

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 1



**Universidad de las Ciencias
Informáticas**

**“Aplicación informática para la gestión de información de la
Práctica Profesional en el Centro de Software Libre”**

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor:

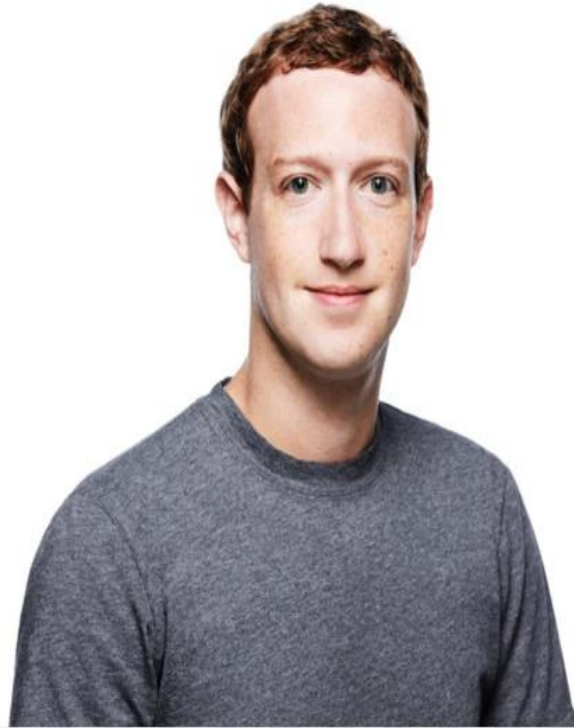
Roberto Javier Ramírez Carreño

Tutor:

Msc. Aneyty Martín García

Ing. Ivaniet Díaz Romeu

LA HABANA, DICIEMBRE 2022



"Los éxitos más importantes se consiguen cuando existe la posibilidad de fracasar"

Mark Zuckerberg

Agradecimientos

Al comenzar a escribir los agradecimientos me di cuenta que son muchas personas que a lo largo de cinco años me han apoyado y ayudado en este camino que no fue fácil de una carrera que desde niño quise ejercer porque era y soy amantes de lo relacionado a la informática. Tuve buenos y malos momentos que gracias a ustedes pude levantarme, mirar adelante y seguir hasta mi meta que era ser ingeniero en ciencias informáticas.

Por eso hoy quiero agradecer a toda mi familia que se resume en mi mamá Nelcy, mi papá Fidel, mis abuelos Elia y Migdalia, mis tías Nancy y Amada, que siempre estuvieron pendiente de mí y fueron el sustento para que yo pudiera lograr mi sueño de convertirme de profesional. Gracias por sus consejos y su apoyo a cada una de mis decisiones.

A mi novia Sandra por estar siempre en las buenas y en las malas, por su paciencia con cada uno de mis berrinches y alentándome a ser mejor persona de la que soy, por darme su amor sin nada a cambio, confianza y lealtad.

A mis tutores Ivaniety y Aneyty por las horas que me dedicaron sin importar el momento ni el día; siempre con sus opiniones y sugerencias bien recibidas y acertadas.

A todos y cada uno de mis profesores durante estos cinco años por sus enseñanzas.

A mis compañeros de aula por la gran amistad que se ha formado entre nosotros en estos cinco años y por esa ayuda incondicional cuando tenía alguna duda en cierta asignatura. Por las bromas y los momentos que pasamos juntos en la convivencia de la residencia.

A mi primo Adrián que desde la distancia siempre estuvo al pendiente de mi desarrollo e incitándome a estudiar para que el día de mañana fuera un mejor profesional, por estar ahí cuando necesitaba de él y darme su apoyo incondicional.

A mis suegros Leticia y Rodolfo por abrirme las puertas de su casa y hacerme sentir en casa, por tratarme como su hijo, eternamente agradecido.

A mi primo Reinier, Yeni, mi primo Angelito, mi tío Alexis gracias por su apoyo en todo momento.

A Alvaro por convertirse como mi hermano y estar también disponible para dar su mano sin nada a cambio.

Agradecimientos

A mis abuelos Pepe y Roberto a los cuales extraño con toda mi alma y hoy no se encuentran físicamente conmigo por todo su amor, consejos y dedicación hacia mí, a ustedes va esta tesis y por ustedes pongo empeño para ser una mejor persona como me enseñaron.

En general a todas las personas con las que he convivido y conocido a lo largo de esta carrera y a las personas que preguntan por mi estado en mi municipio Quemado de Güines.

A todos y cada uno de ustedes, muchas gracias, simple y sencillamente por existir.

Gracias,

Roberto Javier

Dedicatoria

A mi mamá, la más preocupada de toda la madre;

a mi papá, el mejor de todos los padres;

a mis abuelos, los mejores abuelos del mundo;

a mis tías Amada Y Nancy, las mejores tías del mundo;

a mi novia, lo mejor que me ha pasado en la vida;

a mis primos Adrián y Reinier, gracias por estar siempre disponible para mí;

y a mis tutores, los más dedicados de todos los tutores;

DECLARACIÓN DE AUTORÍA

Declaro por este medio que yo **Roberto Javier Ramírez Carreño**, con carné de identidad 97080111349, soy el autor principal del trabajo titulado “**Aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre**” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste firman la presente a los días del mes del 2022.

Roberto Javier Ramírez Carreño

Autor

Msc. Aneyty Martín García

Tutor

Ing. Ivaniel Díaz Romeu Prof. Instructor

Declaración de autoría

Tutor

RESUMEN

El presente documento propone una aplicación informática para la gestión de la información de las prácticas profesionales en el centro de software libre, la cual se fundamenta mediante una investigación del funcionamiento actual de las prácticas profesionales y su problemática en la gestión de la información de la asignatura en el Centro de Software Libre. La solución consiste en un sistema informático que gestione los planes de formación y las tareas de cada estudiante, así como los profesores del centro, Supervisores-Evaluadores y Tutores, y Responsables de PID, brindando un control sobre la información en el centro. El proceso de desarrollo estuvo guiado por la metodología de desarrollo AUP en su versión UCI donde se utiliza el escenario cuatro. Se selecciona como principales tecnologías para la implementación el lenguaje de programación PHP 8.0 y su marco de trabajo correspondiente Laravel en su versión 9.2, para diseñar los prototipos de usuario se utiliza HTML5, CSS3 y su framework Bootstrap 5.2. Con la finalización del sistema se somete a un número de pruebas la cual sentencia la calidad del software, se utilizan pruebas como funcionales, unitarias y de seguridad, las cuales arrojan resultados positivos sobre el rendimiento y funcionalidad de la aplicación informática.

Palabras clave: aplicación informática, gestión, prácticas profesionales

ABSTRAT

This document proposes a computer application for the information management of the professional practices in the free software center, which is based on an investigation of the current operation of the PP and its problems in the management of the information of the subject in the CESOL development center. The solution consists of a computer system that manages the training plans and tasks of each PID student, as well as the PID, SET and PID Head teachers, providing control over the information in the center. The development process was guided by the AUP development methodology in its UCI version using scenario four. The programming language PHP 8.0 and its corresponding work arc Laravel in its version 9.2 are selected as the main technologies for the implementation, HTML5, CSS3 and its Bootstrap 5.2 framework are used to design the user prototypes. With the completion of the system, it is subjected to a number of tests which sentence the quality of the software, tests such as functional, unitary and security are used, which yield positive results on the performance and functionality of the computer application.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: Fundamentación teórica sobre el proceso de gestión de información de la Práctica Profesional en el Centro de Software Libre	7
1.1 Definición de conceptos	7
1.2 Descripción del mecanismo de gestión de la información de la Práctica Profesional en CESOL.....	8
1.3 Caracterización de aplicaciones informáticas para la gestión de información de la Práctica Profesional	10
1.4 Análisis comparativo de aplicaciones informáticas para la gestión de información de la Práctica Profesional	13
1.5 Metodología de desarrollo de software.....	16
1.5.1 Variación de AUP para la UCI	17
1.6 Lenguajes y herramientas para el modelado de la solución	17
1.6.1 Lenguaje de modelado de software UML	18
1.6.2 Herramienta para el modelado de la solución	18
1.7 Lenguajes de programación	18
1.7.1 Lenguaje de marcado de hipertextos.....	19
1.7.2 Hojas de estilos.....	19
1.7.3 Marcos de trabajo	20
1.7.4 Entorno de Desarrollo Integrado	21
1.7.5 Servidor de aplicaciones web.....	21
1.7.6 Servidor de bases de datos	22
1.8 Herramienta de validación	22

1.9 Conclusiones del capítulo.....	23
CAPÍTULO 2: Análisis y diseño de una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre	24
2.1 Descripción del contexto de dominio de la propuesta de solución desarrollada.....	24
2.1.1 Modelo conceptual	24
2.2 Requisitos.....	26
2.2.1 Fuentes para la obtención de requisitos.....	26
2.2.2 Técnicas de Identificación de Requisitos.....	27
2.2.3 Especificación de requisitos de software.....	28
2.2.4 Descripción de requisitos de software mediante Historias de Usuario	34
2.2.5 Validación de requisitos de software.....	36
2.3 Análisis y diseño	40
2.3.1 Diseño de clases	40
2.3.2 Diagrama de despliegue.....	43
2.3.3 Patrones de diseño de software	44
2.3.4 Modelado de datos	47
2.3.5 Diseño arquitectónico	48
CAPÍTULO 3: Implementación y evaluación de la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre.....	49
3.1 Introducción	50
3.2 Diagrama de Componentes.....	50
3.3 Estándares de codificación.....	53
3.4 Validación de la propuesta de solución.....	54
3.4.1 Pruebas Funcionales.....	55
3.4.2 Pruebas de Seguridad.....	59

3.4.3 Pruebas Unitarias	60
3.4.4 Pruebas de aceptación	¡Error! Marcador no definido.
CONCLUSIONES GENERALES	67
RECOMENDACIONES	68
REFERENCIAS BIBLIOGRÁFICAS	69
ANEXOS	75
Anexo 1: Lineamientos de la Política Económica y Social de Cuba	75
Anexo 2: Entrevista realizada al Responsable de la Práctica Profesional en CESOL	76
Anexo 3: Encuesta realizada a 15 Supervisores – Evaluadores - Tutores en CESOL	77
Anexo 4. Estructura del plan de formación del estudiante.....	78
Anexo 5. Guía de observación para analizar cómo se gestiona la información de la Práctica Profesional en aplicaciones informáticas.....	80
Anexo 6. Diagrama de clases del diseño con estereotipos web para la agrupación de requisitos Gestionar plan de formación.....	¡Error! Marcador no definido.

ÍNDICE DE FIGURAS

Figura 1. Modelo Conceptual25

Figura 2. Prototipo de interfaz de usuario correspondiente al RF20. Insertar Plan de Formación35

Figura 3. Prototipo de interfaz de usuario correspondiente al RF22. Eliminar Plan de Formación.....**¡Error!**
Marcador no definido.

Figura 4. Casos de prueba correspondiente al RF20. Insertar Plan de Formación38

Figura 5. Diagrama de Clases42

Figura 6. Diagrama de Despliegue.....43

Figura 7. Patrones de diseño GRASP45

Figura 8. Patrones de diseño GRASP45

Figura 9. Modelado de datos.47

Figura 10. Patrón arquitectónico Modelo-Vista-Controlador49

Figura 11. Diagrama de Componente Tareas.....50

Figura 12. Diagrama de Componente Plan de Formación.52

Figura 13. Caso de Prueba Insertar Estudiante.....56

Figura 14. Caso de Prueba Insertar Profesor.57

Figura 15. Resultado de las pruebas funcionales.....59

Figura 16. Resultado de las pruebas de seguridad.....60

Figura 17. Método store () correspondiente al RF20. Insertar Plan de Formación.61

Figura 18. Grafo de flujo.....62

ÍNDICE DE TABLAS

Tabla 1. Comparación de aplicaciones informáticas para la gestión de información de la Práctica Profesional de estudiantes.....15

Tabla 2: Especificación de requisitos funcionales de la propuesta de solución.....29

Tabla 3. Especificación de requisitos no funcionales de la propuesta de solución.....32

Tabla 4. Historia de usuario del RF 20: Insertar plan de formación.....34

Tabla 5. Descripción de las variables.....38

Tabla 6. Descripción de Diagrama de componentes (Tareas).....51

Tabla 7. Descripción de Diagrama de componentes (Gestionar Plan de Formación).....52

Tabla 8. Validación de la propuesta de solución.....54

Tabla 9. Descripción de las variables. Caso de Prueba Insertar Estudiante.....56

Tabla 10. Descripción de las variables. Caso de Prueba Insertar Profesor.57

Tabla 11. Caso de prueba para el camino 1 de la función store ().....62

Tabla 12. Cuadro lógico de ladov64

Tabla 13. Resultados de la escala de satisfacción65

INTRODUCCIÓN

Las prácticas profesionales (PP) son la prestación de actividades temporales, obligatorias, supervisadas y de carácter formativo mediante la cual el alumno desarrolla, ejerce y perfecciona sus competencias profesionales, vinculándose con los diferentes sectores a nivel regional, nacional e internacional (IESPOH, 2020). Suele constituirse como el primer paso de un estudiante o de un recién graduado en el mercado laboral. Se trata de una etapa que combina cuestiones típicas de un empleo con elementos más vinculados a la formación y al aprendizaje (Salvador Santillan, 2019).

Funcionan en beneficio de las instituciones académicas para convertirse en una base de su prestigio, al acercarse a la realidad social objetiva y hacer partícipes a sus estudiantes en los procesos de desarrollo socioeconómico de sus territorios. Dentro de este escenario existe un vínculo estrecho entre universidad y empresa, vía fundamental para la formación profesional del estudiante. Hace posible la unión de la teoría que recibe el estudiante en la docencia con la Práctica Profesionales que realiza en la empresa (Alonso Betancourt, 2020).

En el contexto de la globalización, las Prácticas Profesionales son elementos integradores de este sistema. Constituyen uno de los motores de trabajo que promueven la integración económica entre países mediante el intercambio no sólo de recursos económicos sino también de talento humano. Son un pilar intelectual que contribuye a que este fenómeno siga ocurriendo, ya que globaliza el conocimiento y lo pone en juego con los otros elementos que conforman la globalización. Además, aumenta el volumen de comercialización de bienes y servicios con calidad internacional debido a la interacción de distintos actores (aacm92, 2019).

La Universidad Cubana y su Sistema Educativo han sido un ejemplo para los países de América Latina y el mundo. La vinculación de la práctica con la teoría en todas las carreras universitarias ha significado la construcción de un país con un alto grado de desarrollo profesional. En el lineamiento 115 de la Política Económica y Social de Cuba (Anexo 1) se evidencia la necesidad de la interacción universidad – empresa y la Práctica Profesional, ya que promueve la interacción entre el sector empresarial y el académico, con el objetivo de desarrollar técnicas, conocimientos al estudiante participante de la práctica (Asamblea Nacional del Poder Popular, 2016).

En la Universidad de las Ciencias Informáticas (UCI), las PP se desarrollan en los laboratorios de los centros productivos e investigativos y está dirigida al proceso de desarrollo de software. En el

Centro de Software Libre (CESOL) de la UCI, contexto en el que se ejecuta la presente investigación, su autor realiza una entrevista (Anexo 2) al Responsable de la PP y una encuesta a 15 Supervisores - Evaluadores - Tutores (SET), de ellos 10 especialistas (5 con categorías docente) y 5 Recién Graduados en Adiestramiento (RGA). La aplicación de estos métodos permitió conocer que los estudiantes de tercero, cuarto y quinto años de la carrera Ingeniería en Ciencias Informáticas, son los que se vinculan a CESOL para desempeñar las PP. En esta entidad realizan actividades según los roles asignados: Analista, Programador y Probador. Cada estudiante es atendido por un SET y el Responsable de la PP, quienes, son los encargados de elaborar un Plan de formación (Anexo 4) a cada educando, para la orientación, supervisión y control de las tareas que se le asignan. Este proceso tiene como objetivo formar habilidades y conocimientos en el estudiante y prepararlo para su vida laboral.

Debido a los cambios tecnológicos y la necesidad de mejorar, controlar e integrar la información de las prácticas profesionales y en general los procesos académicos, se vio la necesidad de desarrollar sistemas informáticos que gestionen el proceso educativo en las instituciones docentes en Cuba y el mundo. Estas aplicaciones brindan mecanismos de control en beneficio de los egresados, docentes y administrativos; logrando agilidad, oportunidad, seguridad y calidad en la información. Permiten llevar registro y seguimiento al ciclo de vida de cada estudiante en su vida universitaria (caseware, 2022).

En el Lineamiento 122 de la Política Económica y Social de Cuba para el periodo 2016-2021 (Anexo 1) se evidencia la necesidad de informatizar el sistema educativo en Cuba para mejorar la gestión de la información del proceso académico en las universidades (Asamblea Nacional del Poder Popular, 2016). En la actualidad se ha logrado grandes avances con respecto a las directrices planteadas por parte de la dirección del país, cada institución universitaria en el territorio nacional, cuenta con un Sistema de Gestión Académica para gestionar las actividades docentes de cada estudiante y profesores (Costa, 2007).

En la Universidad de las Ciencias Informáticas (UCI) se emplea el sistema informático Xauce Akademos para la gestión académica. El sistema fue desarrollado con el objetivo de permitir la gestión automatizada de los elementos que intervienen en la labor docente, mitigar los riesgos que impone el dinamismo del proceso de gestión académica, enfrentando de forma natural los cambios requeridos, adaptándose a las nuevas condiciones y brindar un papel activo a todos los involucrados en el proceso, dígame profesores, estudiantes, directivos y personal de secretaría.

Cuenta con un módulo de plan de estudio, matrícula, expediente, reportes y estudiantes, llevando el control de todas las actividades que se realizan en el sistema (Costa, 2007).

Sin embargo, según los resultados de la entrevista (Anexo 2) y las encuestas (Anexo 3) aplicadas, en CESOL la gestión de la información de la Práctica Profesional, se realiza de forma manual con la ayuda de diferentes herramientas informáticas como: Microsoft Word, Libre Office Writer; Microsoft Excel, Libre Office Calc, la creación de archivos en formato PDF (*Portable Document Format*, Formato de Documento Portátil) y el correo electrónico; ya que Xauce Akademos no dispone de las funcionalidades necesarias para la informatización de las actividades que realizan los estudiantes en la PP, además el Responsable de la PP y los SET de esta área, no disponen de los permisos necesarios en este sistema para el registro de la asistencia y las evaluaciones de los educandos.

El 100% de los encuestados y el entrevistado plantean que esta gestión ineficiente trae como consecuencias: pérdida o duplicidad de la información y retrasos en el proceso de registro de los datos, ya que existen varios documentos con diferentes formatos para estos propósitos; además de demoras en la realización de los reportes solicitados como consecuencias de la dispersión de la información resultante de la ejecución de la Práctica Profesional.

Tales insuficiencias justifican la necesidad de plantear el siguiente **Problema científico**: ¿Cómo contribuir a la gestión de la información de la Práctica Profesional en el Centro de Software Libre?

En relación con el problema científico, se formula el siguiente **Objetivo general**: Desarrollar una aplicación informática para la gestión de la información de la Práctica Profesional en el Centro de Software Libre.

Objetivos específicos:

1. Elaborar el marco teórico referencial de la investigación sobre la gestión de información de la Práctica Profesional.
2. Diseñar una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre.
3. Implementar una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre.

4. Evaluar la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre.

Se establece como **Objeto de estudio** la gestión de información de la Práctica Profesional.

El **Campo de acción** queda enmarcado en la gestión de información de la Práctica Profesional en el Centro de Software Libre.

Preguntas científicas

1. ¿Cuáles son los referentes teóricos fundamentales que sustentan la investigación relacionados con el proceso de gestión de la información de las prácticas profesionales en Centro de Software Libre?
2. ¿Qué elementos se deben tener en cuenta en el diseño de una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre?
3. ¿Qué componentes de software se deben implementar para obtener una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre?
4. ¿Cuáles pruebas de software, métodos y técnicas se deben aplicar para evaluar la aplicación informática que gestione la información de la Práctica Profesional en el Centro de Software Libre?

En el desarrollo de la investigación se emplearon los siguientes **Métodos de la investigación científica**:

Métodos del nivel teórico:

- **Análisis-síntesis:** Para procesar la información obtenida de los métodos utilizados en esta investigación. Además, permitió analizar la situación actual relacionada con la gestión de la información de la Práctica Profesional en el centro de desarrollo Centro de Software Libre.
- **Modelación:** Para el diseño de la aplicación informática que, como aporte, se ofrece en esta investigación.

