

UNIVERSIDAD LUTERANA SALVADOREÑA

FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACION



PROYECTO:

SISTEMA REPOSITORIO DE DOCUMENTOS ACADEMICOS SRDA

EQUIPO DESARROLLADOR:

APELLIDOS, NOMBRES

CARNET

Martínez Cornejo, José Guillermo

MC24342

Melgar Zepeda, Yeimy Elizabeth

MZ24290

Gil Martínez, Johana Marcela

GM24239

Miranda Martínez, Anthony Adalberto

MM24452

Díaz Chávez, María Concepción

DC2412

Carrillo Jorge, Estefanny Yessenia

CJ24340

CÁTEDRA: Programación Orientada a Objetos

HORARIO: sábado 10:00:00 AM a 01:20:00 PM

FACILITADOR: Lic. Rigoberto Israel Orellana Orellana

SAN SALVADOR, 02 DE DICIEMBRE DE 2024

Introducción

¿Qué es un Sistema Repositorio?

Un sistema repositorio es una herramienta digital diseñada para almacenar, organizar y preservar de manera eficiente una gran variedad de información. Esta información puede ser de cualquier tipo: documentos pdf. Los repositorios son fundamentales para compartir conocimiento, facilitar la colaboración y garantizar la preservación a largo plazo de la información.

¿A quién va dirigido?

Los sistemas repositorios pueden estar dirigidos a un público muy amplio, dependiendo de su propósito. Algunos ejemplos incluyen:

- **Instituciones académicas:** Para almacenar y compartir investigaciones, tesis, publicaciones y otros materiales académicos.
- **Organizaciones gubernamentales:** Para gestionar documentos oficiales, datos estadísticos y otros recursos de interés público.
- **Empresas:** Para almacenar y compartir documentos internos, gestionar proyectos y facilitar la colaboración entre equipos.
- **Comunidades científicas:** Para compartir datos de investigación, código fuente y otros recursos científicos.

¿Cuáles son las ventajas de un sistema repositorio?

- **Acceso centralizado:** Toda la información se encuentra en un solo lugar, lo que facilita la búsqueda y el acceso.
- **Organización:** La información se organiza de forma lógica, lo que facilita su localización.
- **Preservación:** Los sistemas repositorios están diseñados para garantizar la preservación a largo plazo de la información.
- **Compartir:** Los repositorios facilitan el intercambio de información con otros usuarios.
- **Versionamiento:** Permite gestionar diferentes versiones de un mismo documento.
- **Metadatos:** La información se enriquece con metadatos, lo que facilita la búsqueda y la recuperación.

Contenido	
Análisis del Problema	4
Tema	4
Objetivos	4
Objetivo General	4
Objetivos Específicos	4
Análisis de beneficios del proyecto	4
Justificación	5
Metodología y ciclo de vida utilizados	6
Fases del Ciclo de Vida Cíclico	6
Características del Ciclo de Vida Cíclico	7
Ventajas del Ciclo de Vida Cíclico	7
Aplicaciones del Ciclo de Vida Cíclico.....	7
Análisis	8
Desarrollo	13
Conclusiones	16
Recomendaciones	20
Fuentes de información (referencias)	21
Anexos	22

Análisis del Problema

Tema:

Desarrollo del Sistema Repositorio de Documentos Académicos
(SRDA)

Problemática a resolver

- **Falta de centralización de documentos**
Los documentos académicos, como tesis, proyectos de investigación, artículos y documentos de trabajo, se encuentran dispersos en múltiples plataformas o en formatos físicos.
- Dificultad para acceder a la información, pérdida de documentos y duplicación de esfuerzos.

Objetivos

Objetivo General

Desarrollar un **Sistema de Repositorio de Documentos Académicos** que permita almacenar, gestionar, clasificar y compartir de manera centralizada documentos académicos, garantizando un acceso fácil, seguro y organizado.

Objetivos Específicos

- Centralizar los documentos académicos: Crear un repositorio único para tesis, artículos, investigaciones y otros documentos académicos.
- Organización eficiente: Diseñar un sistema que permita clasificar los documentos por categorías.

Análisis de beneficios del proyecto

- Mejor acceso a recursos: Los estudiantes e investigadores podrán acceder fácilmente a materiales relevantes para sus estudios y proyectos.
- Colaboración: Facilita el intercambio de conocimientos y fomenta la colaboración entre instituciones educativas y científicos.

Justificación

El desarrollo de un **Sistema de Repositorio de Documentos Académicos** es esencial para mejorar la eficiencia, accesibilidad y visibilidad de los recursos académicos. En el contexto actual, donde el acceso rápido y organizado a la información es vital para la educación e investigación, este sistema ofrece soluciones efectivas a los problemas de centralización, acceso y gestión de documentos.

1. **Relevancia Académica:** Un repositorio centralizado facilita la consulta de investigaciones previas, promoviendo la continuidad y ampliación del conocimiento.
2. **Democratización del conocimiento:** Al permitir un acceso amplio y controlado, el sistema reduce las barreras de entrada al aprendizaje.
3. **Modernización:** Ayuda a instituciones educativas a adaptarse a la era digital, ofreciendo recursos accesibles y organizados.
4. **Sostenibilidad:** Reduce la dependencia de recursos físicos como papel o almacenamiento físico, alineándose con objetivos ambientales.
5. **Competitividad:** Las instituciones educativas que implementan este tipo de soluciones fortalecen su reputación y capacidad para atraer estudiantes e investigadores.

