

EVIDENCIA DE APRENDIZAJE - API

GA7-220501096-AA5-EV02

FASE – EJECUCIÓN

Por:

GEFFRY SANTIAGO LEON JIMENEZ

CENTRO DE LA TECNOLOGÍA DEL DISEÑO Y LA PRODUCTIVIDAD EMPRESARIAL

REGIONAL CUNDINAMARCA – SENA – GIRARDOT

ANÁLISIS Y DESARROLLO DE SOFTWARE

FICHA 2977481

INSTRUCTOR: MILTON IVÁN BARBOSA GAONA

23/09/2025

Tabla de Contenido

1. INTRODUCCIÓN.....	5
2. OBJETIVOS.....	6
2.1 Objetivo general.....	6
2.1.1 <i>Objetivos específicos</i>	6
3. INFORME TÉCNICO DE PLAN DE TRABAJO PARA CONSTRUCCIÓN DE SOFTWARE CON TECNOLOGÍAS SELECCIONADAS	7
3.1 Prototipos	18
3.2 Lenguaje de programación	27
3.3 Gestor de base de datos	27
3.4 Sistema de versionamiento	28
3.5 Conectividad a base de datos	28
3.6 Herramienta de prototipado.....	28
3.7 Herramienta de diagramas	28
3.8 Metodología.....	28
3.9 Módulos del software realizados.....	28
4. CONCLUSIONES.....	46
5. REFERENCIAS BIBLIOGRÁFICAS	47

TABLA DE ILUSTRACIONES

ILUSTRACIÓN 1 DIAGRAMA DE CLASES	7
ILUSTRACIÓN 2 DIAGRAMA DE CASOS DE USO.....	7
ILUSTRACIÓN 3 PROTOTIPO PANTALLA INICIAL.....	18
ILUSTRACIÓN 4 PROTOTIPO INICIO DE SESIÓN	19
ILUSTRACIÓN 5 PROTOTIPO DE REGISTRO	20
ILUSTRACIÓN 6 PROTOTIPO PRÉSTAMO DE LIBROS	21
ILUSTRACIÓN 7 PROTOTIPO ELIMINAR MATERIAL	22
ILUSTRACIÓN 8 PROTOTIPO BUSCAR MATERIAL	23
ILUSTRACIÓN 9 PROTOTIPO RECUPERAR CONTRASEÑA.....	24
ILUSTRACIÓN 10 PROTOTIPO RECUPERAR CONTRASEÑA PRT 2.....	25
ILUSTRACIÓN 11 PROTOTIPO PANTALLA INICIAL USUARIO	26
ILUSTRACIÓN 12 PROTOTIPO PANTALLA INICIAL ADMINISTRADOR.....	27
ILUSTRACIÓN 13 CÓDIGO INICIO	30
ILUSTRACIÓN 14 CÓDIGO INICIO	30
ILUSTRACIÓN 15 CÓDIGO INICIO	31
ILUSTRACIÓN 16 CÓDIGO INICIO	32
ILUSTRACIÓN 17 CÓDIGO INICIO	33

ILUSTRACIÓN 18 SERVLET INICIO	33
ILUSTRACIÓN 19 SERVLET INICIO	34
ILUSTRACIÓN 20 APARTADO VISUAL INICIO	34
ILUSTRACIÓN 21 CÓDIGO REGISTRO.....	36
ILUSTRACIÓN 22 CÓDIGO REGISTRO.....	36
ILUSTRACIÓN 23 CÓDIGO REGISTRO.....	37
ILUSTRACIÓN 24 SERVLET REGISTRO.....	37
ILUSTRACIÓN 25 SERVLET REGISTRO.....	38
ILUSTRACIÓN 26 APARTADO VISUAL REGISTRARSE	38
ILUSTRACIÓN 27 CÓDIGO ÍNDEX	39
ILUSTRACIÓN 28 CÓDIGO ÍNDEX	40
ILUSTRACIÓN 29 CÓDIGO ÍNDEX	40
ILUSTRACIÓN 30 CÓDIGO ÍNDEX	41
ILUSTRACIÓN 31 APARTADO VISUAL ÍNDEX	41
ILUSTRACIÓN 32 BASE DE DATOS MYSQL WORKBENCH.....	42
ILUSTRACIÓN 33 BASE DE DATOS MYSQL WORKBENCH.....	43

