



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

SISTEMA DE CONTROL DE PLAGIO PARA EL ÁMBITO EDUCATIVO
DEL DESARROLLO DE SOFTWARE UTILIZANDO FILOSOFÍA DEVOPS

LOPEZ JARAMILLO EDGAR STEVEN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

SISTEMA DE CONTROL DE PLAGIO PARA EL ÁMBITO
EDUCATIVO DEL DESARROLLO DE SOFTWARE UTILIZANDO
FILOSOFÍA DEVOPS

LOPEZ JARAMILLO EDGAR STEVEN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

SISTEMA DE CONTROL DE PLAGIO PARA EL ÁMBITO EDUCATIVO DEL
DESARROLLO DE SOFTWARE UTILIZANDO FILOSOFÍA DEVOPS

LOPEZ JARAMILLO EDGAR STEVEN
INGENIERO DE SISTEMAS

CARTUCHE CALVA JOFFRE JEORWIN

MACHALA, 25 DE FEBRERO DE 2022

MACHALA
2022

Trabajo de Titulacion

INFORME DE ORIGINALIDAD

5%

INDICE DE SIMILITUD

4%

FUENTES DE INTERNET

0%

PUBLICACIONES

1%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Escuela Politecnica Nacional	<1 %
	Trabajo del estudiante	
2	dspace.utb.edu.ec	<1 %
	Fuente de Internet	
3	doku.pub	<1 %
	Fuente de Internet	
4	Submitted to Universidad Técnica de Machala	<1 %
	Trabajo del estudiante	
5	issuu.com	<1 %
	Fuente de Internet	
6	Submitted to Universidad de Jaén	<1 %
	Trabajo del estudiante	
7	piz.san.edu.pl	<1 %
	Fuente de Internet	
8	openwebinars.net	<1 %
	Fuente de Internet	
9	Submitted to UNIV DE LAS AMERICAS	<1 %
	Trabajo del estudiante	

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, LOPEZ JARAMILLO EDGAR STEVEN, en calidad de autor del siguiente trabajo escrito titulado SISTEMA DE CONTROL DE PLAGIO PARA EL ÁMBITO EDUCATIVO DEL DESARROLLO DE SOFTWARE UTILIZANDO FILOSOFÍA DEVOPS, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 25 de febrero de 2022



LOPEZ JARAMILLO EDGAR STEVEN
0704913201

DEDICATORIA

Dedico este trabajo a mi madre quien con su sacrificio y esfuerzo me ha apoyado incondicionalmente y también ha sido mi pilar fundamental para poder cumplir mis sueños de terminar mi carrera.

A mi tío que por cosas inevitables de la vida pereció este año, pero siempre me guió por el buen camino y estuvo pendiente de mí y apoyándome en los momentos difíciles que atravesaba brindándome su tiempo y consejos para que llegue a ser una persona de bien y un buen profesional.

Sr. Lopez Jaramillo Edgar Steven

AGRADECIMIENTO

Agradezco a mi madre quien nunca dudó de mí, siempre creyó que era capaz de lograrlo y siendo mi pilar fundamental para no darme por vencido.

También quiero agradecer a mi tío quien siempre me inculcó ser una persona de bien y también por ese apoyo moral como económico que siempre estuvo en los momentos más difíciles durante el proceso de mis estudios.

Agradezco a mis compañeros/as que de una u otra manera me apoyaron despejando dudas sobre algunas asignaturas en todos estos años de estudio.

Y a mi tutor Ing. Joffre Jeorwin Cartuche Calva por ser un excelente docente y amigo, ayudándome a realizar de manera exitosa la terminación de este trabajo de titulación.

Sr. Lopez Jaramillo Edgar Steven

RESUMEN

El uso de aplicaciones web ha ido en crecimiento durante la última década debido a que cada vez nos hacemos más dependientes de la tecnología ya que ayudan a las personas a realizar actividades laborales o educativas de forma rápida; además de facilitar y optimizar procesos.

Todo estudiante universitario tiene derechos y obligaciones que debe cumplir dentro del ámbito estudiantil y una de esas obligaciones es la de realizar las tareas que envían los docentes, pero no solo se trata de cumplir sino también de poner en práctica y aplicar valores como la honestidad al momento de realizar las tareas, sin realizar copias de ellas.

Es muy importante que el estudiante aplique valores dentro de sus estudios ya que eso lo hará ser un excelente profesional en el futuro, debido a que aprovechó satisfactoriamente cada enseñanza de manera óptima e hizo desarrollo de sus conocimientos, en cada periodo académico culminado durante su etapa universitaria.

La Universidad Técnica de Machala actualmente posee un sistema de aprendizaje llamado "EVEA" la cual es utilizada por los docentes y estudiantes para la gestión académica en donde se pueden subir las tareas autónomas, intraclases, evaluaciones, etc. Los docentes son los encargados de analizar y evaluar de manera virtual las tareas subidas de los estudiantes; pero la revisión que realizan los docentes sobre las tareas solo se las puede realizar de forma visual y no es posible revisar detalladamente cada punto debido a la gran cantidad de estudiantes y cursos que tiene asignado.

Por tal motivo el docente no puede llevar un buen control sobre el contenido de las tareas que envían los estudiantes; entonces es por eso que se desarrolló una aplicación web que permita al docente calificar las tareas específicamente del área de programación de los alumnos de forma óptima y precisa, para que calcule y a su vez muestre los resultados de similitud entre cada archivo procesado enviado por los estudiantes.

Los resultados obtenidos fueron satisfactorios, la aplicación funciona de forma eficiente y cumple con su propósito establecido en los requerimientos. Logra revisar las tareas de programación y evaluar su contenido para determinar el nivel de plagio.

De esta forma los docentes serán capaces de calificar mejor las tareas de cada uno de sus estudiantes y dar notas que reflejan más el real rendimiento de ellos e ir mejorando y aumentando su capacidad intelectual.

La aplicación web fue desarrollada en lenguaje de programación Python, también frameworks que están en tendencia en la actualidad como son Flask y Bootstrap y también se utilizó como gestor de base de datos no relacional a MongoDB. Aplicando la metodología "SWIRL" como parte fundamental para desarrollar la documentación debido a que contiene las fases de desarrollo web de calidad; Se implementa la arquitectura de tipo cliente-servidor esto permite tener un mejor control y eficiencia en los procesos que realiza la aplicación web, además se realizó la evaluación del sistema utilizando el estándar de métricas de calidad de software ISO/IEC-9126, obteniendo de forma satisfactoria un aporte de desarrollo tecnológico para mejorar el ámbito educativo de la universidad.

Palabras clave: Desarrollo web, Gestión de plagio, Python, SWIRL, Flask.

ABSTRACT

The use of web applications has been growing during the last decade due to the fact that we become increasingly dependent on technology and that they help people to carry out work or educational activities quickly; in addition to facilitating and optimizing processes.

Every university student has rights and obligations that must be fulfilled within the student environment and one of those obligations is to carry out the tasks that the teachers send, but it is not only about complying but also about putting into practice and applying values such as honesty when time to perform the tasks, without making copies of them.

It is very important that the student apply values within their studies since that will make them an excellent professional in the future, because they satisfactorily took advantage of each teaching in an optimal way and developed their knowledge, in each academic period culminating during their stage. university.

The Technical University of Machala currently has a learning system called "EVEA" which is used by teachers and students for academic management where autonomous tasks, intraclassess, evaluations, etc. can be uploaded. The teachers are in charge of analyzing and evaluating the uploaded tasks of the students in a virtual way; but the review that teachers carry out on the tasks can only be done visually and it is not possible to review each point in detail due to the large number of students and courses assigned to it.

For this reason, the teacher cannot have good control over the content of the tasks sent by the students; That is why a web application was developed that allows the teacher to qualify the tasks specifically in the students' programming area in an optimal and precise way, so that it calculates and in turn shows the similarity results between each processed file sent by the students.

The results obtained were satisfactory, the application works efficiently and fulfills its purpose established in the requirements. Manages to review programming tasks and evaluate their content to determine the level of plagiarism. In this way, teachers will be

able to better qualify the tasks of each of their students and give grades that more reflect their real performance and improve and increase their intellectual capacity.

The web application was developed in the Python programming language, as well as frameworks that are currently in trend such as Flask and Bootstrap, and MongoDB was also used as a non-relational database manager. Applying the "SWIRL" methodology as a fundamental part to develop the documentation because it contains the phases of quality web development; The client-server type architecture was implemented, this allows for better control and efficiency in the processes carried out by the web application, in addition, the evaluation of the system was carried out using the ISO/IEC-9126 software quality metric standard, obtaining from a satisfactory way a contribution of technological development to improve the educational field of the university.

Keywords: Web development, Plagiarism management, Python, SWIRL, Flask.

INTRODUCCIÓN

El desarrollo y uso de aplicaciones web ha incrementado con el transcurso de los años, también se han convertido en una herramienta muy útil en el ámbito educativo lo que ha permitido tanto a docentes como estudiantes realizar sus actividades de forma digital. Su fácil accesibilidad y alcance de todas las personas que poseen una computadora.

La implementación de aplicaciones web utilizadas en el aprendizaje de los estudiantes ha ayudado que estos realicen de manera rápida sus tareas por ende los docentes también utilizan ciertas aplicaciones web para revisar las tareas haciendo que sea de forma didáctica la revisión, de forma óptima y con la menor falla de errores.

Los estudiantes son los encargados de realizar sus tareas de forma individual en el cual se ve reflejado sus conocimientos y valores de no tener alguna similitud con tareas de otros compañeros.

Como obligación de cada estudiante es cumplir con las tareas, por ello hemos creado una aplicación que permita al docente poder revisar dichas tareas de forma rápida y visualizar los valores de honestidad que tiene los estudiantes al momento de realizar y enviar sus tareas.

Los docentes de la Universidad Técnica de Machala revisan a diario las tareas que los estudiantes les envían por ello muchas de las veces no se cumple detalladamente el formato estructurado en la revisión de las mismas, y debido a la complejidad de revisión de código de programación estos no son comprobados de la manera correcta al momento de la revisión.

Debido a esta problemática, nació la necesidad de la creación de una aplicación web en donde el estudiante deberá subir los archivos de programación en unas salas específicas creadas por el docente el cual detectará la similitud entre dichos archivos que se encuentren dentro de la sala.

Dicha aplicación web debe tener como principal característica un back-end para que permita funcionamiento correcto para la comprobación y revisión de los documentos a evaluar y que sea de la manera más transparente y equitativa posible, esta aplicación web se la realizará en el lenguaje de programación Python y flask en la parte del back-end y en el front-end utilizar Css, Html y Javascript.

Capítulo 1: Se especifica detalladamente un análisis completo de todos las necesidades y requerimientos de las cuales va a constar el sistema como son los hechos de intereses, establecimientos de requerimientos y la justificación de requerimientos.

Capítulo 2: Esta sección contiene todo lo relacionado con las fases del proceso de desarrollo del prototipo en el cual se especifican detalladamente de manera teórica todos los componentes de cuales va a estar confirmado el sistema como son la arquitectura, objetivos y tecnologías necesarias. También se elaboran los diagramas, modelos del diseño del prototipo y modelos de las ventanas de ejecución del sistema desarrollado.

Capítulo 3: Esta última sección está conformada por toda la parte de evaluación del sistema desarrollado como es el plan de evaluación y los resultados de evaluación utilizando estándares de evaluación. También se especifican las conclusiones y recomendaciones.

ÍNDICE

DEDICATORIA	1
AGRADECIMIENTO	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN	7
1. CAPÍTULO I. NECESIDADES Y REQUERIMIENTOS.	12
1.1. DESCRIPCIÓN DE HECHOS DE INTERESES	12
1.2. ESTABLECIMIENTO DE REQUERIMIENTOS	12
1.3. JUSTIFICACIÓN DE REQUERIMIENTOS	13
2 CAPÍTULO II. DESARROLLO DEL PROTOTIPO DE LA APLICACIÓN WEB.	14
2.1. DEFINICIÓN DEL PROTOTIPO	14
2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO	16
2.3. OBJETIVOS DEL PROTOTIPO	22
2.4. DISEÑO DEL PROTOTIPO	23
2.5. EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO	42
3. CAPÍTULO III. EVALUACIÓN DE LA APLICACIÓN DE WEB.	46
3.1. PLAN DE EVALUACIÓN.	46
3.2. RESULTADOS DE EVALUACIÓN.	47
3.2. CONCLUSIONES	49
3.3. RECOMENDACIONES	50
BIBLIOGRAFÍA	50
ANEXOS	53

ÍNDICE DE ILUSTRACIONES

Figura 1: Arquitectura Cliente-Servidor	14
Figura: 2: Modelado de tres capas	15
Figura 3: Cronograma de actividades	31
Figura 4: Diagrama de Gantt	31
Figura 5: Base de datos No relacional	32
Figura 6: Caso de uso – Inicio de sesión	33
Figura 7: Caso de uso - Registrarse	33
Figura 8: Caso de uso - Crear salas	34
Figura 9: Caso de uso - Resultado.....	34
Figura 10: Caso de uso - Subir archivos	35
Figura 11: Diagrama de secuencia - Inicio de sesión	35
Figura 12: Diagrama de secuencia - Visualizar salas	36
Figura 13: Diagrama de secuencia - Subir archivos	36
Figura 14: Diagrama de secuencia - Crear salas	37
Figura 15: Diagrama de secuencia – Subir archivos	37
Figura 16: Diseño de Interfaces: Inicio de sesión	38
Figura 17: Diseño de Interfaces: Registrarse.....	38
Figura 18: Diseño de Interfaces: Página principal	39
Figura 19: Diseño de Interfaces: Creación de salas	39
Figura 20: Diseño de Interfaces: Subir archivos	40
Figura 21: Diseño de Interfaces: Visualización de resultados por sala	40
Figura 22: Diseño de Interfaces: Visualización de salas	41
Figura 23: Diseño de Interfaces: Visualizar resultados.....	41
Figura 24: Inicio de sesión.....	42
Figura 25: Registrarse	42
Figura 26: Menú principal	43
Figura 27: Crear sala	43
Figura 28: Salas creadas.....	44
Figura 29: Detalle de la sala.....	44
Figura 30: Dashboard general.....	45
Figura 31: Dashboard individual.....	45
Figura 32: Subida de archivos.....	46
Figura 33: Evaluación de calidad - GTmetrix	47
Figura 34: Evaluación de calidad - WAVE	48
Figura 35: Evaluación de calidad - W3C Validator	48
Figura 36: Anexo - Registro de usuario.....	53
Figura 37: Anexo - Validación de usuario	53

ÍNDICE DE TABLAS

Tabla 1: Factibilidad Operativa.....	23
Tabla 2: Factibilidad Técnica.....	23
Tabla 3: Factibilidad Económica.....	24
Tabla 4: Funcionalidades del sistema.....	24
Tabla 5: Limitaciones del sistema.....	25
Tabla 6: Matriz de Interesados.....	25
Tabla 7: Requisitos Funcionales.....	26
Tabla 8: Requerimientos no Funcionales.....	26
Tabla 9: Historia de usuario – Inicio de sesión.....	27
Tabla 10: Historia de usuario – Registro de docentes.....	28
Tabla 11: Historia de usuario – Creación de salas.....	28
Tabla 12: Historia de usuario – Gestión de salas.....	29
Tabla 13: Historia de usuario – Reporte general.....	29
Tabla 14: Historia de usuario – Subida de archivos.....	30
Tabla 15: Evaluación de calidad ISO/IEC 9126.....	47
Tabla 16: Evaluación de calidad - Usabilidad y Portabilidad.....	49
Tabla 17: Anexos - Matriz de riesgos.....	54

1. CAPÍTULO I. NECESIDADES Y REQUERIMIENTOS.

1.1. DESCRIPCIÓN DE HECHOS DE INTERESES

En la Universidad Técnica de Machala, dentro del proceso de desarrollo académico los estudiantes realizan sus tareas de forma diaria las cuales eran revisadas de forma presencial por el docente, pero debido a la emergencia sanitaria a nivel mundial causada por el COVID-19 las clases y por ende las revisiones de las tareas las realizan de manera virtual.