Metodología y ciclo de vida utilizados

El ciclo de vida **cíclico para sistemas** es un enfoque iterativo y repetitivo para el desarrollo, implementación y mantenimiento de sistemas. A diferencia de los modelos lineales, como el modelo en cascada, el modelo cíclico busca mejorar constantemente el sistema a través de ciclos repetidos de desarrollo, evaluación y mejora. Este enfoque es ideal para sistemas que requieren ajustes continuos debido a cambios en las necesidades del usuario, requisitos del negocio o avances tecnológicos.

Fases del Ciclo de Vida Cíclico

1. Identificación de necesidades

- En esta fase, se identifican los problemas o requisitos que el sistema debe resolver o satisfacer.
- Se establecen los objetivos y las expectativas iniciales del sistema.
- Se recopila información de los usuarios y otras partes interesadas.

2. Análisis

- Se analizan los requisitos recopilados para comprender plenamente el problema o las necesidades del sistema.
- Se evalúan los recursos necesarios y las restricciones existentes.

3. Diseño

- Se crea una solución preliminar que aborda los requisitos identificados.
- El diseño inicial puede incluir diagramas de flujo, arquitecturas de sistemas o prototipos.

4. Desarrollo

- En esta fase, se construye o implementa la solución utilizando herramientas y tecnologías adecuadas.
- Se realizan pruebas preliminares para asegurar que el desarrollo inicial cumple con los requisitos.

5. Pruebas

- El sistema se somete a pruebas más rigurosas para detectar errores y asegurarse de que cumple con los estándares de calidad.
- Los usuarios pueden participar en pruebas para validar que el sistema satisface sus necesidades.

6. Implementación

- El sistema se despliega en un entorno real o productivo.
- Se entrena a los usuarios y se ofrece soporte inicial.

7. Evaluación

- Después de la implementación, se evalúa el rendimiento del sistema.
- Se recopila retroalimentación de los usuarios y otros actores involucrados.

8. Mantenimiento y mejora

- Basándose en la evaluación, se realizan ajustes y mejoras al sistema.
- Esta fase puede incluir la corrección de errores, adición de nuevas funcionalidades o actualización de tecnologías.

9. Reinicio del ciclo

- Una vez realizadas las mejoras, el ciclo comienza de nuevo para evaluar el impacto de los cambios y seguir perfeccionando el sistema.
-

Características del Ciclo de Vida Cíclico

- **Iterativo:** Cada fase puede repetirse varias veces para refinar el sistema.
 - **Flexible:** Permite adaptarse a cambios en los requisitos o condiciones externas.
 - **Enfoque incremental:** Las mejoras se implementan en pasos pequeños, facilitando el control y la adaptación.
 - **Colaborativo:** Los usuarios y partes interesadas participan continuamente en el proceso.
-

Ventajas del Ciclo de Vida Cíclico

- Permite mejorar el sistema de forma continua.
 - Minimiza el riesgo de desarrollar un sistema que no cumpla con las expectativas del usuario.
 - Facilita la incorporación de nuevas tecnologías o cambios en los requisitos.
 - Incrementa la satisfacción del usuario al incluirlo en todas las etapas del proceso.
-

Aplicaciones del Ciclo de Vida Cíclico

- Sistemas de software con requisitos cambiantes.
- Proyectos donde se busca desarrollar prototipos antes de una solución final.
- Ambientes de desarrollo ágil o adaptativo.
- Proyectos de larga duración que necesitan mantenimiento y actualizaciones regulares.

Este enfoque asegura que los sistemas evolucionen con el tiempo, manteniéndose relevantes y efectivos.