ÍNDICE DE TABLAS

TABLA 1 HISTORIA DE USUARIO	7
TABLA 2 HISTORIA DE USUARIO	8
TABLA 3 HISTORIA DE USUARIO	8
TABLA 4 HISTORIA DE USUARIO	9
TABLA 5 HISTORIA DE USUARIO	10
TABLA 6 HISTORIA DE USUARIO	10
TABLA 7 HISTORIA DE USUARIO	11
TABLA 8 HISTORIA DE USUARIO	12
TABLA 9 HISTORIA DE USUARIO	12
TABLA 10 HISTORIA DE USUARIO	13
TABLA 11 CASO DE USO	13
TABLA 12 CASO DE USO	13
TABLA 13 CASO DE USO	14
TABLA 14 CASO DE USO	14
TABLA 15 CASO DE USO	15
TABLA 16 CASO DE USO	15
TABLA 17 CASO DE USO	16
TABLA 18 CASO DE USO	16
TABLA 19 CASO DE USO	17
TABLA 20 CASO DE USO	17
TABLA 21 HISTORIA DE USUARIO INICIAR SESIÓN	29
TABLA 22 CASO DE USO INICIO DE SESIÓN	29
TABLA 23 HISTORIA DE USUARIO REGISTRO	35
TABLA 24 CASO DE USO REGISTRO	35
TABLA 25 LISTA DE CHEQUEO	43

1. INTRODUCCIÓN

A lo largo de nuestro curso hemos venido realizando un proyecto de software el cual en mi caso es de una biblioteca, en este informe recolectare lo que llevo y mis avances, prototipos y las tecnologías que usare.

2. OBJETIVOS

2.1 Objetivo general

Poder aprender sobre como conectar la base de datos MySQL con java mediante apache

2.1.1 *Objetivos específicos*

- Aprender más sobre java
- Poder conectar bases de datos
- Avanzar en mi aprendizaje

3. INFORME TÉCNICO DE PLAN DE TRABAJO PARA CONSTRUCCIÓN DE SOFTWARE CON TECNOLOGÍAS SELECCIONADAS

Ilustración 1 Diagrama de clases

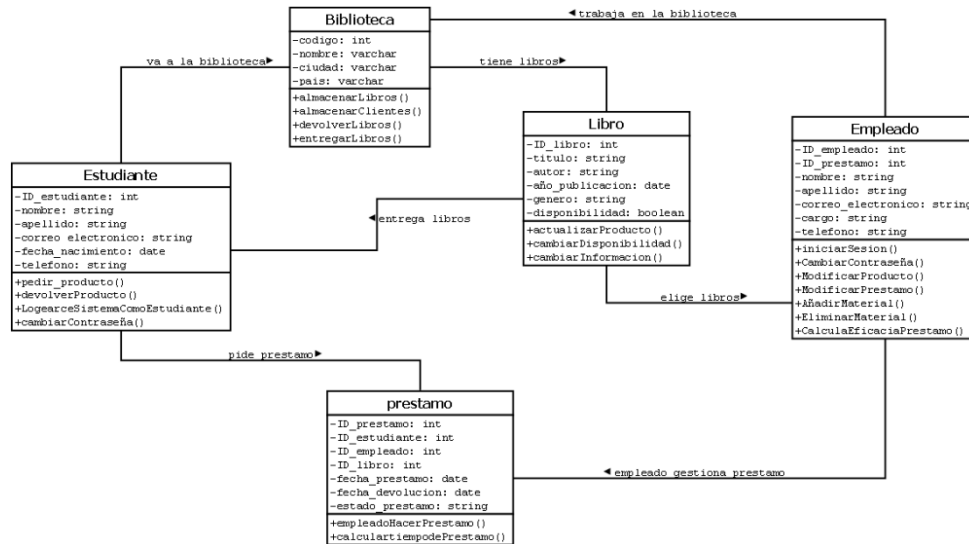
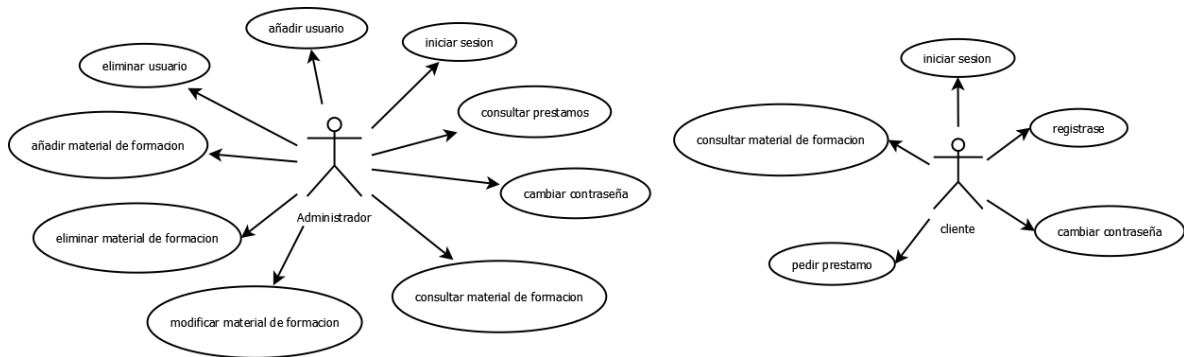