Es bastante complicado revisar detalladamente el código de los archivos de programación debido a la cantidad de líneas, por eso en la mayoría solo se revisa el funcionamiento y que cumpla con todos los requisitos necesarios.

La problemática de satisfacer la necesidad de poder revisar las líneas de código realizadas por los estudiantes en sus en sus tareas es para que los docentes puedan hacer esa revisión de manera eficiente y esto ayudará a los estudiantes a que su nivel académico sea íntegro, de buen rendimiento y que a futuro será un gran profesional que se podrá desenvolver de manera excelente en el ámbito laboral.

Se pretende satisfacer la problemática mediante la creación e implementación de una aplicación web alojada en un servidor gratuito en la nube la cual contiene varias herramientas de desarrollo ágil las cuales son muy utilizadas en el área de desarrollo web es por ello que también se utilizará el modelo de metodología de desarrollo SWIRL.

1.2. ESTABLECIMIENTO DE REQUERIMIENTOS

En la Facultad de Ingeniería Civil específicamente en las carreras de Ingeniería de Sistemas y Tecnologías de la Información, docentes y coordinador realizaron un diálogo referente al método de evaluación que los docentes actualmente están empleando sobre la revisión de archivos de programación de los estudiantes.

Por ello se ha decidido la creación e implementación de una aplicación que optimice el tiempo de revisión en cada uno de los archivos que los estudiantes realizan.

- Registro de estudiantes y docentes utilizando correo institucional.
- Creación de salas que permitirán a los estudiantes poder subir los archivos de sus tareas.

- Realizar la comparación de todos los archivos que se encuentran dentro de la misma clase.
- Descarga de reportes de similitud entre los archivos evaluados dentro de una clase.

1.3. JUSTIFICACIÓN DE REQUERIMIENTOS

El propósito de la realización de esta aplicación web para detección de plagio, es realizar de una manera equitativa y justa la comparación y detección de similitud entre varios archivos que se encuentran subidos en una misma sala la cual es creada por el docente. La cual va a detectar el porcentaje de similitud que encuentre en archivos relacionados directamente con plagio.

Dicha aplicación web adicionalmente de que ayudará a la detección de plagio y de similitud de archivos de programación también permite observar el desarrollo progresivo de los conocimientos de los estudiantes que han ido obteniendo en cada periodo académico y así mismo obtener y desarrollar nuevos conocimientos.

La aplicación web optimiza el tiempo de calificación de las tareas enviadas por los docentes ya que únicamente tendrán que realizar una previa visualización de los resultados obtenidos para comprobar la similitud entre archivos y realizar la calificación correspondiente a cada estudiante.

2 CAPITULO II. DESARROLLO DEL PROTOTIPO DE LA APLICACIÓN WEB.

2.1. DEFINICIÓN DEL PROTOTIPO

PlagSite no será una aplicación web común, porque a diferencia de otros proyectos tecnológicos estará desarrollado mediante un entorno de ejecución específico como es el acceso, comparación y distribución de información, A continuación, se detalla la estructura específica de cómo estará desarrollado el proyecto.

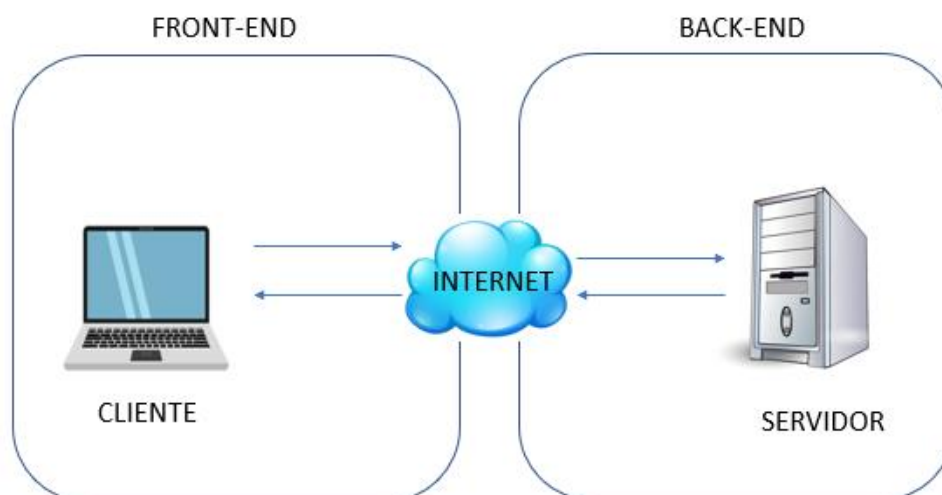
ARQUITECTURA DEL SOFTWARE

Es la forma de cómo está compuesto y estructurado el proyecto que se está realizando ya que cuenta con propiedades visibles y no visibles, también con todas las relaciones que existen entre las entidades con el sistema, mediante esta arquitectura se establecen todos los requerimientos y cambios ya sean estos funcionales o no funcionales que pueden afectar al sistema.[1]

Arquitectura implementada

Para el desarrollo de esta aplicación se utilizó el tipo de arquitectura cliente- servidor, el cual tiene un funcionamiento específico donde la parte del cliente realiza las peticiones al servidor donde está alojada toda la información necesaria que el cliente necesita.[2]

Figura 1: Arquitectura Cliente-Servidor

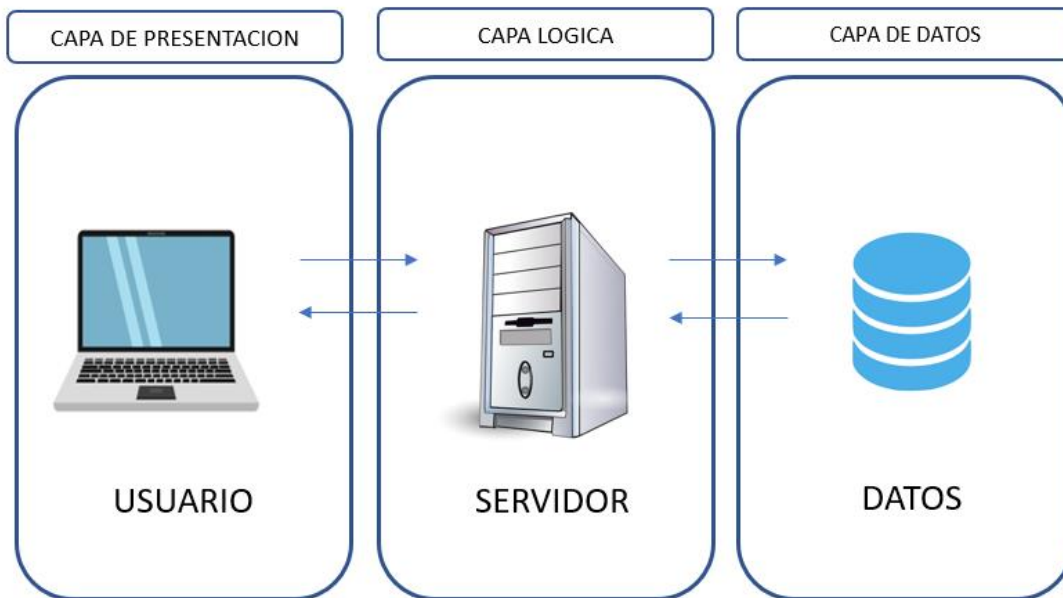


Fuente: Elaboración propia

MODELADO DE TRES CAPAS

Este tipo de modelo es el que se encarga de encapsular todas las funciones del sistema y la separa de la capa de presentación y de datos. [3]

Figura 2: Modelado de tres capas



Fuente: Elaboración propia

Capa de datos

La capa de datos se encarga de todo lo relacionado con el almacenamiento de datos de manera directa e indispensable del sistema, se utilizó el gestor de base de datos no relacional MongoDB para el desarrollo de este prototipo, debido a su fácil usabilidad.[4]

Capa lógica

La capa lógica es la encargada de contener toda la lógica desarrollada en cada fase del sistema, es decir, realiza una interacción entre la capa de datos y la capa de presentación para que se puedan comunicar de manera interna al momento que el usuario realiza una petición.[5]

Existe una gran variedad de lenguajes de programación en los cuales se puede desarrollar la parte lógica del sistema, pero en este caso se utilizó el lenguaje de programación Python tomando en cuenta la estabilidad y utilización de dicho lenguaje.

Capa de presentación

Dentro de esta capa se visualizan todas las interfaces desarrolladas para la interacción del usuario con el sistema conocida también como “Front-end” mediante estándares de Bootstrap y CSS, en la cual se observarán todas las funcionalidades del mismo.[6]

2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO

A continuación, se destaca la fundamentación teoría del prototipo del presente proyecto con sus respectivas aplicaciones de web, metodologías, herramientas CASE y la norma ISO/IEC 9126.

DESARROLLO DEL SOFTWARE

El software en sí se puede decir que es un conjunto de instrucciones o programas que le dicen a la computadora qué hacer, a su vez que sea programable e independiente del hardware.[7]

Comprendiendo lo antes mencionado se manifiesta que el desarrollo de software se refiere a un cúmulo de acciones informáticas consagradas al proceso de creación, implementación, diseño y soporte de un producto software, porque permite escribir, solucionar y mantener su código fuente.

APLICACIONES WEB:

El software en sí se puede decir que es un conjunto de instrucciones o programas que le dicen a la computadora qué hacer, a su vez que sea programable e independiente del hardware.[8]

Comprendiendo lo antes mencionado se manifiesta que el desarrollo de software se refiere a un cúmulo de acciones informáticas consagradas al proceso de creación, implementación, diseño y soporte de un producto software, porque permite escribir, solucionar y mantener su código fuente.[9]

METODOLOGÍAS DE DESARROLLO DE SOFTWARE

Las metodologías de Desarrollo de Software en el transcurso del tiempo han tenido algunos cambios pasando de ser una organización arcaica a una base importante con

la cual se trabaja de manera fructífera y eficaz al momento de querer obtener soluciones.[10]

Diferentes autores apuntan a que estas metodologías de DS, son el conjunto de métodos organizativos profesionales que ayudan a solucionar diferentes problemas de software, teniendo siempre en cuenta que metodología se debe usar en los diferentes proyectos.[11]

Hay que tener muy claro que el desarrollo de software no es algo simple. Por ello existen algunas propuestas metodológicas que influyen en las dimensiones de su proceso, en este caso se ha tomado en cuenta las siguientes:

METODOLOGÍA TRADICIONAL:

A parte de las metodologías de DS actuales existe una que se consagra como “la tradicional”, la cual es un proceso que tiene como enfoque la disciplina y la predicción, es lineal y una vez que se empieza no hay marcha atrás, por ello es indispensable saber qué tipo de recursos, análisis y diseño se utilizará para que el cliente quede satisfecho. Lo más acertado es que este tipo de metodología se incline a proyectos que no tengan cambios y lo adopte alguien con mucha experiencia en el campo tecnológico. [12]

METODOLOGÍAS ÁGILES

Son las que permiten entregar un producto software de calidad, con rapidez, pero con menos costo y tiempo, esta metodología es flexible y moldeable lo que hace que el proyecto se acople a las circunstancias y entorno, a su vez tiene la capacidad de ser predecible en la demanda y respuesta del cliente. Cabe mencionar que el trabajo colaborativo es esencial en el marco de las metodologías ágiles, las cuales en la actualidad son las más solicitadas por los múltiples beneficios que ya se mencionaron anteriormente. [13]

METODOLOGÍAS HÍBRIDAS

Hoy en día al momento de producir software es importante de hacerlo de la manera más rápida y ágil, pero si se requiere actualizaciones o modificar algo es necesario la documentación que refleje lo que se ha avanzado en el desarrollo de software. Y es

así como nace la metodología híbrida que no es más que una combinación de las ágiles con las tradicionales combinando así sus procesos.[14]

Este tipo de metodología se dio por el incremento de las páginas webs y se adapta a la necesidad del usuario debido a que hace cualquier proyecto fácil de entender, mejorar y modificarlo las veces que se requiera, eso sí teniendo en cuenta los criterios internos y externos del cliente, que todo en conjunto logran un resultado óptimo y mejor.[15]

METODOLOGÍA SWIRL

Software Web Iterativo Relacional Lógico es utilizada para las aplicaciones que se basan en la web, su enfoque es híbrido y también iterativo. Toma en cuenta al cliente en todo el proceso, permite que se hagan modificaciones y un post mantenimiento del DS.[16]

Esta metodología es tomada en cuenta para grandes o pequeños proyectos por su carácter flexible y modificable a las necesidades, a su vez que permite ahorrar tiempo y recursos, aunque tiene dos desventajas: Se trabaja con una interacción y la ejecución de una próxima revisión, el usuario tiene que estar definido. [17]

Se basa en ciertos criterios como tiempo, costo, alcance, calidad y comunicaciones determinadas por las partes interesadas.

A continuación, se explica mediante un gráfico los criterios y las 6 fases de esta metodología que son:

- **Análisis:** Una perspectiva acerca del estado y lo que se quiere lograr del proyecto, además de la funcionalidad y análisis de riesgo del sistema.
- **Planificación:** Es necesario que el desarrollador tome en cuenta todos los criterios dentro de las actividades de cada fase.
- **Modelado:** De acuerdo a la base de información se crean los diseños, modelos ya sean conceptuales o navegacionales. A su vez se inclina por diagramas y actividades [18]

En esta fase se realizan los diseños correspondientes a las bases de datos, modelos conceptuales, y modelos navegacionales. Además, dentro de esta metodología se aplica el uso de diagramas UML (Lenguaje de Modelado Unificado) para el modelado de datos.

- **Implementación:** Se tiene que emplear las herramientas necesarias de programación para que haya una interconexión entre páginas web, base y enlaces.
- **Verificación y pruebas:** En esta fase se controla la correcta funcionalidad del producto software, se detecta errores, cambios y cierre.
- **Lanzamiento:** Esto ocurre al final del proceso del producto y puede o no ser aceptado por el cliente, no debe poseer errores al momento de ejecutarse, y por último es indispensable trabajar con SEO, con el fin de posicionarlo en los diferentes buscadores.

