$V(G) = A - N + 2 = (10 - 9) + 2 = 3$$

$V(G) = P + 1$, donde P es el número de nodos predicados, es decir, nodos donde emergen 2 o más aristas.

$$V(G) = 2 + 1 = 3$$

3. **Determinar un conjunto básico de rutas linealmente independientes:** el valor de $V(G)$ indica el número de rutas linealmente independientes de la estructura de control del programa (57), por lo que se obtuvieron las siguientes 3 rutas independientes:

Ruta independiente 1: 1-2-3-4-5-7-8-9

Ruta independiente 2: 1-2-3-4-6-7-8-9

Ruta independiente 3: 1-2-8-9

4. **Obtención de casos de pruebas:** cada ruta independiente es un caso de prueba a realizar, de forma que los datos introducidos provoquen que se visiten las sentencias vinculadas a cada nodo del camino (57). Las 3 rutas independientes obtenidas se corresponden con la confección de casos de prueba.

Tabla 7. Caso de prueba. Ruta independiente 1

Caso de prueba: Ruta independiente 1	
Descripción: Esta función exporta el reporte que se corresponde con el id seleccionado.	
Entrada	
Resultados esperados	Se exporta el reporte generado en el formato que fue seleccionado.
Condiciones	Se debe haber generado un reporte, debe ser de tipo jrxml y el campo filtro no debe estar vacío.

Tabla 8. Caso de prueba. Ruta independiente 2

Caso de prueba: Ruta independiente 2	
Descripción: Esta función exporta el reporte que se corresponde con el id seleccionado.	
Entrada	
Resultados esperados	Se exporta el reporte generado en el formato que fue seleccionado.
Condiciones	Se debe haber generado un reporte, debe ser de tipo jrxml y no contiene el campo filtro.

Tabla 9. Caso de prueba. Ruta independiente 3

Caso de prueba: Ruta independiente 3	
Descripción: Esta función exporta el reporte que se corresponde con el id seleccionado.	
Entrada	
Resultados esperados	No se exportará el reporte generado en el formato que fue seleccionado.
Condiciones	Se debe haber generado un reporte que no es de tipo jrxml.

Una vez ejecutados todos los casos de pruebas obtenidos a través de la aplicación de la técnica Ruta Básica, se concluye que los mismos fueron probados satisfactoriamente.

➤ **Pruebas de integración:**

Las pruebas de integración son una técnica sistemática para construir la arquitectura del software mientras se llevan a cabo pruebas para descubrir errores asociados con la interfaz. El objetivo es tomar los componentes probados de manera individual y construir una estructura de programa que se haya dictado por diseño (57). Consiste en validar las conexiones e integración entre dos o más componentes de software.

Para realizar las pruebas de integración se utilizó el método de integración ascendente, ya que permite comenzar la construcción y la prueba con módulos atómicos, es decir, pruebas unitarias. Puesto que los componentes se integran de abajo hacia arriba, la funcionalidad que proporcionan los componentes subordinados en determinado nivel siempre está disponible.

Cada vez que se agrega un nuevo componente como parte de las pruebas de integración, el software cambia. Se establecen nuevas rutas de flujo de datos, ocurren nuevas operaciones de entrada/salida y se invoca nueva lógica de control. Para corregir las nuevas fallas a causa de la integración de los nuevos componentes se realizan las pruebas de regresión, que constituyen la nueva ejecución de algún subconjunto de pruebas que ya se realizaron a fin de asegurar que los cambios no propagaron efectos colaterales no deseados (57).

Luego de realizadas las pruebas de integración, el módulo de reportes queda en el sistema como muestran las siguientes figuras:

<input type="checkbox"/>	Nombre	Tipo	Acciones
<input type="checkbox"/>	Acciones en la plataforma	Reporte del sistema	Previsualizar
<input type="checkbox"/>	Acciones de los participantes en la plataforma	Reporte del sistema	Previsualizar
<input type="checkbox"/>	Actividades de un curso	Reporte de aprendizaje	Previsualizar
<input type="checkbox"/>	Comportamiento de los estudiantes por actividad en un curso	Reporte de aprendizaje	Previsualizar
<input type="checkbox"/>	Copias de seguridad de un curso	Reporte de aprendizaje	Previsualizar
<input type="checkbox"/>	Cuestionarios por curso	Reporte de aprendizaje	Previsualizar
<input type="checkbox"/>	Datos del usuario	Reporte del sistema	Previsualizar
<input type="checkbox"/>	Estado de wiki de un curso	Reporte de aprendizaje	Previsualizar
<input type="checkbox"/>	Estudiantes activos por curso	Reporte de aprendizaje	Previsualizar

Figura 24. Elaboración propia. Vista para el administrador.