Ilustración 2 Diagrama de casos de uso



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 1 Historia de usuario

Identificador:	HU-01
Nombre:	Iniciar sesión
Actores:	Administrador - cliente

Objetivo:	Iniciar sesión en el sistema
Descripción:	Como cliente quiero que dejen el sistema de inicio de sesión como esta ya que me parece muy bueno
Caso de prueba:	El inicio de sesión es un éxito
Puntos estimados de esfuerzo	El administrador calcula que el tiempo mínimo para iniciar sesión es de 30 segundos, ya que solo es poner el correo y la contraseña
observaciones	Este sistema hará que la persona pueda ingresar a la plataforma
Criterios de aceptación	Para ser aceptado debe permitir que la persona ingrese a la plataforma

Nota: tabla hecha por Geffry Santiago Leon Jimenez

Tabla 2 Historia de usuario

Identificador:	HU-02
Nombre:	Añadir usuario - Registrarse
Actores:	Administrador - cliente
Objetivo:	Añadir un usuario al sistema
Descripción:	Como administrador quiero tener un sistema de un botón donde se permita crear un usuario para poder tener un perfil en el que puedan pedir sus préstamos a partir de las características (nombre, edad, celular, Gmail) y que se genere un mensaje en donde me confirme si la operación salió correctamente para un mejor sistema
Caso de prueba:	El usuario se ha añadido correctamente
Puntos estimados de esfuerzo	El administrador a calculado cuando podría demorar en agregar un usuario, el cual dice que demora 1 minuto
observaciones	Esta función ayudara a que la persona acceda al sistema
Criterios de aceptación	Para ser aceptado se necesita que al añadir el usuario se le dé la notificación al cliente de que su usuario ha sido activado correctamente

Nota: tabla hecha por Geffry Santiago Leon Jimenez

Tabla 3 Historia de usuario

Identificador:	HU-03
Nombre:	Eliminar usuario
Actores:	administrador
Objetivo:	Eliminar un usuario del sistema

Descripción:	Como administrador me gustaría tener un sistema en el cual me aparezcan los usuarios que no se han utilizado con un botón que diga eliminar usuarios lo cual sería que eliminaría todos, donde me vote un mensaje de alerta de que si estoy seguro de borrar el usuario y que me aparezca un mensaje de que el usuario fue borrado exitosamente para tener más orden
Caso de prueba:	El usuario se ha eliminado correctamente
Puntos estimados de esfuerzo	El administrador después de poner en desarrollo la eliminación de usuarios a determinado que se demora aproximadamente 10 minutos en eliminar un usuario
observaciones	También debería tener un botón en el cual le permita seleccionar cuales usuarios debería borrar
Criterios de aceptación	Para aceptarlo necesito saber si se realizo correctamente es decir que me vote un mensaje el cual me diga que los usuarios seleccionados han sido eliminados