Visual studio code

Es un editor de texto que permite a todos los programadores crear sus proyectos en diferentes plataformas, es de fácil uso que hace que cada vez sea más utilizado. Brinda un ambiente confortable para desarrollar aplicaciones no muy complejas porque a su vez cuenta con terminales que permiten ejecutar y comprobar lo proyectos creados por los desarrolladores.[19]

PYTHON

Es un lenguaje de programación de alta demanda porque está respaldado por desarrolladores y vendedores. A si mismo con poco código realiza muchas tareas gracias a que es entendible, limpio y sencillo por lo que muchos profesionales pueden manejarlo. Por último, cabe mencionar que crea aplicaciones, páginas web, analiza y visualiza datos, efectúa estadísticas y gestiona bases de datos.[20]

FLASK

Es un microframework minimalista compatible con el lenguaje de Python, especialmente con Python3, aunque también es posible con wsgi, ayuda a crear páginas web dinámicas–creativas, esto ocurre gracias a la ayuda del código HTML, en el cual se asientan las bases e interconexión entre páginas o sitios webs. Un punto importante es tomar en cuenta el controlador para ver que pide el usuario, a su vez que permite dar un seguimiento óptimo de ruta.[21]

BOOTSTRAP

Ayuda a moldear los componentes de una página con código HTML. Permite al usuario la interconectividad entre menús de navegación, barras de progreso y controles de página. Es indispensable el uso de BOOTSTRAP porque es compatible con todos los dispositivos actuales del mercado. Para hacer más fácil y organizado el trabajo en el DS emplea dos directorios: Css y Js. Cabe destacar que para que funcione eficazmente se debe usar las bibliotecas como: JQuery y Popper.js.[22]

BASE DE DATOS NO RELACIONAL

Esta base de datos se está utilizando actualmente y no descarta totalmente al SQL, porque si se apoya de vez en cuando en este. Trabaja con un sistema de gestión bastante amplio capaz de almacenar datos no estructurados. [23] Tiene algunas características entre ellas en que es gil cuando no se sabe que datos se va a guardar, tiene un esquema dinámico, flexible porque si el programador quiere modificar alguna estructura de un dato lo puede hacer, debido a q no afecta en lo absoluto a todo el sistema y por último tienen escalabilidad horizontal. [24]

MONGODB

Es una base de datos NoSQL (humongous “enorme), líder e importante del mercado y que se usa en diferentes aplicaciones porque permite almacenar poca o bastante información en documentos BSON, la cual es similar a JSON, con el fin de ocupar menos espacio. [25] No se maneja con tablas, esquemas, SQL, columnas o filas, lo cual lo hace factible para trabajar en proyectos gubernamentales y empresariales. Entre las principales características que reúne MONGODB se encuentran las siguientes: consultas ad hoc, Indexación, replicación, balanceo de carga, almacenamiento de archivos y ejecución de JavaScript del lado del servidor.[26]

Por otro lado, esta base de datos se trabaja con un código abierto y se escribe en C++, por lo cual se adapta fácilmente al Windows, GNU/Linux, OS X y Solaris, además cuenta con la licencia GNU AGPL 3.0. Cabe destacar que empresas como MTV, Craigslist, BCI o Foursquare han adoptado esta base para sus diferentes objetivos.

HERRAMIENTAS CASE:

Se refieren a ingeniería de Software, son las diferentes aplicaciones que ayudan a la productividad a los desarrolladores, porque reducen costos y tiempo, gracias al diseño, cálculo, implementación, compilación, documentación y detección de errores.[27]

Para el desarrollo de software es indispensable usar las siguientes herramientas:

- **BALSAMIQ MOCKUPS**

Es una herramienta wireframing de escritorio que no necesariamente tiene que ser descargada para su uso posterior, cuando se está creando un proyecto. Lo más usual es la modificación de ciertas ideas, mediante diseños para llegar al prototipo del proyecto final y es ahí cuando BALSAMIC entra en acción pues es fácil, rápido e inteligente para trabajar con demostraciones y pruebas. Se programa con FLEX y AIR, tiene una interfaz accesible, se adapta a Linux y Mac OS X.[28]

- **STARUML**

Es una herramienta que ayuda en el moldeamiento de los proyectos gracias a que se construye en base a diagramas UML, como los de caso, de clase, de secuencia, de Estado y relación entidad. StarUml es de free software, trabaja con C#, C++ y también con Java, a su vez produce información en diferentes formatos y permite producir plugins. En pocas palabras es una gran alternativa eficaz y gratis para un desarrollador. [29]

- **BIZAGI**

Es una aplicación gratuita de Gustavo Gómez que permite crear, modelar, documentar información acerca del proyecto gracias a BPMN, además de que goza de cierta popularidad debido a que tiene una interfaz intuitiva y eficiente. Además, se puede ejecutar desde la nube por ende ahorra tiempo y recursos.

- **MICROSOFT PROJECT**

Es una aplicación costosa de escritorio que tiene el objetivo de administrar proyectos, pertenece a Microsoft, ayuda a muchos administradores en el diseño, asignación de recursos, administración de presupuesto, evaluación de las tareas, y así poder calcular el tiempo de cada proyecto. Utiliza Kanban, hojas de rutas, diagramas entre

otras herramientas, para que así también el usuario tenga más fácil el acceso a la información y sea comprendida. Cabe mencionar que las diferentes versiones de Ms Project tienen sus propias características.[30]

- **TOAD DATA MODELER**

Es una herramienta muy práctica para todos los que trabajan con datos porque permite tener rápidamente cambios en las bases de datos, en muchas plataformas. Usa SQL y DDL, lo cual hace fácil el diseño, mantenimiento, documentación a un precio accesible. Hoy en día se emplea con Oracle Database, Microsoft SQL Server, MySQL y IBM DB2.

- **NORMA ISO/IEC 9126:**

Es un estándar de calidad que todo producto software debe pasarlo, por ello se basa en 7 características que son: función, uso, eficiencia, mantenimiento, portabilidad y satisfacción, además la evaluación de recursos humanos y procedimientos de administración, para ello utiliza métricas externas e internas de acuerdo al modelo McCall. En definitiva, es beneficioso tanto para el usuario como para el desarrollador.[31]

2.3. OBJETIVOS DEL PROTOTIPO

Objetivo general

Crear una aplicación web utilizando metodología Swirl, lenguaje de programación Python y MongoDB como gestor de base datos para la detección de similitud y control de plagio en los archivos de programación.

Objetivos específicos

- Recopilar información sobre los principales requerimientos funcionales y no funcionales del sistema mediante historias de usuarios.
- Determinar las principales fases de la metodología SWIRL en el desarrollo de la estructura de la aplicación web.
- Diseñar la arquitectura funcional de la aplicación web utilizando herramientas CASE.
- Desarrollar la aplicación web mediante el uso del lenguaje de programación Python.

- Evaluar la aplicación web utilizando la norma de calidad ISO/IEC 9126 para la evaluación de la plataforma web.

2.4. DISEÑO DEL PROTOTIPO

Estudio de factibilidad

El presente estudio de factibilidad determinará qué tan factible es la implementación de la aplicación móvil para la gestión de portafolios académicos.

Factibilidad Operativa

Tabla 1: Factibilidad Operativa

Factibilidad Operativa		
No.	Actividad	Priorización
1	Capacitación indispensable para entender el funcionamiento correcto del sistema	Media
2	Conexión a internet indispensable para el funcionamiento	Alta
3	Poseer conocimientos técnicos sobre desarrollo de aplicaciones web	Alta

Fuente: Elaboración propia

Factibilidad Técnica

Tabla 2: Factibilidad Técnica

Factibilidad Técnica		
Cantidad	Nombre	Descripción
1	Electricidad	Instalaciones de energía eléctrica.
1	Internet	Instalación de conexión a internet.
1	Computador	Intel core i5 – 16 Gb RAM – SSD 480 Gb

Fuente: Elaboración propia

Factibilidad Económica.

Tabla 3: Factibilidad Económica

Factibilidad Operativa		
Hardware		
Cantidad	Descripción	Costo (\$)
1	Laptop Intel Core i5 10ma generación, 16 Gb memoria RAM, 480 SSD.	\$1000.00
Software		
Cantidad	Descripción	Costo (\$)
1	Lenguaje de programación Python.	\$ 0.00
1	Visual Studio Code.	\$ 0.00
Total de Costos		\$ 1000.00

Fuente: Elaboración propia

Fase de análisis

Esta fase presenta un enfoque claro sobre los requerimientos y funcionalidades del sistema que se está desarrollando el cual permite analizar de manera correcta la viabilidad que tiene el proyecto.

Alcance del proyecto

El alcance de un proyecto está conformado por todas las funcionalidades necesarias que contiene el sistema y todas las gestiones de los módulos que realiza definidas por niveles de mayor a menor prioridad.

Tabla 4: Funcionalidades del sistema

FUNCIONALIDADES DE PLAGSITE
Permitir el registro de personas naturales o docentes en el sistema.
Permitir la gestión de salas.
Permitir la visualización de los reportes generados de forma general e individual.
Permitir la subida de archivos en formatos específicos preestablecidos.
Permitir compartir la sala creada.

Fuente: Elaboración propia

También se establecen ciertos tipos de limitaciones dependiendo los objetivos propuestos desde un comienzo, estos tipos de limitaciones no son de forma fija ya que se los puede ir modificando en un futuro.

Tabla 5: Limitaciones del sistema

LIMITACIONES DE PLAGSITE
No permite generar un reporte en archivo pdf.
No se puede subir más de un archivo por usuario.
No contiene un administrador para gestionar usuarios

Fuente: Elaboración propia

Fase de planificación

Interesados

También llamados stakeholders son las personas involucradas o relacionadas en el desarrollo del sistema ya sea con las actividades y decisiones importantes que toma una organización. [32]

Tabla 6: Matriz de Interesados

MATRIZ GENERAL DE STAKEHOLDER			
Nombre	Rol	Profesión	Responsabilidad
Ing. Joffre Cartuche Calva	Docente/ Gerente de proyecto	Ingeniero de sistemas	Requerimientos, limitaciones y revisión de entregables.
Edgar López Jaramillo	Desarrollador	Estudiante	Planificación, Diseño, Desarrollo, Implementación y Pruebas.

Fuente: Elaboración propia

Requerimientos

En esta fase se desarrollan principalmente todas las actividades que ayudan a la creación del sistema, en donde se definen todos los requisitos funcionales y no

funcionales mediante la información que se ha obtenido de todos los interesados del proyecto

Funcionales

Los requerimientos funcionales permiten el análisis, identificación y la gestión del software mediante las actividades que cumple cada una, en caso de que los requerimientos sean deficientes puede causar el fracaso del proyecto. [33]

Tabla 7: Requisitos Funcionales

REQUERIMIENTOS FUNCIONALES (RF)	
Código	Descripción
RF - 01	Página principal de bienvenida a los usuarios del sitio web y autenticación de usuarios
RF - 02	Registro de docentes
RF - 03	Creación de salas respectivas organizada por el docente
RF - 04	Gestión de salas (Compartir, Editar, Evaluar y Reporte)
RF - 05	Reporte detallado de las salas creadas de forma individual y general
RF - 06	Gestión de subida de archivos de parte de los estudiantes

Fuente: Elaboración propia

No Funcionales

Este tipo de requerimientos son utilizados para juzgar las funciones del sistema, corresponden a todos los requisitos indirectos del sistema que son importantes, pero no indispensables. [34]

Tabla 8: Requerimientos no Funcionales

REQUERIMIENTOS NO FUNCIONALES (RNF)	
Código	Descripción
RNF-01	El sistema debe ser responsivo y que se adapte a todo tipo de dispositivos.

RNF-02	El sitio web debe contener mensajes de errores, advertencia y satisfacción en lugares visibles.
RNF-03	La base de datos no relacional debe estar estructurada y validada de forma correcta.

Fuente: Elaboración propia

- Historias de usuarios

Las historias de usuarios son una forma más explícita de mostrar la presentación de las funciones de un software de manera general y también informal, la función principal de las historias de usuarios es proporcionar valor al cliente dependiendo de las funciones.[35]

Tabla 9: Historia de usuario – Inicio de sesión

Historia de usuario	
Número: 1	Usuario: Docente
Nombre de la historia: Inicio de sesión.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-01	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: La aplicación web contiene una ventana de inicio de sesión sólo para docentes en donde se ingresará con el correo y la contraseña respectiva colocada por el docente al momento de registrarse.	
Observaciones: se muestra un mensaje de error en caso de que cualquiera de los dos parámetros sea ingresado incorrectamente por el usuario, debido a que cuando se envió la petición dichos datos no se encuentran almacenados en la base de datos utilizada por el sistema.	

Fuente: Elaboración propia

El ingreso será controlado por el sistema y solo permitirá el acceso al personal autorizado que en este serán los docentes mediante la verificación si datos ingresados son correctos y así habilitar las funcionalidades respectivas.

Tabla 10: Historia de usuario – Registro de docentes

Historia de usuario	
Número: 2	Usuario: Docente
Nombre de la historia: Registro de docentes.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-02	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: Como docente va a poder registrarse ingresando sus datos personales como nombres, apellidos, correo institucional y clave de manera obligatoria.	
Observaciones: El docente será el único usuario que podrá tener acceso a todas las funcionalidades del sistema.	

Fuente: Elaboración propia

El sitio web debe poder permitir registrarse a los docentes para que puedan tener acceso a todas las funcionalidades del sistema

Tabla 11: Historia de usuario – Creación de salas

Historia de usuario	
Número: 3	Usuario: Docente
Nombre de la historia: Creación de salas	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-03	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: La aplicación web permitirá al docente poder crear una o varias salas como sean necesarias y requeridas por el usuario.	
Observaciones: las salas se podrán crear dependiendo de los parámetros elegidos por el docente.	

Fuente: Elaboración propia

El docente será la única persona que luego de haberse registrado, ingresa al sistema y podrá crear las salas especificando todos los parámetros necesarios como el nivel

máximo de estudiantes, tamaño de archivo, cantidad máxima de archivos para subir, etc.

Tabla 12: Historia de usuario – Gestión de salas

Historia de usuario	
Numero: 4	Usuario: Docente
Nombre de la historia: Gestión de salas (Compartir, Editar, Evaluar y Reporte)	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-04	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: Como Docente tendrá habilitado todas las funcionalidades como compartir, editar, evaluar y reporte de la sala seleccionada.	
Observaciones: El sistema permitirá compartir la sala, realizar cualquier cambio funcional de la sala, así como evaluar y presentar un reporte de la sala seleccionada.	

Fuente: Elaboración propia

El sitio web debe permitir al docente visualizar la información general de la sala, los estudiantes que van subiendo los archivos a la sala y también obtener un link que permita ingresar de forma directa a los estudiantes.

Tabla 13: Historia de usuario – Reporte general

Historia de usuario	
Número: 5	Usuario: Docente
Nombre de la historia: Visualización detallada de las salas creadas de forma individual y general.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-05	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: El sistema debe permitir al docente poder visualizar la información detallada y general de la evaluación de los archivos que se encuentran en la sala.	
Observaciones: Se visualiza los datos de los estudiantes que hayan subido los archivos, y también el porcentaje de similitud que contengan cada archivo.	

Fuente: Elaboración propia

El sistema va a ir mostrando gráficamente barras de forma general todos los estudiantes que hayan subido el archivo, así también como de forma individual bien detallado todo el proceso.

Tabla 14: Historia de usuario – Subida de archivos

Historia de usuario	
Número: 6	Usuario: Estudiantes
Nombre de la historia: Gestión de subida de archivos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Requerimiento funcional: RF-06	Iteración asignada: 1
Programador responsable: Steven Lopez	
Descripción: El sistema permitirá a los estudiantes mediante el enlace compartido poder ingresar los datos personales y seleccionar el archivo que va a subir.	
Observaciones: solo se podrá subir un archivo y con las extensiones establecidas por el docente al momento de crear la sala.	

Fuente: Elaboración propia

El sistema permitirá al estudiante solo subir un archivo y con cierto tipo de extensiones establecidas, también deberá ingresar sus datos personales.