Análisis Caso de Uso General

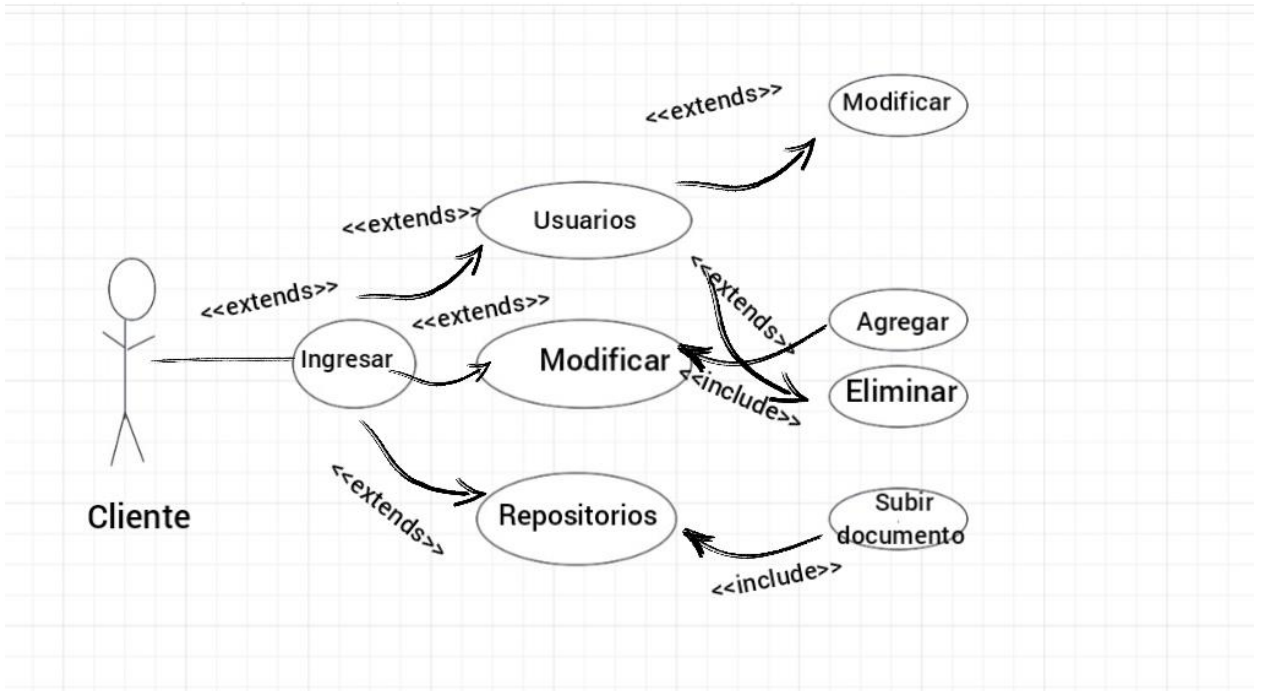


Diagrama de Clase o DER

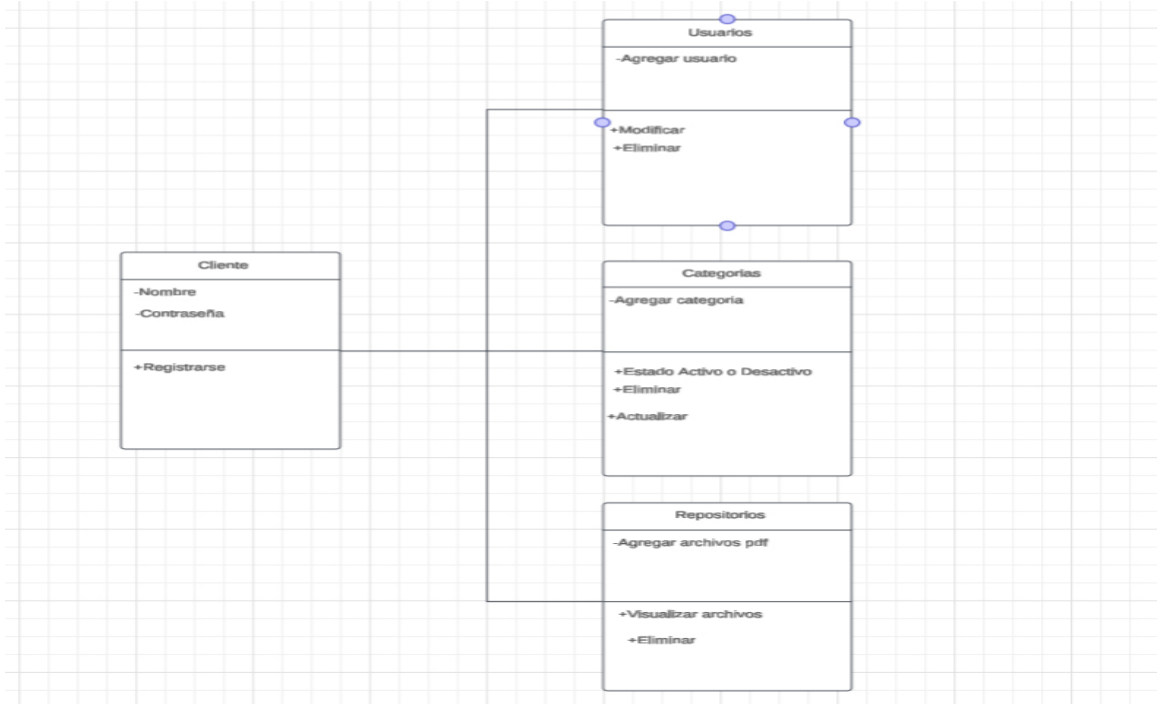


Diagrama de Estado