Nota: tabla hecha por Geffry Santiago Leon Jimenez

Tabla 4 Historia de usuario

Identificador:	HU-04
Nombre:	Añadir material de formación
Actores:	Administrador
Objetivo:	Añadir material de formación al sistema
Descripción:	Como administrador quiero tener un apartado en donde pueda agregar el material de formación a partir de las siguientes características (nombre, autor, número de serie, grado del libro de material de formación) y que se genere un mensaje de confirmación en donde diga que el material de formación se ha añadido correctamente para un mejor orden
Caso de prueba:	El material se ha añadido correctamente
observaciones	Una observación sería que al añadir un material de formación se les notifique a los usuarios
Puntos estimados de esfuerzo	El administrador dice que demora 10 minutos en agregar un material, pero a la hora de hacerlo demoro 15 minutos
Criterios de aceptación	Para ser aceptado se necesita que al momento de querer agregar el libro y ya haber puesto toda la información vote un mensaje de que el material si fue agregado y fue notificado a los usuarios.

Nota: tabla hecha por Geffry Santiago Leon Jimenez

Tabla 5 Historia de usuario

Identificador:	HU-05
Nombre:	Eliminar material de formación
Actores:	Administrador
Objetivo:	Eliminar un material de formación
Descripción:	Como administrador quiero tener un sistema de un botón que me permita eliminar un material de formación que yo seleccione, y que al darle me aparezca una advertencia de que si quiero eliminar ese material acompañado de un mensaje de confirmación que me diga que el material de formación ha sido eliminado exitosamente para poder organizar más el sistema
Caso de prueba:	El material de formación se ha eliminado correctamente
observaciones	Una observación seria que el material que vayas a eliminar sea notificado a los usuarios
Puntos estimados de esfuerzo	El administrador a analizado la forma de eliminar usuarios y a determinado que entre buscar los materiales que borrar se llevaría aproximadamente 20 minutos
Criterios de aceptación	Para que esto sea aceptado al momento de eliminar el material deberá aparecer un mensaje en el que me confirme que el material ha sido eliminado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 6 Historia de usuario

Identificador:	HU-06
Nombre:	Modificar material de formación
Actores:	Administrador
Objetivo:	Modificar un material de formación
Descripción:	Como administrador quiero un sistema en el cual me aparezca un listado en el cual estén todos los materiales de formación el cual tengan algún error, después quiero un botón que me permita modificar el material y que al darle de nuevo genere un mensaje de confirmación en la cual diga que la modificación ha sido un éxito para poder de algún modo ayudar a que el sistema fluya mejor
Caso de prueba:	La modificación ha sido un éxito

observaciones	Una observación para agregar es que hubiera un auto correcto automático en el cual se hiciera parte de este progreso automáticamente para así ahorrar más tiempo
Puntos estimados de esfuerzo	Teniendo en cuenta la observación pasada el administrador del sistema a estimado un tiempo en el cual le toma modificar todos los materiales el cual sería de 30 minutos
Criterios de aceptación	Para esta ser aceptada debe ser puesta a prueba si esta funciona correctamente debe aparecer un mensaje en el cual me diga que el proceso de modificación ha sido terminado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 7 Historia de usuario

Identificador:	HU-07
Nombre:	Consultar material de formación
Actores:	Administrador
Objetivo:	Consultar cuales son los materiales de formación
Descripción:	Como administrador quiero un sistema más avanzado el cual con un solo código pueda encontrar el material de formación que necesite, así se puede ahorrar más tiempo en el momento de buscar un material de formación
Caso de prueba:	El material de formación ha sido encontrado
observaciones	Para sumar a la descripción sería que ese código se les den a los usuarios para que ellos también puedan tener un sistema de búsqueda más rápido
Puntos estimados de esfuerzo	El administrador a detallado la nueva forma que el quiere en encontrar el material para esto analiza cuanto él se podría demorar y cuanto los usuarios se podrían demorar, esto le da que para encontrar un material con el código se demoraría 1 minuto ya que es rápido el proceso de encontrar el material
Criterios de aceptación	Para esto ser aceptado y ya ser considerado terminado se necesita verificar si funciona bien lo cual para esto al momento de buscar con el código el material te ponga un mensaje de que el material a sido encontrado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 8 Historia de usuario