Gestión del cronograma

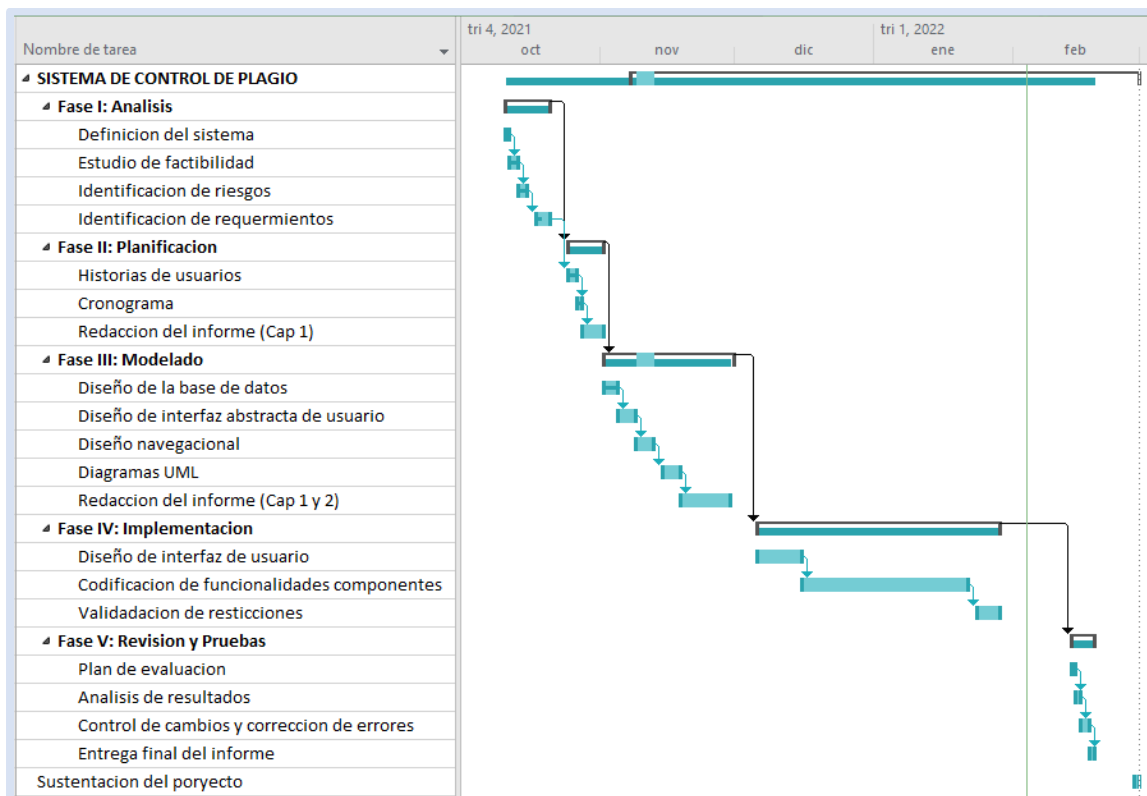
El siguiente cronograma desarrollado contiene todas las fases de la metodología SWIRL utilizada para el desarrollo de este prototipo en el cual se detallan todas las actividades que se deben realizar, así como indicando el tiempo de duración y la fecha en que se inicia y termina todo el proyecto. También se muestra el correspondiente diagrama de Gantt.

Figura 3: Cronograma de actividades

Nombre de tarea	Duración	Comienzo	Fin
▲ SISTEMA DE CONTROL DE PLAGIO	81 días	lun 8/11/21	lun 28/2/22
▲ Fase I: Analisis	8 días	lun 11/10/21	mié 20/10/21
Definición del sistema	1 día	lun 11/10/21	lun 11/10/21
Estudio de factibilidad	2 días	mar 12/10/21	mié 13/10/21
Identificación de riesgos	2 días	jue 14/10/21	vie 15/10/21
Identificación de requerimientos	3 días	lun 18/10/21	mié 20/10/21
▲ Fase II: Planificación	6 días	lun 25/10/21	lun 1/11/21
Historias de usuarios	2 días	lun 25/10/21	mar 26/10/21
Cronograma	1 día	mié 27/10/21	mié 27/10/21
Redacción del informe (Cap 1)	3 días	jue 28/10/21	lun 1/11/21
▲ Fase III: Modelado	21 días	mar 2/11/21	mar 30/11/21
Diseño de la base de datos	3 días	mar 2/11/21	jue 4/11/21
Diseño de interfaz abstracta de usuario	2 días	vie 5/11/21	lun 8/11/21
Diseño navegacional	4 días	mar 9/11/21	vie 12/11/21
Diagramas UML	4 días	lun 15/11/21	jue 18/11/21
Redacción del informe (Cap 1 y 2)	7 días	vie 19/11/21	lun 29/11/21
▲ Fase IV: Implementación	40 días	lun 6/12/21	vie 28/1/22
Diseño de interfaz de usuario	8 días	lun 6/12/21	mié 15/12/21
Codificación de funcionalidades componentes	27 días	jue 16/12/21	vie 21/1/22
Validación de restricciones	5 días	lun 24/1/22	vie 28/1/22
▲ Fase V: Revisión y Pruebas	5 días	lun 14/2/22	vie 18/2/22
Plan de evaluación	1 día	lun 14/2/22	lun 14/2/22
Análisis de resultados	1 día	mar 15/2/22	mar 15/2/22
Control de cambios y corrección de errores	2 días	mié 16/2/22	jue 17/2/22
Entrega final del informe	1 día	vie 18/2/22	vie 18/2/22
Sustentación del proyecto	1 día	lun 28/2/22	lun 28/2/22

Fuente: Elaboración propia

Figura 4: Diagrama de Gantt



Fuente: Elaboración propia

En la **figura 3 y 4**, se muestra un desglose completo y secuencial de todos los procesos requeridos por la metodología “SWIRL” en la cual se establece de manera obligatoria una estructura de las fechas en que se realizó o se debe realizar cada actividad establecida en el cronograma y en el diagrama de Gantt.

Fase de modelado

Diseño de base de datos

Figura 5: Base de datos No relacional

```
{
  "_id" : ObjectId("620f9c8b5ad325ca9aa7035f"),
  "nombre" : "Optativa II",
  "cant_estudiantes" : "10",
  "peso" : "1",
  "lenguaje" : "python",
  "comentario" : "Evaluacion Unidad 1",
  "creador" : ObjectId("620f9aff5ad325ca9aa7035e"),
  "fecha_inicio" : "2022-02-18",
  "fecha_cierre" : "2022-02-19",
  "clave" : "123456",
  "extensiones" : ".py",
  "cant_archivos" : "1",
  "seguidores" : [
    {
      "nombre" : "Juan",
      "apellido" : "Quezada",
      "correo" : "juan123@utmachala.edu.ec",
      "comentario" : "Evaluacion 1 Quezada",
      "archivo" : "65fd577c-90be-11ec-a2db-5ce0c509c69d.py",
      "resultado" : [
        {
          "correo" : "nicole123@utmachala.edu.ec",
          "porcentaje" : 30.2
        },
        {
          "correo" : "edison123@utmachala.edu.ec",
          "porcentaje" : 9.07
        }
      ]
    },
    "plagio" : 19.635
  ],
}
{
  "_id" : ObjectId("620f9aff5ad325ca9aa7035e"),
  "nombre" : "steven",
  "apellido" : "lopez",
  "correo" : "eslopez_est@utmachala.edu.ec",
  "clave" : "1234"
}
```

Fuente: Elaboración propia

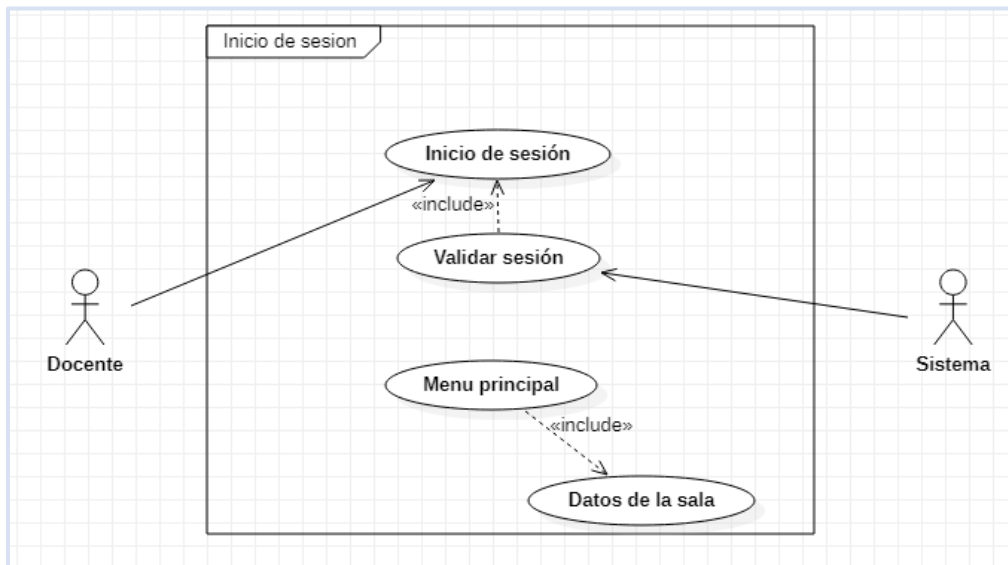
Como se puede observar en la figura 5, toda la información ingresada en la base de datos se encuentra estructurada de manera individual por cada docente registrado, por ejemplo, cada docente será único y todo lo que haga en relación a su cuenta no afectará de ninguna manera a otro usuario.

Diagramas UML

Los diagramas uml están conformados por muchos elementos gráficos que se los puede combinar de diferentes maneras, la finalidad de estos diagramas es observar de forma gráfica todas las diversas perspectivas que tiene el sistema.[36]

Diagramas de casos de uso

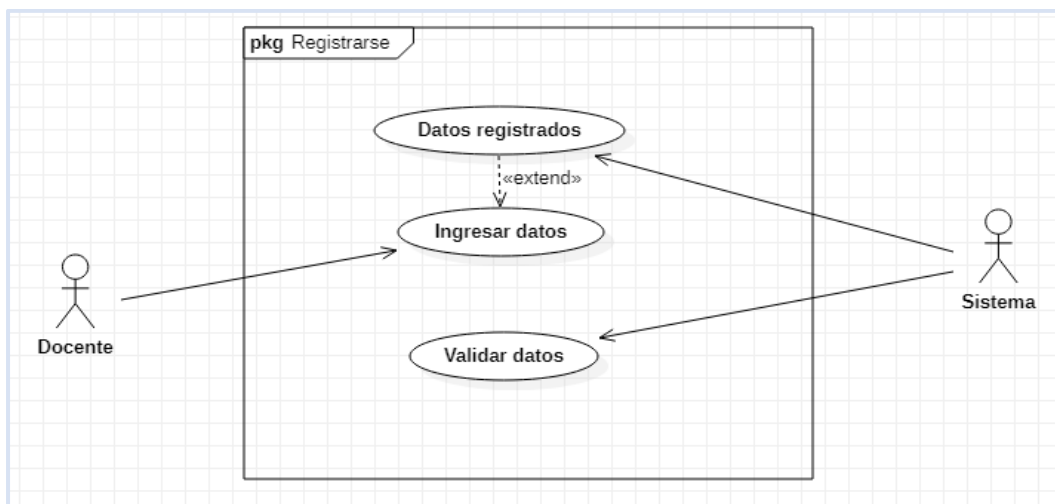
Figura 6: Caso de uso – Inicio de sesión



Fuente: Elaboración propia

En la **figura 6**, se observa que solo el docente podrá ingresar al sistema mediante la comprobación de sus datos como la clave, las mismas que son validadas por el sistema de manera interna, permitirá el ingreso de forma correcta al menú principal y tener acceso a todas las funcionalidades del sitio web.

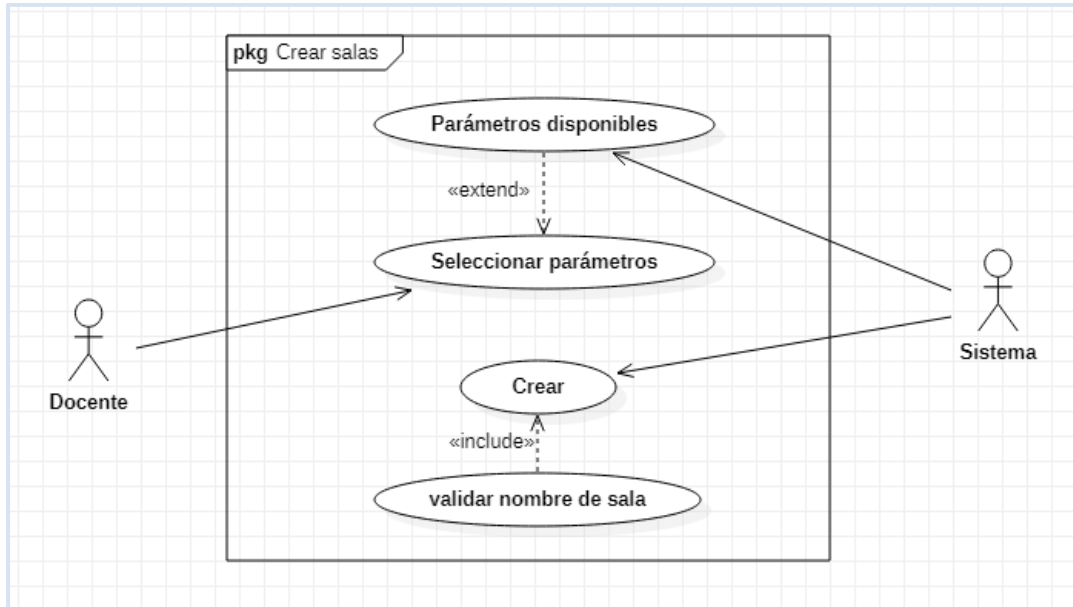
Figura 7: Caso de uso - Registrarse



Fuente: Elaboración propia

En la **figura 7**, se visualiza el módulo de registrarse el cual es solo para docentes que necesitan evaluar archivos de programación, este módulo requiere de manera obligatoria ingresar los datos personales como nombres, apellidos, correo y una clave.

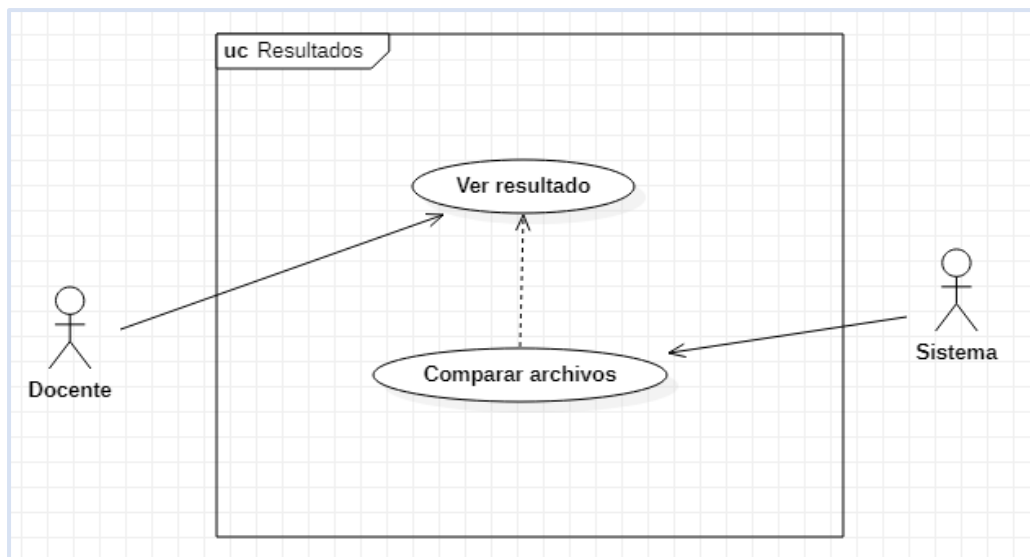
Figura 8: Caso de uso - Crear salas



Fuente: Elaboración propia

En la **figura 8**, el proceso de creación de salas se realiza mediante la selección del botón Crear sala, requiere de manera obligatoria seleccionar todos los parámetros necesarios.