Métodos del nivel empírico:

- **Entrevista:** Se realizó al Responsable de la PP (Anexo 2) para conocer las características e importancia que posee el desarrollo de la Práctica Profesional en los estudiantes de la UCI y CESOL, así como obtener información sobre la gestión de la información obtenida de

la ejecución de la Práctica Profesional en el Centro de Software Libre en cuanto a: importancia y características de su desarrollo, los objetivos que persigue, los años académicos que se vinculan al centro, las personas que interactúan y sus funciones; las actividades que realizan los estudiantes, su planificación y evaluación; los roles del proceso productivo de software que estos desempeñan; su vinculación a los proyectos de investigación y desarrollo del centro y los documentos que se elaboran para el cumplimiento de la PP. Permitió además, conocer las características del contexto del negocio informatizado y requisitos funcionales y no funcionales.

- **Encuesta:** Se aplicó a 15 Supervisores, Evaluadores o Tutores (SET) del Centro de Software Libre (Anexo 3), para conocer la necesidad del desarrollo de la Práctica Profesional en el proceso de formación de los estudiantes, las actividades que realizan los educandos, su planificación y evaluación, las características y deficiencias de la gestión de la información que se obtiene de la realización de la PP. Se obtuvo información sobre el dominio de la propuesta de solución, así como la obtención de funcionalidades y características incluidas en la aplicación.
- **Observación:** se empleó una guía de observación (Anexo 6) en la búsqueda de aplicaciones informáticas que gestionan información sobre la Práctica Profesional para conocer cómo realizan esta actividad e identificar las funciones y características que tienen implementadas.

La presente investigación está estructurada en: resumen, introducción, tres capítulos, conclusiones, referencias bibliográficas y anexos. A continuación, se presenta una breve descripción de los capítulos:

Capítulo 1. Fundamentación teórica sobre proceso de gestión de información de la Práctica Profesional en el Centro de Software Libre: En este capítulo queda reflejada la fundamentación teórica de la investigación sobre el proceso de gestión de información de la Práctica Profesional en el Centro de Software Libre. Contiene un estudio de sistemas homólogos; además se definen la metodología de desarrollo de software, y los lenguajes y las herramientas para la modelación e implementación de una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre.

Capítulo 2. Análisis y diseño una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre: El capítulo presenta las diferentes tareas desarrolladas y los productos de trabajo obtenidos en el análisis y diseño de una aplicación

informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre. Se muestra el modelo conceptual elaborado para comprender el contexto del negocio informatizado. Se especifican los requisitos de software y se muestra el diseño de la propuesta de solución según lo establecido por la metodología de desarrollo de software Variación de AUP para la UCI.

Capítulo 3. Implementación y evaluación de la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre: En este capítulo se muestran las tareas y los resultados obtenidos de la implementación, y la evaluación de la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre. Contiene los estándares de codificación empleados, así como el diagrama de despliegue elaborado. Se presentan las pruebas de software aplicadas y la evaluación del índice de satisfacción grupal con respecto a la propuesta de solución.

CAPÍTULO 1: Fundamentación teórica sobre el proceso de gestión de información de la Práctica Profesional en el Centro de Software Libre

Introducción

El presente capítulo aborda aspectos sobre la fundamentación teórica de la investigación. Contiene un estudio sobre la gestión de la información de la Práctica Profesional y un análisis de soluciones informáticas existentes para lograr su realización. Se define la metodología de desarrollo de software, así como los lenguajes y herramientas empleados en el modelado e implementación del sistema informático para la gestión de la información de la Práctica Profesional en CESOL.

1.1 Definición de conceptos

Aplicaciones Informáticas: Son un tipo de software que permite al usuario realizar uno o más tipos de trabajo. Permiten la interacción entre usuario y computadora, dando opción a elegir opciones y ejecutar acciones. Existen innumerable cantidad de tipos de aplicaciones como los procesadores de texto y las hojas de cálculo, mientras que los sistemas operativos o los programas de utilidades que cumplen tareas de mantenimiento no forman parte de estos. Pueden haber sido desarrolladas a medida para satisfacer las necesidades específicas de un usuario o formar parte de un paquete integrado como el caso de Microsoft Office (Benítez Jiménez, 2012).

Gestión Académica: Es el conjunto de estrategias establecidas en el plan de gestión de calidad que se desarrolla a través de una dinámica de procesos académicos, administrativos y de control, orientados a la formación integral de los estudiantes con la mediación de docentes que permiten el logro de los objetivos institucionales. La gestión académica se vale de la aplicación de estrategias que posibiliten que ésta sea de beneficio y coadyuven a la formación profesional de los estudiantes; es decir, promueva el aprendizaje significativo de los estudiantes a través de la participación de todos los actores de la acción educativa en su conjunto, siendo para ello indispensable que los docentes se encuentren satisfechos, especialmente con la labor docente que realizan (León Horna, 2019).

Gestión de información: Es la denominación convencional de un conjunto de procesos por los cuales se controla el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final. Tales procesos también comprenden la extracción, combinación,

depuración y distribución de la información a los interesados. El objetivo de la gestión de la información es garantizar la integridad, disponibilidad y confidencialidad de la información. Se puede identificar como la disciplina que se encargaría de todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al coste adecuado, en el momento oportuno, en el lugar apropiado y articulando todas estas operaciones para el desarrollo de una acción correcta (Infomed, 2017).

1.2 Descripción del mecanismo de gestión de la información de la Práctica Profesional en CESOL

En la Universidad de las Ciencias Informáticas la Práctica Profesional tiene como objetivos: ejecutar actividades en los proyectos productivos desempeñando roles del proceso de desarrollo de software con determinados parámetros de calidad y mostrando las competencias técnicas; aplicar los conocimientos recibidos en entrenamientos y capacitaciones, que complementan la formación académica y desarrolla una conciencia de productores de software, que incluya un comportamiento disciplinado, los hábitos de organización personal y la responsabilidad que requieren las actividades relacionadas con su desempeño como profesional.

En una entrevista (Anexo 2) realizada al Ing. Ivaniel Díaz Romeu, profesor Responsable de planificar, organizar, dirigir y controlar las actividades relacionadas con el desarrollo de las Prácticas Profesionales en CESOL, se pudo conocer las características que tiene el desempeño de las prácticas y cómo se gestiona la información resultante de su ejecución.

En las PP de CESOL se desarrollan las asignaturas Proyecto de Investigación y Desarrollo (PID) desde la III a la VII. Existen cuatro categorías de personas: los estudiantes, los SET, el Responsable de las PP y los profesores de la facultad de las distintas asignaturas PID.

- **Los estudiantes:** Son de tercero, cuarto y quinto años
- **Los SET:** Pueden ser Especialistas o RGA con categoría docente o no. Son los responsables de planificar y evaluar las tareas de los estudiantes que tienen asignados y orientarlos y guiarlos para poner en práctica el conocimiento adquirido en las diferentes asignaturas recibidas y la formación de nuevos saberes y habilidades
- **Responsable de la PP:** Coordina todas sus tareas, controla la atención que brindan los SET a los educandos, gestiona toda la información relacionada con la ejecución de las PP, elabora y entrega los reportes de asistencia y evaluación a la Dirección del Centro, el Departamento Docente de Informática de la Facultad y los Profesores de PID

- **Profesores de PID:** Encargado de dar los cursos de capacitación en el centro y evalúa las tareas que pertenecen a los planes de formación.

En el epígrafe 2.1 se presenta la descripción de este contexto de dominio mediante un modelo conceptual, que refleja las relaciones existentes entre los componentes de la Práctica Profesional en CESOL.

Los estudiantes en tercer año reciben diferentes capacitaciones sobre la metodología de desarrollo de software utilizada en CESOL, así como tecnologías empleadas en la elaboración de los productos informáticos. A partir de cuarto año comienzan a desempeñar roles relacionados con el proceso productivo como: Analista, Programador y Probador. Realizan tareas afines con los proyectos de investigación y desarrollo que se ejecutan en el centro, encaminadas a la construcción de las variantes de la distribución cubana GNU/Linux Nova, herramientas para el apoyo al diagnóstico y migración a software libre y de código abierto y aplicaciones para Android. Tienen planificadas un conjunto de horas semanales, en las que debe ir cumpliendo con las tareas programadas y demostrar los conocimientos y habilidades obtenidos mediante la elaboración de productos de trabajo, evaluados por los SET y Responsable de la PID. Toda esta información está contenida en un Plan de formación (Anexo 4) confeccionado por los SET y el Responsable de la PP para cada estudiante.

Información contenida en el plan de formación:

1-Datos generales del estudiante

- a) Nombre
- b) Grupo

2-Datos de asignatura

- a) Disciplina
- b) Asignatura
- c) Cantidad de horas total
- d) Cantidad de horas semanales

3-Datos específicos

- a) Centro de Desarrollo
- c) Grupo de Investigación
- e) Responsable de la PID
- b) Tipo de proyecto
- d) Proyecto
- f) SET
- g) Profesor PID

4-Roles y tareas

- a) Rol (es) que desempeñará b) Tarea

5-Objetivos específicos para cada semana

- a) Fecha b) Semana c) Actividades d) Habilidades e) Cantidad de horas
- f) Resultado esperado g) Tipo de resultado h) Evaluación

El compendio de asistencia y evaluaciones de todos los estudiantes lo realiza el Responsable de la PP en los Registros de Asistencia y Evaluaciones (Anexo 5), empleando herramientas informáticas como: Microsoft Word, Libre Office Writer; Microsoft Excel, Libre Office Calc, Archivos en formato PDF y el correo electrónico. A su vez tramita esta información con la Facultad. Los Profesores de PID son los autorizados a insertar la asistencia y evaluaciones de cada estudiante en Xauce Akademos, ya que tienen los permisos para estas operaciones. Los SET y el Responsable de PP de CESOL no pueden realizar estas tareas ya que carecen de los privilegios necesarios.

El sistema informático Xauce Akademos, destinado para procesar la información docente de los estudiantes no tiene las funcionalidades necesarias para la generación de los planes de formación de los estudiantes, tampoco permite realizar un seguimiento de las tareas que realizan los educandos y los resultados que obtienen. Solamente permite insertar la asistencia y las evaluaciones.

Ante estas deficiencias, el autor de la investigación realizó una búsqueda de aplicaciones informáticas que gestionan información de estudiantes, con el objetivo de encontrar alguna herramienta que solucione estos problemas. Las mismas fueron descritas en los epígrafes 1.3 y 1.4, presentados a continuación.

1.3 Caracterización de aplicaciones informáticas para la gestión de información de la Práctica Profesional

En la actualidad existen diversas aplicaciones informáticas dedicadas a la gestión académica que brindan una mejor organización, control y seguimientos de información de los datos de cada estudiante y trabajador docente de la institución docente. A continuación, se presenta una breve caracterización y las herramientas o beneficios que brindan cada una de estas.

1.3.1 Xauce Akademos: Es el Sistema de Gestión Académica empleado para almacenar las evaluaciones, proyectos asignados y asistencias de cada estudiante. Este radica en la Universidad de las Ciencias Informáticas y solo se emplea para profesores y estudiantes de la institución (Costa, 2007).

- Módulo de plan de estudio
- Módulo de matrícula
- Módulo de expediente
- Módulo de registro
- Módulo de profesor
- Módulo de estudiantes

1.3.2 UNIANDES: Es un sistema para la gestión académica de la unidad educativa Universidad de Los Andes, Bogotá, (Colombia) donde se almacena el rendimiento académico de cada estudiante, asistencia a clases, proyectos vinculados, tesis de pregrado y la cantidad de alumnos por facultad. Brinda herramientas las cuales se mostraran a continuación (Uniandes, 2022).

- Inscripciones en línea (pregrado)
- Inscripciones en línea (postgrado)
- Sistema de evaluación al desempeño docente
- Sistema de prácticas pre-profesionales
- Test de admisión estudiantil.
- Seguimientos a graduados
- Botón de pagos del estudiante
- Repositorio de revistas científicas y tesis.
- Seguimiento y control de estudiantes

1.3.3 Sistema Escolar Edcon: El Sistema de Control Escolar Edcon ofrece a las instituciones la posibilidad de monitorear y evaluar el progreso de sus procesos, recopilando información sobre las actividades de los estudiantes y docentes: fechas de ingreso, tiempo de permanencia, exámenes realizados y actividades como tutorías. Todos estos elementos forman un conjunto de herramientas que permiten garantizar el servicio para los usuarios (Edcon, 2022).

- Matricular estudiantes en línea
- Administración de grupos
- Calificaciones e inasistencias

- Bajas temporales y académicas
- Planes y guía de estudio
- Control de expedientes
- Guías de estudio
- Cambio de carrera
- Revalidaciones y equivalencias
- Seguimiento de los egresados
- Tramitación escolar
- Constancias, certificados y títulos.

1.3.4 EduColombia: Es una aplicación web que sirve para administrar las instituciones educativas. No requiere instalación e importa datos iniciales como: Lista de Alumnos, Lista de Maestros, Lista de Grados e Inventarios físicos. Brinda herramientas que hacen fácil manejar tareas comunes como (EduColombia, 2022):

- Matricular estudiantes en línea
- Calificación de alumnos
- Manejo de inventarios del Colegio
- Traslado de estudiantes a otro grupo
- Registrar información en el observador del alumno
- Manejo de estudiantes cancelados y desertores
- Certificados finales
- Impresión de Boletines de Notas
- Envío de tareas al estudiante
- Temas de estudio diferenciados por estudiante

1.3.5 Sistacnet: Es un sistema informático que brinda servicios e información para los profesores. Permite el ingreso de notas, consulta y generación de reportes, estadística de notas, envío de correos a alumnos, profesores; envío de tareas, archivos para descarga, comunicados a alumnos, padres de familia. Realiza reportes especializados si es tutor de aula. Dispone de un banco de preguntas para realizar exámenes en línea y una infraestructura web para realizar cursos virtuales. Brinda herramientas tales como (Sistacnet, 2022):

- Módulo Académico
- Módulo de Clases Online

- Proceso de Postulación y Admisión
- Calificación del estudiante
- Impresión de boletines de notas
- Registro de faltas y tardanzas del estudiante
- Envío de tareas al estudiante
- Envío de comunicados a profesores
- Recepcionar tareas del estudiante
- Historial de pagos actualizados
- Posee un módulo intranet

1.3.6 UAN: El sistema de gestión académica de la Universidad Autónoma de Nayarit permite registrar y almacenar de forma segura la información del estudiante de tal manera que pueda ser de fácil y rápido acceso para responder con agilidad, eficiencia y exactitud en el momento que está sea requerida por los miembros de la comunidad educativa. Es un sistema para el registro de la información académica de cada uno de los estudiantes de la universidad (UAN, 2022).

- Módulo Académico
- Registro de calificaciones del estudiante
- Envío de tareas al estudiante
- Impresión de boletines de notas
- Registro en línea de estudiantes
- Inscripción de cursos complementarios de enfermería
- Repositorio de revistas científicas y tesis
- Reinscripciones de nivel medio superior y superior

En el siguiente epígrafe se realiza un análisis comparativo de las aplicaciones informáticas para la gestión académica de estudiantes de las instituciones docentes caracterizadas en el epígrafe 1.3.

1.4 Análisis comparativo de aplicaciones informáticas para la gestión de información de la Práctica Profesional

La comparación de las aplicaciones estudiadas, mostrada en la Tabla 1, se efectuó en correspondencia a un conjunto de criterios definidos a partir del contexto del problema de la

investigación y las características del dominio del negocio a informatizar. A continuación, se muestra la descripción de los criterios de comparación:

- **Interfaz gráfica de usuario:** La Interfaz Gráfica de Usuario (GUI, *Graphical User Interface*) es parte fundamental de cualquier aplicación; al comenzar a trabajar con una computadora el usuario interactúa con la interfaz, ya sea la del sistema operativo, la de un software en particular o la de cualquier sitio web. Es donde inicia la interacción hombre-computadora (Albornoz, Berón y Montejano, 2017).
- **Definición de roles del proceso productivo de software:** Describen sus responsabilidades de cada participante en el proceso productivo de software. Se definen las tareas que van a realizar cada uno, sin dejar lugar para las ambigüedades sobre las responsabilidades que cada uno tiene. Se definen estos roles en administrador de proyecto, analista, diseñador, programador y probador.
- **Brindar la posibilidad de especificar proyectos de investigación y desarrollo:** Asigna a cada estudiante un proyecto de investigación y desarrollo para el desarrollo de la propuesta de solución asignada. Posee especificaciones explícitas que caracterizan al proyecto como el nombre, campo de acción, tipo de proyecto, tutor o coordinador y objetivos específicos y generales.
- **Evaluación de desempeño de las actividades:** Gestiona información sobre la evolución de un alumno en distintos momentos y a través de diversas tareas. Es la manera de obtener una visión más clara y completa del proceso de aprendizaje de cada estudiante y así tomar las decisiones más acertadas para impulsar su desempeño (Unir, 2022).
- **Definición de grupos de investigación:** Define el grupo de investigación, Desarrollo Tecnológico o Innovación a un conjunto de estudiantes que interactúan para investigar y generar productos de conocimientos en uno o varios temas, teniendo en cuenta un plan de trabajo de corto, mediano o largo plazo (Universidad Francisco José de Caldas, 2022).
- **Orientación de tareas personalizadas:** Integra en sí mismo recursos y componentes del proceso enseñanza-aprendizaje como los objetivos, los contenidos, estrategias metodológicas, los recursos de apoyo a las estrategias, las formas de organizar el proceso y las estrategias de evaluación, las cuales se personalizan por el trabajo de planificación del docente y las posibilidades, carencias y necesidades satisfechas por los estudiantes. Por su amplitud, las guías pueden organizar una tarea docente, una clase con varias tareas, una unidad, un curso, disciplinas integradoras (Torrens, Arbolaez, 2020).

- **Centralización de la información:** Evita duplicidad de archivos y materiales, ofreciendo un control definitivo sobre los datos permitiendo trabajar con workflows (flujo de trabajo) claros y organizados. Evita cambios, borradores innecesarios y crea copias de seguridad garantizan poder recuperar versiones antiguas de cualquier información de producto, lo cual eleva el control de calidad (Muñoz, 2021).
- **Definición de Supervisor, Evaluador, Tutor y Profesor:** Define Supervisor, Evaluador, Tutor (SET) o Profesor que posteriormente se vincula al proceso de la PP en el centro de desarrollo de la institución académica. Los roles de Supervisor, evaluador o Profesor gestionan la información relacionada con cada uno de los estudiantes que reciben las prácticas profesionales, en cambio los Tutores tienen tareas integradas con los proyectos de investigación y desarrollo (Patterson Peña, Puentes Puentes y Martínez Leyet, 2015).
- **Basado en software libre:** Aplicación informática cuyo código fuente se puede estudiar, modificar y utilizar libremente con cualquier fin, e incluso, copiar y redistribuir el programa con cambios. Posee libertades de ser utilizado en cualquier sistema de computación, con cualquier tipo de trabajo y finalidad y sin la obligación de comunicar a cualquier entidad específica y programador su uso (de Souza, 2019).

Tabla 1. Comparación de aplicaciones informáticas para la gestión de información de la Práctica Profesional de estudiantes

Fuente: Elaboración propia

Criterios de comparación	Aplicaciones informáticas estudiadas					
	Aplicación (1.3.1)	Aplicación (1.3.2)	Aplicación (1.3.3)	Aplicación (1.3.4)	Aplicación (1.3.5)	Aplicación (1.3.6)
Interfaz gráfica de usuario	Sí	Sí	Sí	Sí	Sí	Sí
Permitir la definición de roles del proceso productivo de software	No	No	No	No	No	No
Brindar la posibilidad de especificar proyectos de investigación y desarrollo	No	No	No	No	No	No
Evaluación del desempeño de las actividades	Sí	Sí	Sí	Sí	Sí	Sí
Definir grupo de investigación	No	No	No	No	No	No

Orientación de tareas personalizadas	No	No	No	No	Sí	Sí
Centralización de la información	Sí	Sí	Sí	Sí	Sí	Sí
Definir Supervisor-Evaluador-Tutor-Profesor que atiende al estudiante	No	No	No	No	No	No
Basados en software libre	No	No	No	No	No	No

El análisis comparativo anterior permite evidenciar que las aplicaciones estudiadas no satisfacen en su totalidad la problemática planteada al inicio de la investigación. Demuestra la importancia de desarrollar una aplicación informática que mejore el proceso de gestión de la información de la Práctica Profesional en el Centro de Software Libre. La caracterización de estas aplicaciones y la comparación efectuada contribuye en la obtención de requisitos funcionales y no funcionales, reflejados en el Capítulo 2. Permite identificar, las limitaciones presentes en las aplicaciones mencionadas, tomadas en cuenta para la elaboración de un diseño más robusto de la propuesta de solución.