Identificador:	HU-08
Nombre:	prestamos
Actores:	administrador
Objetivo:	Consultar los préstamos de los materiales de formación
Descripción:	Como administrador quiero poner un listado en el cual me aparezcan todos los préstamos que la gente está pidiendo, y poner un botón que diga aceptar para ahorrar más tiempo y poder hacer todos los prestamos más rápido
Caso de prueba:	El préstamo ha sido aprobado
observaciones	Los usuarios deberían tener un tiempo estimado de cuanto podría tardar en aceptar su préstamo
Puntos estimados de esfuerzo	El administrador ha analizado cuanto se podría demorar en aceptar todos los préstamos y a calculado que en unas 5 horas podría aceptar los prestamos ya que al tener que leerlos todos se lleva mucho tiempo
Criterios de aceptación	Para ser aceptado se necesita probar este sistema así que para esto cuando se acepten los prestamos se debe notificar con un mensaje a los usuarios de que el préstamo ha sido aceptado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 9 Historia de usuario

Identificador:	HU-09
Nombre:	Pedir préstamo
Actores:	cliente
Objetivo:	Pedir un préstamo
Descripción:	Como cliente quiero que agreguen un sistema en el cual los prestamos sean automáticos, es decir que no lleven tanto tiempo en ser aceptados esto para poder tener el material más rápido
Observaciones	Los administradores deberían tener un estimado de tiempo en el cual acepten las solicitudes
Puntos estimados de esfuerzos	El usuario a dicho que para hacer un préstamo se deberían llevar entre 5 a 6 minutos
Criterios de aceptación	Para poder aceptar esto necesitamos que a la hora de hacer el préstamo se ejecute de forma correcta y alerte de que si se hizo

Tabla 10 Historia de usuario

Identificador:	HU-10
Nombre:	Cambiar contraseña
Actores:	administrador
Objetivo:	Cambiar la contraseña de usuario
Descripción:	Como administrador quiero agregar un sistema de seguridad el cual el usuario pueda cambiar la contraseña con solo el correo si esta se le olvido
Caso de prueba:	Su contraseña se ha cambiado correctamente
Observaciones	También se podría agregar un sistema en el cual si el usuario no se acuerda de su contraseña tiene la oportunidad de cambiarla con un botón
Puntos estimados de esfuerzo	Para cambiar una contraseña el administrador a analizado cuanto se podría demorar y a estimado que se demoraría unos 2 minutos ya que es super fácil con el sistema que tiene
Criterios de aceptación	Para ser aceptado se necesita que a la hora de cambiar la contraseña me vote un mensaje de confirmación el cual diga que la contraseña se ha cambiado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 11 Caso de uso

Identificador:	CU-01
Nombre:	Iniciar sesión
Actores:	Administrador-usuario
Objetivo:	Añadir un usuario al sistema
Descripción:	<p>1 el usuario administrador entra a la pagina</p> <p>2 el usuario administrador ingresa el usuario y la contraseña</p> <p>3 el usuario administrador le da a iniciar sesión</p>
Caso de prueba:	El usuario se ha añadido correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 12 Caso de uso

Identificador:	CU-02
Nombre:	Añadir usuario - Registrarse

Actores:	Administrador - cliente
Objetivo:	Añadir un usuario al sistema
Descripción:	<p>4 el usuario administrador entra a la pagina</p> <p>5 el usuario administrador le da a donde dice registrarse</p> <p>6 el usuario administrador pone los datos</p> <p>7 el usuario administrador le da al botón que dice registrase</p> <p>8 se devuelve el mensaje de que la operación ha sido correcta</p>
Caso de prueba:	El usuario se ha añadido correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 13 Caso de uso

Identificador:	CU-03
Nombre:	Eliminar usuario
Actores:	Administrador
Objetivo:	Eliminar usuario del sistema
Descripción:	<p>1 el usuario administrador entra a la pagina</p> <p>2 el usuario administrador revisa los usuarios registrados</p> <p>3 el usuario administrador ve un usuario que no ha usado su cuenta</p> <p>4 el usuario administrador le da al botón de eliminar usuario</p> <p>5 el usuario administrador le da en que está seguro de eliminar el usuario</p> <p>6 se devuelve un mensaje de que la operación ha sido correcta</p>
Caso de prueba:	El usuario se ha eliminado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 14 Caso de uso