Mostrando registros del 1 al 10 de un total de 14 registros

Nombre	Acciones
Actividades de un curso	
Comportamiento de los estudiantes por actividad en un curso	
Copias de seguridad de un curso	
Cuestionarios por curso	
Estado de wiki de un curso	
Estudiantes activos por curso	
Glosario de términos en un curso	
Roles activos en un curso	
Roles inactivos en un curso	
Uso de foros en un curso	

Figura 25. Elaboración propia. Vista para el profesor.

➤ **Pruebas funcionales:**

Al integrar software y hardware, se lleva a cabo un amplio rango de pruebas funcionales con la intención de descubrir errores en la interfaz (57). Para realizar estas pruebas se hace una verificación dinámica del comportamiento de un sistema, basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba (62).

Para el desarrollo de las pruebas funcionales se utilizó el método de caja negra, el cual se lleva a cabo en la interfaz del software. Este método examina algunos aspectos fundamentales de un sistema con poca preocupación por la estructura lógica interna del software. Este método se divide en tres técnicas que se enfocan en el dominio de información del software, y derivan casos de prueba mediante la partición de los dominios de entrada y salida de un programa en forma que proporciona cobertura de prueba profunda, estos métodos son: partición de equivalencia, análisis de valor de fronteras y prueba de rutas (57).

De los métodos mencionados anteriormente, se utilizó la partición de equivalencia, ya que permite dividir el dominio de entrada en clases de datos que ejecutan una función de software específica. Al aplicar esta técnica se generaron casos de pruebas, que fueron

diseñados con la intención de descubrir errores en el procesamiento, antes de comenzar las pruebas con el cliente.

Algunos de los casos de prueba que fueron diseñados para aplicar el método de caja negra al sistema se presentan en la siguiente tabla. El resto de los artefactos de este tipo se encuentran en el directorio Casos de Prueba.

Tabla 10. Caso de prueba. Actividades de un curso

Escenario	Descripción	Filtro	Respuesta del sistema	Flujo central
		Cursos		
EC 1.1 Previsualizar reporte	El usuario selecciona el reporte a generar: - Actividades de un curso.	N/A	El sistema muestra la vista previa del reporte. Además permite seleccionar la opción Cancelar y la opción Previsualizar.	Curso/Reportes/ Previsualizar Actividades de un curso
EC 1.2 Opción Cancelar	El usuario selecciona la opción Cancelar.	N/A	El sistema regresa a la vista anterior.	Curso/Reportes/ Previsualizar Actividades de un curso/Cancelar
EC 1.3 Adicionar filtro	El usuario selecciona el filtro cursos.	V	El sistema muestra una lista desplegable con todos los cursos .	Curso/Reportes/ Previsualizar Actividades de un curso/ Adicionar filtro
EC 1.4 Seleccionar curso	El usuario selecciona el curso del cual desea generar el reporte.	V	El sistema permite previsualizar el reporte .	Curso/Reportes/ Previsualizar Actividades de un curso/Adicionar filtro
EC 1.5 Previsualizar	El usuario selecciona la opción Previsualizar.	N/A	El sistema genera el reporte.	Curso/Reportes/ Previsualizar Actividades de un curso/Previsualizar

EC 1.6 Opción Exportar	El usuario selecciona la opción Exportar.	N/A	El sistema exporta el reporte.	Curso/Reportes/ Previsualizar Actividades de un curso/ Previsualizar/Exportar
------------------------------	---	-----	--------------------------------	--