Después de demostrar que las aplicaciones encontradas y analizadas no responden a las necesidades que dieron origen a la investigación, el autor del trabajo de diploma procede a definir la metodología de desarrollo de software empleada en la elaboración de la propuesta de solución, así como las tecnologías utilizadas en su construcción, mostradas en los epígrafes siguientes.

1.5 Metodología de desarrollo de software

La metodología de desarrollo de software es el conjunto de técnicas y métodos que se utilizan para diseñar una solución de software informático. Sirven para controlar el desarrollo del trabajo ya que minimiza los márgenes de errores y se anticipa a esa situación. Mejora la gestión de recursos disponibles y ahorra tiempo de trabajo, esto sucede tanto a corto como largo plazo (Calermany Universidad, 2020). Asegura la uniformidad y calidad tanto del desarrollo como del sistema en sí y consigue un mayor nivel de rendimiento y eficiencia del personal asignado al desarrollo. Ayuda a identificar, lo antes posible, cualquier cambio que sea necesario realizar dentro del proceso de desarrollo y permite construir un sistema documentado y que sea fácil de mantener (Uladech, 2017).

1.5.1 Variación de AUP para la UCI

La metodología de desarrollo de software Variación AUP para la UCI es una variación de AUP (*Agile Unified Process*, Proceso Unificado Ágil), se adapta perfectamente al ciclo de vida de la actividad productiva de los diferentes centros de la institución, logra estandarizar los productos de trabajo para todas las entidades desarrolladoras de la universidad. Dispone de tres fases las cuales muestran a continuación (Rodríguez Sánchez, 2015):

- **Inicio:** Durante el inicio del proyecto 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 el proyecto.
- **Ejecución:** En la fase 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.
- **Cierre:** En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

De las disciplinas propuestas por esta metodología, en la fase de Ejecución, se desarrolla en la investigación: Requisitos, Análisis y Diseño, Implementación, Pruebas internas y Pruebas de aceptación. Se emplea, el escenario 4 ya que aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio muy bien definido. El cliente estará siempre acompañando al equipo de desarrollo para convenir los detalles de los requisitos y así poder implementarlos, probarlos y validarlos. Se ajustan a las características del contexto de dominio de la propuesta de solución.

1.6 Lenguajes y herramientas para el modelado de la solución

En el desarrollo de la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre se emplea el lenguaje de modelado UML (*Unified Modeling Language*, Lenguaje de Modelado Unificado) y la herramienta Visual Paradigm, ambos se describen a continuación.

1.6.1 Lenguaje de modelado de software UML

El Lenguaje de Modelado Unificado fue concebido por Grady Booch, Ivar Jacobson y Jim Rumbaugh y la colaboración de otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores (Ferré Grau y Sánchez Segura, 2011). Es un lenguaje de modelado, que se trabaja con cajas, flechas y diagramas. Esto nos permite poder representar las ideas de cómo queremos estructurar nuestros programas de una forma mucho más visual (Sanchez Quiróz, 2018). Visualiza, especifica, construye y documenta partes de un sistema software orientado a objetos desde distintos puntos de vista (García Peñalvo y Pardo Aguilar, 2018).

1.6.2 Herramienta para el modelado de la solución

El modelado de software es una técnica de ingeniería donde se visualiza la aplicación informática antes de ser desarrollada mediante una herramienta informática. Esta proporciona el diseño de la arquitectura, componentes, modelado de base de datos, actividades y flujos a través de una serie de diagramas. Tiene como objetivo dejar listo el terreno para que el desarrollador codifique el sistema desde una base sólida (Lago, 2022).

Visual Paradigm 8.0

Visual Paradigm es una herramienta UML que está diseñada para una amplia gama de usuarios, incluidos ingenieros de software, analistas de sistemas, analistas de negocios y arquitectos de sistemas, o para cualquiera que esté interesado en construir de manera confiable sistemas de software a gran escala utilizando un enfoque orientado a objetos. Además, admite los últimos estándares de notación UML (Visual Paradigm 8.0 (formerly VP-UML 8.0) Released 2022).

1.7 Lenguajes de programación

Los lenguajes de programación son instrucciones que permiten dar soluciones a distintos tipos de problemas de negocios, educativos y científicos. Estos lenguajes han evolucionado a través del tiempo, desde el denominado lenguaje máquina, pasando por los lenguajes ensambladores, hasta llegar a los lenguajes de alto nivel. A continuación se muestra una breve descripción de los lenguajes utilizados en la propuesta de solución (Cordova Quispe, 2018).

PHP 8.1.1

PHP (*Hypertext Pre-processor*, Preprocesador de Hipertexto) es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. La característica principal es su uso para la comunicación de un sitio web con un servidor de datos. Gracias a esta funcionalidad es posible crear un contenido dinámico que permite trabajar con bases de datos como MySQL y con HTML. Con este lenguaje de programación de propósito general también es posible proceder a la recopilación de datos de formularios, la modificación de base de datos o la administración de archivos en el servidor (KeepCoding, 2021).

JavaScript 8.0

El lenguaje JavaScript es un lenguaje de programación de propósito general que ha sido adoptado como lenguaje específico para implementar funcionalidades ya sea frontend o backend. Posee características imperativas (if-then-else, bucles, funciones, etc.) y orientadas a objetos. Con este lenguaje de programación multiplataforma es posible dar una mayor interactividad y dinamismo a los sitios web. Entre sus distintas funcionalidades, permite crear tanto animaciones como objetos, localizar errores en formularios, cambiar elementos web de manera intuitiva, crear cookies, etcétera. Además, también permite desarrollar aplicaciones tan potentes como Facebook o Twitter (KeepCoding, 2021).

1.7.1 Lenguaje de marcado de hipertextos

HTML 5

HTML (*HyperText Markup Language*, Lenguaje de marcado de hipertexto) es un lenguaje de marcado usado para la creación de sitios web. Consiste en un conjunto de códigos cortos, que se clasifican como archivos de textos en las etiquetas. Cada una de las etiquetas generadas tiene diferentes funciones. HTML 5 es la última versión del estándar que incorpora algunas novedades interesantes. Una de las notables, es la posibilidad de añadir archivos multimedia a la web, como vídeos o audios, y que estos no tengas que insertarlos utilizando otros plug-ins, todo está incluido dentro del código (Fernández, 2021).

1.7.2 Hojas de estilos

CSS 3

CSS (*Cascade Style Sheets*, Hoja de estilo en cascada) es un lenguaje de diseño gráfico que permite definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web e interfaces de usuario escritas en HTML. CSS 3 es una especificación que se viene desarrollando desde 1999 que permite añadir con respecto a los demás estándares gradientes de colores, maquetar de manera mucho más fácil utilizando contenedores Flex y Grid y utilizar las *media-queries* (Pérez Jiménez, 2019).

1.7.3 Marcos de trabajo

Un *framework* es un esquema o marco de trabajo que ofrece una estructura base para elaborar un proyecto con objetivos específicos, una especie de plantilla que sirve como punto de partida para la organización y desarrollo de software. Generalmente, son usados porque permiten acelerar el trabajo y favorecer que este sea colaborativo, reducir errores y obtener un resultado de calidad. Sirve para acometer un proyecto en menos tiempo, y en el sector de la programación, con un código más limpio y consistente, de manera rápida y eficaz. Ofrece una estructura base que los programadores pueden complementar o modificar según sus objetivos (Unir, 2021).

Laravel 9

Laravel es un marco de trabajo de aplicaciones web PHP de código abierto creado por Taylor Otwell. Utiliza arquitectura Modelo Vista Controlador (MVC) para construir aplicaciones web simples y complejas utilizando el lenguaje de programación PHP. Ofrece un entorno de desarrollo altamente funcional, así como interfaces de línea de comandos intuitivas y expresivas. Laravel 9 requiere la última versión de PHP 8, posee mejoras y características significativas, desde la compilación JIT (*Just-In-Time*) hasta la promoción de propiedades del constructor. Posee una nueva interfaz del generador de consultas y establece que la migración anónima de *stub* es el comportamiento por defecto cuando se ejecuta el comando de migración (Solomon, 2021).

Bootstrap 5.1.3

Bootstrap es un marco de trabajo CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía. Inicialmente, se llamó Twitter Blueprint y, un poco más tarde, en 2011, se transformó en código abierto y su nombre cambió para Bootstrap, desde entonces fue actualizado varias veces. Una de las características más destacadas es la existencia de

muchísima documentación en la red sobre su manejo y muchos blogs especializados en ello. Posee compatibilidad con los principales navegadores web, se integra con las librerías de JavaScript y posee un sistema de cuadrículas que te permite crear el diseño de una web insertando el contenido en bloques o columnas (Casas, 2019).

1.7.4 Entorno de Desarrollo Integrado

Un entorno de desarrollo integrado (*Integrated Development Environment*, IDE) es una herramienta digital utilizada para desarrollar software. Ofrece integración desde los pasos más básicos del desarrollo de software, como escribir su código, depurar o incluso compilar sus aplicaciones en un lenguaje que las computadoras puedan entender (Delgado, 2021). Proporciona acceso a bibliotecas y plantillas preconstruidas para tareas de programación estándar que acelera nuestro flujo de trabajo de desarrollo y aplica las mejores prácticas de una manera fluida y fácil (Geek, 2020).

Visual Studio Code 1.67

Visual Studio Code es un editor de texto plano desarrollado por Microsoft totalmente gratuito y de código abierto para ofrecer a los usuarios una herramienta de programación avanzada como alternativa al Bloc de Notas. Está escrito totalmente en Electron, un framework utilizado para unir Chromium y Node.js en forma de aplicación de escritorio. No se caracteriza precisamente por un bajo consumo de memoria, pero es muy sencillo de programar, potente y flexible. Una de las mejores características es IntelliSense, esta función permite resaltar la sintaxis de todo el código fuente y, además, nos permite usar funciones como la de autocompletar, basándose en variables, definiciones y módulos (Velasco, 2021).

1.7.5 Servidor de aplicaciones web

Un servidor de aplicaciones web es un programa en una red distribuida que proporciona la lógica de negocio para un programa de aplicación. Se observa frecuentemente como parte de una aplicación de tres niveles, que consta de un servidor gráfico de interfaz de usuario (Interfaz Gráfica de Usuario, *Graphical User Interface*), un servidor de aplicaciones (lógica empresarial) y un servidor de bases de datos y transacciones. Proporciona varias formas diferentes de reenviar una solicitud a un servidor de aplicaciones y de reenviar una página Web nueva o modificada al usuario (Ferguson, 2021).

Apache 2.4.51

Apache HTTP Server es un software de servidor web gratuito y de código abierto para plataformas Unix con el cual se ejecutan el 46% de los sitios web de todo el mundo. Es mantenido y desarrollado por la Apache Software Foundation y fue lanzado en 1995. Su trabajo es establecer una conexión entre un servidor y los navegadores de los visitantes del sitio web (Firefox, Google, Chrome) mientras envían archivos entre ellos (estructura cliente-servidor). Posee módulos de seguridad, almacenamiento en caché, reescritura de URL (*Uniform Resource Locator*, Localizador de Recursos Uniforme) y autenticación de contraseña (Gustavo, 2018).

1.7.6 Servidor de bases de datos

Un servidor de base de datos, también conocido RDBMS (*Relational DataBase Management Systems*, Sistemas de Gestión de Base de Datos Relacionales) en caso de bases de datos relacionales, es un tipo de software de servidor que permiten la organización de la información mediante el uso de tablas, índices y registros. A nivel de hardware, es un equipo informático especializado en servir consultas a clientes remotos o locales que solicitan información o realizan modificaciones a los registros y tablas que existen dentro de las bases de datos del sistema, en muchos casos desde un servidor web o de aplicaciones (Borges, 2019).

MySQL 8.0.27

MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por Sun Microsystems en 2008 y a su vez comprada por Oracle Corporation en 2010, la cual ya era dueña de un motor propio InnoDB (mecanismo de almacenamiento de datos) para MySQL. Presenta algunas ventajas que lo hacen muy interesante para los desarrolladores. La más evidente es que trabaja con bases de datos relacionales, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente (Robledano, 2019).

1.8 Herramienta de validación

Acunetix 14.8

Acunetix es una herramienta de seguridad de aplicaciones web automatizada. Es capaz de escanear cualquier sitio o aplicación web que sea accesible a través del protocolo HTTP/HTTPS,

sin embargo, no todas las pruebas se pueden realizar de forma automática. Proporciona herramientas de penetración manuales para pruebas particulares y comprueba más de 500 tipos de vulnerabilidades, por ejemplo, la inyección de SQL. (Baró, 2019).

1.9 Conclusiones del capítulo

El análisis del marco teórico sobre el proceso de gestión de la información de las prácticas profesionales y el estudio de los principales conceptos asociados al problema planteado, permitieron establecer las bases para el desarrollo de la investigación y conocer las características del objeto de estudio. A partir de la caracterización y comparación de varias aplicaciones para la gestión de la Práctica Profesional se demuestra que se debe desarrollar una aplicación informática que responda a las necesidades planteadas al inicio de la investigación. En elaboración de la propuesta de solución se definió la metodología utilizada, así como las tecnologías y herramientas empleadas en la construcción de sistema para la gestión de la información de la PP en CESOL.

CAPÍTULO 2: Análisis y diseño de una aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre

Este capítulo aborda los aspectos técnicos de la investigación referentes al análisis y diseño de la aplicación informática para la gestión de la información de la Práctica Profesional en el Centro de Software Libre CESOL. Contiene un modelo conceptual realizado para comprender el contexto del negocio informatizado. Se definen los requisitos a través de la especificación de requisitos de software, su descripción mediante historias de usuario, además de realizar la validación de los mismos. Se realiza el diagrama de clases del diseño, en el que se representan las restricciones de implementación de las funcionalidades del sistema a partir de los patrones del diseño. Se presenta el modelo de datos que describe la estructura de persistencia de la información y el diseño arquitectónico definido para la propuesta de solución.

2.1 Descripción del contexto de dominio de la propuesta de solución desarrollada

La descripción del contexto del negocio presentada en el epígrafe se realizó utilizando un modelo conceptual. Este permite al autor de la investigación comprender el funcionamiento del dominio del negocio informatizado. A continuación, se muestra el modelo conceptual en el siguiente epígrafe.

2.1.1 Modelo conceptual

Un modelo conceptual consiste en el diseño de un plan sistemático donde se exponen claramente los propósitos, las ideas, y los aspectos que deben tenerse en cuenta para la creación de un negocio, es una etapa preliminar al plan de negocio que tiene como principal objetivo modelar y madurar una idea u oportunidad de negocio, en este caso, se refiere a la gestión de la información de la Práctica Profesional en el Centro de Desarrollo CESOL (León y Andrés, 2017).

En la Figura 1 se presenta el modelo conceptual del dominio del negocio referente a la gestión de la Práctica Profesional en CESOL. El mismo le permitió al autor de la investigación comprender los conceptos que interactúan en el contexto informatizado. La información que se utiliza de cada uno de estos y las relaciones que subsisten entre ellos.

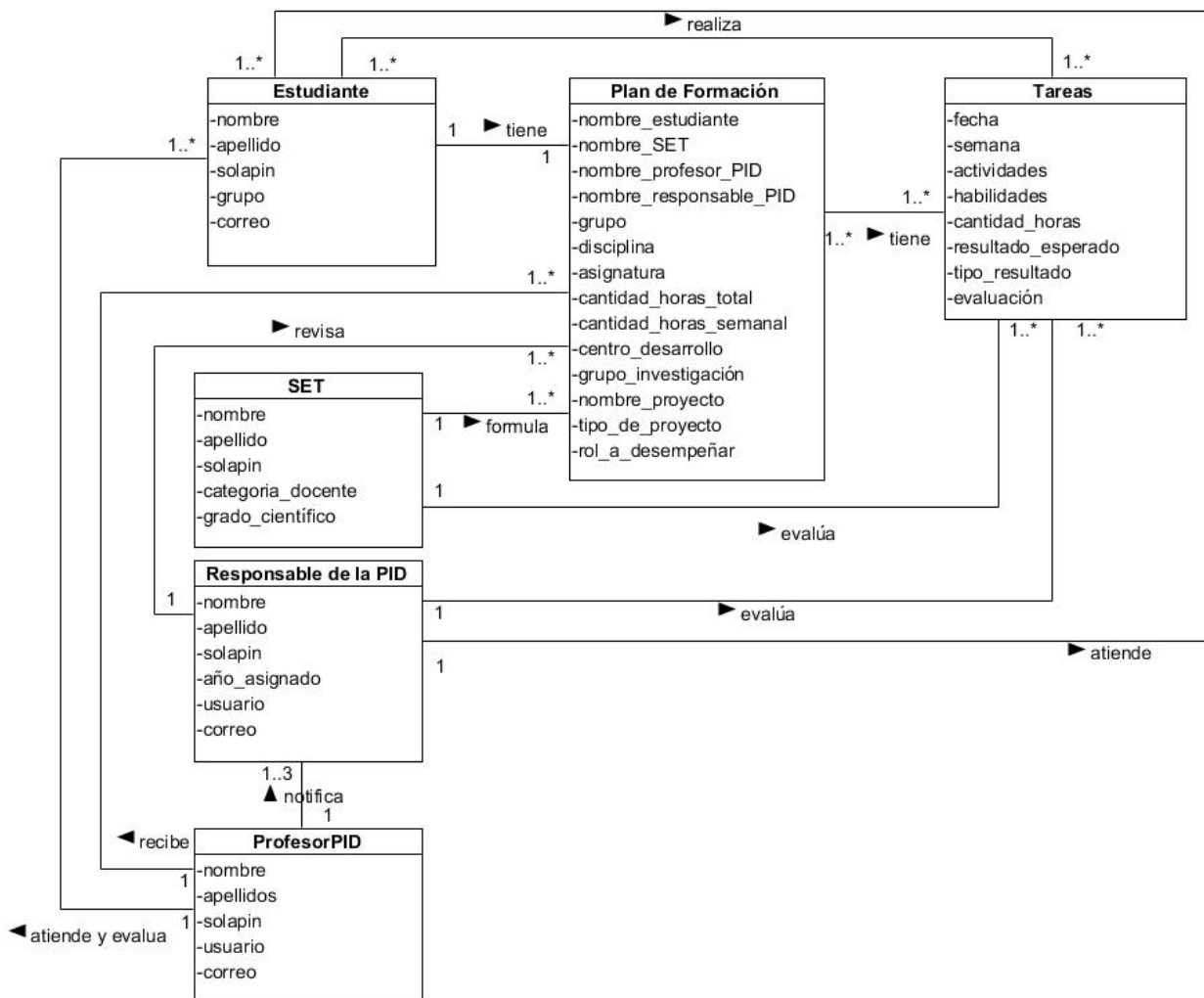


Figura 1. Modelo Conceptual

Fuente: elaboración propia

Las definiciones que se detallan a continuación permiten comprender el significado de los conceptos representados en el modelo conceptual en el dominio del negocio informatizado.

Conceptos del contexto del negocio:

- **SET:** Supervisor, evaluador y tutor: es el especialista de la atención, orientación y evaluación de los estudiantes en el centro.
- **Estudiante:** Son alumnos de tercero, cuarto o quinto año que realizan la Práctica Profesional en CESOL.

- **Profesor:** Es el profesor responsable de impartir los diferentes cursos de capacitación de la PP en el centro y evaluar a los estudiantes mediante las distintas tareas.
- **Responsable de PID:** Encargado de coordinar todas las tareas de los estudiantes y SET en el centro.
- **Plan de formación:** Documento que contiene todos los elementos necesarios para gestionar y organizar toda la información pertinente a los estudiantes como son: tareas asignadas, cantidad de horas semanales, centro de desarrollo, tipo de proyecto y rol.
- **Tarea:** Actividad que tiene como objetivo sistematizar el conocimiento del estudiante.

2.2 Requisitos

El esfuerzo principal en la disciplina Requisitos es desarrollar un modelo del sistema que se va a construir. Esta disciplina comprende la administración y gestión de los requisitos funcionales y no funcionales del producto (Rodríguez Sánchez, 2015). Los siguientes subepígrafes describen las tareas y los productos de trabajo desarrollados en esta disciplina.