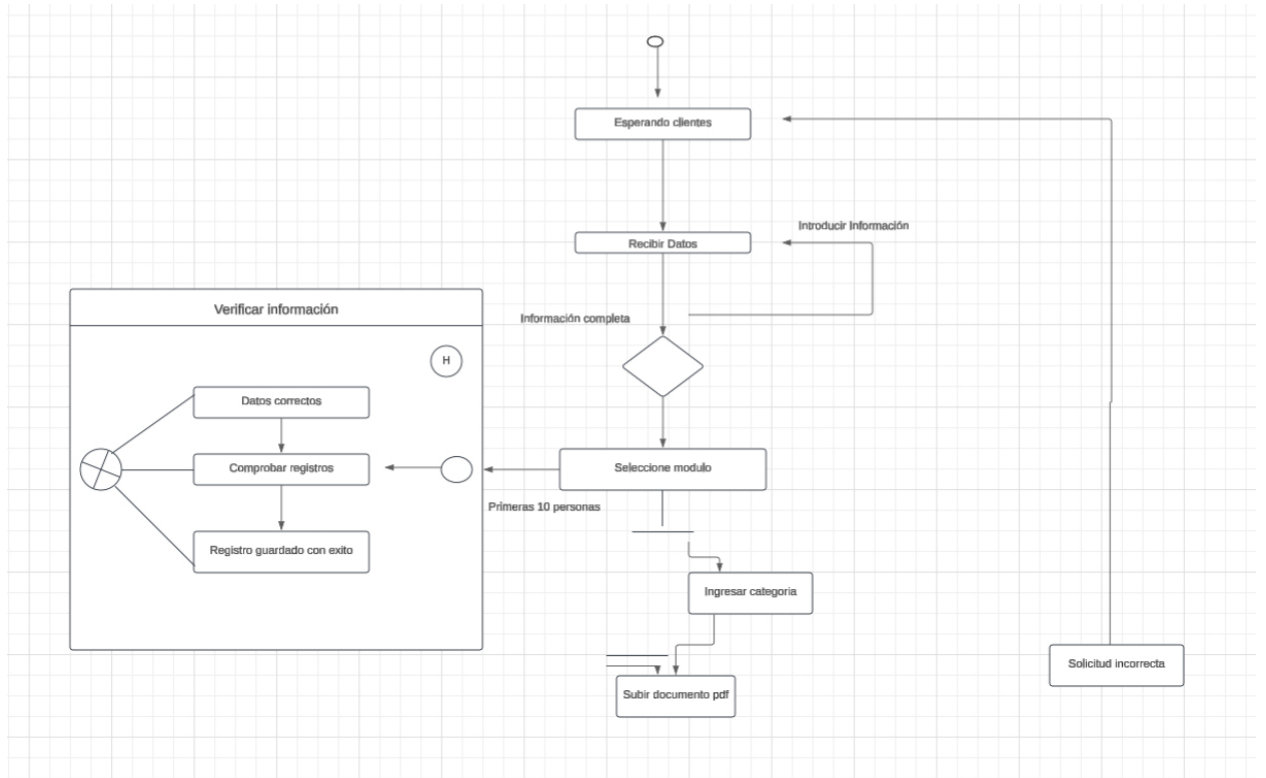


Diagrama de Actividades

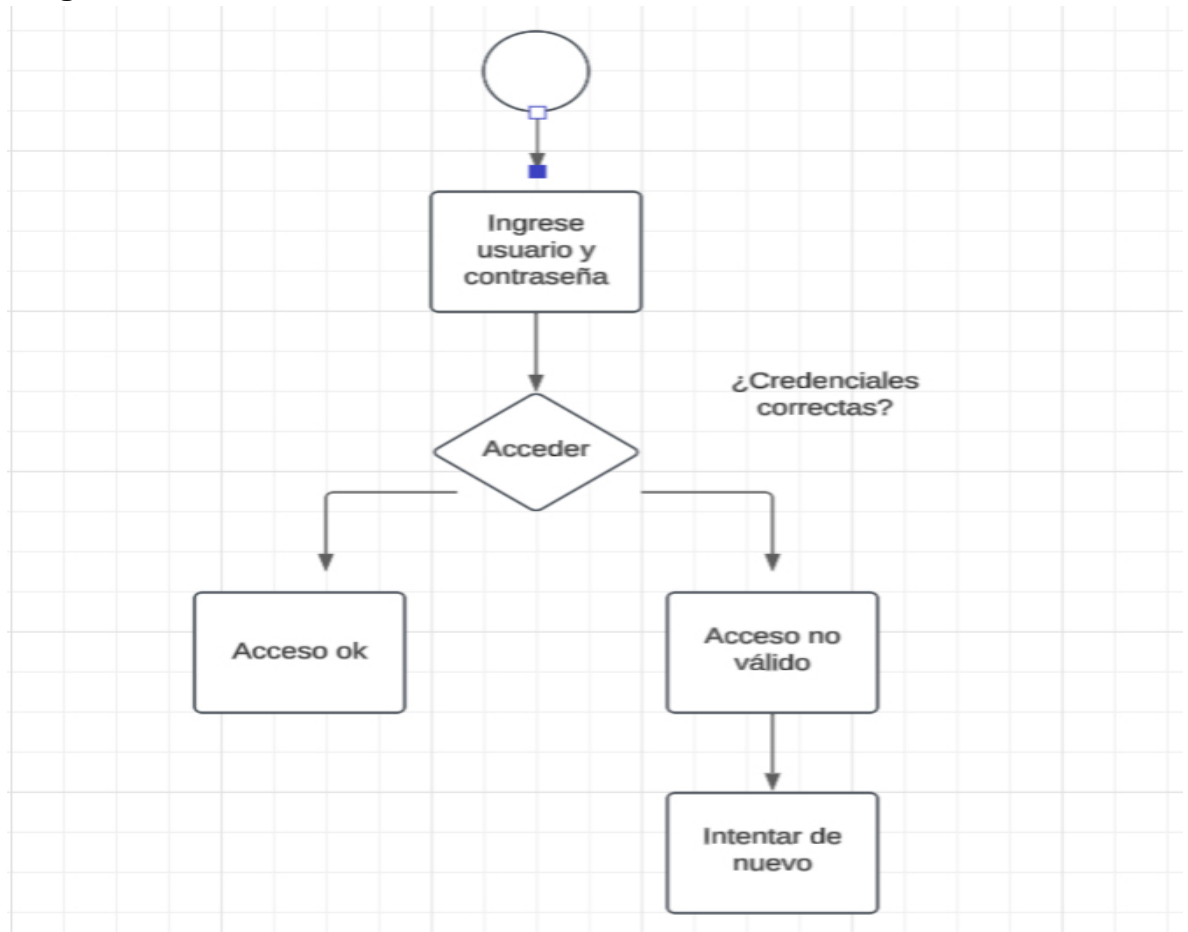
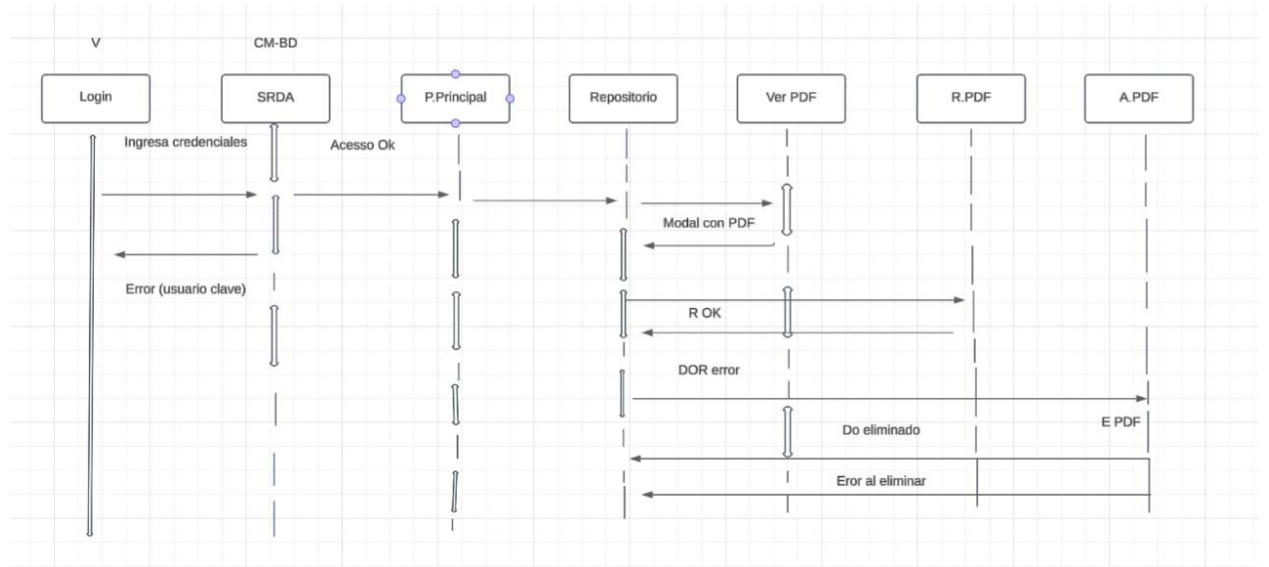