Tabla 11. Caso de prueba. Usuarios registrados en la plataforma

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Previsualizar reporte	El usuario selecciona el reporte a generar: - Usuarios registrados en la plataforma.	El sistema muestra la vista previa del reporte. Además permite seleccionar la opción Cancelar y la opción Previsualizar.	Administración/Reportes/ Todos los reportes/ Previsualizar Usuarios registrados en la plataforma
EC 1.2 Opción Cancelar	El usuario selecciona la opción Cancelar.	El sistema regresa a la vista anterior.	Administración/Reportes/ Todos los reportes/ Previsualizar Usuarios registrados en la plataforma/Cancelar
EC 1.5 Previsualizar	El usuario selecciona la opción Previsualizar.	El sistema genera el reporte.	Administración/Reportes/ Todos los reportes/ Previsualizar Usuarios registrados en la plataforma/Previsualizar
EC 1.6 Opción Exportar	El usuario selecciona la opción Exportar.	El sistema exporta el reporte.	Administración/Reportes/ Todos los reportes/ Previsualizar Usuarios registrados en la plataforma/ Previsualizar/ Exportar

Después de realizadas las pruebas funcionales fueron detectadas 8 No Conformidades (NC) en la primera iteración. Para darle solución a las mismas se realizaron las pruebas de regresión antes de pasar a la próxima iteración, quedando solucionadas 7 de las NC

detectadas. Posteriormente se procedió a hacer una segunda iteración y fueron encontradas 3 NC que quedaron corregidas luego de aplicarles pruebas de regresión a las mismas. Luego, en una tercera iteración, se no detectaron NC y fue corregida la NC pendiente. Finalmente, se realizó una cuarta iteración en la que no se detectaron NC.

Iteración 1

- **Funcionalidad:**

NC1. No se mostraban todos los tipos de actividades asociadas a un curso.

NC2. No se conocía el rol de los usuarios registrados.

NC3. No se conocía el nombre de los usuarios inactivos.

NC4. No se conocía el tipo de respaldo que se le realizan a los cursos.

NC5. No se conocía el total de usuarios registrados.

NC6. No se conocía la cantidad de usuarios activos e inactivos.

NC7. No se mostraba la cifra correcta de comentarios por foro.

NC8. El reporte de los estudiantes activos por cursos mostraba todos los roles registrados.

Iteración 2

- **Error de interfaz:**

NC9. No se visualizaba correctamente las letras ni la estructura del reporte.

- **Opción que no funciona:**

NC10. La opción previsualizar de la vista del profesor no funciona.

- **Ortografía:**

NC11. Omisiones de tildes y cambio de mayúscula por minúscula.

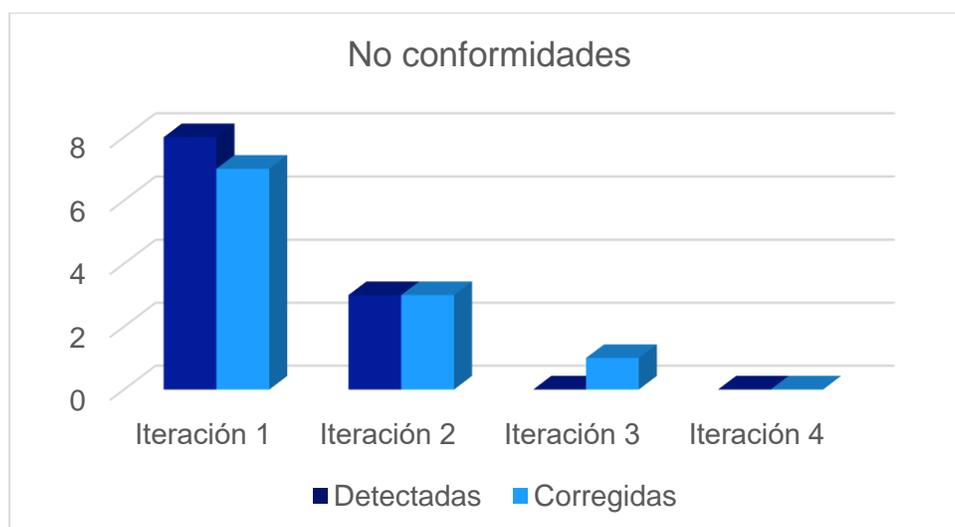


Figura 26. Elaboración propia. No conformidades.

Pruebas aceptación

Es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido (32). Estas pruebas son realizadas por el cliente, acompañado del equipo de desarrollo y se orientan a las funcionalidades del sistema.

Luego de realizadas las pruebas de aceptación, se generó el Acta de aceptación del producto como constancia de la conformidad por parte del cliente de la solución propuesta, dando de esta manera culminación al proceso de pruebas de software.