2.2.1 Fuentes para la obtención de requisitos

Uno de los elementos más significativos del proceso de desarrollo del software es la obtención de los requisitos porque ajusta intereses entre los involucrados y determina qué software se desea desarrollar (Toledo Rodríguez, 2019). En este proceso intervienen diferentes fuentes que permiten identificar los requisitos que forman parte de una aplicación informática. Durante esta etapa de la investigación se tuvieron en cuenta como fuentes de obtención de requisitos:

- Descripción del mecanismo de gestión de la información de la Práctica Profesional en CESOL (Epígrafe 1.2)
- Caracterización y Análisis de los sistemas homólogos (Epígrafes 1.3 y 1.4)
- Descripción del contexto de dominio de la propuesta de solución desarrollada (Epígrafe 2.1)
- 15 SET y el Responsable de la Práctica Profesional de CESOL

Una adecuada comprensión de los requisitos beneficia el desarrollo de las aplicaciones informáticas y que estos cumplan con las necesidades y expectativas del cliente. Para realizar

este procedimiento existen diversas técnicas que rigen al equipo de proyecto en el proceso de comunicación con el cliente (Toledo Rodríguez, 2019). En el siguiente epígrafe se relacionan las técnicas empleadas para la obtención de los requisitos de la aplicación informática para la gestión de información de la PP en el Centro de Software Libre.

2.2.2 Técnicas de Identificación de Requisitos

En la selección de las Técnicas de Identificación de Requisitos (TIR) se debe elegir la técnica, método, herramienta o enfoque de elicitación de requisitos adecuada para obtener y expresar las necesidades de los usuarios, donde se debe considerar la naturaleza del proyecto que se va a implementar, por ejemplo: entrevistas, prototipos, escenarios, reuniones (Wong Portillo, 2019). Las TIR utilizadas en el desarrollo de la propuesta de solución se describen a continuación.

- **Entrevista:** Es una técnica muy utilizada para recoger información de otra persona mediante preguntas abiertas que son útiles para identificar, documentar, confirmar o validar información detallada. Direccionan al usuario hacia aspectos específicos de los requisitos o requerimientos a levantar e informatizar (Sarmiento Cuervo y Hernández, 2017).

La técnica se aplicó a través de una serie de preguntas realizadas al Responsable de la PP en CESOL para conocer cuáles los requerimientos y las características que debe tener la aplicación de gestión de la información de la Práctica Profesional en el centro.

- **Encuesta:** La encuesta es un conjunto de procedimientos estandarizados de investigación mediante los cuales se recoge y analiza una serie de datos de una muestra de casos representativa de una población, del que se pretende explorar, describir y explicar una serie de características a través de la interrogación a los miembros una comunidad (Falcón, Pertile y Ponce, 2019).

Esta técnica se ejecutó mediante un cuestionario realizado por 15 SET de CESOL para conocer las funcionalidades y las características que el autor de la investigación debía incluir en la propuesta de solución.

- **Tormenta de ideas:** Es una sesión de trabajo estructurada, orientada para obtener la mayor cantidad de ideas posibles. Es recomendable limitarlas en el tiempo, utilizar ayudas visuales y designar un facilitador. Tiene como objetivo identificar la mayor cantidad de ideas, para luego evaluarlas, estas deben ser consideradas y limitarse a crítica antes de desarrollarla (Sarmiento Cuervo y Hernández, 2017).

Se realizó una tormenta de ideas con el Responsable de PP y los 15 SET de CESOL, para concretar las funcionalidades de la propuesta de solución, ya que se habían identificado un primer conjunto de requisitos mediante la entrevista y las encuestas.

- **Análisis de documentación:** Consiste en obtener la información sobre los requerimientos funcionales y requerimientos no funcionales de software a partir de documentos que ya están elaborados. Utiliza la documentación que sea relevante al requerimiento que se está levantando por ejemplo tesis, libros, secciones de revista o sitios web (Sarmiento Cuervo y Hernández, 2017).

La aplicación de esta técnica le permitió al autor de la investigación analizar los informes: Plan de formación (Anexo 4) y Registro de Asistencia y Evaluación (Anexo 5), referentes a la gestión de información de la Práctica Profesional en el Centro de Software Libre para definir requisitos de software.

2.2.3 Especificación de requisitos de software

La Especificación de Requisitos de Software (ERS) es una representación de las características del sistema que se va a implementar. Describe claramente lo que se desea alcanzar de un determinado software y permite a los desarrolladores entender qué pide el cliente. Incluye un conjunto de requisitos funcionales y no funcionales que sirven para relatar las necesidades de los clientes y de los usuarios, así como los requerimientos que debe ostentar el sistema para tener su funcionamiento óptimo (Toledo Rodríguez, 2019). A continuación, se relacionan los requisitos funcionales y no funcionales de la propuesta de solución.

Requisitos funcionales

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer (Sommerville, 2011).

Los requisitos funcionales (RF) identificados para el desarrollo del sistema de gestión de la información de la Práctica Profesional de CESOL son 38 en su totalidad, y se muestran en la Tabla 2:

Tabla 2: Especificación de requisitos funcionales de la propuesta de solución

Fuente: elaboración propia

Número	Requisito	Descripción	Complejidad
RF1	Autenticar usuario	El sistema permite que se autentique un usuario.	Alta
RF2	Insertar estudiante	El sistema permite al administrador insertar un usuario seleccionando el botón insertar estudiante.	Alta
RF3	Modificar estudiante	El sistema permite al administrador modificar un estudiante seleccionando el botón modificar estudiante.	Medio
RF4	Eliminar estudiante	El sistema permite al administrador eliminar un estudiante seleccionando el botón eliminar estudiante.	Bajo
RF5	Listar estudiantes	El sistema permite al administrador listar los estudiantes registrados en la base de datos.	Medio
RF6	Consultar estudiante	El sistema permite al administrador consultar los datos de un estudiante seleccionando el botón consultar estudiante.	Bajo
RF7	Buscar estudiante	El sistema permite al administrador buscar a un estudiante ingresando sus datos.	Bajo
RF8	Insertar profesor	El sistema permite al administrador insertar un profesor seleccionando el botón insertar profesor.	Alto
RF9	Modificar profesor	El sistema permite al administrador modificar un profesor seleccionando el botón modificar profesor.	Medio
RF10	Eliminar profesor	El sistema permite al administrador eliminar un profesor seleccionando el botón eliminar profesor.	Bajo
RF11	Listar profesores	El sistema permite al administrador listar los profesores registrados en la base de datos.	Medio

RF12	Consultar profesor	El sistema permite al administrador consultar los datos de un profesor seleccionando el botón consultar profesor.	Bajo
RF13	Buscar profesor	El sistema permite al administrador buscar a un profesor ingresando sus datos.	Bajo
RF14	Insertar SET	El sistema permite al administrador insertar un set seleccionando el botón insertar set.	Alto
RF15	Modificar SET	El sistema permite al administrador modificar un set seleccionando el botón modificar set.	Medio
RF16	Eliminar SET	El sistema permite al administrador eliminar un set seleccionando el botón eliminar set.	Bajo
RF17	Listar SET	El sistema permite al administrador listar los sets registrados en la base de datos.	Medio
RF18	Consultar SET	El sistema permite al administrador consultar los datos de un set seleccionando el botón consultar set.	Bajo
RF19	Buscar SET	El sistema permite al administrador buscar a un set ingresando sus datos.	Bajo
RF20	Insertar plan de formación	El sistema permite al administrador insertar un plan de formación seleccionando el botón insertar plan de formación.	Alto
RF21	Modificar plan de formación	El sistema permite al administrador modificar un plan de formación seleccionando el botón modificar plan de formación.	Medio
RF22	Eliminar plan de formación	El sistema permite al administrador eliminar un plan de formación seleccionando el botón eliminar plan de formación.	Bajo
RF23	Consultar plan de formación	El sistema permite al administrador consultar los datos de un plan de formación seleccionando el botón consultar plan de formación.	Bajo

RF24	Listar plan de formación	El sistema permite al administrador listar los planes de formación registrados en la base de datos.	Bajo
RF25	Buscar plan de formación	El sistema permite al administrador buscar a un plan de formación ingresando sus datos.	Bajo
RF26	Insertar tarea	El sistema permite al administrador insertar una tarea seleccionando el botón insertar tarea.	Alta
RF27	Modificar tarea	El sistema permite al administrador modificar una tarea seleccionando el botón modificar tarea.	Medio
RF28	Eliminar tarea	El sistema permite al administrador eliminar una tarea seleccionando el botón eliminar tarea.	Bajo
RF29	Consultar tarea	El sistema permite al administrador consultar los datos de una tarea seleccionando el botón consultar tarea.	Bajo
RF30	Listar tareas	El sistema permite al administrador listar las tareas registradas en la base de datos.	Bajo
RF31	Buscar tarea	El sistema permite al administrador buscar a una tarea ingresando sus datos.	Bajo
RF32	Insertar Responsable PID	El sistema permite al administrador insertar un Responsable PID seleccionando el botón insertar Responsable PID.	Alta
RF33	Modificar Responsable PID	El sistema permite al administrador modificar un Responsable PID seleccionando el botón modificar Responsable PID.	Medio
RF34	Eliminar Responsable PID	El sistema permite al administrador eliminar una Responsable PID seleccionando el botón eliminar responsable PID.	Bajo
RF35	Listar Responsable PID	El sistema permite al administrador listar las Responsable PID registradas en la base de datos.	Bajo

RF36	Consultar Responsable PID	El sistema permite al administrador consultar los datos de un Responsable PID seleccionando el botón consultar tarea.	Bajo
RF37	Buscar Responsable PID	El sistema permite al administrador buscar a un Responsable PID ingresando sus datos.	Bajo
RF38	Asociar tarea a plan de formación	El sistema permite asignar tarea a un plan de formación determinado	Medio

Requisitos no funcionales

Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requerimientos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema (Sommerville, 2011).

La definición de los requisitos no funcionales (RNF) de la propuesta de solución se realizó mediante el estándar de calidad ISO/IEC-25000 (ISO, *International Organization for Standardization*, Organización Internacional de Normalización; IEC, *International Electrotechnical Commission*, Comisión Electrotécnica Internacional). En la Tabla 3 se describen estos requisitos no funcionales:

Tabla 3. Especificación de requisitos no funcionales de la propuesta de solución

Fuente: elaboración propia

Número	Atributo de calidad	Subatributo de calidad	Descripción del Requisito
RNF1	Adecuación funcional	Compleitud funcional	El conjunto de funcionalidades de la aplicación cubre todas las tareas y los objetivos del usuario especificados.
RNF2		Corrección funcional	La aplicación desarrollada provee resultados correctos con el nivel de precisión requerido.
RNF3		Pertinencia funcional	La aplicación desarrollada proporciona un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.
RNF4	Usabilidad	Capacidad para	La aplicación desarrollada permite al usuario

		reconocer su adecuación	entender si el software es adecuado para sus necesidades.
RNF5		Capacidad de aprendizaje	La aplicación desarrollada permite al usuario aprender su aplicación.
RNF6		Capacidad para ser usado	La aplicación desarrollada permite al usuario operarlo y controlarlo con facilidad.
RNF7		Protección contra errores de usuario	La aplicación desarrollada permite proteger a los usuarios de hacer errores.
RNF8		Estética de la interfaz de usuario	La aplicación desarrollada dispone de una interfaz de usuario que agrada y satisface la interacción con el cliente.
RNF9	Fiabilidad	Madurez	La aplicación desarrollada satisface las necesidades de fiabilidad en condiciones normales.
RNF10		Disponibilidad	La aplicación desarrollada tiene la capacidad de estar operativo y accesible para su uso cuando se requiere.
RNF11		Tolerancia a fallos	La aplicación desarrollada tiene la capacidad para operar según lo previsto en presencia de fallos hardware o software.
RNF12		Capacidad de recuperación	La aplicación desarrollada posee la capacidad para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de fallo.
RNF13	Seguridad	Confidencialidad	La aplicación desarrollada tiene la capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
RNF14		Integridad	La aplicación desarrollada tiene la capacidad de prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.

RNF15	Mantenibilidad	Modularidad	La aplicación desarrollada tiene la capacidad de permitir que un cambio en un componente tenga un impacto mínimo en los demás.
RNF16		Capacidad para ser modificado	La aplicación desarrollada tiene la capacidad de ser modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.
RNF17		Capacidad para ser probado.	La aplicación desarrollada brinda la facilidad para establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.
RNF18	Calidad en uso	Satisfacción	La aplicación desarrollada garantiza la satisfacción de los usuarios en la gestión de la información de la PP.
RNF19		Productividad	La aplicación desarrollada permite aumentar la productividad de los usuarios en la gestión de la información de la PP.

Para describir las características de los requisitos funcionales definidos en el epígrafe 2.2.3, así como sus restricciones en el contexto del negocio informatizado, se utilizó el producto de trabajo Historia de Usuario. En el epígrafe 2.2.4 se presenta un ejemplo de este artefacto de la ingeniería de software.

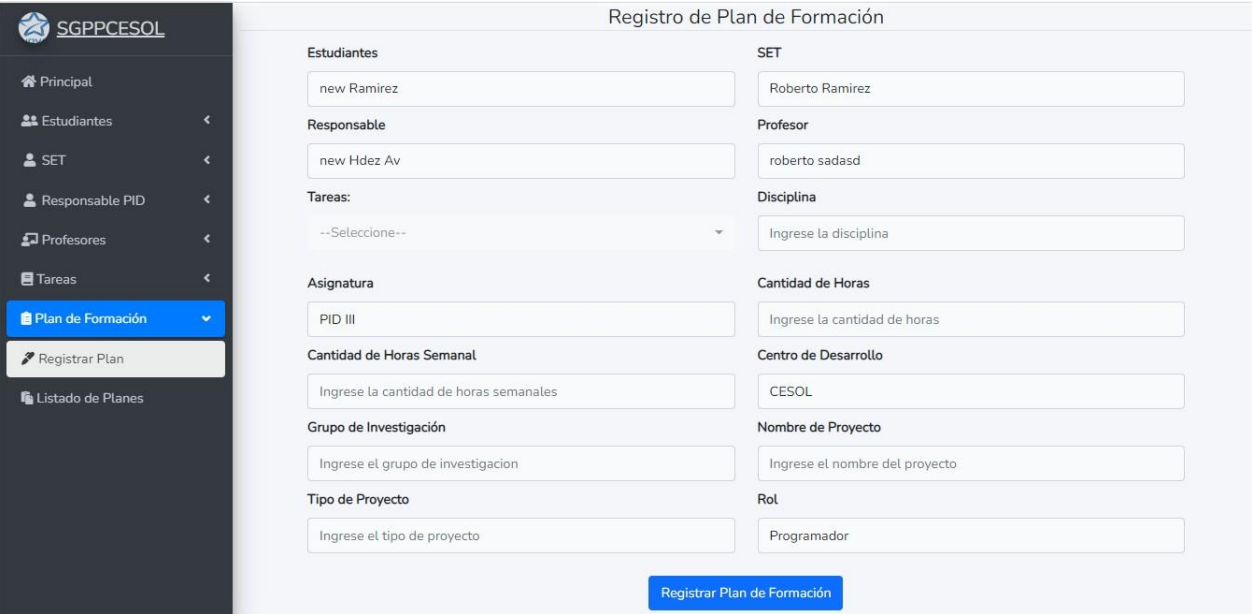
2.2.4 Descripción de requisitos de software mediante Historias de Usuario

En la descripción de los requisitos se elaboraron 38 Historias de Usuarios. En la tabla 5 se muestra la descripción de la historia de usuario correspondiente al RF 20: Insertar plan de formación.

Tabla 4. Historia de usuario del RF 20: Insertar plan de formación

Fuente: elaboración propia

HISTORIA DE USUARIOS	
Número: RF20	Nombre del requisito: Insertar plan de formación
Programador: Roberto Ramírez Carreño	Iteración Asignada: 1

Prioridad: Alta	Tiempo Estimado: 15 horas
Riesgo en Desarrollo: No aplica	Tiempo Real: 15 horas
<p>Descripción:</p> <p>Permite al administrador (Profesor de PID) crear un plan de formación donde llena los campos de texto del formulario (Disciplina, Cantidad de Horas, Centro de Desarrollo, Nombre de Proyecto, Rol, Asignatura, Cantidad de Horas Semanal, Grupo de Investigación y Tipo de Proyecto). Una vez llenados los campos se selecciona la opción Registrar Plan de Formación la cual es un botón que se encuentra en la parte inferior del formulario.</p>	
<p>Observaciones:</p>	
<p>Prototipo elemental de interfaz gráfica de usuario:</p> 	
<p><i>Figura 2. Prototipo de interfaz de usuario correspondiente al RF20. Insertar Plan de Formación</i></p> <p><i>Fuente: elaboración propia</i></p>	

Los requisitos identificados para el desarrollo de la propuesta de solución fueron validados mediante las técnicas prototipado de interfaz de usuario y diseño de caso de prueba (DCP) para verificar que satisfacen las necesidades del negocio informatizado. En el epígrafe 2.2.5 se describen cómo se aplicaron las técnicas.

2.2.5 Validación de requisitos de software

La validación de requisitos de software es un proceso para determinar si los requerimientos del sistema o producto software, tal como se ha construido, cumplen con el objetivo previsto. Es un conjunto de actividades encaminadas a llegar a un acuerdo entre todos los participantes en el que se ratifique que los requisitos elicitados y analizados representan realmente las necesidades de los involucrados y contribuyendo a la construcción de un software útil. Permite garantizar que los requisitos han sido enunciados sin ambigüedades; que se detectaron y corrigieron las inconsistencias, las omisiones y los errores, y que los productos de trabajo se presentan conforme a los estándares establecidos para el proceso, el proyecto y el producto (Canchari Cuyutupac, 2018).

Prototipado de interfaz de usuario

Es el medio que le permite al usuario de un sistema informático su interacción con este de forma gráfica. Su principal función es la exploración de aspectos interactivos del sistema como la usabilidad, la accesibilidad y la funcionalidad. Permite ver los flujos de entrada y salida, ya sea en una sola sección o en un producto digital completo. Los prototipos de interfaz de usuario ayudan a ordenar ideas, explorar diferentes caminos de conceptos o diseños y detectar posibles problemas o defectos antes de empezar la fase de programación (Lara Galicia, 2020).

Los prototipos de interfaz de usuario elaborados de acuerdo a los requisitos obtenidos fueron analizados con el Responsable de la PP, lo cual permitió conocer las deficiencias existentes con el objetivo de realizar las correcciones necesarias antes de implementar las funcionalidades de la propuesta de solución.

Diseño de caso de prueba (DCP)

Los diseños de casos de pruebas tienen como objetivo determinar si una funcionalidad o requisito funcional particular de un sistema informático es correcto. Permite crear escenarios para probar los posibles defectos que pudiera tener una aplicación y demostrar que se satisfacen los requisitos obtenidos (Hoogenraad, 2018).

La elaboración de casos de prueba le permitió al autor de la investigación definir los posibles escenarios que se ejecutan en cada requisito, y las respuestas de la aplicación informática a los diferentes datos de entrada. En total se generaron 38 diseños de casos de pruebas, uno para cada requisito funcional. Estos DCP fueron revisados con el Responsable de la PP de CESOL

contribuyendo a la identificación y corrección de las deficiencias encontradas para mejorar la calidad de las pruebas de software mostradas en el capítulo 3. A continuación, se presenta el DCP que corresponde al RF 20. Insertar plan de formación.

Caso de prueba correspondiente al RF 20. Insertar plan de formación

Descripción general: Permite al profesor de PID insertar plan de formación.

Precondiciones:

Precondición 1: Insertar al estudiante que se le asigna el plan de formación.

Precondición 2: Insertar al Responsable de la PP que revisa el plan de formación.

Precondición 3: Insertar al SET que formula el plan de formación.

Precondición 4: Insertar al profesor que elabora el plan de formación.

Precondición 5: Insertar las tareas asignadas al plan de formación.