Diagrama de secuencia



Lista de Requerimientos Funcionales y No funcionales:

- **Requerimientos funcionales**

Req1. El sistema solicitara Usuario y Contraseña.

Req2. El sistema debe tener 3 módulos: Usuarios, Categorías y Repositorios.

Req3. El sistema permitirá que el administrador pueda agregar usuarios, modificar o eliminarlos, agregar equipo, serie, modelo y estado.

Req4. El sistema permitirá al administrador ver las categorías registradas, modificarlas o eliminarlas.

Req5. El sistema permitirá en el módulo de repositorios registrar documentos pdf. Y modificarlos o eliminarlos.

Req6. Disponibilidad, El sistema estará disponible las 24 horas, para el uso de administrador y usuarios.

Req7. Accesibilidad, El sistema será accesible de forma local para los usuarios.

Req8. Rendimiento, El sistema debe estar preparado para aceptar múltiples sesiones simultáneas de usuarios ingresando a la vez.

Req9. Mantenibilidad, El sistema es susceptible de ser ampliado. Por lo tanto, deberá diseñarse para darle un soporte adecuado fácilmente, aplicando para su desarrollo los métodos que sean necesarios.

Req10. La interfaz debe estar en el idioma nativo Español para el usuario que lo utilice.

Req11. El sistema debe permitir la visualización de la información consultada.

Req12. El Diseño del sistema debe de permitir un entorno grafico amigable y de fácil utilización para el usuario y administrador.

Req13. El sistema debe visualizarse y funcionar correctamente en cualquier navegador, tales como Google Chrome, Mozilla Firefox, Safari, Opera, Microsoft Edge entre otros.

Req14. El sistema deberá acoplarse a los diferentes Sistemas Operativos correctamente tales como, Microsoft Windows, Linux mint, Ubuntu, Macos, Android, entre otros de software libre.

Req15. El sistema permitirá al administrador Agregar, Modificar o eliminar usuarios.

Req16. El diseño de sistema debe ser definido a través de un conjunto de clases /objetos y componentes re utilizables.

Req17. Para la Utilización del sistema previamente tienen que estar instalados en el servidor que se vaya a montar una serie de programas tales como: Apache2,

PHP, MySQL y Visual Studio Code.

Req18. El Sistema estará expuesto a los cambios a futuro tales como actualizaciones, mejoras de interfaz, nuevos módulos y funciones de administración actualizadas.

- **Requerimientos no funcionales**

Req1. El sistema asignara un color para cada botón dentro del programa en el sistema.

Req2. El sistema permitirá despegar la ventana hacia las 4 direcciones: arriba, abajo, derecha e izquierda.

Req3. El sistema resaltara en negrita el texto seleccionarlo por el puntero.

Req4. Los cuadros del sistema tendrán una sombra de iluminación al seleccionarlos.

Req5. Al dar clic a un botón cambiara de un color suave a uno más intenso.

Req6. El sistema permitirá copiar el texto de la página actual al seleccionar el texto con el puntero.

Desarrollo

- Herramientas de desarrollo utilizadas.

Los sistemas repositorio son esenciales para el manejo eficiente de grandes volúmenes de información, brindando accesibilidad, seguridad y control sobre los datos. Son herramientas cruciales en diversos sectores, desde el desarrollo de software hasta la investigación académica. A medida que la tecnología avanza, los repositorios continúan evolucionando, con nuevas características y capacidades que los hacen más poderosos y adaptados a las necesidades de los usuarios. un sistema repositorio implica la creación de una plataforma que permita almacenar, organizar, gestionar y compartir información de manera eficiente y segura. Este proceso requiere una planificación estratégica, diseño técnico, e implementación de herramientas que cumplan con las necesidades específicas de los usuarios y del entorno en el que se utilizará.