Conclusiones del capítulo

Después de realizado este capítulo se puede concluir que:

- Fue descrito el proceso de implementación del módulo de reportes para la Plataforma Educativa XAUCE ZERA a través los artefactos utilizados para llevar a cabo la misma; los diagramas de componentes y despliegue.
- Los estándares de codificación utilizados fueron los establecidos por el proyecto, para así obtener un estilo de programación homogéneo.
- Se presentó la estrategia de prueba tomada, que comprendió la realización de las pruebas unitarias, pruebas de integración, pruebas funcionales y pruebas de aceptación como métrica para erradicar los fallos y medir el correcto funcionamiento y eficiencia del producto.

Conclusiones generales

Al finalizar el presente trabajo de diploma, se cumple con el objetivo general planteado al inicio de la investigación, por lo que se puede arribar a las siguientes conclusiones:

- Con el estudio de plataformas educativas que gestionan su información a través de reportes, se demostró la importancia de contar un módulo de reportes en la Plataforma Educativa XAUCE ZERA, que facilite la gestión de la información y la toma de decisiones tanto a los profesores como a los administradores.
- La selección de las herramientas, tecnologías y la metodología de desarrollo de software AUP-UCI, permitió el desarrollo de un producto acorde con las necesidades establecidas por el cliente.
- Se diseñó una solución capaz de gestionar la información que se genera en la plataforma para la creación de reportes. El producto fue desarrollado haciendo uso de estándares y buenas prácticas de programación de acuerdo a lo descrito en el acápite 3.1.3 de la investigación, garantizando limpieza y organización en el código del sistema.
- Para comprobar la calidad del módulo de reportes desarrollado fueron realizadas las pruebas de software utilizando los métodos de caja blanca y caja negra, obteniéndose resultados satisfactorios tras la corrección de los errores existentes en cuatro iteraciones.

Recomendaciones

Una vez terminado el módulo de reportes para la Plataforma Educativa XAUCE ZERA y analizados los resultados obtenidos de la investigación, se mantiene la idea de continuar perfeccionando la solución final, por lo que se recomienda:

1. Añadir un reporte en el que se muestren los cursos más activos, para así saber cuáles resultan más interesantes.

Referencias bibliográficas

1. **Belloch, Consuelo.** *Las Tecnologías de la Información y Comunicación en el Aprendizaje.* Departamento de Métodos de Investigación , Universidad de Valencia. 2013.
2. **Kovanović V, Joksimović S, Gašević D, Siemens G, Hatala M.** *What public media reveals about MOOCs: A systematic analysis of news reports.* 2015. págs. 510–527.
3. **Ponce, Viviana Mercedes.** *Plataformas virtuales y herramientas informáticas evaluativas con sentido formativo: alcances y limitaciones.* 2013.
4. *E-learning Platforms in Higher Education. Case Study.* **D. Benta, G. Bologna, I. Dzitac.** 2014, Procedia Computer Science.
5. *Los desafíos de las TIC para el cambio educativo.* **Roberto Carneiro, Juan Carlos Toscano, Tamara Díaz.** 2014.
6. **Mincom, Dirección de Comunicación Institucional del.** Palabras del Ministro de Comunicaciones en la inauguración del 6to Congreso Latinoamericano de Telecomunicaciones. *República de Cuba Ministerio de Comunicaciones.* 15 de Febrero de 2019.
7. **UCI, Universidad de las Ciencias Informáticas.** La UCI de un vistazo. [En línea] [Citado el: 29 de Diciembre de 2018.] <https://www.uci.cu/universidad/la-uci-de-un-vistazo>.
8. **Informática, Universidad de las Ciencias.** Plataforma Educativa XAUCE ZERA. [En línea] [Citado el: 16 de Diciembre de 2018.] <https://eva.uci.cu/es/aboutAs>.
9. *Las plataformas en la educación en línea.* **Peñaloza, Alberto Domingo Robles.** 4, México : s.n., Diciembre de 2004, Revista Electrónica e-formadores.
10. *Implementación de una plataforma educativa en una institución de nivel medio superior como apoyo en las actividades docentes.* **Alma Isabel Arias Hurtado, Rosa Icela Gracia Nolasco, Cinthya Grisel Talamantes Molina.** 2, Enero de 2015, Revista Iberoamericana de Producción Académica y Gestión Educativa. 2007 - 8412.
11. *La gestión de la información: Herramienta esencial para el desarrollo de habilidades en la comunidad estudiantil universitaria.* **A. Suárez Alfonso, I. Cruz Rodríguez, Y. Pérez Macías.** 2, 2015, Universidad y Sociedad, Vol. 7, págs. 72-79. ISBN 2218-3620.
12. **Quiroga, Lourdes Aja.** *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.* La Habana : ACIMED, 2002. Vol. 10. ISBN 1024-9435.
13. **José Alejandro Bidot Hernández, Fabiel León Oliva.** *Extensión al iReport para la gestión de reportes en el Servidor Dinámico de Reportes.* Universidad de las Ciencias Informáticas. 2015. Tesis de diploma.