Escenario	Descripción	Nombre estudiante	Nombre SET	Nombre responsable	Nombre profesor	Tareas	Disciplina	Tipo de proyecto	Respuesta del sistema	Flujo central
1-Insertar plan de formación con datos correctos	El SET rellena los campos necesarios correctamente y selecciona el botón Ingresar plan de formación.	Roberto	Edgardo	Julio	Ramón	-Pregunta Escrita -Evaluación Oral	Programación web	Seminario	El plan de formación se registró con éxito	1.Seleccionar botón "Insertar plan de formación"
		Asignatura	Cantidad horas total	Cantidad horas semanal	Centro de desarrollo	Grupo de investigación	Nombre proyecto	Rol		
		PID 5	14	10	CESOL	Grupo 1501	Diseño Web.	Probador		
2-Insertar plan de formación con campos vacíos	El SET rellena algunos campos de texto y selecciona el botón Ingresar plan de formación.	Nombre estudiante	Nombre SET	Nombre responsable	Nombre profesor	Tareas	Disciplina	Tipo de proyecto	Este campo es obligatorio	1.Seleccionar botón "Insertar plan de formación"
		Daniel	Richard	Julio	Ramón	-Pregunta Escrita -Evaluación Oral		875		
		Asignatura	Cantidad horas total	Cantidad horas semanal	Centro de desarrollo	Grupo de investigación	Nombre proyecto	Rol		
		PID 4	CESOL		CESOL		Programación web con PHP	Probador		

Escenario	Descripción	Nombre estudiante	Nombre SET	Nombre responsable	Nombre profesor	Tareas	Disciplina	Tipo de proyecto	Respuesta del sistema	Flujo central
3-Insertar plan de formación existente	El SET rellena los campos necesarios correctamente y selecciona el botón Ingresar plan de formación.	Roberto	Edgardo	Julio	Ramón	-Pregunta Escrita -Evaluación Oral	Programación web	Seminario	El plan de formación ya existe	1.Seleccionar botón "Insertar plan de formación"
		Asignatura	Cantidad horas total	Cantidad horas semanal	Centro de desarrollo	Grupo de investigación	Nombre proyecto	Rol		
		PID 5	14	10	CESOL	Grupo 1501	Programación web con PHP.	Probador		
4-Insertar plan de formación con datos incorrectos	El SET rellena algunos campos de texto y selecciona el botón Ingresar plan de formación.	Nombre estudiante	Nombre SET	Nombre responsable	Nombre profesor	Tareas	Disciplina	Tipo de proyecto	Los datos son incorrectos	1.Seleccionar botón "Insertar plan de formación"
		Daniel	Richard	Julio	Ramón	-Pregunta Escrita -Evaluación Oral	5678	875		
		Asignatura	Cantidad horas total	Cantidad horas semanal	Centro de desarrollo	Grupo de investigación	Nombre proyecto	Rol		
		PID 4	CESOL	Programación	CESOL	Grupo 1501	Programación web con PHP	Probador		

Figura 3. Casos de prueba correspondiente al RF20. Insertar Plan de Formación

Fuente: elaboración propia

Descripción de las variables

Tabla 5. Descripción de las variables

Fuente: elaboración propia

No	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Nombre del estudiante	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
2	Nombre del SET	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
3	Nombre Profesor	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

4	Nombre Responsable PID	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
5	Tareas	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
6	Disciplina	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
7	Cantidad de Horas	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores numéricos
8	Centro de Desarrollo	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
9	Nombre del Proyecto	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
10	Rol	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
11	Asignatura	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
12	Cantidad de Horas Semanal	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite

				valores numéricos
13	Grupo de investigación	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
14	Tipo de Proyecto	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

Las necesidades del contexto de dominio de la propuesta de solución identificadas y relacionadas en los diferentes productos de trabajo elaborados durante la disciplina de requisitos, fueron materializadas en modelos de diseño de software, y se muestran en el epígrafe 2.3, descrito a continuación.

2.3 Análisis y diseño

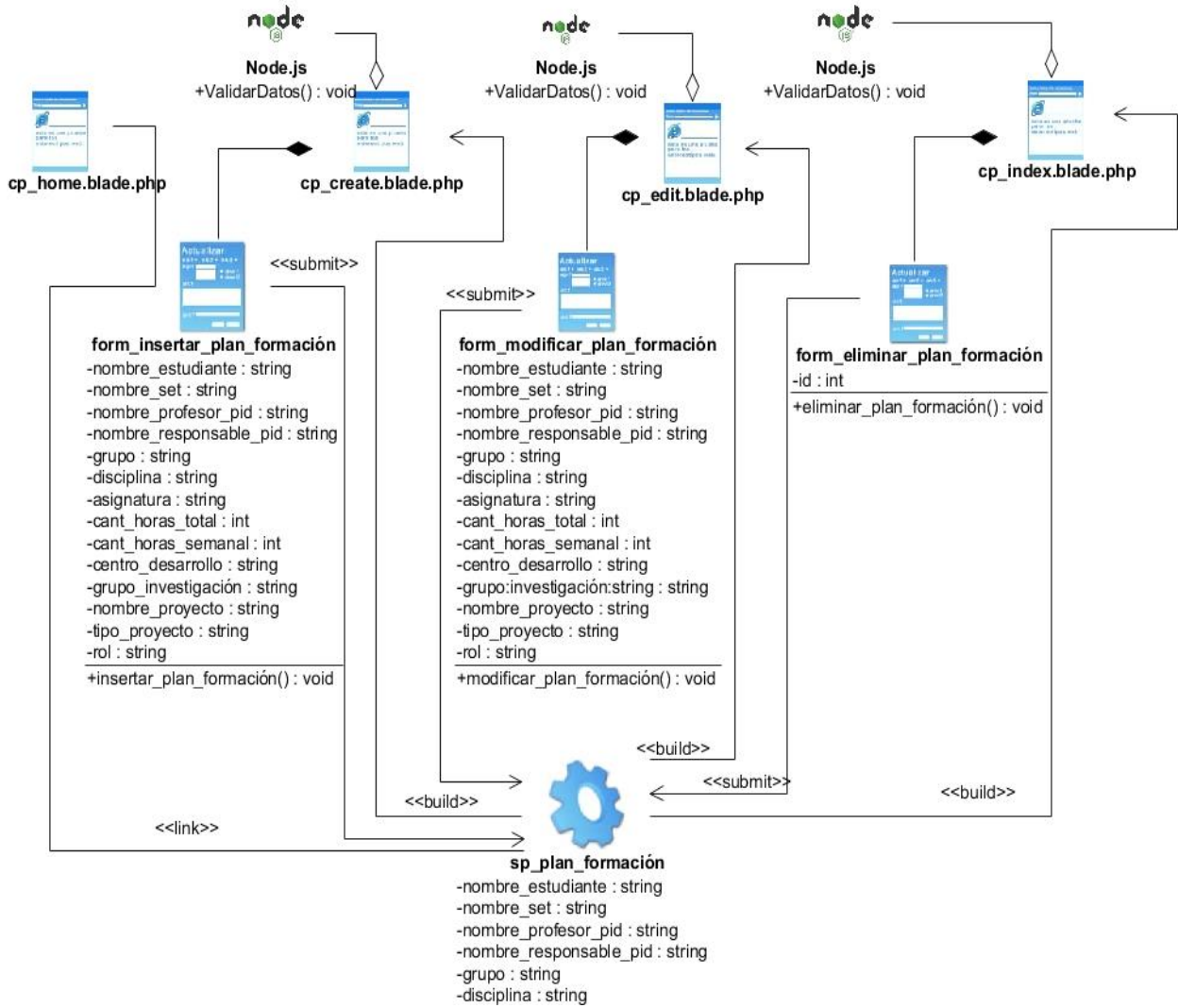
En esta disciplina los requisitos pueden ser refinados y estructurados para adquirir una comprensión y una descripción precisa que sea fácil de mantener y ayude a la estructuración del sistema. Se modela el sistema, su forma y su arquitectura para que soporte todos los requisitos, incluyendo los requisitos no funcionales (Rodríguez Sánchez, 2015). En los siguientes epígrafes se muestran las actividades desarrolladas, así como los productos de trabajo elaborados en el proceso de gestión de la información de las PP en el centro de software libre.

2.3.1 Diseño de clases

El diseño de clases de la aplicación informática para la gestión de la información de las PP en el Centro de Software Libre, se realizó mediante estereotipos web. En la Figura 6 se presenta el diagrama de clases del diseño para la agrupación de requisitos plan de formación.

Diagrama de clases de diseño con estereotipos web

Los diagramas de clases basados en estereotipos web son unos de los principales diagramas utilizados en el modelado de negocio de las aplicaciones informáticas. Perfilan la estructura de un sistema al modelar sus clases, atributos, operaciones y relaciones entre objetos. A continuación, en la figura 6 se muestra el diagrama de clases del requisito plan de formación (Anexo 7).



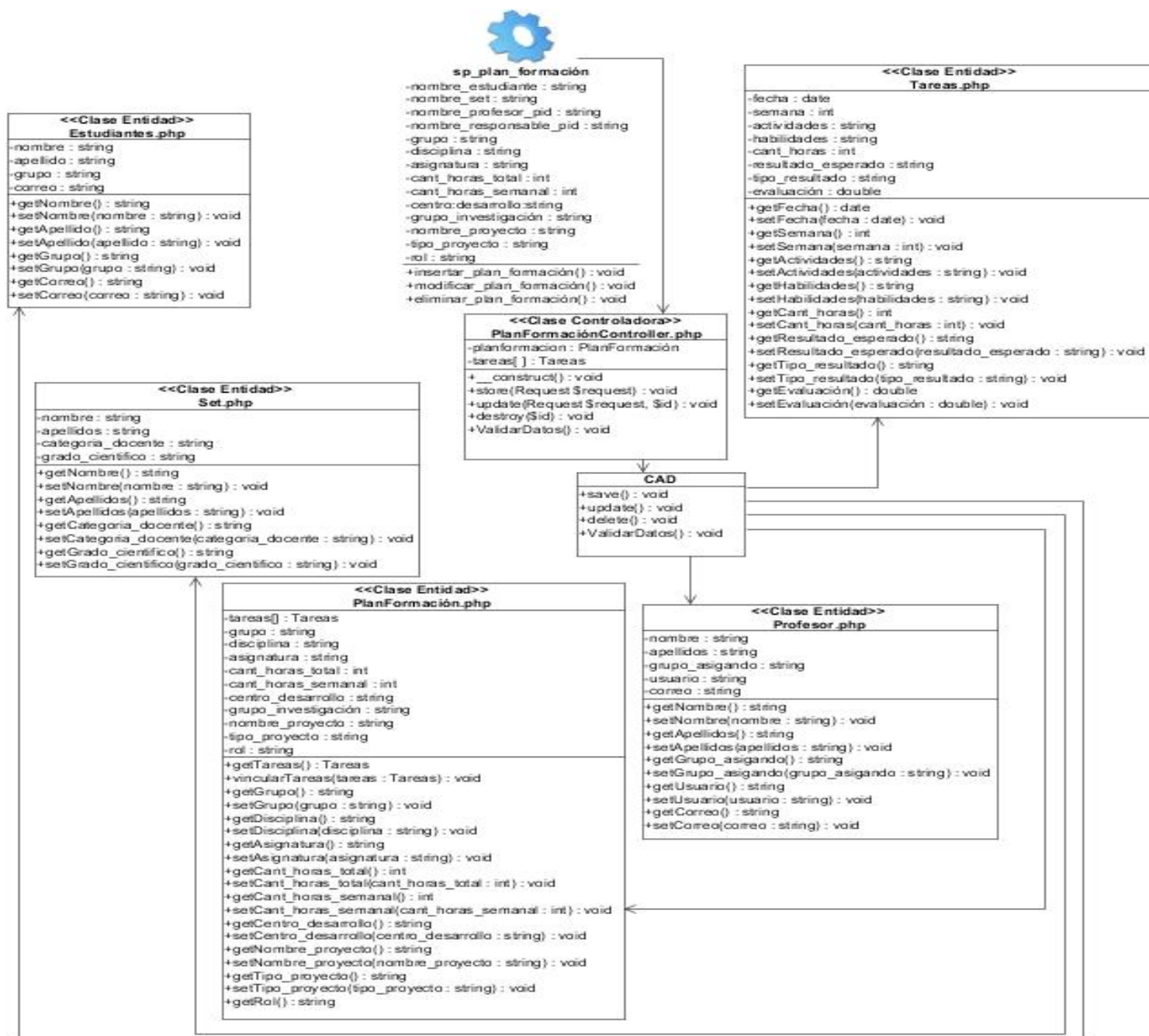


Figura 4. Diagrama de Clases

Fuente: elaboración propia

Página Cliente: Son las vistas las cuales permiten la interacción del usuario con la aplicación informática. Está conformado por home.blade.php, create.blade.php, index.blade.php, edit.blade.php y show.blade.php.

Clase Controladora: Contiene las funcionalidades para interactuar con las clases interfaces, las clases entidades y de acceso a los datos. Está conformada por la clase PlanFormacionController.php.

Clase Acceso a Datos: Permite la interacción entre la clase controladora del negocio de la aplicación y la base de datos.

Clase Entidad: Representa los datos de cada elemento necesario en el negocio. Está representada por las clases Tareas.php, PlanFormacion.php, Estudiantes.php, Sets.php y Responsable.php.

2.3.2 Diagrama de despliegue

Los elementos del diseño al nivel del despliegue revelan cómo se ubicarán la funcionalidad y los subsistemas dentro del entorno computacional físico que soportará al software. En el modelo de despliegue se define la organización física del sistema en términos de nodos de cómputo y se contrasta que los casos de uso puedan efectuarse como mecanismos que se ejecutan en esos nodos (González Rodríguez, 2017).

El siguiente diagrama de despliegue describe la arquitectura física del sistema de gestión de la información de las prácticas profesionales en el Centro de Desarrollo de Software Libres durante la ejecución, en términos de procesadores, dispositivos y componentes de software. Permite una mejor comprensión entre la correspondencia de la arquitectura de software y la arquitectura de hardware.

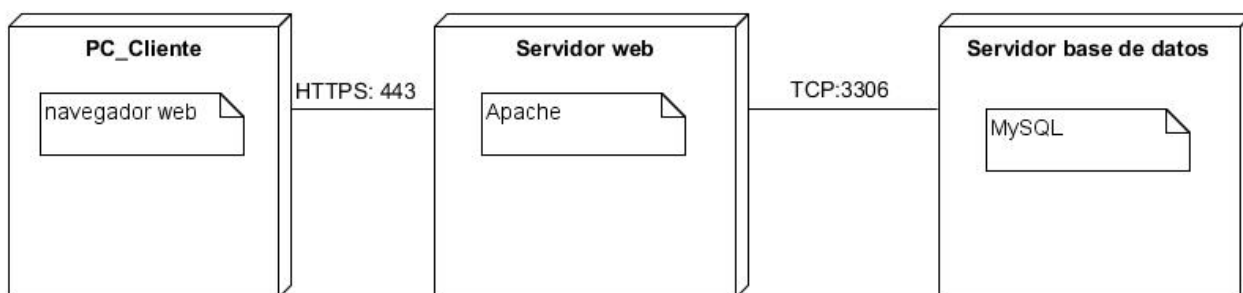


Figura 5. Diagrama de Despliegue.

Fuente: elaboración propia

PC_Cliente: Contiene la estación de trabajo cliente que necesita un navegador Chrome versión 83.0 o superior; o Mozilla Firefox versión 84.0 o superior que conecte con el servidor web

(*Apache*) mediante computadora personal, Tablet o teléfono celular, el cual utiliza para conectarse el protocolo de comunicación HTTPS (*Protocolo Seguro de Transferencia de Hipertexto*), tiene presente un Sistema operativo GNU/Linux Ubuntu 20.04 o superior, o Windows 10 Home. Requiere una tarjeta de red con velocidad de 100 Mbps, 2 GB de memoria RAM como mínimo y un procesador de 2 GHZ como mínimo.

Servidor Web: Aquí se encuentra el código fuente de la aplicación, brinda a los usuarios las interfaces necesarias durante todos los procesos del negocio. Esta estación se comunica con el servidor de base de datos MySQL donde se almacenan los datos de las prácticas profesionales, realizando la comunicación a través del protocolo TCP:3306. El servidor de aplicaciones web debe poseer una capacidad mínima de 80 GB como mínimo 2 GB de memoria RAM, requieren de una tarjeta de red con velocidad de 100 Mbps, un procesador de 2 GHZ como mínimo y el Servidor Web Apache con PHP 8.1.3

Servidor de BD: Este nodo es el encargado del almacenamiento de los datos de las prácticas profesionales. Se comunica con el servidor de aplicaciones a través del protocolo TCP, permite el acceso a la información que tanto el administrador como la aplicación pueden manipular, es el resultado almacenado de las iteraciones del software con una capacidad mínima de 20 GB, requieren de una tarjeta de red con velocidad de 100 Mbps, 2 GB de memoria RAM como mínimo, un procesador de 2 GHZ como mínimo y MySQL 8.0 como gestor de bases de datos.

2.3.3 Patrones de diseño de software

Los patrones de diseño (*design patterns*) son elementos reutilizables creados para resolver problemas comunes. Con su uso podremos corregir diferentes problemas que presenta nuestro código de una manera segura, estable y testada por cientos de programadores de todo el mundo (Soto, 2021). A continuación, se muestran los patrones de diseño utilizados en la propuesta de solución.

Patrones GRASP:

Los patrones GRASP (*General Responsibility Assignment Software Patterns*, Patrones de software de asignación de responsabilidad general) son un método de enseñanza que ayuda a entender el diseño de la programación orientada a objetos. Entre los más usados destacan los patrones Creador, Controlador, Experto, Bajo Acoplamiento y Alta Cohesión. A continuación, se

muestra en las siguientes figuras los patrones GRASP utilizados en el sistema de gestión de la información de las PP en CESOL.

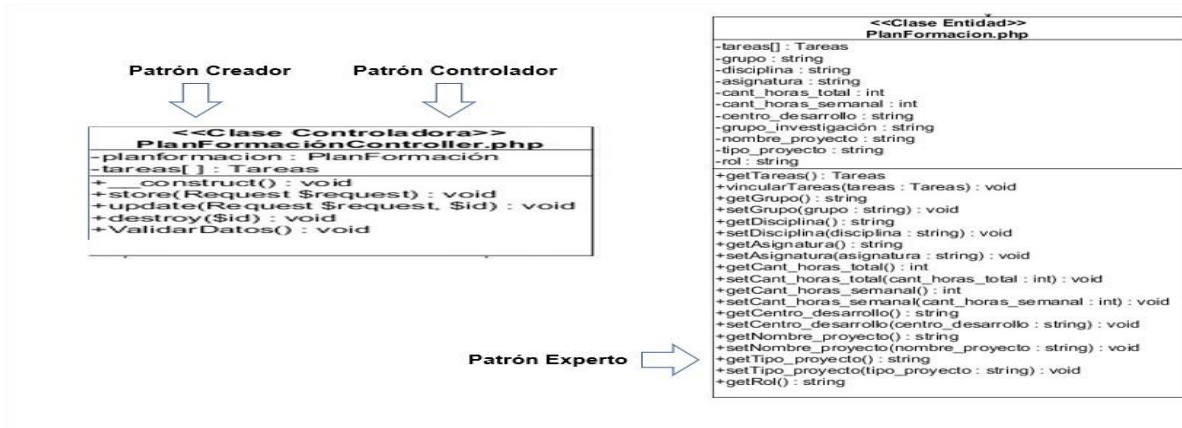


Figura 6. Patrones de diseño GRASP

Fuente: elaboración propia



Figura 7. Patrones de diseño GRASP

Fuente: elaboración propia

- En la clase *PlanFormacion.php* se evidencia el uso del patrón Experto ya que contiene las funcionalidades necesarias para acceder a la información de un plan de formación.
- La clase *PlanFormacionController.php* está presente el patrón Creador ya que es la única clase que posee una funcionalidad para crear un plan de formación mediante el método store.

- El patrón Controlador se observa en la clase *PlanFormacionController.php* ya que es responsable de gestionar la información relacionada a un plan de formación, provenientes del cliente.
- El patrón Alta Cohesión se evidencia en la clase *Estudiantes.php* porque es responsable solo de ejecutar sus funcionalidades y de poca complejidad.
- El patrón Bajo Acoplamiento se visualiza en la clase *Set.php* ya que posee poca dependencia con las demás clases.

Patrones GOF

Los patrones GoF (*Gang of Four*, Pandilla de Cuatro) describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos y se agrupan en tres grandes categorías: creacionales, estructurales y de comportamiento (Pressman, 2010).

Factory: Ayuda a crear instancias de otras clases, con la finalidad de ocultar la complejidad que se requiere para crearlas. Uno de los casos en el cual se debería utilizar Factory en Laravel es cuando el objeto que necesitamos crear, puede cambiar y afectar su utilización, por lo que en cada modelo se utiliza el uso de HasFactory ayudando a que algún cambio en los atributos de esta clase no afectara en el controlador en el cual será llamado. Este patrón de diseño se encuentra en el Modelo de nuestro sistema.