- Lenguaje de programación y gestores de bases de datos u otras herramientas utilizadas en el desarrollo del proyecto.

1 visual studio code

2 MySQL

3 XAMPP

4 Google Chrome

5 conexiones

6 My admin

7 apache

Características de visual studio code:

Control de versiones integrado: VS Code incluye herramientas nativas para el control de versiones, como Git.

Depuración integrada: VS Code permite establecer puntos de interrupción, inspeccionar variables y ejecutar líneas de código paso a paso.

Terminal integrado: VS Code permite ejecutar comandos sin necesidad de salir del editor.

Autocompletado inteligente: VS Code cuenta con IntelliSense, que ayuda a reducir errores tipográficos y sintácticos.

Ligero y rápido: VS Code es un editor ágil que consume pocos recursos.

Multiplataforma: VS Code está disponible para Windows, macOS y Linux.

Característica de MySQL:

1. Modelo Relacional

MySQL utiliza un modelo relacional, lo que significa que organiza los datos en tablas estructuradas con filas y columnas. Este enfoque facilita la consulta, actualización y mantenimiento de los datos.

2. Código Abierto

MySQL es un software de código abierto, lo que permite a los usuarios descargarlo, modificarlo y utilizarlo de manera gratuita bajo la Licencia Pública General de GNU (GPL). También está disponible una versión comercial para uso empresarial con soporte técnico.

3. Multiplataforma

MySQL es compatible con varios sistemas operativos, incluidos:

Windows

Linux

macOS

Unix

Esto permite que sea utilizado en diferentes entornos de desarrollo y producción.

Características de xampp:

1. Paquete Todo en Uno

XAMPP incluye las herramientas esenciales para configurar un servidor web local:

Apache: Servidor HTTP que permite ejecutar aplicaciones web.

MySQL/MariaDB: Sistema de gestión de bases de datos relacionales para almacenar y gestionar datos.

PHP: Lenguaje de programación del lado del servidor, utilizado en la mayoría de las aplicaciones web.

Perl: Lenguaje de programación para tareas específicas de scripting.

Multiplataforma.

Compatible con Windows, Linux, y macOS, lo que permite a los desarrolladores trabajar en cualquier sistema operativo.

Su instalación y configuración es prácticamente idéntica en todas las plataformas.

Característica de Google Chrome:

Velocidad:

Motor de renderizado rápido: Utiliza el motor Blink, que ofrece una carga ágil de páginas.

Optimización del rendimiento: Mejora constante para reducir el uso de recursos y tiempo de carga.

Interfaz de usuario sencilla.

Diseño minimalista que prioriza el contenido de las páginas web. Barra de direcciones multifuncional (omnibox), que permite búsquedas y navegación en un solo lugar.

Característica de myadmin:

Interfaz Gráfica.

Fácil de usar: Permite realizar tareas comunes en bases de datos con una interfaz amigable.

Multiplataforma: Accesible desde cualquier navegador web.

Soporte multilenguaje, incluyendo español, con opciones de personalización.

Ejecución de Consultas SQL.

Editor SQL integrado:

Ejecutar sentencias SQL manualmente.

Mostrar resultados en tiempo real.

Historial de consultas:

Almacena consultas recientes para reutilizarlas.

Soporte para consultas complejas, como uniones, subconsultas y vistas.

Característica de apache:

Compatibilidad Multiplataforma.

Funciona en múltiples sistemas operativos, como Windows, Linux, macOS y Unix.

Es una pieza clave en pilas de servidores como LAMP (Linux, Apache, MySQL, PHP).

Código Abierto y Gratuito.

Desarrollado y mantenido por la Apache Software Foundation.

Totalmente gratuito bajo licencia de código abierto, lo que permite modificarlo y adaptarlo según las necesidades.

Conclusiones

Conclusiones sobre el Desarrollo del Sistema Repositorio de Documentos Académicos (SRDA)

El desarrollo del **Sistema Repositorio de Documentos Académicos (SRDA)** ha sido una experiencia enriquecedora que ha permitido no solo la creación de una herramienta útil para la gestión de la información académica, sino también la reflexión sobre el proceso de diseño y ejecución de un sistema tecnológico al servicio de la comunidad educativa. A continuación, se presentan las principales conclusiones sobre el logro de los objetivos, los aprendizajes obtenidos, los logros alcanzados y la utilidad futura del proyecto.

1. Logro de Objetivos del Proyecto

Los principales **objetivos** del proyecto SRDA eran crear un sistema de gestión documental que permitiera almacenar, organizar y facilitar el acceso a documentos académicos de manera eficiente. Los **objetivos alcanzados** incluyen:

- **Centralización de documentos:** El sistema ha logrado centralizar y organizar los documentos académicos, lo que facilita su acceso y gestión. Con el SRDA, se ha logrado reunir trabajos de investigación, tesis, artículos académicos y otros materiales educativos en una plataforma común.
- **Facilidad de búsqueda y acceso:** La implementación de funcionalidades de **búsqueda avanzada**, filtros por categorías, y organización en carpetas o temáticas ha permitido que los usuarios encuentren los documentos de manera rápida y eficiente.
- **Seguridad y gestión de derechos de autor:** El sistema incorpora medidas para controlar el acceso a los documentos, asegurando que solo los usuarios autorizados puedan cargar, editar o eliminar contenido, respetando así los derechos de autor y las licencias de los documentos almacenados.
- **Optimización del acceso remoto:** El SRDA ha permitido que los documentos sean accesibles desde cualquier lugar con conexión a Internet, lo que facilita el trabajo remoto y el acceso a la información por parte de estudiantes e investigadores.