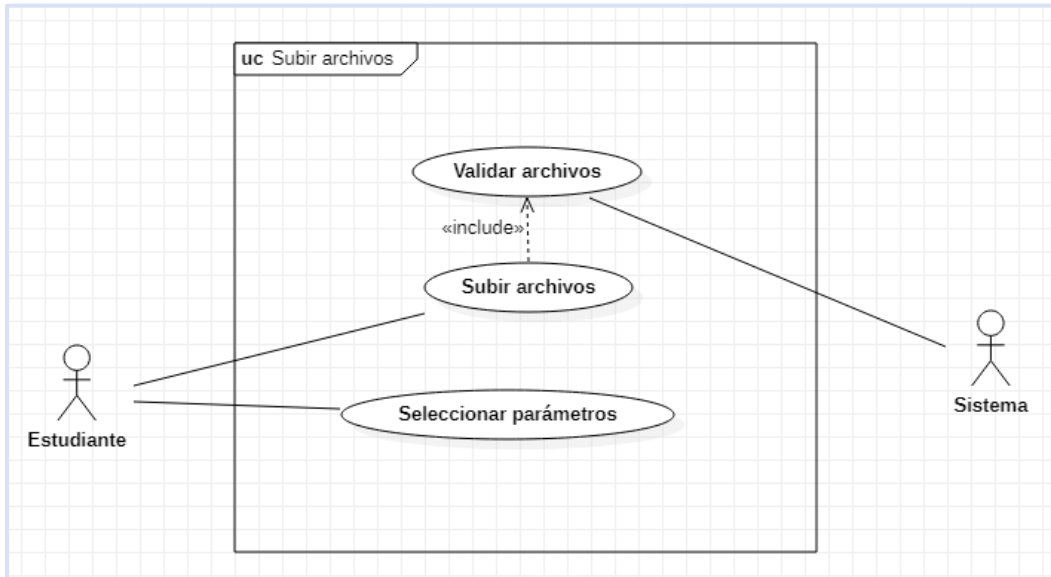
Figura 9: Caso de uso - Resultado



Fuente: Elaboración propia

En la **figura 9**, se visualiza de forma gráfica y general el porcentaje de plagio de cada uno de los estudiantes que subieron los archivos para evaluarlos.

Figura 10: Caso de uso - Subir archivos

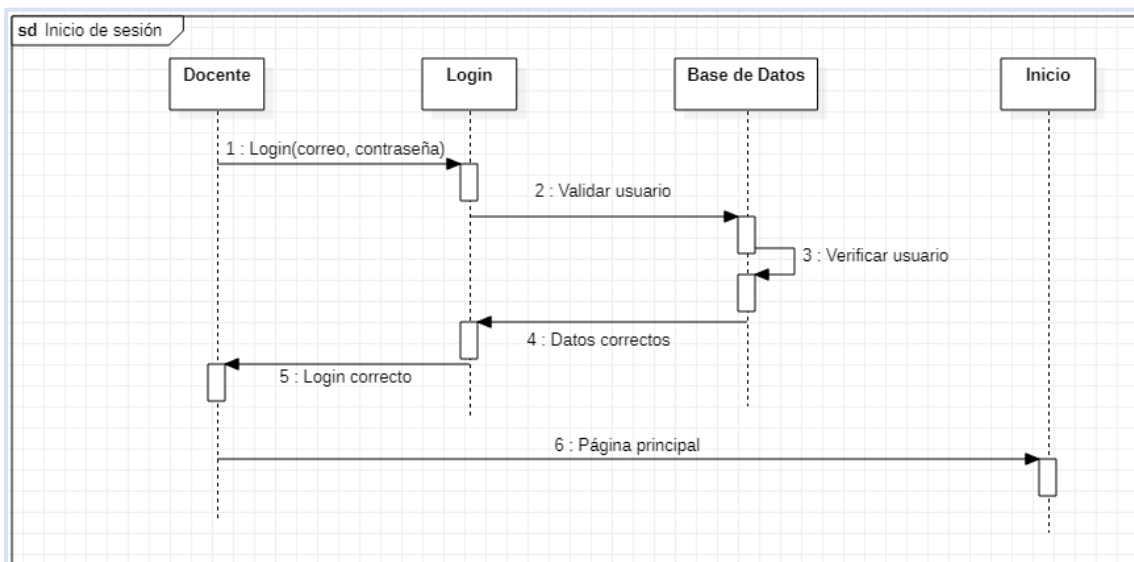


Fuente: Elaboración propia

En la **figura 10**, el estudiante puede subir el archivo y el sistema realiza la validación respectiva que corresponda para que se cumplan con los parámetros requeridos.

Diagramas de secuencia

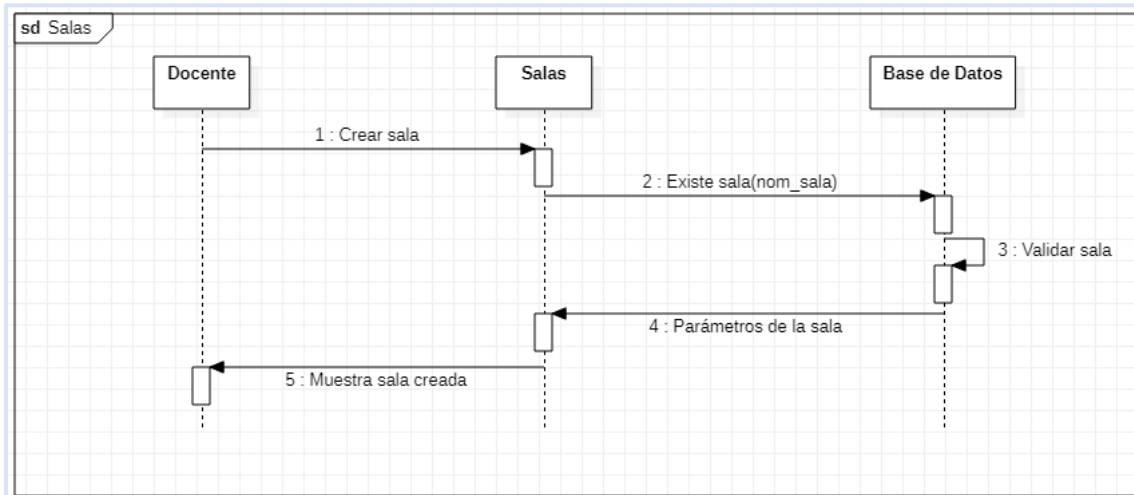
Figura 11: Diagrama de secuencia - Inicio de sesión



Fuente: Elaboración propia

En la **figura 11**, se visualiza el diagrama de secuencia cuando se inicia sesión en donde se presentan todas las secuencias de los pasos realizados por el docente en cada proceso mientras realiza esa actividad.

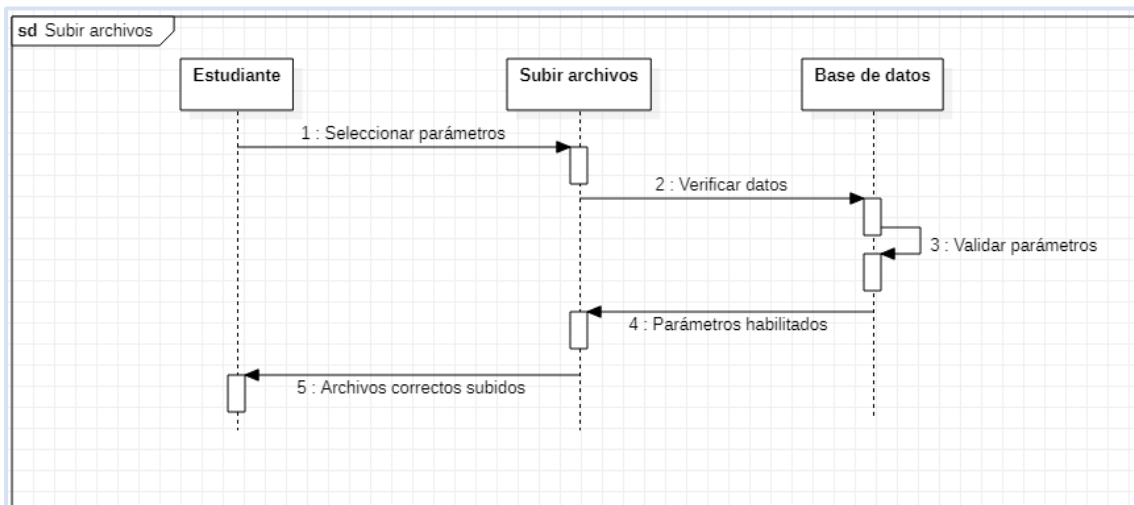
Figura 12: Diagrama de secuencia - Visualizar salas



Fuente: Elaboración propia

Se visualiza en la **figura 12**, todos los pasos de forma secuencial del proceso “Crear sala” que realiza el docente en donde esté especificado paso por paso en donde se comprueba si la sala existe y también donde se especifica todos los parámetros requeridos.

Figura 13: Diagrama de secuencia - Subir archivos

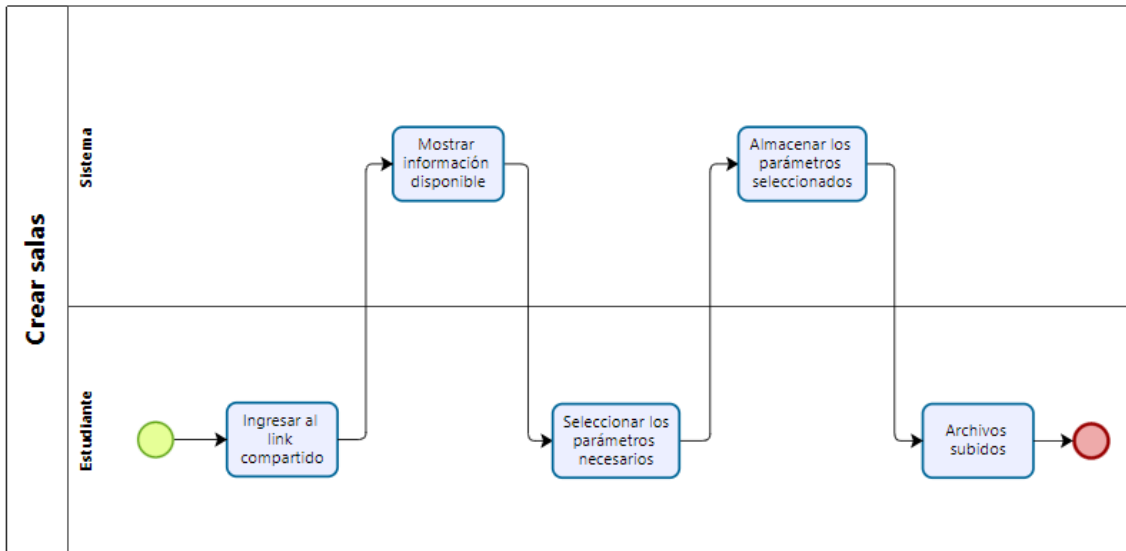


Fuente: Elaboración propia

En la **figura 13**, se observa los procesos de subida de archivos que el estudiante la realiza, así mismo debe ingresar los datos personales.

Diagramas de actividades

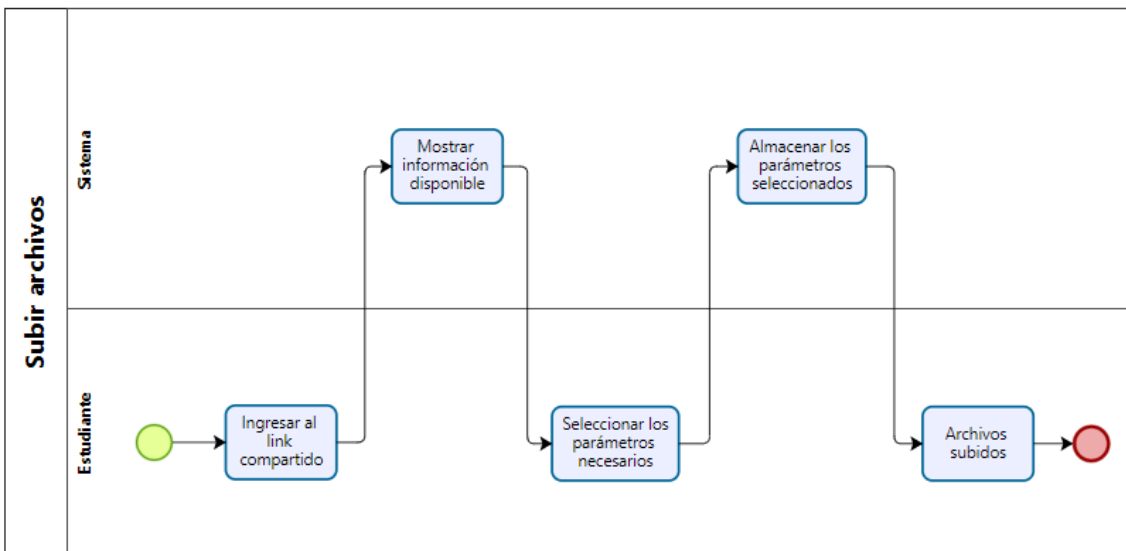
Figura 14: Diagrama de secuencia - Crear salas



Fuente: Elaboración propia

En la **figura 14**, se observa la validación que realiza el sistema al momento de crear una nueva sala ya que no puede repetirse algún nombre que ya esté en uso.

Figura 15: Diagrama de secuencia – Subir archivos



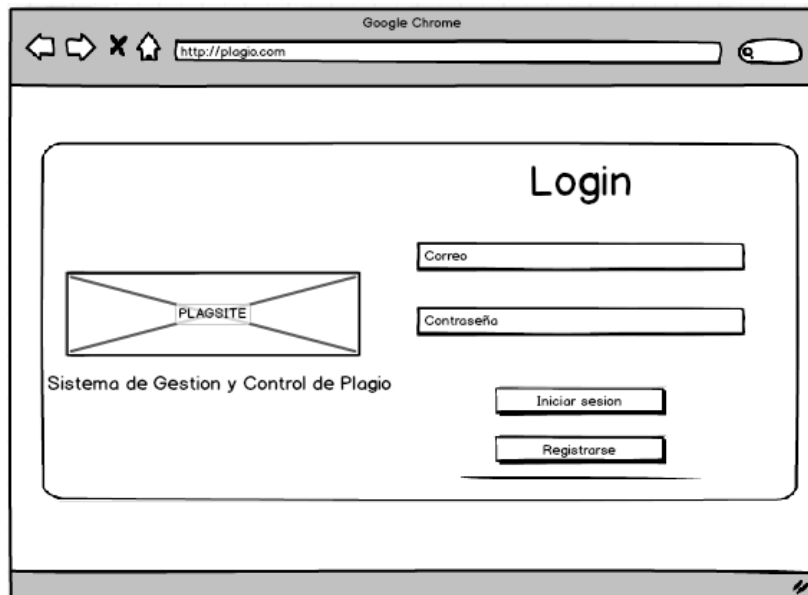
Fuente: Elaboración propia

En la **figura 15**, se componen las acciones que realiza el sistema al momento que el estudiante realiza la actividad de subir archivos.

Diseño de Interfaces

Se utilizó la herramienta Balsamiq Mockups para realizar el diseño de interfaces, esta herramienta es muy útil para crear modelos similares a la realidad de cómo será la presentación final de la aplicación web, a continuación, se mostrará todas las vistas con las que consta el sitio web tomando en cuenta los requisitos establecidos anteriormente.