Use HasFactory;

Patrón decorador: Añade responsabilidades adicionales a un objeto dinámicamente proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades (González Rodríguez, 2017). Se observa su uso en la propuesta de solución en todas las vistas, ya que todas heredan de una plantilla padre determinada. En el presente segmento de código fuente se ejemplifica lo antes descrito:

- En la vista estudiantes en los archivos *create.blade.php*, *edit.blade.php*, *index.blade.php*, *show.blade.php* y *home.blade.php*:

```
@extends('adminlte::page')
```

2.3.4 Modelado de datos

Un modelo de datos es un esquema que detalla las expresiones permitidas por el propio modelo, anuncia las reglas y definiciones esenciales de los datos a los usuarios. Describe la semántica a través de tablas representada por una tecnología de manipulación de datos tal como el lenguaje SQL. Lo conciertan entidades, atributos y relaciones donde se pueden realizar un conjunto de operaciones que permiten especificar consultas y actualizaciones de la base de datos (Toledo Rodríguez, 2019).

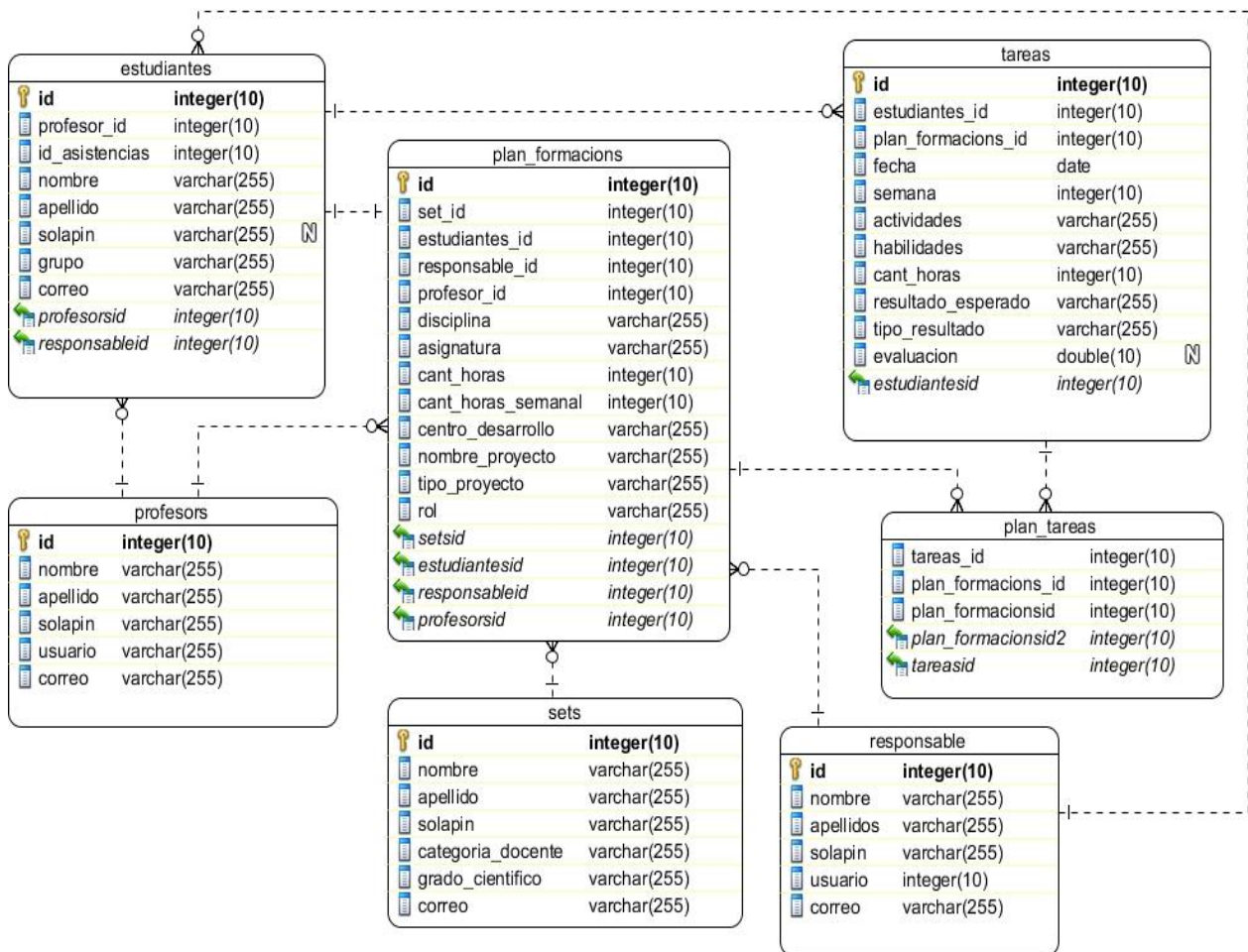


Figura 8. Modelado de datos.

Fuente: elaboración propia

El modelo anterior muestra de forma gráfica los datos que se capturan en el sistema, así como las relaciones existentes entre ellos. Refleja cómo serán guardados los datos persistentes del negocio informatizado, en la base de datos.

2.3.5 Diseño arquitectónico

El diseño arquitectónico consiste en la estructura o sistema de estructuras que comprenden los elementos de software, las propiedades externas visibles de esos elementos y la relación entre ellos. 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. Es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y los principios que orientan su diseño y evolución (Ospino Pinedo, 2018).

Patrón arquitectónico MVC

El patrón arquitectónico Modelo Vista Controlador (*Model View Controller, MVC*) separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos de forma que las modificaciones al componente de la vista, o a cualquier parte del sistema puedan ser hechas con un mínimo impacto en el componente del modelo de datos o en los otros componentes del sistema (Espitia, Armao y Carbajo, 2016).

Los tres principales componentes del patrón MVC son:

- **Modelo:** Representa los datos que el usuario está esperando ver y que persisten después de la ejecución del sistema informático.
- **Vista:** Se encarga de transformar el modelo para que sea visualizada por el usuario, ya sea un archivo de texto normal o en una página Web (*HTML* o *JSP*) que el navegador pueda desplegar. El propósito de la Vista es convertir los datos para que el usuario le sea significativos y los pueda interpretar fácilmente. La vista no debe trabajar directamente con los parámetros de request, debe delegar esta responsabilidad al controlador.
- **Controlador:** Es la parte lógica que es responsable de procesamiento y comportamiento de acuerdo a las peticiones (*request*) del usuario, construyendo un modelo apropiado y pasándolo a la vista para su correcta visualización.

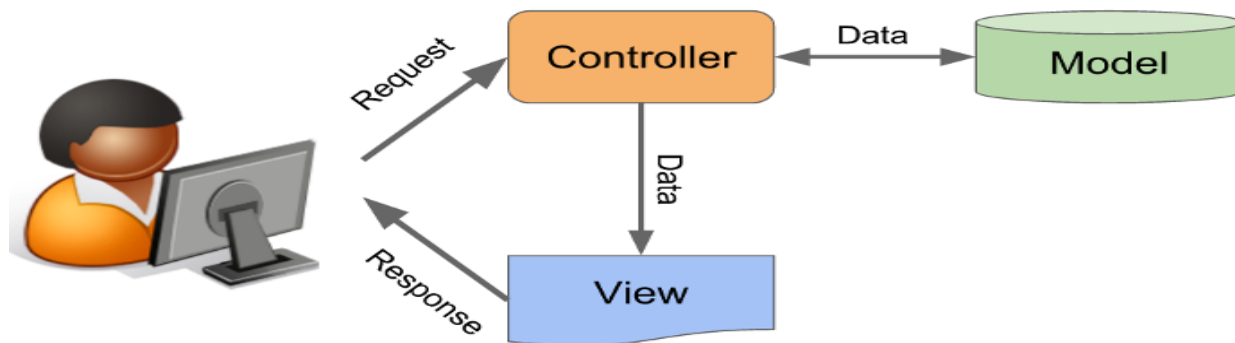


Figura 9. Patrón arquitectónico Modelo-Vista-Controlador

Fuente: gocnhinso.com, 2022

Conclusiones del capítulo

El análisis y diseño de la aplicación informática para la gestión de la información de la PP en CESOL permitió identificar 38 requisitos funcionales y 19 no funcionales, respecto a las necesidades del cliente. Se modela el sistema donde se incluye los atributos y relaciones entre clases, su forma y arquitectura de la propuesta de solución. El empleo del patrón arquitectónico MVC permitió validar la organización y la comunicación entre los diferentes componentes modelo, vista y controlador. Los patrones de diseño brindaron una mayor calidad y estructura al sistema informático.

CAPÍTULO 3: Implementación y evaluación de la aplicación informática para la gestión de información de la Práctica Profesional en el Centro de Software Libre

En el presente capítulo se describe el proceso de implementación de la aplicación informática para la gestión de la información de la PP en CESOL, además, se realiza la validación y evaluación del software a través del plan de pruebas.

3.1 Introducción

En el proceso de implementación se desarrolla cada funcionalidad del sistema y requerimiento solicitado por el cliente, las cuales deben funcionar como un todo. Esto tiene como propósito definir la organización del código, implementar clases, objetos y subsistemas en términos de componentes, los cuales se prueban y se vinculan a un sistema funcional. Durante el flujo de prueba se verifica la verdadera funcionalidad descrita y la satisfacción de los requisitos, arrojando un resultado de implementación (Suárez Peña, Broche Guevara y Alfaro Castro, 2016).

3.2 Diagrama de Componentes

El diagrama de componentes ilustra la relación que existe entre componentes de software, así como la ubicación de cada uno de ellos dentro del módulo y la implementación de las clases en término de componentes. Describe también como se organizan los componentes de acuerdo con los mecanismos de estructuración, modularizan el entorno de implementación y en el lenguaje de programación utilizado. Además, muestra las dependencias entre componentes (González Rodríguez, 2017).

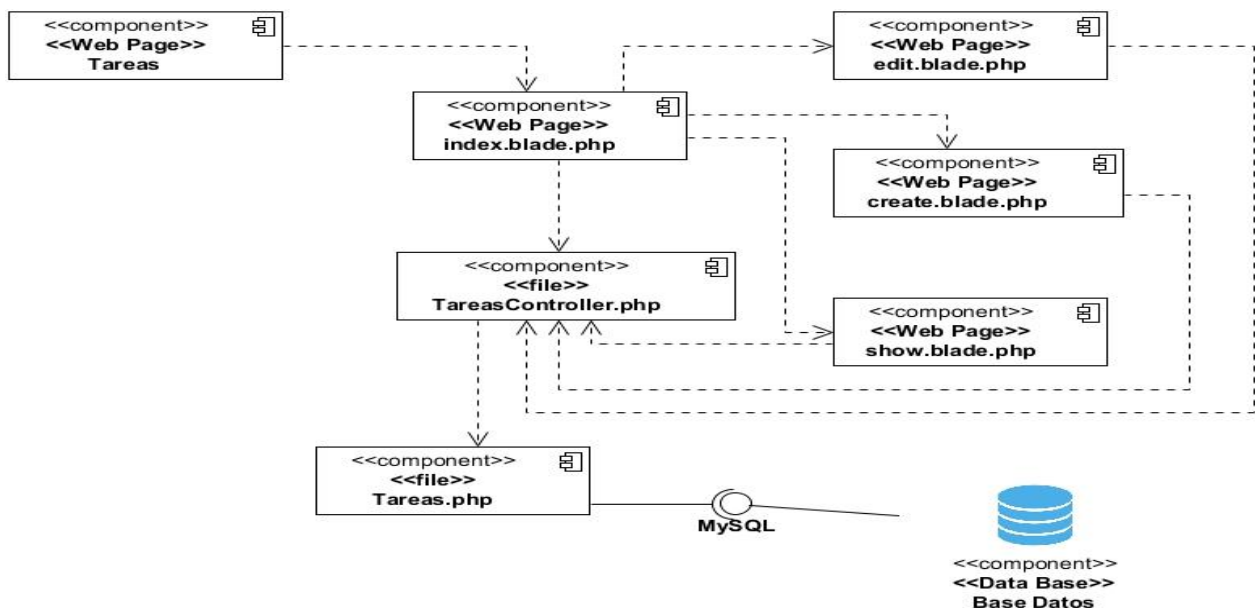


Figura 10. Diagrama de Componente Tareas.

Fuente: elaboración propia

A continuación, se describen los elementos que componen el diagrama de componentes mostrado:

Tabla 6. Descripción de Diagrama de componentes (Tareas).

Fuente: elaboración propia

Componentes		Descripción
Modelo	Tareas.php	Clase entidad que almacena las Tareas en el sistema.
Vistas	Tareas	Este componente se encarga de redireccionar desde la página principal de la aplicación informática hacia la vista donde se gestionan las Tareas.
	index.blade.php	Este componente se encarga de mostrar un listado de las Tareas registradas en la base de datos, así como mostrar las opciones necesarias para gestionar los Tareas.
	create.blade.php	Este componente se encarga de registrar una nueva Tarea en la base de datos cuando se ingresan correctamente los datos.
	edit.blade.php	Este componente se encarga de modificar una Tarea que el usuario previamente debe seleccionar.
	show.blade.php	Este componente se encarga de mostrar los detalles de una Tarea que el usuario previamente seleccionó.
Controlador	TareasController.php	Este componente se encarga de realizar todos los métodos relacionados a la gestión de las tareas del software

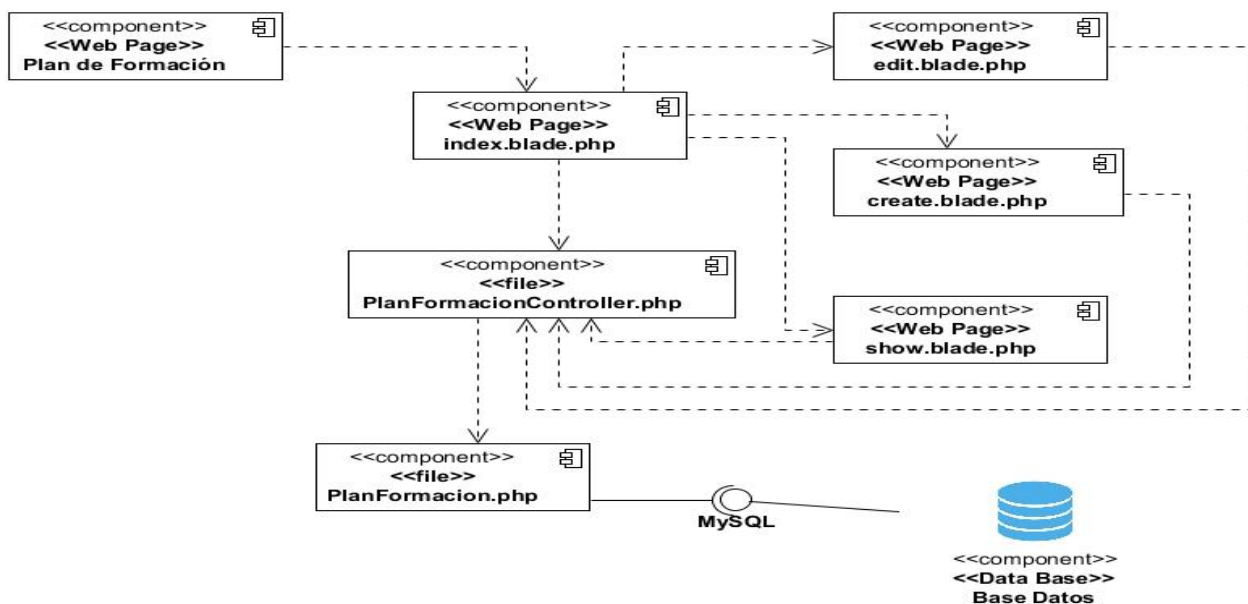


Figura 11. Diagrama de Componente Plan de Formación.

Fuente: elaboración propia

A continuación, se describen los elementos que componen el diagrama de componentes mostrado:

Tabla 7. Descripción de Diagrama de componentes (Gestionar Plan de Formación).

Fuente: elaboración propia

Componentes		Descripción
Modelo	PlanFormacion.php	Clase entidad que almacena los Planes de Formación en el sistema.
Vistas	Plan de Formación	Este componente se encarga de redireccionar desde la página principal de la aplicación informática hacia la vista donde se gestionan los Planes de Formación.

	index.blade.php	Este componente se encarga de mostrar un listado de los Planes de Formación registradas en la base de datos, así como mostrar las opciones necesarias para gestionar los Tareas.
	create.blade.php	Este componente se encarga de registrar una nuevo Planes de Formación en la base de datos cuando se ingresan correctamente los datos.
	edit.blade.php	Este componente se encarga de modificar un Planes de Formación que el usuario previamente debe seleccionar.
	show.blade.php	Este componente se encarga de mostrar los detalles de un Planes de Formación que el usuario previamente seleccionó.
Controlador	PlanFormacionController.php	Este componente se encarga de realizar todos los métodos relacionados a la gestión de los Planes de Formación del software

3.3 Estándares de codificación

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código. Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento (Guilarte Domínguez, 2019).

Nomenclatura según el tipo de clases

- ✓ **Clases controladora:** En estas clases después del nombre que identifica esta clase le sigue la palabra Controller. Un ejemplo de ello es la clase TareasController.
- ✓ **Clases de los modelos:** En estas clases que se encuentran dentro del negocio se representan sin un nombre que identifique el tipo de clase que es. Un ejemplo de ello es la clase Tareas.

Nomenclatura de las clases

- ✓ Se puede evidenciar que los nombres de cada clase comienzan con la primera letra en mayúscula auto seguido de letras en minúscula. Un ejemplo de ello es la clase PlanFormacion.

Estándar para Comentarios

- ✓ Los comentarios de una línea para aclaraciones del código aparecerán seguidos de los caracteres “//” en caso de código JavaScript y deben ubicarse en la misma línea que se desea comentar.
- ✓ ¡Los comentarios en caso de código HTML aparecerán dentro de estos caracteres <! -- -- > y pueden ubicarse donde desee el desarrollador
- ✓ Los comentarios deben ser oraciones completas y en caso de ser corto puede omitirse el punto final.
- ✓ Los comentarios de varias líneas en caso de JavaScript o PHP aparecerán dentro de los siguientes caracteres /* */.

Uso de nombres entendibles

- ✓ Use palabras entendibles y descriptivas para nombrar a las variables, no utilice abreviaciones.

3.4 Validación de la propuesta de solución

Tabla 8. Validación de la propuesta de solución

Fuente: elaboración propia

Tipo de prueba	Técnica de prueba	Validación
----------------	-------------------	------------

Funcional	Casos de prueba (Caja Negra)	Valida las funcionalidades diseñadas para el sistema.
Seguridad	Software Acunetix	Valida la confidencialidad, integridad y disponibilidad de los datos en el sistema.
Pruebas Unitarias	Casos de prueba (Caja Blanca)	Valora el código de la aplicación y su funcionalidad y desempeño

3.4.1 Pruebas Funcionales

Las pruebas funcionales se basan en la comprobación de los resultados de la ejecución de las funcionalidades que se han diseñado previamente. Para llevarlas a cabo, hay que diseñar un plan de pruebas que se pueda ejecutar y revisar a lo largo de las validaciones que se realicen en la funcionalidad entregada. Se divide en una serie de etapas nos aportarán la correcta realización de las mismas (QALovers, 2022):

- ✓ **Analizar requisitos y documentación**

En esta etapa, se revisa toda la documentación de las funcionalidades a probar. Una vez que se ha revisado todo, se realiza un esqueleto de plan de pruebas y cuando los requisitos están finalizados, se completa.

- ✓ **Diseño del plan de prueba**

Cuando concluye la revisión de la documentación, se diseña y especifica las pruebas que garantizarán las funcionalidades. Las pruebas tienen que ser claras y concisas y sobre todo determinar una cobertura total.

- ✓ **Ejecución del plan de pruebas**

Se ejecuta el plan de pruebas que se ha diseñado anteriormente. Para ello, hay que seguir cada paso de los casos de prueba y reportar el resultado del mismo.

- ✓ **Gestión de defectos**

Una vez que se ejecuta el plan de pruebas, aparecerán unos defectos que serán solucionados por las personas que han realizado la funcionalidad entregada.