2. Aprendizajes Durante el Desarrollo del Proyecto

A lo largo del proceso de desarrollo del SRDA, se adquirieron una serie de **aprendizajes importantes** que enriquecieron tanto el proyecto como la experiencia del equipo de trabajo:

- **Importancia de una buena estructura de base de datos:** El diseño adecuado de la **base de datos** y la **estructura de metadatos** fue esencial para asegurar que los documentos se organizaran correctamente y fueran fácilmente recuperables. Este aspecto resultó clave para la eficiencia y escalabilidad del sistema.
- **Desafíos en la integración de tecnologías:** Integrar tecnologías como **sistemas de gestión de bases de datos, servidores web y sistemas de autenticación** resultó ser un desafío técnico importante. Sin embargo, este desafío también fue una oportunidad para aprender sobre la interoperabilidad de distintas tecnologías y cómo hacerlas trabajar de manera conjunta.
- **Importancia de la usabilidad:** Durante las pruebas del sistema, se observó que una **interfaz de usuario intuitiva y fácil de usar** es fundamental para que los usuarios adopten la plataforma. Esto refuerza la idea de que no solo se deben considerar aspectos técnicos, sino también las necesidades y expectativas del usuario final.
- **Mantenimiento y escalabilidad:** El proyecto destacó la importancia de **planificar la escalabilidad** desde el principio. Si bien el sistema está diseñado para manejar una cantidad moderada de documentos, se debe continuar mejorando la infraestructura para soportar una mayor cantidad de usuarios y datos a medida que el sistema crece.

3. Logros Alcanzados

El desarrollo del SRDA ha permitido lograr varios hitos importantes, tanto a nivel técnico como institucional. Entre los logros alcanzados, se incluyen:

- **Creación de un repositorio funcional:** El sistema se encuentra operativo y cumple con las funcionalidades básicas de almacenamiento, búsqueda y acceso a documentos académicos, lo que es un logro significativo desde el punto de vista funcional.

- **Mejora en la accesibilidad de la información:** La plataforma facilita el acceso a documentos de manera rápida y ordenada, eliminando barreras geográficas y temporales para los usuarios. Este acceso remoto ha sido especialmente útil para investigadores y estudiantes que necesitan consultar materiales académicos fuera del horario o ubicación de la institución.
- **Adopción y uso del sistema:** El sistema ha sido adoptado de manera efectiva por un número significativo de usuarios (estudiantes, profesores, investigadores), lo que demuestra la utilidad del SRDA dentro del contexto académico.
- **Protección de derechos de autor:** A través de las funcionalidades de control de acceso, el SRDA ha logrado implementar políticas de gestión de derechos de autor que protegen el trabajo de los autores mientras garantizan el acceso adecuado a los documentos.

4. Propósito y Utilidad Futura del Proyecto

El **SRDA** no solo tiene relevancia en el contexto académico actual, sino que también tiene un gran potencial para evolucionar y servir a más usuarios a lo largo del tiempo. Algunos de los principales **beneficios futuros** que se derivan de este proyecto incluyen:

- **Facilitar la colaboración académica:** El SRDA permitirá que los académicos colaboren de manera más eficiente, compartiendo documentos, referencias y datos de investigación en un solo lugar. Esto puede generar nuevas oportunidades para investigaciones colaborativas y trabajos conjuntos.
- **Contribución a la educación abierta:** Si el SRDA es configurado para permitir acceso abierto a documentos (Open Access), puede contribuir significativamente a la democratización del conocimiento, permitiendo que más personas, fuera del ámbito institucional, puedan acceder a investigaciones académicas valiosas.
- **Adaptación a nuevas tecnologías:** A medida que la tecnología avance, el SRDA podrá integrarse con otras plataformas académicas o adoptar nuevas tecnologías (como **inteligencia artificial** para mejorar las recomendaciones de documentos o mejorar la búsqueda), ampliando sus capacidades.
- **Expansión a otras instituciones:** El SRDA puede ser ampliado y adaptado a otras instituciones académicas, sirviendo como modelo de repositorio digital académico para universidades, centros de investigación y bibliotecas, facilitando la integración de contenido y colaboraciones entre diferentes entidades.

5. Reflexiones Finales

En conclusión, el **Desarrollo del Sistema Repositorio de Documentos Académicos (SRDA)** ha sido un proyecto exitoso que ha logrado establecer una plataforma sólida para la gestión y acceso a documentos académicos. El proceso de desarrollo ha proporcionado aprendizajes valiosos sobre el diseño de sistemas, la importancia de la usabilidad y la escalabilidad, y la necesidad de un enfoque centrado en el usuario.

El SRDA tiene un gran **potencial de crecimiento y mejora**. Si bien el sistema cumple su propósito de manera efectiva en su estado actual, su utilidad a largo plazo dependerá de la capacidad para **adaptarse a nuevas necesidades**, incorporar tecnologías emergentes y **expandir su alcance** dentro de la comunidad académica y más allá.

Finalmente, el éxito del SRDA no solo se mide por la creación de un sistema tecnológico, sino por el impacto que tendrá en el **acceso y la difusión del conocimiento académico**, un elemento fundamental en el proceso educativo y en el avance de la ciencia.