Figura 16: Diseño de Interfaces: Inicio de sesión

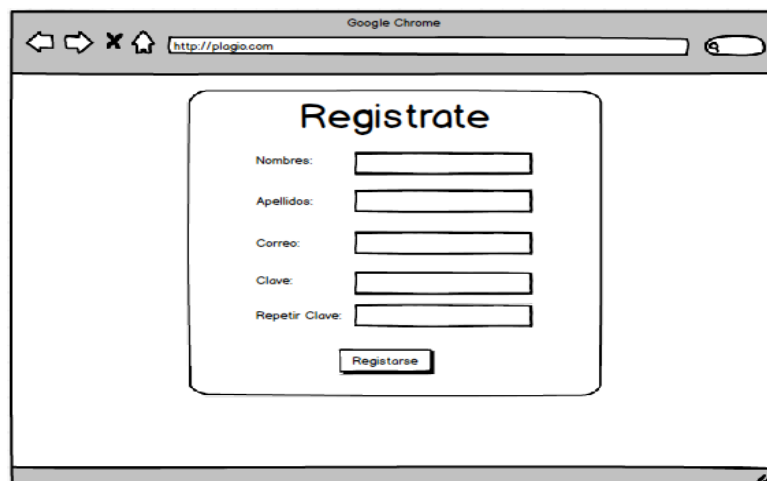


The image shows a wireframe of a login page in a Google Chrome browser window. The browser's address bar contains 'http://plagio.com'. The page content is enclosed in a rounded rectangle and features the following elements: a logo on the left consisting of a trapezoidal shape with the text 'PLAGSITE' inside; the text 'Sistema de Gestion y Control de Plagio' below the logo; the title 'Login' centered at the top; two input fields labeled 'Correo' and 'Contraseña'; and two buttons at the bottom labeled 'Iniciar sesion' and 'Registrarse'.

Fuente: Elaboración propia

En la **figura 16**, se observa la página del proceso de ingreso al sistema mediante el correo institucional y clave colocada por el estudiante al momento de registrarse y para finalizar también consta con un botón de ingresar.

Figura 17: Diseño de Interfaces: Registrarse

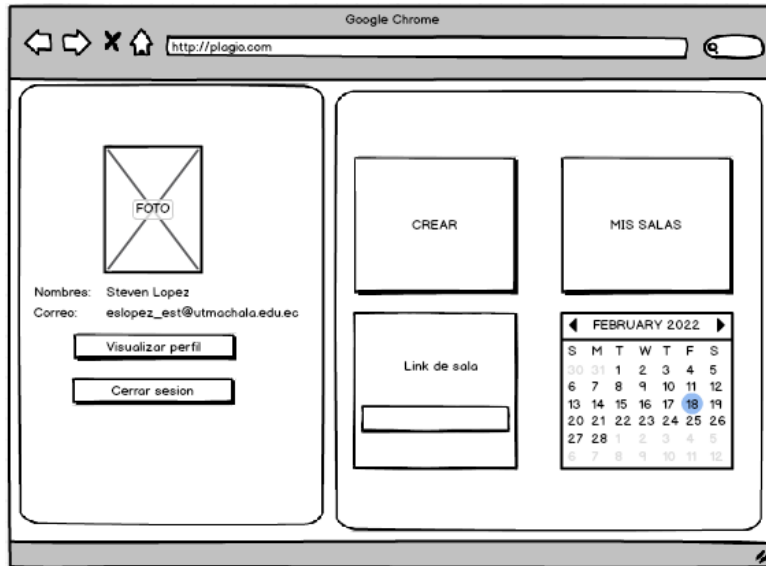


The image shows a wireframe of a registration page in a Google Chrome browser window. The browser's address bar contains 'http://plagio.com'. The page content is enclosed in a rounded rectangle and features the following elements: the title 'Registrate' centered at the top; five input fields labeled 'Nombre:', 'Apellidos:', 'Correo:', 'Clave:', and 'Repetir Clave:'; and a single button at the bottom labeled 'Registrarse'.

Fuente: Elaboración propia

En la **figura 17**, se muestra la ventana que permite registrar a un usuario (estudiante), la cual necesita los siguientes datos como son los nombres, apellidos, correo, clave y foto, para el registro correcto del usuario.

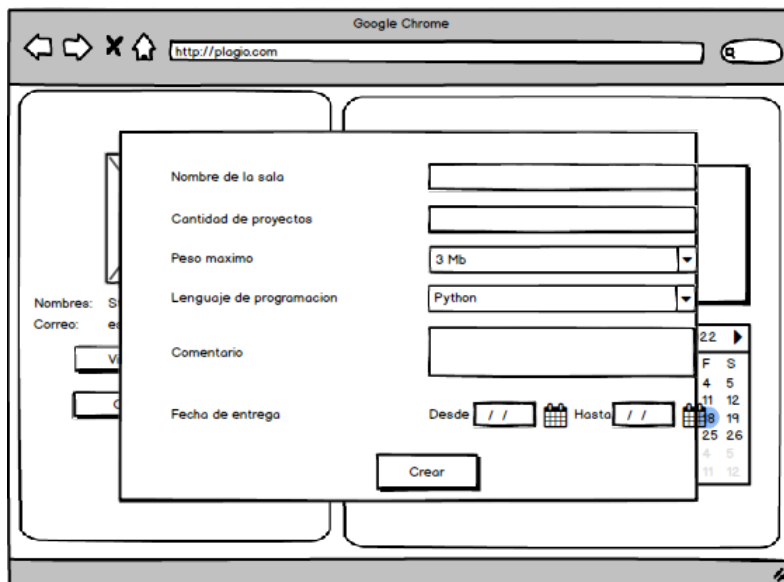
Figura 18: Diseño de Interfaces: Página principal



Fuente: Elaboración propia

En la **figura 18**, se muestra una ventana con las opciones principales que cuenta el docente que son: crear salas, ver las salas creadas y agendar una sala, adicionalmente se observa la información personal del docente.

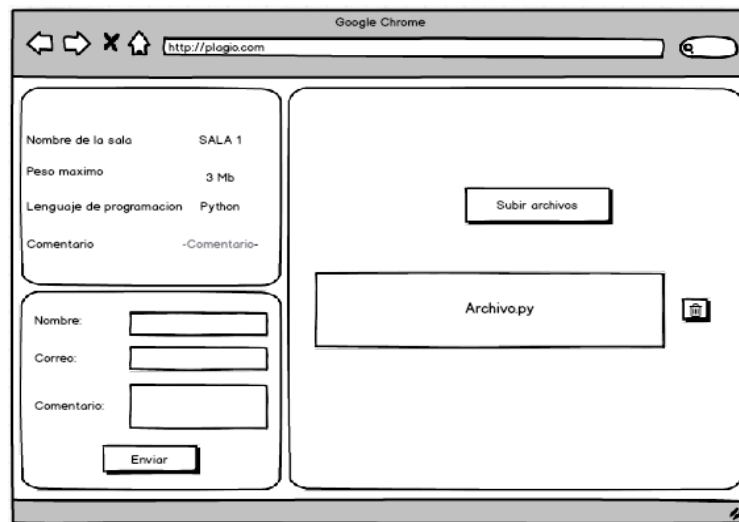
Figura 19: Diseño de Interfaces: Creación de salas



Fuente: Elaboración propia

En la **figura 19**, se procede a realizar el proceso de creación de una sala la cual va a ser utilizada por los estudiantes para subir los archivos, en esta sala se pedirán los siguientes datos como son: la cantidad de proyectos, peso máximo del archivo, tipo de lenguaje de programación que se van a comparar, algún comentario visible para los estudiantes y la fecha de entrega.

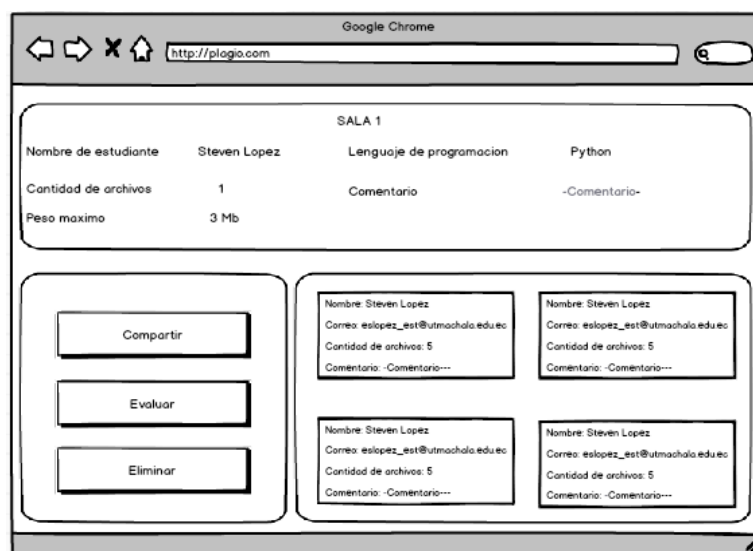
Figura 20: Diseño de Interfaces: Subir archivos



Fuente: Elaboración propia

En la siguiente interfaz de la **figura 20**, la aplicación web se observa ventana que se abrirá con el link que compartirá el docente con sus estudiantes, la cual consta con información de la sala creada, datos del estudiante que deberá ingresar y los archivos que va a subir.

Figura 21: Diseño de Interfaces: Visualización de resultados por sala



Fuente: Elaboración propia

En la **figura 21**, se podrá observar la información de la sala, así como los datos de cada estudiante que ya haya subido los archivos en la sala y también consta de unos botones que permitirán compartir, editar, evaluar y crear reporte de la sala.

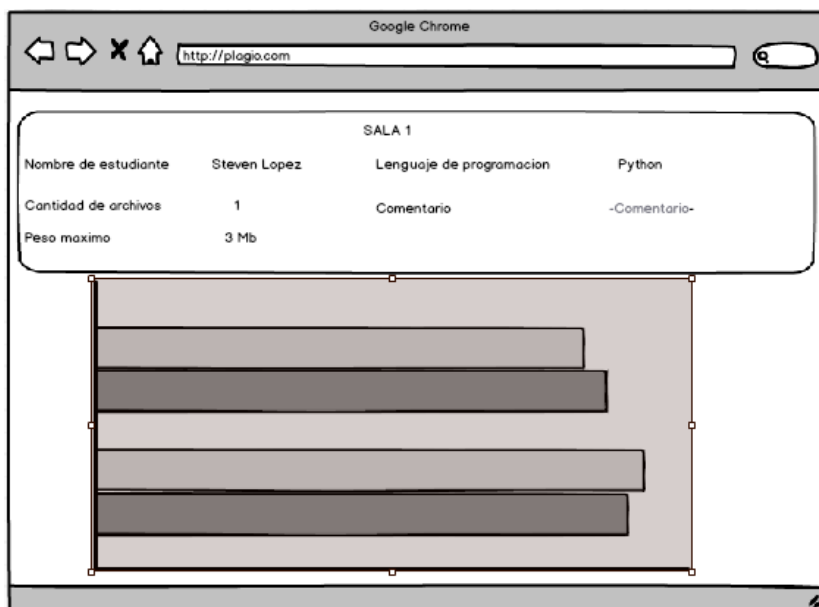
Figura 22: Diseño de Interfaces: Visualización de salas



Fuente: Elaboración propia

En la **figura 22**, la interfaz permitirá visualizar todas las salas que haya creado el docente para ir evaluando de manera progresiva a los estudiantes a lo largo del periodo académico.

Figura 23: Diseño de Interfaces: Visualizar resultados



Fuente: Elaboración propia

En la **figura 23**, permitirá al docente poder visualizar un reporte general de la sala creada con todos los datos de los estudiantes, fecha de entrega y porcentaje de similitud.

2.5 EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO

Página Principal de bienvenida

Figura 24: Inicio de sesión



PLAGSITE
Sistema Informático para la Gestión de Plagio

Ingreso al Sistema

Correo electrónico

Clave de seguridad

Iniciar Sesión

Crear Nueva Cuenta

Autor Steven Lopez

Fuente: Elaboración propia

Se muestra la página principal del prototipo en donde se puede apreciar el inicio de sesión y así también el botón que permite registrarse.

Figura 25: Registrarse



[Volver](#)

Registro de Usuarios

* Nombres: Nombres Completos

* Apellidos: Apellidos Completos

* Correo: Correo Electrónico

* Clave: Clave de Acceso

* Repetir Clave: Confirmar Clave

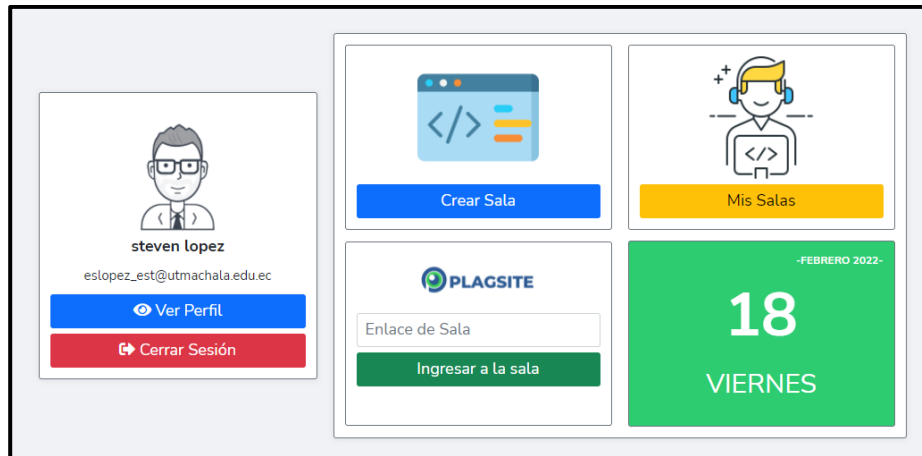
[+ Crear Usuario](#)

Fuente: Elaboración propia

En la figura 25, se muestra la ventana de registrarse en la cual los docentes realizaron ese proceso para tener una cuenta y poder tener acceso a más funcionalidades del sistema.