Escenario	Nombres	Apellidos	Solapín	Grupo	Correo	Respuesta del sistema	Flujo central
1-Insertar un estudiante con datos correctos	Roberto Javier	Ramírez Carreño	E178598	FI1501	rjramirez@estudiantes.uci.cu	El estudiante se registró con éxito	1.Selecciona el botón "Insertar estudiante"
2- Insertar un estudiante con campos vacíos	Sandra	Barlet		FI1301		Existen campos vacíos	1.Selecciona el botón "Insertar estudiante"
3- Insertar un estudiante existente	Roberto Javier	Ramírez	E178598	FI1501	rjramirez@estudiantes.uci.cu	El estudiante ya existe	1.Selecciona el botón "Insertar estudiante"
4- Insertar un estudiante con datos incorrectos	Lían	Fernandez	567	12	lian	Los datos son incorrectos	1.Selecciona el botón "Insertar estudiante"

Figura 12.Caso de Prueba Insertar Estudiante.

Fuente: elaboración propia

Descripción de las variables. Caso de Prueba Insertar Estudiante.

Tabla 9. Descripción de las variables. Caso de Prueba Insertar Estudiante.

Fuente: elaboración propia

No	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
2	Apellidos	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
3	Solapín	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

4	Usuario	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
5	Correo	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

Escenario	Nombres	Apellidos	Solapín	Grupo	Correo	Respuesta del sistema	Flujo central
1-Insertar un profesor con datos correctos	Lázaro Paul	Fernandez	E124568	FI1501	lpfernandez@uci.cu	El profesor se registró con éxito	1.Selecciona el botón "Insertar profesor"
2- Insertar un profesor con campos vacíos	Alejandro	Rodríguez		FI1301		Existen campos vacíos	1.Selecciona el botón "Insertar profesor"
3- Insertar un profesor existente	Lázaro Paul	Fernandez	E124568	FI1501	lpfernandez@uci.cu	El profesor ya existe	1.Selecciona el botón "Insertar profesor"
4- Insertar un profesor con datos incorrectos	Lían	Fernandez	567	12	lian	Los datos son incorrectos	1.Selecciona el botón "Insertar profesor"

Figura 13. Caso de Prueba Insertar Profesor.

Fuente: elaboración propia

Descripción de las variables. Caso de Prueba Insertar Profesor.

Tabla 10. Descripción de las variables. Caso de Prueba Insertar Profesor.

Fuente: elaboración propia

No	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
2	Apellidos	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

3	Solapín	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
4	Usuario	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto
5	Correo	Campo de texto, radio button, campo de selección	No	El campo de texto solo admite valores de texto

Las pruebas funcionales se realizaron en cuatro (4) iteraciones a medida que se ejecutaba la aplicación informática, las cuales determinaron la calidad del software. Como resultado final, se obtuvo en una primera iteración un total de diez (10) no conformidades, divididas en cuatro (4) de ortografía, dos (2) de redacción, dos (2) de funcionalidad y dos (2) de validación. De estas, se resolvieron cinco (5), y cinco (5) quedaron pendientes. En una segunda iteración, no se identifican nuevas no conformidades y de las cinco (5) pendientes, se solucionan cuatro (4) quedando una (1) pendiente. En la tercera iteración la pendiente se resolvió satisfactoriamente. La siguiente gráfica, muestra los resultados antes descritos:

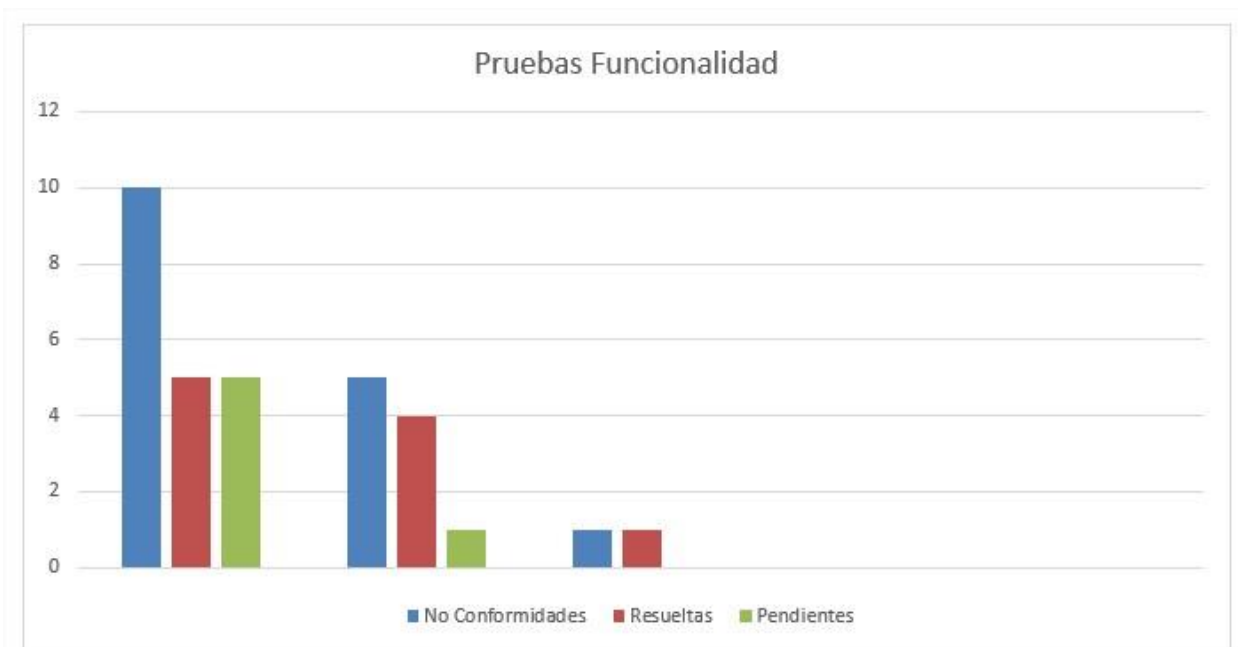


Figura 14. Resultado de las pruebas funcionales.

Fuente: elaboración propia

3.4.2 Pruebas de Seguridad

Cualquier sistema informático es un blanco de penetración inadecuada o ilegal. Esta abarca un amplio rango de actividades: hackers que intentan penetrar en las aplicaciones informáticas en busca de información o por diversión. Para contrarrestar los ataques o minimizar los daños se realizan pruebas de seguridad a los sistemas informáticos, las mismas intentan verificar que los mecanismos de protección que se construyen en un sistema en realidad lo protegerán de cualquier penetración impropia. Durante la prueba de seguridad, quien la realiza juega un papel importante ya que se introduce al sistema para intentar adquirir contraseñas por medios administrativos externos ataca con software a la medida diseñado para romper cualquier defensa que se haya construido, abrumba al sistema, y por tanto negar el servicio a los demás; causando a propósito errores con la esperanza de penetrar durante la recuperación (Pérez Bejerano, 2019).

Con el objetivo de realizar una satisfactoria prueba de seguridad de la propuesta de solución desarrollada, se utilizó la herramienta *Acunetix Web Vulnerability Scanner*. Durante la primera iteración realizada, la aplicación detectó 15 no conformidades, las cuales se dividen en cuatro secciones: nivel alto (0 no conformidades), nivel medio (10 no conformidades), nivel bajo (3 no conformidades) y de carácter informal (2 no conformidades). Las de nivel medio se deben al uso de protocolos no seguros para el envío de datos, y el resto se deben al envío de mensajes de

mensajes de error utilizados por el *framework* utilizado en la propuesta de solución. Las de nivel bajo, están relacionadas generalmente a los posibles ataques a la aplicación informática y redirecciones que pueden surgir que el usuario está dentro del sistema. Se obtuvo varios errores de tipo Informativo, correspondientes a los campos de texto de las contraseñas y verificar contraseña.

Todas las no conformidades encontradas en la primera iteración de la aplicación informática fueron corregidas, logrando que en la segunda iteración no se encontrara ninguna deficiencia en la seguridad del sistema, permitiendo su uso y funcionamiento de una manera correcta. A continuación, en la siguiente figura se pueden observar todos los resultados antes descritos:



Figura 15. Resultado de las pruebas de seguridad.

Fuente: elaboración propia

3.4.3 Pruebas Unitarias

Las pruebas unitarias se realizaron a través del método de prueba caja blanca y la técnica del camino básico. Para aplicar la prueba de camino básico se debe realizar un análisis de la complejidad ciclomática de cada procedimiento que componen las clases del sistema. El procedimiento de mayor valor tiene una alta probabilidad de contener errores, además de que ofrece una medida del número de pruebas que deben diseñarse para validar la correcta

implementación de una determinada función (Pressman, 2010). En la figura 19 se presenta el método store () correspondiente al RF20. Insertar Plan de Formación.

```

56 public function store(Request $request)
57 {
58     $rules=[
59         'estudiantes_id'=>'required|unique:plan_formacions',
60         'set_id'=>'required',
61         'responsable_id'=>'required',
62         'profesor_id'=>'required',
63         'tareas_id'=>'required',
64         'disciplina'=>'required',
65         'asignatura'=>'required',
66         'cant_horas'=>'required',
67         'cant_horas_semanal'=>'required',
68         'centro_desarrollo'=>'required',
69         'grupo_investigacion'=>'required',
70         'nombre_proyecto'=>'required|unique:plan_formacions',
71         'tipo_proyecto'=>'required',
72         'rol_desempena'=>'required',
73     ];

74     $message=[
75         'estudiantes_id.required'=>'Este campo es obligatorio',
76         'estudiantes_id.unique'=>'El estudiante ya tiene plan de formación asignado',
77         'set_id.required'=>'Este campo es obligatorio',
78         'responsable_id.required'=>'Este campo es obligatorio',
79         'profesor_id.required'=>'Este campo es obligatorio',
80         'tareas_id.required'=>'Este campo es obligatorio',
81         'disciplina.required'=>'Este campo es obligatorio',
82         'asignatura.required'=>'Este campo es obligatorio',
83         'cant_horas.required'=>'Este campo es obligatorio',
84         'cant_horas_semanal.required'=>'Este campo es obligatorio',
85         'centro_desarrollo.required'=>'Este campo es obligatorio',
86         'grupo_investigacion.required'=>'Este campo es obligatorio',
87         'nombre_proyecto.required'=>'Este campo es obligatorio',
88         'nombre_proyecto.unique'=>'El nombre del proyecto ya esta en uso',
89         'tipo_proyecto.required'=>'Este campo es obligatorio',
90         'rol_desempena.required'=>'Este campo es obligatorio',
91     ];
92
93     $this->validate($request,$rules,$message);
94     $plan=PlanFormacion::create($request->all());
95     $plan->tareas()->sync($request->get('tareas_id'));
96     return redirect('/plan')->with('info','adicionar-plan');
97 }
    
```

Figura 16. Método store () correspondiente al RF20. Insertar Plan de Formación.

Fuente: elaboración propia

1. Dibujar el grafo de flujo de la funcionalidad

En la figura 20 se utilizó la notación de grafo de flujo para representar el código de la validación del formulario para insertar estudiante:

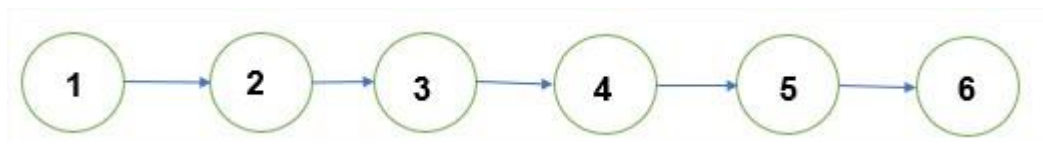


Figura 17. Grafo de flujo.

Fuente: elaboración propia

2. Determinar la complejidad ciclomática

$$V(G) = A - N + 2$$

$$V(G) = 5 - 6 + 2$$

$$V(G) = 1$$

A: Número de aristas del grafo de flujo.

N: Cantidad de nodos del grafo.

3. Determinar los caminos linealmente dependientes

Camino básico 1: 1, 2, 3, 4, 5, 6

4. Definir los casos de prueba para comprobar la ejecución

En la tabla se presenta el diseño de caso de prueba del camino 1 correspondientes a la funcionalidad store ():

Tabla 11. Caso de prueba para el camino 1 de la función store ()

Fuente: elaboración propia

Diseño de caso de prueba para el camino 2	
Descripción	Función para crear un nuevo plan de formación.
Condiciones	El administrador (Profesor PID) selecciona la opción de adicionar un nuevo plan de formación, verificando antes si existe ya insertado un estudiante al cual se le asigna el plan de formación, el Responsable de PID, un SET, el profesor de la PID y las tareas correspondientes.
Entradas	Object: {

	<p>id: "1",</p> <p>estudiantes_id: "1",</p> <p>responsablePID_id: "1",</p> <p>profesor_id: "1",</p> <p>disciplina: "1",</p> <p>asignatura: "PID IV",</p> <p>cant_horas: "2",</p> <p>cant_horas_semanal: "3",</p> <p>centro_desarrollo: "CESOL",</p> <p>grupo_investigacion: "CESOL",</p> <p>nombre_proyecto: "Sistema de Gestión de Activos en CESOL",</p> <p>tipo_proyecto: "Tesis",</p> <p>rol_desempeña: "Programador",</p>
Resultados	<p>Crea un nuevo plan de formación y se incluye en el listado de planes de formaciones ya creados.</p>

Luego de realizadas las pruebas unitarias se obtuvo como resultado que el flujo de trabajo de las funcionalidades de la aplicación informática para la gestión de la información de las PP en CESOL es correcto; pues se comprueba que cada sentencia del código fuente se ejecuta al menos una vez.

3.5 Evaluación de la satisfacción del cliente sobre la propuesta de solución

La satisfacción del cliente se ha transformado en uno de los objetivos más importantes dentro del sumario de desarrollo de software en la validación de cualquier investigación científica. La investigación brindada resulta útil para conocer las fortalezas y debilidades de la propuesta realizada (Barroso Rodríguez, 2017). Con la finalidad de determinar el índice de satisfacción

personal y grupal de los usuarios sobre la aplicación informática de gestión de las prácticas profesionales en el Centro de Software Libre, se aplica la Técnica de V. A. Iadov.

Esta técnica establece una vía para el estudio de la satisfacción en dependencia del campo donde se aplique, ya que los criterios que se utilizan se fundamentan en las relaciones que se establecen entre tres preguntas que se intercalan dentro de un cuestionario y cuya relación el sujeto desconoce. Estas preguntas permiten ubicar al o los encuestados en una escala de satisfacción tomando valores del -1 a 1 de la siguiente forma (Barroso Rodríguez, 2017):

- +1. Máximo de satisfacción
- 0,5. Más satisfecho que insatisfecho
- 0. No definida
- 0,5. Más insatisfecho que satisfecho
- 1. Máxima insatisfacción

Para conocer el grado de satisfacción de los usuarios de la propuesta de solución se emplea una encuesta (ver Anexo 3) a 15 SET relacionados al proceso de prácticas profesionales en el Centro de Software Libre. A continuación, en la tabla 15 se evidencia la relación entre las preguntas del cuestionario.

Tabla 12. Cuadro lógico de Iadov

Fuente: elaboración propia

11. ¿Qué opina acerca de los beneficios que trae consigo la aplicación para la gestión de la información de la Práctica Profesional en el Centro de Software Libre?	9. ¿Actualmente se satisface la gestión de las prácticas profesionales de manera correcta en el Centro de Software Libre?								
	No			No Se			Sí		
	10. ¿Considera usted que es necesario contar con una aplicación informática para la gestión de la información de la Práctica Profesional en el Centro de Software Libre?								
	Sí	No Sé	No	Sí	No Sé	No	Sí	No Sé	No
Me gusta mucho	4	2	3	3	2	1	6	6	1

Me gusta más de lo que me disgusta	4	6	1	1	6	3	1	3	2
Me da lo mismo	3	5	6	4	5	4	5	2	3
Me disgusta más de lo que me gusta	2	4	4	6	1	6	3	1	6
No me gusta	1	5	3	5	2	3	2	4	5
Me da lo mismo	6	1	5	2	4	1	4	5	1

El número de la interrelación de las tres preguntas indica la posición en el nivel de satisfacción siguiente: clara satisfacción (A), más satisfecho que insatisfecho (B), no definida (C), más insatisfecho que satisfecho (D) y clara insatisfacción (E). A partir de la cantidad de respuestas en cada categoría se puede calcular el Índice de Satisfacción Grupal (ISG) utilizando la siguiente fórmula:

$$ISG = \frac{A(+1)+B(+0.5)+C(0)+D(-0.5)+E(-1)}{N}$$

Donde N es la cantidad total de respuestas.

El valor del ISG permite identificar las siguientes categorías grupales:

- ✓ Máxima insatisfacción: desde -1 hasta -0.49
- ✓ Más insatisfecho que satisfecho: desde -0.5 hasta -0.1
- ✓ No definido y contradictorio: 0
- ✓ Más satisfecho que insatisfecho: desde 0.1 hasta 0.49
- ✓ Máximo de satisfacción: desde 0.5 hasta 1

Los valores que se encuentran comprendidos entre -1 y -0.5 indican insatisfacción; los comprendidos entre -0.49 y +0.49 evidencian contradicción y los que están entre 0.5 y 1 indican que existe satisfacción (Fabre, Sánchez Ortega y Reyes Farrat, 2019).

Resultados obtenidos

Tabla 13. Resultados de la escala de satisfacción

Fuente: elaboración propia

Categorías grupales de satisfacción	N =15	Escala
Clara satisfacción	10	A
Más satisfecho que insatisfecho	5	B
Indefinido	0	C
Más insatisfecho que satisfecho	0	D
Máxima insatisfacción	0	E

Cálculo del Índice de Satisfacción Grupal (ISG)

$$ISG = A (+1) + B (+0.5) / 15$$

$$ISG = (10 (+1) + 5 (+0.5)) / 15$$

$$ISG = 0.83$$

El valor obtenido del ISG fue de 0.83, lo que indica que entre los encuestados existe la mayor satisfacción con respecto a la aplicación informática para la gestión de la información de las prácticas profesionales en el Centro de Software Libre. A través de las respuestas conseguidas, se pudieron ratificar los beneficios que trae consigo la solución desarrollada. Por lo que se puede afirmar que se cumplió el objetivo de la investigación.

CONCLUSIONES GENERALES

La presente investigación documenta el proceso de desarrollo de la aplicación informática para la gestión de información de la Práctica Profesional en CESOL donde se arriba a las siguientes conclusiones:

- El análisis de los referentes teóricos sobre el proceso de gestión de la Práctica Profesional, así como el estudio de sistemas homólogos, demostró la necesidad de desarrollar un sistema informático que contribuya con la gestión de información de la Práctica Profesional en el Centro de Software Libre.
- Los 38 requisitos funcionales y 19 no funcionales definidos responden a las exigencias del cliente. El análisis y diseño propició la elaboración de un sistema informático para la gestión de información de la Práctica Profesional en CESOL. El uso del patrón arquitectónico MVC y los patrones de diseño GRASP, brindó una mayor calidad al software desarrollado.
- Se obtuvo la implementación de 38 funcionalidades del sistema informático para la gestión de información de la Práctica Profesional en CESOL, que cumple con los requisitos de software determinados en correspondencia de las necesidades del cliente y las restricciones del diseño desarrollado.
- Las pruebas de software permitieron evaluar las funcionalidades del sistema informático para la gestión de información de la Práctica Profesional en CESOL implementadas y garantizar su correcto funcionamiento.

RECOMENDACIONES

Se recomienda implementar en la aplicación informática el sistema de pasarela de la UCI, para la autenticación al sistema de los estudiantes. Además de extender el uso del sistema para la totalidad de centros de desarrollo existentes en la universidad.