14. **Julián Pérez Porto, María Merino.** Definición de reporte. [En línea] 2013. [Citado el: 16 de Noviembre de 2018.] <https://definicion.mx/reporte/>.
15. **Bouly, Yanet Ennis.** *Componentes para la Generación de Reportes Dinámicos en el GDR sobre Bases de Datos en Access y Oracle 11g.* 2015.
16. **Anayda Fernández Naranjo, Miguel Rivero López.** *Las plataformas de aprendizajes, una alternativa a tener en cuenta en el proceso enseñanza aprendizaje.* UNIVERSIDAD DE ALCALÁ. s.l. : Revista Cubana de Informática Médica, 2014.
17. *La importancia del uso de plataformas educativas.* **Viñas, Mariela.** 6, s.l. : Centro de Investigación en Lectura y Escritura (CILE), Junio de 2017, Letras, págs. 157-169.
18. *Análisis Comparativo de las Plataformas Educativas Virtuales Moodle y Dokeos.* **Rogelio Estrada Lizárraga, Aníbal Zaldívar Colado, Juan Francisco Peraza Garzón.** 10, Enero-Junio de 2013, Revista Iberoamericana para la Investigación y el Desarrollo Educativo.
19. **Lina Guzman Medina, Pablo Arturo Alvarez, Raul Alberto Celis, Eduardo Romero.** *Selección de la herramienta Moodle vs. Sakai LMS.* UNIVERSIDAD NACIONAL DE COLOMBIA – INVIMA. 2014. INFORME DE ESCOGENCIA DE PLATAFORMA LMS.
20. **Paulo Alves, Luís a Miranda, Carlos Morais, Evandro Alves.** *Proposal of a Learning Styles Identification Tool for Sakai Virtual Learning Environment.* 2013.
21. **Aberdour, Mark.** *Open Source Learning Management Systems.* 2013.
22. **Elene Raquel Garcés Arguello, Cecilia Jenny Rivera Enriquez.** *Evaluación de plataformas tecnológicas para la teleformación o e-learning para el ámbito universitario, tomando como caso de estudio e-educativa.* Departamento de Ciencias de la Computación, Escuela Politécnica del ejército. 2010. Trabajo de diploma.
23. **Salvador, Patricia Gissela Pereyra.** *Uso de la plataforma virtual para el aprendizaje del curso pensamiento orrego en los estudiantes de ingeniería del programa carreras para gente que trabaja en la Universidad Privada Antenor Orrego en el semestre académico 2015-39.* 2017. Tesis.
24. **Cervantes, Ing. Beatriz Hernández.** *Servidor Dinámico de Reportes V2.0.* Universidad de las Ciencias Informáticas. 2016. Manual de Usuario.
25. *GeReport: Sistema de Gestión de Reportes Dinámicos.* **Rafael Felipe Bomate Gavio, Yenía Román Bu, Cinthya Rodríguez Hernández, Carlos Manuel DelgadoRivero, Manuel Cortés Cortés.** 4, s.l. : Ediciones Futuro, 2014, Revista Cubana de Ciencias Informáticas, Vol. 8, págs. 116-129.