Páginas de Docentes

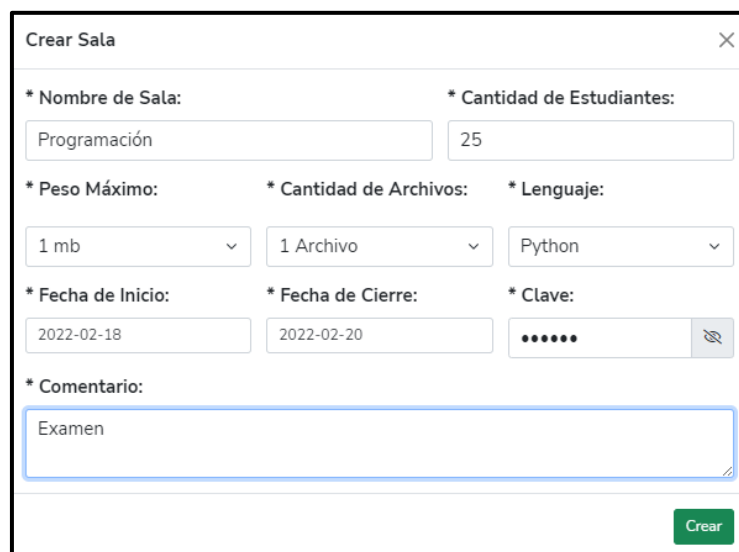
Figura 26: Menú principal



Fuente: Elaboración propia

Muestra el menú principal al docente luego de haber iniciado sesión en donde se visualiza la información personal, crear salas, ver las salas creadas y agendar

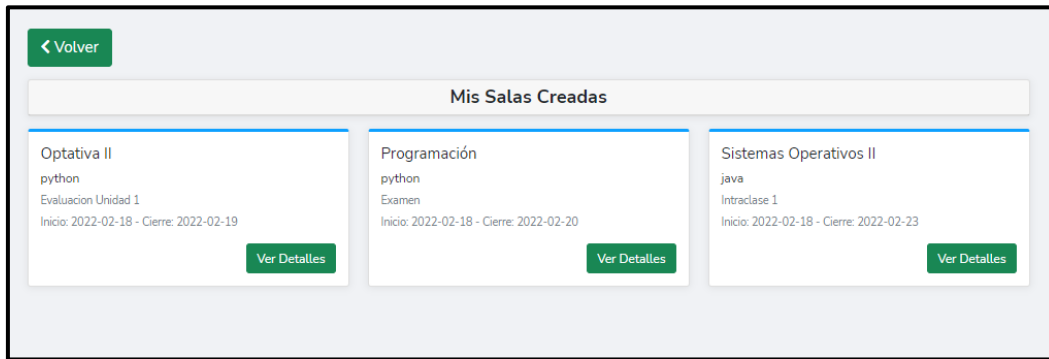
Figura 27: Crear sala



Fuente: Elaboración propia

Al presionar el botón de crear sala se muestra una ventana models que permitirá al docente ingresar toda la información necesaria para la creación de una sola.

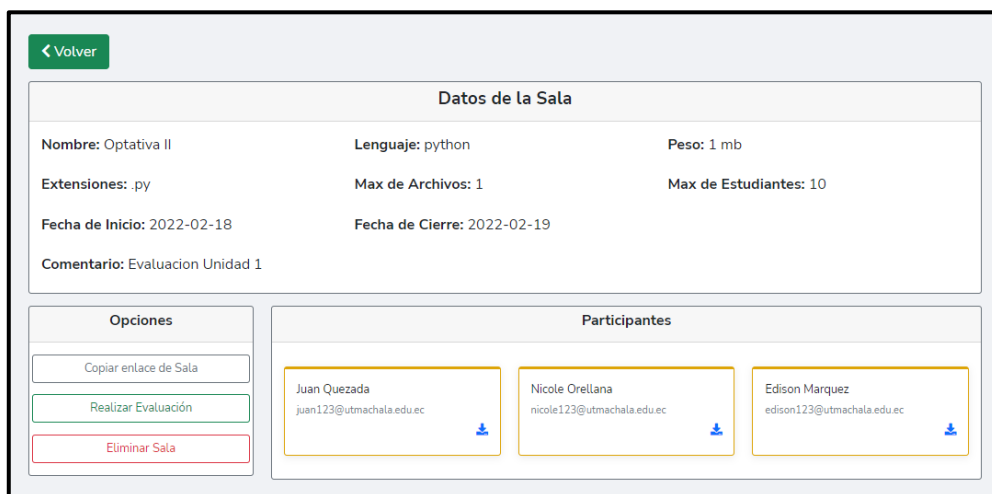
Figura 28: Salas creadas



Fuente: Elaboración propia

Al presionar el botón mis salas, se observa una nueva ventana que va a contener todas las salas creadas por el docente.

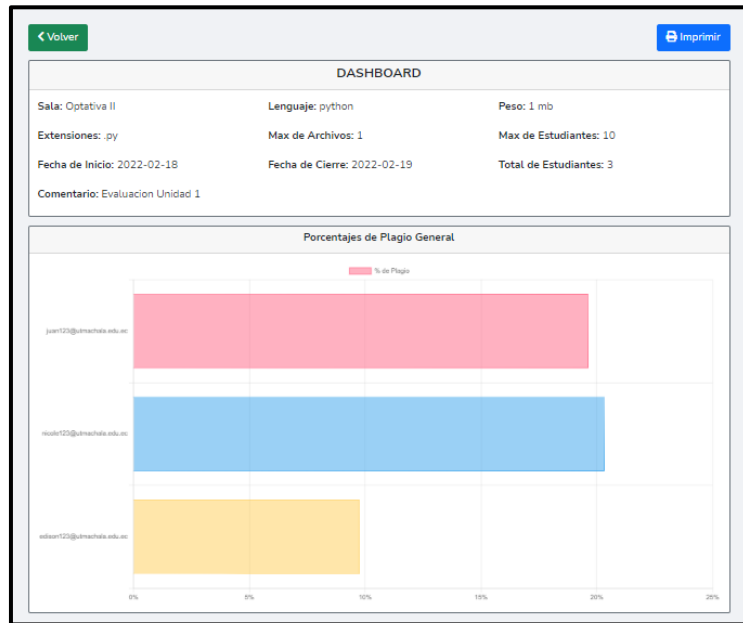
Figura 29: Detalle de la sala



Fuente: Elaboración propia

En la **figura 29**, se muestra el detalle de la sala creada con todos los usuarios que han subido los archivos para luego proceder con su comparación.

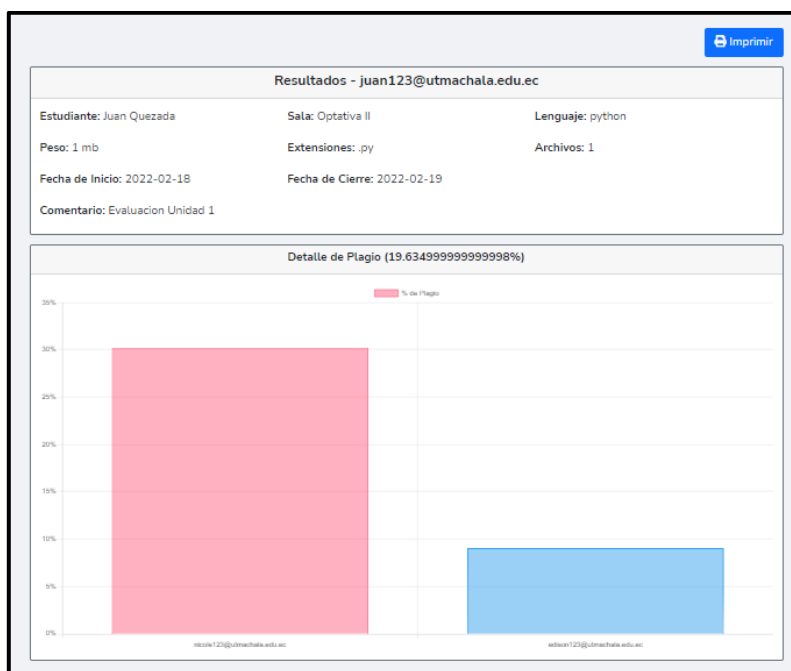
Figura 30: Dashboard general



Fuente: Elaboración propia

En la **figura 30**, se presenta un dashboard que en la parte superior contiene la información de la sala y en la parte inferior contiene gráficos de barras de forma general con el porcentaje de plagio de todos los estudiantes que han realizado el proceso de subir archivos.

Figura 31: Dashboard individual



Fuente: Elaboración propia

En la **figura 31**, se muestra un dashboard más detallado en el cual también presenta unos gráficos de barras para ver el nivel de porcentaje de plagio que el estudiante seleccionado tiene en relación con los demás.

Páginas de Estudiantes

Figura 32: Subida de archivos

The screenshot shows a web form titled "Datos de la Sala". At the top, it displays course information: "Sala: Optativa II", "Creador de Sala: steven lopez", and "Lenguaje: python". Below this is a "Comentario: Evaluacion Unidad 1".

The form is divided into two main sections:

- Datos Personales:** This section contains fields for "Nombres" (Edison), "Apellidos" (Marquez), "Correo" (edison123@utmachala.edu.ec), and "Comentario" (Evaluacion 1 Marquez).
- Subir Archivos:** This section shows upload settings: "Tipo de Archivos: .py", "Peso Max: 1 mb", and "Cantidad Max Archivos: 1". It also displays "Archivos Subidos: 1" and "Peso total de Archivos: 1.844 kb". A visual representation of the uploaded file shows a card with "1.8 KB" and "2222.py", with a "Remover archivo" link below it.

At the bottom of the form is a green button labeled "Enviar Trabajo".

Fuente: Elaboración propia

En la figura 32, se puede observar la ventana que abre con el link que se comparte la sala creada por el docente la cual permite ingresar los datos del estudiante y subir el archivo.

3. CAPÍTULO III. EVALUACIÓN DE LA APLICACIÓN DE WEB.

3.1. PLAN DE EVALUACIÓN.

Considerando la propuesta del proyecto en donde se definió que para el desarrollo de la documentación sería la metodología "SWIRL" se aplicaran los tres tipos de evaluación sobre el prototipo desarrollado para comprobar el nivel de calidad utilizando herramientas de evaluación que miden el nivel de SEO. Las métricas de calidad y de evaluación.

3.2. RESULTADOS DE EVALUACIÓN.

Evaluación de Calidad de la aplicación web

Se utilizó la norma estándar internacional ISO/IEC 9126 para la realizar la evaluación de calidad, esta norma contiene varias sub categorías que permiten evaluar los parámetros de fiabilidad, funcionalidad, rendimiento, velocidad, accesibilidad, etc.

Existen muchas herramientas en la web de medición de software dedicadas a la evaluación de los parámetros de las aplicaciones web, se utilizó un servidor web gratuito para poder subir la aplicación y poder realizar la evaluación requerida de forma satisfactoria utilizando las herramientas web.

Tabla 15: Evaluación de calidad ISO/IEC 9126

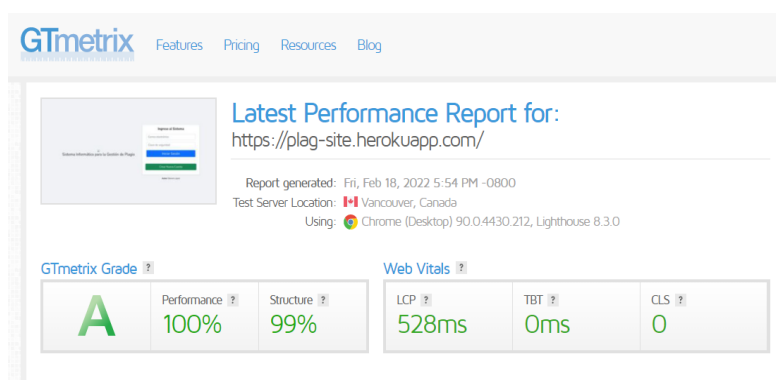
Estandar ISO/IEC 9126	funcionalidad	Accesibilidad	Fiabilidad	Eficiencia
GTmetrix	X		X	
WAVE		X		
W3C Validator				X

Fuente: Elaboración propia

GTmetrix

Es una página web orientada a medir la fiabilidad de las páginas web dependiendo del servidor en donde esta alojada la aplicación como el navegador en el que se está ejecutando.

Figura 33: Evaluación de calidad - GTmetrix



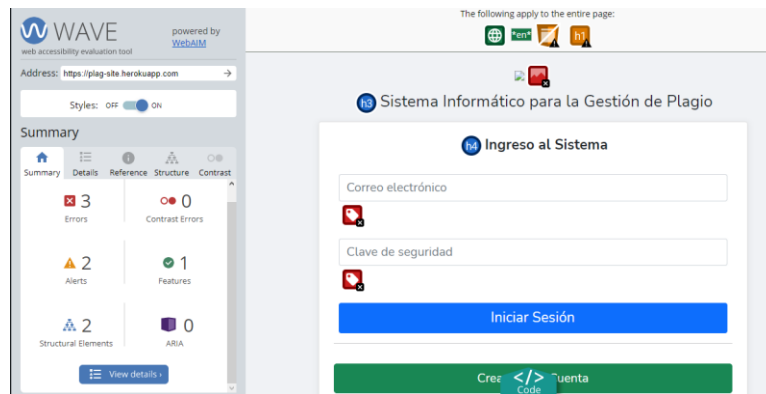
Fuente: Elaboración propia

En la figura 33, la aplicación utilizada muestra de forma gráfica la evaluación realizada al sitio web.

WAVE

Es una aplicación completa tiene buena portabilidad ya que se la puede ejecutar en el escritorio, móvil o web, permite medir la accesibilidad de las aplicaciones web.

Figura 34: Evaluación de calidad - WAVE



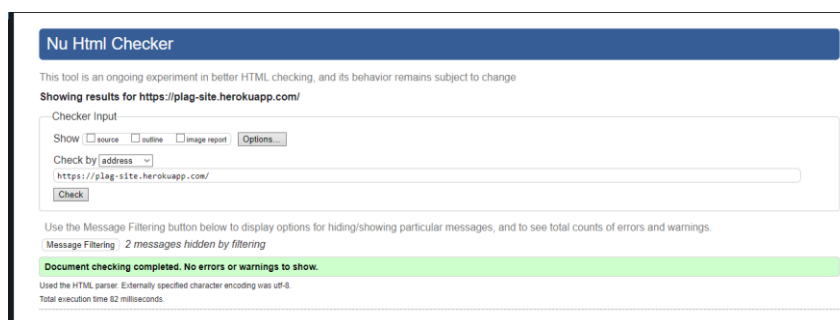
Fuente: Elaboración propia

En la figura 34, se observan varias opciones de estadísticas que reflejan muchas características para medir la accesibilidad de la aplicación web validando la cantidad de errores, alertas y elementos estructurales de los que está conformado.

W3C Validator

Este tipo de aplicación dedicada a la evaluación de eficiencia y seguridad de las páginas web considera de manera primordial el análisis a los archivos html de los cuales está conformada, así mismo detecta vulnerabilidades en JavaScript, etc.

Figura 35: Evaluación de calidad - W3C Validator



Fuente: Elaboración propia

Se puede observar en la **figura 36**, que la prueba realizada mediante esta herramienta fue de manera satisfactoria en donde no se mostró ningún tipo de error.

Tabla 16: Evaluación de calidad - Usabilidad y Portabilidad

EVALUACION CON BASE A LA NORMA ISO/IEC 9126		Muy Malo	Malo	Regular	Bueno	Excelente
Características	Módulo	1	2	3	4	5
Usabilidad	Comprensión				X	
	Atractividad				X	
	Aprendizaje					X
	Operatividad				X	
Portabilidad	Capacidad de instalación		X			
	Capacidad de reemplazamiento			X		

Fuente: Elaboración propia

En la **tabla 16** se muestra una tabla que permita la evaluación de portabilidad y la usabilidad de aplicaciones web.

3.2. CONCLUSIONES

- Se creó una plataforma web utilizando lenguaje de programación Python y mongoDB que permite comparar archivos con un tipo de extensión específico satisfaciendo la necesidad de evaluar las tareas de programación subidas por los estudiantes logrando obtener los porcentajes de plagio de forma general e individual.
- Se recopiló todos los requerimientos funcionales y no funcionales del sistema mediante la creación de historias de usuarios que permitan determinar de una manera clara y concisa los procesos.
- Se implementó la metodología “SWIRL” para realizar de forma bien estructurada la documentación incluyendo todas las fases por las que está

compuesta esta metodología que permite el desarrollo web de una manera eficiente.

- Se utilizó múltiples herramientas CASE para la creación de diagramas de actividades, de casos de uso, Cronograma y prototipado que permiten ver de manera gráfica y como está estructurado el proyecto.
- Se desarrolló una plataforma web utilizando Python, flask y Bootstrap que permitió su desarrollo e implementación de una forma eficiente y amigable para los usuarios.
- Se utilizó la norma de estándar internacional de calidad ISO/IEC 9126 para la realizar la evaluación de calidad de la plataforma web garantizando que cumpla con todas las acciones que requiera el sistema.