Recomendaciones

- ✚ Se recomienda crear un Sistema de autenticación y autorización robusto.
- ✚ Esta elaborado para Subida de documentos académicos (tesis, artículos, investigaciones, reportes, etc.).
- ✚ Capacidad para agregar descripciones y metadatos (título, autor, resumen, fecha, palabras clave, etc.).
- ✚ Búsqueda avanzada por palabras clave, autor, fecha, categoría, etc.
- ✚ Visualización de documentos dentro del sistema (lector PDF/visor online).
- ✚ Registro de versiones de documentos.
- ✚ Posibilidad de revertir a versiones anteriores.
- ✚ Comentarios y reseñas por parte de los usuarios.
- ✚ Alertas de nuevos documentos en categorías de interés.

Fuentes de información (referencias)

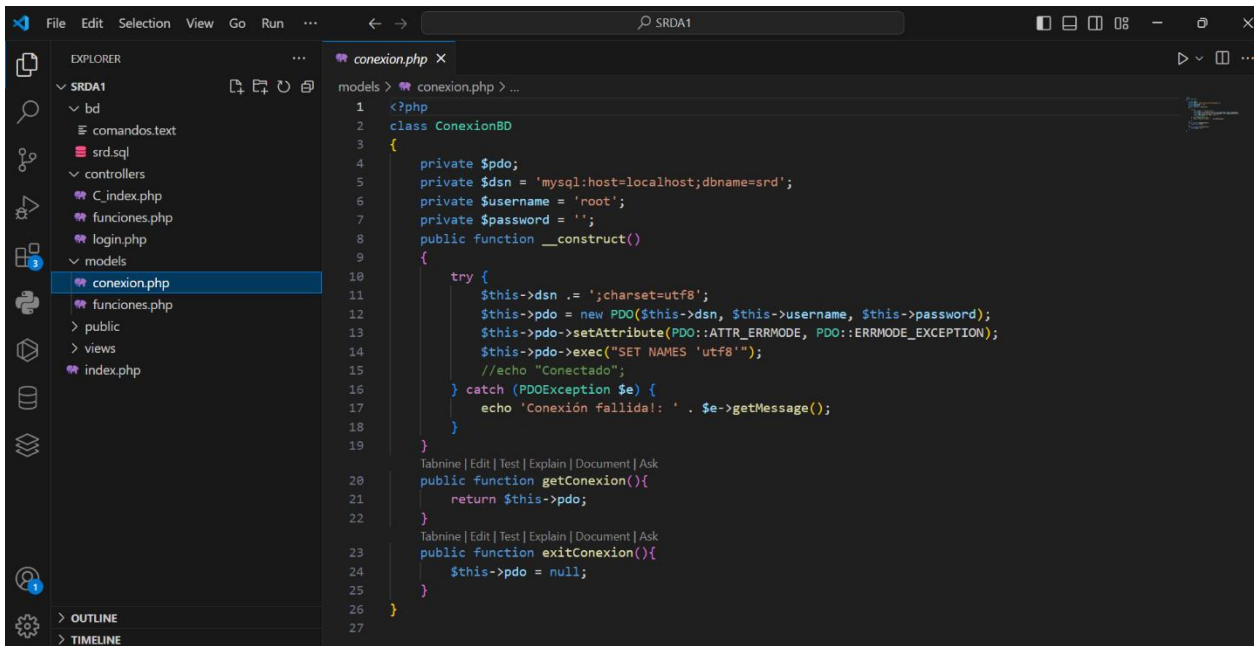
colaboradores de Wikipedia. (2024, October 4). Repositorio (contenido digital). Wikipedia, La Enciclopedia Libre. [https://es.wikipedia.org/wiki/Repositorio_\(contenido_digital\)](https://es.wikipedia.org/wiki/Repositorio_(contenido_digital))

adminUnlp, & adminUnlp. (2023, May 2). Qué son los repositorios institucionales y cómo utilizarlos » UNLP. UNLP » Universidad Nacional de la Plata. <https://unlp.edu.ar/investigacion/recursos/como-funcionan-los-repositorios-institucionales-14545-19545/>

Shinohara, R. P. M. (2023, November 20). Repositorios Institucionales: ¿qué son y cuáles son sus características? Abierto Al Público. <https://blogs.iadb.org/conocimiento-abierto/es/repositorios-institucionales/>

QUÉ ES UN REPOSITORIO y PARA QUÉ SIRVE. (2019, October 25). Río Subterráneo. <https://www.riosubterraneo.com.mx/que-es-un-repositorio-y-para-que-sirve/>

Anexos



```
1 <?php
2 class ConexionBD
3 {
4     private $pdo;
5     private $dsn = 'mysql:host=localhost;dbname=srd';
6     private $username = 'root';
7     private $password = '';
8     public function __construct()
9     {
10
11         try {
12             $this->dsn .= ';charset=utf8';
13             $this->pdo = new PDO($this->dsn, $this->username, $this->password);
14             $this->pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
15             $this->pdo->exec("SET NAMES 'utf8'");
16             //echo "Conectado";
17         } catch (PDOException $e) {
18             echo 'Conexión fallida: ' . $e->getMessage();
19         }
20     }
21     public function getConexion(){
22         return $this->pdo;
23     }
24     public function exitConexion(){
25         $this->pdo = null;
26     }
27 }
```

MANUAL DE USUARIO

<https://drive.google.com/file/d/1YaCNLTdMIZpm0iGbZ6IIHVAr4QIrgis-/view?usp=sharing>

MANUAL DE PROGRAMADOR DE IMPLEMENTACION

<https://drive.google.com/file/d/1ldDUkfMXAoHiNDgFvyvS9P2h3E7CR5ZE/view?usp=sharing>