REFERENCIAS BIBLIOGRÁFICAS

- Albornoz, Berón y Montejano. (2017). *Interfaz Gráfica de Usuario: El Usuario como Protagonista del Diseño*.
http://sedici.unlp.edu.ar/bitstream/handle/10915/62078/Documento_completo.pdf?sequence=1
- Asamblea Nacional del Poder Popular. (2016). *Lineamientos 2016-2021 Versión Final*.
<https://misiones.cubaminrex.cu/es/articulo/lineamientos-2016-2021-version-final>
- Baró, C. A. (2019). *Sistema para la gestión del banco de problemas científicos de la Facultad 1*.
<https://repositorio.uci.cu/jspui/handle/123456789/9998>
- Barroso Rodríguez. (2017). *Sistema de apoyo al diagnóstico a distancia de enfermedades genéticas basado en mapas cognitivos difusos*. *Revista Ciencias Médicas de Pinar del Rí*o. <http://www.revcmpinar.sld.cu/index.php/publicaciones/article/view/3135>
- Benítez Jiménez, E. (2012). *Aplicaciones Informáticas*.
<https://elisainformatica.files.wordpress.com/2012/11/aplicaciones-informc3a1ticas.pdf>
- Borges, E. (2019, marzo 17). *Servidor Base de Datos: ¿Qué es? Funciones, Tipos y Ejemplos*. Infranetworking. <https://blog.infranetworking.com/servidor-base-de-datos/>
- Calermany Universidad. (2020, abril 6). *Metodologías de desarrollo de software | Universitat Carlemany*. <https://www.universitatcarlemany.com/actualidad/metodologias-de-desarrollo-de-software>
- Canchari Cuyutupac, L. (2018). *LA VALIDACIÓN DE REQUISITOS DE SOFTWARE COMO BASE DEL ÉXITO DE LOS PROYECTOS DE SISTEMAS INFORMÁTICOS DESARROLLADOS E IMPLEMENTADOS EN LA COMISIÓN NACIONAL PARA EL DESARROLLO Y VIDA SIN DROGAS - DEVIDA [PARA OPTAR EL GRADO ACADÉMICO DE: MAESTRO EN INGENIERÍA DE SISTEMAS, UNIVERSIDAD NACIONAL DEL CENTRO DEL PERÚ]*.
https://repositorio.uncp.edu.pe/bitstream/handle/20.500.12894/5002/T010_20089835_M.pdf?sequence=1&isAllowed=y
- Casas, V. (2019, octubre 28). *¿Qué es Bootstrap y para qué sirve?* *Blog de LucusHost*.
<https://www.lucushost.com/blog/que-es-bootstrap/>
- caseware. (2022). *Sistema de Gestión Académica. CASEWARE - TECNOLOGÍA*.
<http://www.casewaresa.com.co/sistema-gestion-academica/>

Referencias bibliográficas

- Cordova Quispe, J. A. (2018). *Herramientas de programación y su efecto en el aprendizaje del lenguaje de programación Visual Basic en estudiantes del 5to grado de educación secundaria del colegio nacional de Vitarte UGEL N° 06 del distrito de ATE-2018* [Tesis]. <https://repositorio.une.edu.pe/bitstream/handle/20.500.14039/6282/TEISIS%20-%20CORDOVA%20QUISPE%20JUNIOR%20ALEXIS%20-%20FAC.pdf?sequence=1&isAllowed=y>
- Costa, G. I. (2007). Akademos, un Sistema Automatizado para la Gestión Académica. *Serie Científica de la Universidad de las Ciencias Informáticas*, 1(1), Article 1. <https://publicaciones.uci.cu/index.php/serie/article/view/253>
- de Souza, I. (2019, noviembre 23). Software libre: Definición, tipos y ventajas de esos programas. *Rock Content - ES*. <https://rockcontent.com/es/blog/software-libre/>
- Delgado, C. (2021, marzo 29). *¿Qué es un IDE (entorno de desarrollo integrado)?* Our Code World. <https://ourcodeworld.co/articulos/leer/1469/que-es-un-ide-entorno-de-desarrollo-integrado>
- Edcon. (2022). *Sistema Escolar Edcon*. www.edcon.com.mx
- EduColombia. (2022). *Software para Colegios 1290. Sistema Integral para la Gestión educativa*. <https://educolombia.org/>
- Espitia, Armao y Carbajo. (2016). *MODELO VISTA-CONTROLADOR (MVC)*. <https://espejodeantagno.files.wordpress.com/2016/04/modelo-vista-controlador-mvc.pdf>
- Fabre, Sánchez Ortega y Reyes Farrat. (2019). El proceso de validación mediante la Técnica de ladov en cursos por encuentros. *Revista Ingeniería Agrícola*, 10(1). <https://www.redalyc.org/journal/5862/586262449010/html/>
- Falcón, Pertile y Ponce. (2019). *La encuesta como instrumento de recolección de datos*. https://www.memoria.fahce.unlp.edu.ar/trab_eventos/ev.13544/ev.13544.pdf
- Ferguson, K. (2021). *¿Qué es Servidor de aplicaciones y proveedor de servicios de aplicaciones? - Definición en WhatIs.com*. ComputerWeekly.es. <https://www.computerweekly.com/es/definicion/Servidor-de-aplicaciones-y-proveedor-de-servicios-de-aplicaciones>
- Fernández, Y. (2021, enero 13). *Qué es el HTML5 y qué novedades ofrece*. Xataka. <https://www.xataka.com/basics/que-html5-que-novedades-ofrece>
- Ferré Grau y Sánchez Segura, X. y I. (2011). *Desarrollo Orientado a Objetos con UML*. <https://docplayer.es/131743-Desarrollo-orientado-a-objetos-con-uml.html>
- García Peñalvo y Pardo Aguilar, F. J. y C. (2018). *Un lenguaje de modelado estándar para los métodos de ADOO*.

Referencias bibliográficas

- https://gredos.usal.es/bitstream/handle/10366/122032/DIA_GarciaPenalvo_PardoAguilar_uml11.pdf;jsessionid=CA36F05CC44508E722C0202CEACE1833?sequence=1
- Geek. (2020, diciembre 2). *Qué es un Entorno de desarrollo integrado*. IfgeekthenNTTdata. <https://ifgeekthen.nttdata.com/es/que-es-un-entorno-de-desarrollo-integrado>
- González Rodríguez. (2017a). *Subsistema de gestión de perfil de usuario para el buscador Orión*. <https://repositorio.uci.cu/jspui/handle/123456789/7963>
- González Rodríguez. (2017b). *Subsistema de gestión de perfil de usuario para el buscador Orión*. https://repositorio.uci.cu/jspui/bitstream/123456789/7963/1/TD_08785_17.pdf
- Guilarte Domínguez, C. R. (2019). *Sistema para la gestión de procesos en la Dirección de Extensión Universitaria*. https://repositorio.uci.cu/jspui/bitstream/123456789/10152/1/TD_09567_19%20.pdf
- Gustavo, B. (2018, agosto 31). *¿Qué es Apache? Descripción completa*. Tutoriales Hostinger. <https://www.hostinger.es/tutoriales/que-es-apache/>
- Hoogenraad, W. (2018). *Casos de prueba, ejemplos y mejores prácticas*. <https://www.itpedia.nl/es/2018/06/01/testcases-voorbeelden-en-best-practices/>
- IESPOH. (2020). *Prácticas Profesionales*. INSTITUTO DE ESTUDIOS SUPERIORES DE PROGRESO DE OBREGÓN HIDALGO. <https://www.iespoh.com/home/practicas-y-servicio/practicas-profesionales/>
- Infomed. (2017, abril 16). *¿QUE ES GESTIÓN DE LA INFORMACIÓN?* <https://instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion/>
- Jiménez Bibián. (2017). *Pruebas de calidad aplicadas al sitio web Allison* [Tesis profesional]. <https://dspace.itcolima.edu.mx/bitstream/handle/123456789/725/OSCAR%20PAUL%20ruebas%20de%20Calidad%20Aplicadas%20al%20Sitio%20Web%20Allison.pdf;jsessionid=DB934EA9E173944AACCD42D86F35FC0B?sequence=1>
- KeepCoding, R. (2021, noviembre 25). *7 Tecnologías para Desarrollo Web | KeepCoding Tech School*. <https://keepcoding.io/blog/7-tecnologias-para-desarrollo-web/>
- la practica profesional en el contexto de la globalizacion—Brainly.lat*. (s. f.). Recuperado 4 de mayo de 2022, de <https://brainly.lat/tarea/12488990>
- Lago, N. (2022). *Seis herramientas de modelado de software*. <https://saasradar.net/modelado-de-software/>
- Lara Galicia, F. P. (2020). *Prototipo de interfaz de usuario: Qué es y qué herramientas hay para crearlo*. <https://mx.godaddy.com/blog/prototipo-interfaz-usuario-definicion-herramientas/>

Referencias bibliográficas

- León, A., & Andrés, G. (2017). *Diseño de un modelo conceptual para la creación de un centro gastronómico temático* [Tesis, Universidad Nacional de La Plata].
<https://doi.org/10.35537/10915/63415>
- León Horna, L. H. (2019). *LA GESTIÓN ACADÉMICO – ADMINISTRATIVA Y LA SATISFACCIÓN LABORAL DE LOS DOCENTES DE LA UNIVERSIDAD PRIVADA “ARZOBISPO LOAYZA”* [Tesis de Maestría en Gestión Educativa, UNIVERSIDAD PRIVADA ARZOBISPO LOAYZA].
http://repositorio.uigv.edu.pe/bitstream/handle/20.500.11818/3865/TESIS_MAEST_GEST.EDUCATIVA_Adolfo%20Jes%C3%BAs%20Le%C3%B3n%20Horna.pdf?sequence=2&isAllowed=y
- Muñoz, A. (2021). *Centralizar información de producto dispersa en muchas fuentes*.
<https://blog.saleslayer.com/es/centralizar-informacion-producto-dispersa>
- Ortega Díaz y Aguilera Santos. (2017). *Módulo para exportar datos desde el Sistema de Gestión de Ingreso a la Educación Superior (SIGIES)* [Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas, UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS].
https://repositorio.uci.cu/bitstream/123456789/8099/1/TD_08935_17.pdf
- Ospino Pinedo, M. (2018). *ARQUITECTURA DE SOFTWARE PARA LA GESTIÓN Y PROYECCIÓN FINANCIERA DE PLANES DE NEGOCIOS* [UNIVERSIDAD DE CARTAGENA].
<https://repositorio.unicartagena.edu.co/bitstream/handle/11227/8503/TESIS%20DE%20GRADO%20RUIZ%20HERN%C3%81NDEZ.pdf?sequence=1&isAllowed=y>
- Patterson Peña, Puentes Puentes y Martínez Leyet. (2015). La superación de actores que inciden en la formación del ingeniero en la Universidad de Ciencias Informáticas: Una perspectiva reflexiva. *Revista Cubana de Ciencias Informáticas*, 15.
- Pérez Bejerano, Y. (2019). *Sistema de Gestión de Licencias del Personal Aeronáutico del Instituto de la Aeronáutica Civil de Cuba versión 2.0*.
<https://repositorio.uci.cu/jspui/handle/123456789/10288>
- Pérez Jiménez, J. D. (2019, enero 20). *Qué es CSS3 y sus fundamentos*. OpenWebinars.net.
<https://openwebinars.net/blog/que-es-css3/>
- Pressman. (2010). *Ingeniería de software enfoque práctico* (7ma ed.).
- QALovers. (2022). *Introducción al Testing: Pruebas funcionales*.
<https://www.qalovers.com/2021/04/pruebas-funcionales.html>

Referencias bibliográficas

- Robledano, A. (2019, septiembre 24). *Qué es MySQL: Características y ventajas*. OpenWebinars.net. <https://openwebinars.net/blog/que-es-mysql/>
- Rodríguez Sánchez, T. (2015). *Metodología de desarrollo para la Actividad productiva de la UCI*. Salvador Santillan, G. (2019). *PRACTICA PRE PROFECIONAL*. <https://sites.google.com/site/gabysalvadorsantillan/practica-pre-profecional>
- Sanchez Quiróz, D. (2018, septiembre 11). *Qué es UML: Unified Modeling Language*. OpenWebinars.net. <https://openwebinars.net/blog/que-es-uml-unified-modeling-language/>
- Sarmiento Cuervo y Hernández. (2017). *METODOLOGÍA PARA LA OPTIMIZACIÓN DE LOS PROCESOS DE RECOLECCIÓN DE INFORMACIÓN Y ANÁLISIS EN LA ETAPA DE ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE* [Para optar el Grado Académico de Doctor en Ingeniería de Sistemas e Informática]. UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS.
- Sistacnet. (2022). *SISTAC & SISTACNET EN LA WEB*. <https://www.sistacweb.com/servicios/>
- Solomon, E. (2021, septiembre 9). *Novedades de Laravel 9: Una Inmersión Profunda en la Próxima Gran Versión*. Kinsta. <https://kinsta.com/es/blog/laravel-9/>
- Sommerville. (2011). *Ingeniería de Software—Ian Sommerville 7a Edición*. (7a Edición). https://www.academia.edu/15059886/Ingenieria_de_Software_Ian_Sommerville_7a_Edicion
- Soto, N. (2021, julio 2). *¿Qué son los patrones de diseño en programación? 2022*. Craft - Code | La Academia de las Buenas Prácticas. <https://craft-code.com/que-son-los-patrones-de-disenio/>
- Suárez Peña, Broche Guevara y Alfaro Castro. (2016). *Componente para la autorización, la administración de perfiles de usuario y la multi-institución en el marco de trabajo Xalix*. [UCI]. <https://repositorio.uci.cu/jspui/handle/ident/9020>
- Toledo Rodríguez, M. (2019). *Módulo para la gestión de copias de seguridad en Nova 360* [Universidad de las Ciencias Informáticas. Facultad 1.]. <https://repositorio.uci.cu/jspui/handle/123456789/10098>
- Torrens, R. E. P., & Arbolaez, G. de la C. U. (2020). *Guías didácticas en el proceso enseñanza-aprendizaje: ¿Nueva estrategia?* *Revista Científica*, 5(18), 371-392. <https://doi.org/10.29394/Scientific.issn.2542-2987.2020.5.18.20.371-392>
- UAN, M. D. (2022). *Acerca de la UAN*. Universidad Autónoma de Nayarit. <https://www.uan.edu.mx/es/universidad>

Referencias bibliográficas

- Uladech. (2017). *METODOLOGÍA DE DESARROLLO DE SOFTWARE*.
<https://www.uladech.edu.pe/images/stories/universidad/documentos/2018/metodologia-desarrollo-software-v001.pdf>
- Uniandes. (2022). *Universidad de los Andes – Colombia – Sitio oficial*. <https://uniandes.edu.co/>
- Unir. (2021, agosto 19). *Framework: Qué es, para qué sirve y algunos ejemplos*.
<https://www.edix.com/es/instituto/framework/>
- Unir. (2022). *Evaluación educativa: En qué consiste y cuál es su importancia*. Unir.
<https://www.unir.net/educacion/revista/evaluacion-educativa/>
- Universidad Francisco José de Caldas. (2022). *Definición de grupo de investigación—Universidad Distrital Francisco José de Caldas*. <http://www1.udistrital.edu.co:8080/web/unidad-de-investigaciones-de-la-facultad-tecnologica/definicion-de-grupo-de-investigacion>
- Velasco, R. (2021, mayo 26). *Visual Studio Code: Editor de texto de código abierto para programar*. SoftZone. <https://www.softzone.es/programas/utilidades/visual-studio-code/>
- Visual Paradigm. (2022). *Compare Visual Paradigm Product Editions*. <https://www.visual-paradigm.com/editions/>
- Wong Portillo, L. R. (2019). *Contribuciones en el proceso de elicitación de requisitos: Factores, actividades y cualidades* [Para optar el Grado Académico de Doctor en Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos].
<https://core.ac.uk/download/pdf/323343068.pdf>

ANEXOS

Anexo 1: Lineamientos de la Política Económica y Social de Cuba

Lineamiento 115

Promover y propiciar la interacción entre los sectores empresarial, presupuestado, académico, el sistema educativo y formativo, y las entidades de ciencia, tecnología e innovación, incentivando que los resultados científicos y tecnológicos se apliquen y generalicen en la producción y los servicios(Asamblea Nacional del Poder Popular, 2016).

Lineamiento 122

Avanzar en la informatización del sistema de educación. Desarrollar, de forma racional, los servicios en el uso de la red telemática y la tecnología educativa, así como la generación de contenidos digitales y audiovisuales(Asamblea Nacional del Poder Popular, 2016).

Anexo 2: Entrevista realizada al Responsable de la Práctica Profesional en CESOL

Estimado compañero:

La presente entrevista tiene como objetivo conocer las características e importancia que posee el desarrollo de la Práctica Profesional para los estudiantes de la UCI y cómo se realiza la gestión de la información resultante de su ejecución. Su colaboración es de vital importancia para la presente investigación ya que contribuye con el desarrollo de una aplicación informática que mejore la gestión de la información de la PP.

Cuestionario:

1. ¿Cree usted que es importante el desarrollo de la Práctica Profesional para los estudiantes de la UCI? ¿Por qué?
2. ¿Cuáles son los objetivos que tiene la Práctica Profesional en la Universidad y el Centro de Software Libre?
3. ¿Cuáles son las personas que interactúan en el desarrollo de la Práctica Profesional en CESOL? ¿Qué funciones desempeñan?
4. ¿Qué años son los que se vinculan a la PP en CESOL?
5. ¿Qué actividades realizan los estudiantes? ¿Cómo se les planifican y evalúan?
6. ¿Los estudiantes desempeñan algún rol del proceso productivo de software? ¿Cuáles?
7. ¿Los estudiantes se vinculan a los proyectos de investigación y desarrollo del centro?
8. ¿Existe algún documento que recoja toda esta información y les sirva de guía a los estudiantes para su desenvolvimiento?
9. ¿Cómo se lleva el control de la asistencia y el cumplimiento de los objetivos de las Prácticas Profesionales?
10. ¿Cómo se gestiona la información que se genera en la PP?

Anexo 3: Encuesta realizada a 15 Supervisores – Evaluadores - Tutores en CESOL

Estimado(a) compañero(a):

La presente encuesta tiene como objetivo conocer y analizar cómo se realiza la PP en CESOL, su importancia, qué información se obtiene de su ejecución, así como las deficiencias que presenta. Sus respuestas permitirán además obtener requisitos de software para la elaboración de la propuesta de solución. Su colaboración es de vital importancia para la presente investigación ya que contribuye con el desarrollo de una aplicación informática que mejore la gestión de la información de la PP, por lo que se necesita de su total colaboración y responsabilidad.

Cuestionario:

1. ¿Considera usted necesario el desarrollo de la Práctica Profesional en el proceso de formación de los estudiantes?
2. ¿De qué año académico son los estudiantes que usted atiende?
3. ¿Qué tipos de tareas realizan los estudiantes?
4. ¿Cuáles son los objetivos que tienen estas tareas?
5. ¿Cómo planificas y evalúas las tareas a los estudiantes que supervisas?
6. ¿Cómo gestionas la información que se genera de la atención a los estudiantes?
7. ¿Cuáles son los pasos que ejecutas para obtener y notificar esta información?
8. ¿Qué deficiencias se han presentado durante la gestión de la información de la PP en CESOL?
9. ¿Actualmente se satisface la gestión de las prácticas profesionales de manera correcta en el Centro de Software Libre?
10. ¿Considera usted que es necesario contar con una aplicación informática para la gestión de la información de la Práctica Profesional en el Centro de Software Libre?
11. ¿Qué opina acerca de los beneficios que trae consigo la aplicación para la gestión de la información de la Práctica Profesional en el Centro de Software Libre?

Anexo 4. Estructura del plan de formación del estudiante

PLAN DE FORMACIÓN DEL ESTUDIANTE			
1-DATOS GENERALES DEL ESTUDIANTE			
Nombre completo:			
Número de Solapín:		Grupo:	

2-DATOS DE LA ASIGNATURA			
Disciplina:		Asignatura:	
Cantidad de horas:		Cantidad de horas semanales:	

3-DATOS ESPECÍFICOS				
Centro de Desarrollo o Estudio:		Tipo de Proyecto		Proyecto con cliente
Grupo de Investigación:				
Proyecto:				Proyecto I+D
Responsables PID:				
SET:				Proyecto sin cliente
Profesor de PP:				

4-ROLES Y TAREA

Rol (es) que desempeñará:	
Tarea:	

5-OBJETIVOS ESPECÍFICOS PARA CADA SEMANA							
Fecha	Semana	Actividades	Habilidades	Cant. Horas	Resultado esperado	Tipo de resultado	Evaluación

Anexo 5. Guía de observación para analizar cómo se gestiona la información de la Práctica Profesional en aplicaciones informáticas

Observador: Roberto Javier Ramírez Carreño

Lugar: Laboratorio 102 de CESOL

Objetivos: Conocer cómo se realiza la gestión de información de la Práctica Profesional en diferentes aplicaciones informáticas e identificar sus funciones y características.

Elementos a observar:

1. Datos de las aplicaciones informáticas:
 - Nombre de las aplicaciones
 - Tipo de licencia
 - ¿Qué herramientas se necesitan para instalarlas?
 - ¿Cuáles son los pasos a seguir para instalar estas herramientas?
2. Características de las aplicaciones informáticas:
 - ¿Cómo son las aplicaciones informáticas que permiten la gestión de copias de seguridad?
 - Tienen disponibilidad de interfaz gráfica de usuario
 - Garantiza la centralización de la información
 - Define roles del proceso productivo de software
 - Especifica proyectos de investigación y desarrollo y/o grupos de investigación
 - Permite la orientación de tareas personalizadas
3. Impacto social de las aplicaciones informáticas