3.3. RECOMENDACIONES

De acuerdo a lo implementado a lo largo del desarrollo de esta aplicación web tanto en la práctica como en lo teórico, considero recomendar ciertos criterios:

- Utilizar lenguajes de desarrollo de programación que sean flexibles y potentes, para que permitan utilizar de manera óptima los recursos de la máquina.
- Utilizar bases de datos no relacionales porque se adaptan a los cambios que se van generando en el transcurso del desarrollo de la aplicación web.
- Realizar pruebas individuales de cada funcionalidad para ir comprobando que no existan errores.
- Utilizar Bootstrap ya que facilita la creación de diseños de las plantillas en cada archivo html creado en la aplicación web.

BIBLIOGRAFÍA

- [1] J. Calvo, J. Gracia, y E. Bayo, «Robust design to optimize client–server bi-directional communication for structural analysis web applications or services», *Adv. Eng. Softw.*, vol. 112, pp. 136-146, oct. 2017, doi: 10.1016/j.advengsoft.2017.04.010.
- [2] L. A. Acosta, F. A. Becerra, y D. Jaramillo, «Sistema de Información Estratégica para la Gestión Universitaria en la Universidad de Otavalo (Ecuador)», *Form. Univ.*, vol. 10, n.º 2, pp. 103-112, 2017, doi: 10.4067/S0718-50062017000200011.
- [3] A. D. Rodríguez Peña y L. G. Silva Rojas, «Arquitectura de software para el sistema de visualización médica Vismedic», *Rev. Cuba. Informática Médica*, vol. 8, n.º 1, pp. 75-86, jun. 2016.
- [4] Z. M. Rodríguez, L. D. P. Rodríguez, y J. C. G. Suarez, «Arquitectura basada en Microservicios y DevOps para una ingeniería de software continua», *Ind. Data*, vol. 23, n.º 2, pp. 141-149, 2020, doi: 10.15381/idata.v23i2.17278.

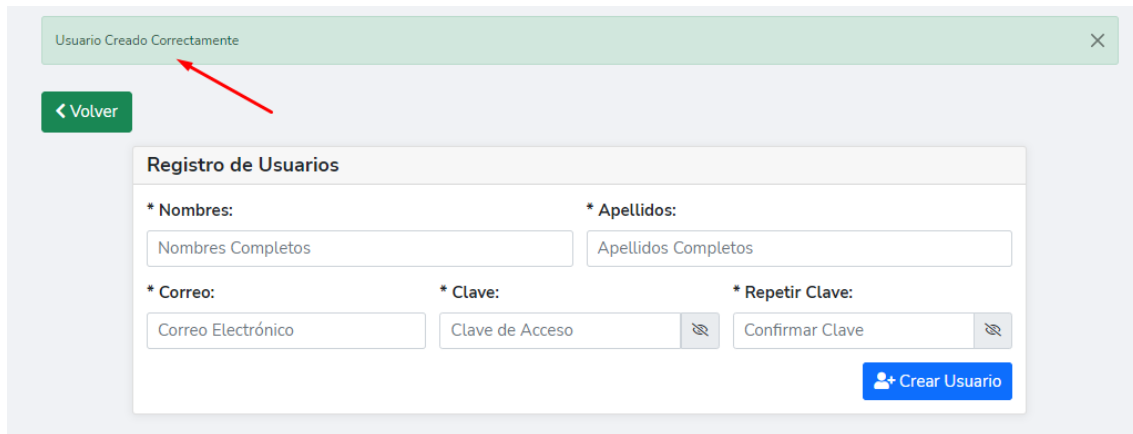
- [5] E. Gonzaga, J. Alvarez-Cedillo, y A. Mejía, «Arquitecturas en n-Capas: Un Sistema Adaptivo», *Polibits*, vol. 34, pp. 34-37, jul. 2006, doi: 10.17562/PB-34-7.
- [6] L. F. Maldonado-Granados *et al.*, «Software for ontological and collaborative online representation of knowledge», *Ing. Investig. Tecnol.*, vol. 19, n.º 2, pp. 147-158, jun. 2018, doi: 10.22201/fi.25940732e.2018.19n2.013.
- [7] O. Villavicencio, M. Ordóñez, M. Pardo, y R. Ramón, «Comparativa de tendencias de desarrollo de software móvil», *3C TIC Cuad. Desarro. Apl. Las TIC*, vol. 10, pp. 123-147, mar. 2021, doi: 10.17993/3ctic.2021.101.123-147.
- [8] J. Llamuca-Quinaloa, Y. Vera-Vincent, V. Tapia-Cerda, J. Llamuca-Quinaloa, Y. Vera-Vincent, y V. Tapia-Cerda, «Análisis comparativo para medir la eficiencia de desempeño entre una aplicación web tradicional y una aplicación web progresiva», *Tecnológicas*, vol. 24, n.º 51, pp. 164-185, ago. 2021, doi: 10.22430/22565337.1892.
- [9] J. R. Molina Ríos, M. P. Zea Ordóñez, M. J. Contento Segarra, y F. G. García Zerda, «COMPARACIÓN DE METODOLOGÍAS EN APLICACIONES WEB», *3C Tecnol. Innov. Apl. Pyme*, vol. 7, n.º 1, pp. 1-19, mar. 2018, doi: 10.17993/3ctecno.2018.v7n1e25.1-19.
- [10] L. de la C. Delgado Olivera, L. M. Díaz Alonso, L. de la C. Delgado Olivera, y L. M. Díaz Alonso, «Modelos de Desarrollo de Software», *Rev. Cuba. Cienc. Informáticas*, vol. 15, n.º 1, pp. 37-51, mar. 2021.
- [11] J. P. Zumba y C. A. L. Arreaga, «Evolución de las metodologías y modelos utilizados en el desarrollo de software.», *INNOVA Res. J.*, vol. 3, n.º 10, Art. n.º 10, oct. 2018, doi: 10.33890/innova.v3.n10.2018.651.
- [12] C. Boaventura José, E. Peña Herrera, P. Verdecia Vicet, y Y. Fustiel Alvarez, «Elección entre una metodología ágil y tradicional basado en técnicas de soft computing», *Rev. Cuba. Cienc. Informáticas*, vol. 10, pp. 145-158, 2016.
- [13] F. Sáenz Blanco, F. Gutiérrez Sierra, y J. C. Ramos Rivera, «ESTABLISHMENT OF AGILE TEAMS FOR SOFTWARE DEVELOPMENT: REVIEW OF LITERATURE», *Dimens. Empres.*, vol. 16, n.º 2, pp. 39-54, dic. 2018, doi: 10.15665/rde.v16i2.1042.
- [14] M. A. Santos-Romero y N. E. Escudero-López, «Propuesta de Metodología Híbrida y Base de Documentación para el Desarrollo de Software Actual», *Concienc. Tecnológica*, n.º 60, 2020, Accedido: 9 de febrero de 2022. [En línea]. Disponible en: <https://www.redalyc.org/journal/944/94465715002/html/>
- [15] P. Patanakul y R. Rufo-McCarron, «Transitioning to agile software development: Lessons learned from a government-contracted program», *J. High Technol. Manag. Res.*, vol. 29, n.º 2, pp. 181-192, nov. 2018, doi: 10.1016/j.hitech.2018.10.002.
- [16] J. Ríos y N. Pedreira, «SWIRL», *metodología para el diseño y desarrollo de aplicaciones web*. 2019. doi: 10.17993/IngyTec.2019.55.
- [17] J. R. M. Ríos, «Modelo de evaluación de metodologías de desarrollo de software web», <http://purl.org/dc/dcmitype/Text>, Universidade da Coruña, 2021. Accedido: 17 de febrero de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=295964>
- [18] V. Cortellessa, R. Eramo, y M. Tucci, «From software architecture to analysis models and back: Model-driven refactoring aimed at availability improvement», *Inf. Softw. Technol.*, vol. 127, p. 106362, nov. 2020, doi: 10.1016/j.infsof.2020.106362.
- [19] «Documentation for Visual Studio Code». <https://code.visualstudio.com/docs> (accedido 17 de febrero de 2022).
- [20] M. S. Prokopyev *et al.*, «Desarrollo de un curso de programación para estudiantes de una institución de educación superior de formación docente utilizando el lenguaje de programación Python», *Propósitos Represent.*, vol. 8, n.º 3, sep. 2020, doi: 10.20511/pyr2020.v8n3.484.

- [21] J. M. Valencia-Henao, G. R. Solarte-Martínez, y L. E. Muñoz-Guerrero, «Análisis y Diseño de Interfaz para conexión de PLC basado en Arduino con PLC Allen Bradley a través de Ethernet», *Sci. Tech.*, vol. 24, n.º 3, pp. 504-511, 2019.
- [22] D. J. Bastidas-Logroño, Espíndola-Lara, y A. D. Palma-Rivera, «Implementación del bootstrap como una metodología ágil en la web», *Rev. Arbitr. Interdiscip. Koinonía*, vol. 5, n.º 9, pp. 268-287, 2020, doi: 10.35381/r.k.v5i9.648.
- [23] S. Bjeladinovic, Z. Marjanovic, y S. Babarogic, «A proposal of architecture for integration and uniform use of hybrid SQL/NoSQL database components», *J. Syst. Softw.*, vol. 168, p. 110633, oct. 2020, doi: 10.1016/j.jss.2020.110633.
- [24] G. E. Vega, M. T. Villalobos, L. V. Acuña, y R. Q. Oviedo, «Una comparación de rendimiento entre bases de datos NoSQL: MongoDB y ArangoDB», *Tecnol. En Marcha*, vol. 32, n.º Extra 9, pp. 5-15, 2019, doi: 10.18845/tm.v32i6.4223.
- [25] A. Celesti, M. Fazio, y M. Villari, «A Study on Join Operations in MongoDB Preserving Collections Data Models for Future Internet Applications», *Future Internet*, vol. 11, p. 83, mar. 2019, doi: 10.3390/fi11040083.
- [26] F. Moreno, J. Rendón, y R. Vásquez, «Una comparación de rendimiento entre Oracle y MongoDB», *Cienc. E Ing. Neogranadina*, vol. 26, p. 109, abr. 2016, doi: 10.18359/rcin.1669.
- [27] R. P. Fuentes, A. R. Morffi, y F. R. García, «Una herramienta para la evaluación de la calidad de modelos de procesos de negocio», vol. 14, n.º 4, p. 17, 2020, doi: <https://rcci.uci.cu/?journal=rcci>.
- [28] E. Kurilovas y S. Kubilinskiene, «Lithuanian case study on evaluating suitability, acceptance and use of IT tools by students – An example of applying Technology Enhanced Learning Research methods in Higher Education», *Comput. Hum. Behav.*, vol. 107, p. 106274, jun. 2020, doi: 10.1016/j.chb.2020.106274.
- [29] G. R. Solarte-Martínez, F. Silva Castro, y L. E. Muñoz-Guerrero, «Análisis, diseño y desarrollo de un prototipo de software para la administración de parqueaderos», *Ing. Compet.*, vol. 22, n.º 1, pp. 1-13, ene. 2020, doi: 10.25100/iyc.v22i1.8752.
- [30] C. Raibulet y F. Arcelli Fontana, «Collaborative and teamwork software development in an undergraduate software engineering course», *J. Syst. Softw.*, vol. 144, pp. 409-422, oct. 2018, doi: 10.1016/j.jss.2018.07.010.
- [31] Universidad Pedagógica y Tecnológica de Colombia, Tunja - Colombia, M. Callejas-Cuervo, A. C. Alarcón-Aldana, Universidad Pedagógica y Tecnológica de Colombia, Tunja - Colombia, A. M. Álvarez-Carreño, y Universidad Pedagógica y Tecnológica de Colombia, Tunja - Colombia, «Modelos de calidad del software.», *ENTRAMADO*, vol. 13, n.º 1, pp. 236-250, 2017, doi: 10.18041/entramado.2017v13n1.25125.
- [32] M. E. Uribe Macías, «Analysis of construction projects stakeholders from corporate social responsibility», *Cuad. Adm.*, vol. 33, n.º 58, pp. 58-76, ago. 2017, doi: 10.25100/cdea.v33i58.4517.
- [33] C. A. Győrödi, D. V. Dumșe-Burescu, D. R. Zmaranda, R. Ș. Győrödi, G. A. Gabor, y G. D. Pecherle, «Performance Analysis of NoSQL and Relational Databases with CouchDB and MySQL for Application's Data Storage», *Appl. Sci.*, vol. 10, n.º 23, Art. n.º 23, ene. 2020, doi: 10.3390/app10238524.
- [34] K. I. Gómez Sotelo, C. Baron, P. Esteban, C. Y. A. G. Estrada, y L. de J. Laredo Velázquez, «How to find non-functional requirements in system developments», *IFAC-Pap.*, vol. 51, n.º 11, pp. 1573-1578, ene. 2018, doi: 10.1016/j.ifacol.2018.08.272.
- [35] A. J. Gregory, J. P. Atkins, G. Midgley, y A. M. Hodgson, «Stakeholder identification and engagement in problem structuring interventions», *Eur. J. Oper. Res.*, vol. 283, n.º 1, pp. 321-340, may 2020, doi: 10.1016/j.ejor.2019.10.044.
- [36] J. Soler, I. Boada, F. Prados, y J. Poch, «Un Nuevo Enfoque para la Puntuación Automática de Problemas cuya Resolución se basa en Diagramas», *Form. Univ.*, vol. 10, n.º 1, pp. 47-60, 2017, doi: 10.4067/S0718-50062017000100006.

ANEXOS

ANEXO A: MENSAJES DE VALIDACIONES

Figura 36: Anexo - Registro de usuario



The screenshot displays a web interface for user registration. At the top, a green notification bar contains the text "Usuario Creado Correctamente" with a close button (X) on the right. Below this, a green button labeled "Volver" with a left-pointing arrow is visible. The main form, titled "Registro de Usuarios", includes several input fields: "Nombres Completos" (with a red arrow pointing to the success message), "Apellidos Completos", "Correo Electrónico", "Clave de Acceso" (with a strength indicator icon), and "Confirmar Clave" (with a strength indicator icon). A blue button labeled "Crear Usuario" with a user icon is positioned at the bottom right of the form.

Fuente: Elaboración propia

Figura 37: Anexo - Validación de usuario



The screenshot shows the login page for "PLAGSITE". At the top, a red notification bar displays the error message "Correo o Clave incorrecto" with a close button (X). The page features the "PLAGSITE" logo and the text "Sistema Informático para la Gestión de Plagio". On the right, there is a login form titled "Ingreso al Sistema" with fields for "Correo electrónico" and "Clave de seguridad", a blue "Iniciar Sesión" button, and a green "Crear Nueva Cuenta" button. The footer of the page reads "Autor Steven Lopez".

Fuente: Elaboración propia

ANEXO B: ANÁLISIS DE RIESGOS

Tabla 17: Anexos - Matriz de riesgos

N.	RIESGO	CONSECUENCIA	PREVENCION
RG-1	Recursos tecnológicos escasos.	Mala definición de la arquitectura del proyecto	Estructuración de forma optimizada
RG-2	Daños físicos de los equipos informáticos.	Desastre natural	Respaldar información en la nube
RG-3	Incompatibilidad de herramientas utilizadas	Malas prácticas en el desarrollo de los procesos del proyecto.	Verificar la guía de documentación.
RG-4	Procesos de diseño y desarrollo demorados	Disponibilidad de tiempo	Cumplir actividades del cronograma.

Fuente: Elaboración propia