26. **JasperReports.** JasperReports. [En línea] Copyright © 2018. [Citado el: 16 de Noviembre de 2018.] <http://community.jaspersoft.com/project/jasperreports-server>.
27. **Reports, Crystal.** Crystal Reports. [En línea] © 2018. [Citado el: 17 de Noviembre de 2018.] <https://www.crystalreports.com/>.
28. **Mariaca, Ricardo.** DynamicReports. [En línea] © 2018. <http://www.dynamicreports.org/>.
29. **Jaspersoft.** DynamicReports. [En línea] @2018. [Citado el: 26 de Noviembre de 2018.] <https://community.jaspersoft.com/project/dynamicreports>.
30. **Yanoski Agneri Martínez Hernández, José Rolando Lafaurie Olivares.** *Sistema para la generación de reportes en la plataforma alasGRATO*. 2009.
31. **DBxtra.** DBxtra Diseñador de Reportes. [En línea] © 2006 - 2018. [Citado el: 26 de Noviembre de 2018.] <http://www.dbxtra.com/es/reportes-disenador.htm>.
32. **Sánchez, Tamara Rodríguez.** *Metodología de desarrollo para la para la Actividad productiva de la UCI*. 2015.
33. **Falcón, Ing. Yuniel Suárez.** DeProgramación. *Qué es un IDE?* [En línea] 2015. [Citado el: 26 de Noviembre de 2018.] <http://deprogramacion.cubava.cu/2016/02/01/que-es-un-ide/>.
34. **Corporation, Oracle.** Apache NetBeans. [En línea] 2015. [Citado el: 26 de Diciembre de 2018.] <https://netbeans.org/>.
35. **Foundation, The Apache Software.** Apache NetBeans. [En línea] 2017. [Citado el: 26 de Diciembre de 2018.]
36. **PostgreSQL.** PostgreSQL. [En línea] 2016. [Citado el: 23 de Noviembre de 2018.] <http://www.postgresql.org.es>.
37. **PgAdmin.** PgAdmin. [En línea] 2016. [Citado el: 23 de Noviembre de 2018.] <http://www.pgadmin.org>.
38. **Morales, Ricardo.** Colombia Digital. [En línea] 1 de Septiembre de 2014. [Citado el: 26 de Noviembre de 2018.] <https://colombiadigital.net/actualidad/articulos-informativos/item/7669-lenguajes-de-programacion-que-son-y-para-que-sirven.html>.
39. **PHP.** php. [En línea] Copyright © 2001-2019. [Citado el: 26 de Noviembre de 2018.] <http://php.net/manual/es/intro-what-is.php>.
40. **BBVAOpen4U.** ¿Desarrollador en PHP? Prepárate para la revolución de PHP 7. [En línea] 15 de Febrero de 2016. [Citado el: 26 de Noviembre de 2018.] <https://bbvaopen4u.com/es/actualidad/desarrollador-en-php-preparate-para-la-revolucion-de-php-7>.
41. **Pergamini.** Pergamini. [En línea] ©1998-2018. [Citado el: 23 de Noviembre de 2018.] <http://www.pergaminovirtual.com.ar/definicion/JavaScript.html>.

42. **Gauchat, J uan Diego.** *El gran libro de HTML5, CSS3 y Javascript.* 2012.
43. **w3c.** w3c. [En línea] [Citado el: 26 de Noviembre de 2018.] <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>.
44. **Soft112.** Visual Paradigm for UML Standard 8.0- Free Download. [En línea] © Copyright 2018. [Citado el: 26 de Noviembre de 2018.] <http://visual-paradigm-for-uml-standard.soft112.com/>.
45. **Paradigm, Visual.** Visual Paradigm. [En línea] 2016. [Citado el: 26 de Noviembre de 2018.] <https://online.visual-paradigm.com/es/>.
46. **Lucid.** Qué es el lenguaje unificado de modelado (UML). [En línea] © 2018. [Citado el: 26 de Noviembre de 2018.] <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>.
47. **Alvaréz, Miguel Angel.** *Manual de jquery.* 2013.
48. **Bootstrap.** [En línea] Copyright © 2018. [Citado el: 26 de Noviembre de 2018.] <http://getbootstrap.com>.
49. **Cuesta, Diego García.** ¿Qué es Nginx? Características y diferencias con Apache. [En línea] 16 de Marzo de 2018. [Citado el: 26 de Noviembre de 2018.] <https://linuxenespañol.com/noticias/que-es-nginx-caracteristicas/>.
50. **Quintana, Roylan Tumbarell.** *Componente sobre el marco de trabajo Xalix, para el trabajo colaborativo en la creación de recursos educativos.* Universidad de las Ciencias Informáticas. 2016. Trabajo de Diploma.
51. **Sommerville, Ian.** *Ingeniería del software.* séptima. s.l. : Pearson Educación, 2005.
52. **Cohn, Mike.** *User Stories Applied for Agile Software Development.* 2009.
53. **José H. Canós, Patricio Letelier, María Carmen Penadés.** *Métodologías ágiles en el Desarrollo de Software.* Universidad Politécnica de Valencia. 2013.
54. **Reynoso, Carlos Billy.** *Introducción a la Arquitectura de Software.* s.l. : Universidad de Buenos Aires, 2004. pág. 11.
55. **Adriana Sandra Almeida, Vanina Perez Cavenago.** *Arquitectura de Software: Estilos y Patronos.* Universidad Nacional de la Patagonia San Juan Bosco. 2007. Tesina.
56. **Fabien Potencier, François Zaninotto.** *Symfony 1.4, la guía definitiva.* [En línea] © 2006-2018. [Citado el: 26 de Noviembre de 2018.] <https://uniwebsidad.com/libros/symfony-1-4?from=librosweb>.
57. **Chu, Sijing Xi y Bei.** *Herramienta de Optimización y Filtrado de Búsqueda de Internet.* Universidad Politécnica de Madrid. 2016. Tesis Máster.

58. **Pressman, Roger S.** *Ingeniería del software. Un enfoque práctico.* Séptima. Nueva York : McGraw-Hill, 2010. pág. 191.
59. **Larman, Craig.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos y al proceso unificado.* Segunda. Madrid : Pearson Educación, S.A, 2003.
60. **Carlos A. Guerrero, Johanna M. Suárez, Luz E. Gutiérrez.** *Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web.* Grupo de Investigación de Ingeniería del Software. Santander, Colombia : s.n., 2013. pág. 2, Investigación.
61. **Grady Booch, Ivar Jacobson, James Rumbaugh.** *El lenguaje unificado de modelado: manual de referencia.* Primera. Madrid : Pearson Educación, 2000.
62. **López, Nancy.** *Estándares de codificación Java. Ingeniería de Software.* 2013.
63. *Método para generar casos de prueba funcional en el desarrollo de software.* **Palacio, Liliana González.** 15, Medellín, Colombia : s.n., Diciembre de 2009, Revista Ingenierías Universidad de Medellín, Vol. 8, págs. 29-36. 1692 -3324.
64. **Cantón, Alejandro Castillo.** *Manual de HTML5 en español.* 2015.
65. **Dayana Mendoza Peña, Lionel Rodolfo Baquero Hernández.** *Extensión de la herramienta Visual Paradigm for UML para la evaluación y corrección de Diagramas de Casos de Uso.* 2016. Trabajo de Diploma.
66. **Formación, Centro de Tecnologías para la.** Plataforma Educativa XAUCE ZERA. [En línea] [Citado el: 29 de Noviembre de 2018.] <https://eva.uci.cu/es/aboutAs>.
67. **Jasper, Dynamic.** Dynamic Jasper. [En línea] Copyright © 2018. [Citado el: 29 de Noveiembre de 2018.] <http://dynamicjasper.com/features/>.
68. **JasperReports.** JasperReports Library. [En línea] Copyright © 2018. [Citado el: 16 de Noviembre de 2018.] <https://community.jaspersoft.com/project/jasperreports-library>.
69. **Carhuavilca, Hermes Yesser Pantoja.** *Plataforma virtual para el desarrollo de asignaturas usando el enfoque constructivista.* s.l. : Universidad Nacional de Ingeniería, 2013. Maestría.
70. **Sánchez, Daniela Moreno.** *Componente para la gestión y evaluación de actividades grupales en la Plataforma Educativa XAUCE ZERA 2.1.* Universidad de las Ciencias Informáticas. 2018. Tesis de diploma.
71. **Pressman, Roger S.** *Ingeniería de Software. Un enfoque práctico.* Séptima . México DF : McGraw-Hill INTERAMERICA EDITORES, 2010.
72. **Aretio, Lorenzo García.** *Wiki en contextos educativos.* s.l. : BENED, 2006.

Anexos

Entrevista al cliente Miguel Medina Ramírez para el levantamiento de requisitos.

1. A partir de la encuesta realizada por el CENED, ¿cuáles fueron los principales problemas detectados en la plataforma?
2. ¿Cómo es el funcionamiento de la plataforma para los roles de administrador y profesor?
3. ¿Con qué información almacenada en la base de datos interactúan estos roles?
4. ¿Qué información les sería útil consultar en la plataforma para resolver los problemas detectados en la encuesta realizada por el CENED?
5. ¿Qué otra información les sería útil a profesores y administradores para apoyarlos en la toma de decisiones?