Identificador:	CU-04
Nombre:	Añadir material de formación

Actores:	Administrador
Objetivo:	Añadir un material de formación al sistema
Descripción:	<ol style="list-style-type: none"> 1 el usuario administrador entra a la pagina 2 el usuario administrador inicia sesión 3 el usuario administrador pulsa sobre el botón añadir material de formación 4 el usuario administrador los datos del material que desea añadir 5 el usuario administrador le da al botón agregar libro 6 al usuario administrador le llega un mensaje que confirma que el material se añadió al sistema
Caso de prueba:	El material se ha añadido correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 15 Caso de uso

Identificador:	CU-05
Nombre:	Eliminar material de formación
Actores:	Administrador
Objetivo:	Eliminar un material de formación
Descripción:	<ol style="list-style-type: none"> 1. el usuario administrador ingresa a el sistema 2. el usuario administrador busca el material que desea eliminar 3. el usuario administrador le da el botón de eliminar material de formación 4. el usuario administrador le da al botón de aceptar del mensaje de advertencia 5. el usuario administrador recibe una notificación donde se notifica que el material ha sido eliminado correctamente
Caso de prueba:	El material ha sido eliminado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 16 Caso de uso

Identificador:	CU-06
----------------	-------

Nombre:	Modificar material de formación
Actores:	Administrador
Objetivo:	Modificar un material de formación
Descripción:	<ol style="list-style-type: none"> 1. el usuario administrador ingresa al sistema 2. el usuario administrador selecciona que material de formación modificara 3. el usuario administrador le da al botón de guardar cambios 4. el usuario administrador recibe un mensaje de que el material de formación a sido modificado
Caso de prueba:	La modificación ha sido un éxito

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 17 Caso de uso

Identificador:	CU-07
Nombre:	Consultar material de formación
Actores:	Administrador
Objetivo:	Consultar cuales son los materiales de formación
Descripción:	<ol style="list-style-type: none"> 1. el usuario administrador accede al sistema 2. el usuario administrador accede al listado de los materiales de formación 3. el usuario administrador usa la barra de búsqueda para buscar su material de formación 4. el usuario administrador encuentra el material de formación
Caso de prueba:	El material de formación ha sido encontrado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 18 Caso de uso

Identificador:	CU-08
Nombre:	prestamos
Actores:	Administrador
Objetivo:	Aceptar los préstamos de los materiales de formación
Descripción:	<ol style="list-style-type: none"> 1. el usuario administrador accede al sistema

	<ol style="list-style-type: none"> 2. el usuario administrador se va al apartado de préstamos 3. el usuario administrador empieza a leer y detallar que es lo que los clientes piden 4. el usuario administrador acepta los préstamos y entrega los materiales de formación
Caso de prueba:	El préstamo ha sido aprobado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 19 Caso de uso

Identificador:	CU-09
Nombre:	Pedir préstamo
Actores:	cliente
Objetivo:	Pedir un préstamo
Descripción:	<ol style="list-style-type: none"> 1. el usuario cliente accede al sistema 2. el usuario cliente ingresa al apartado de prestamos 3. el usuario cliente pasa los materiales que necesita 4. el usuario cliente recibe respuesta del administrador 5. el usuario cliente recibe un mensaje el cual notifica que el préstamo fue aceptado
Caso de prueba:	El préstamo fue aceptado

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 20 Caso de uso

Identificador:	CU-10
Nombre:	Cambiar contraseña
Actores:	administrador
Objetivo:	Cambiar la contraseña del usuario
Descripción:	<ol style="list-style-type: none"> 1. el usuario administrador accede de acceder al sistema 2. el usuario administrador le da al botón de cambiar contraseña 3. el usuario administrador pone su nueva contraseña y le da al botón de cambiar contraseña

	4. el usuario administrador recibe un mensaje afirmativo en el cual dice que la contraseña se ha cambiado
Caso de prueba:	La contraseña se ha cambiado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

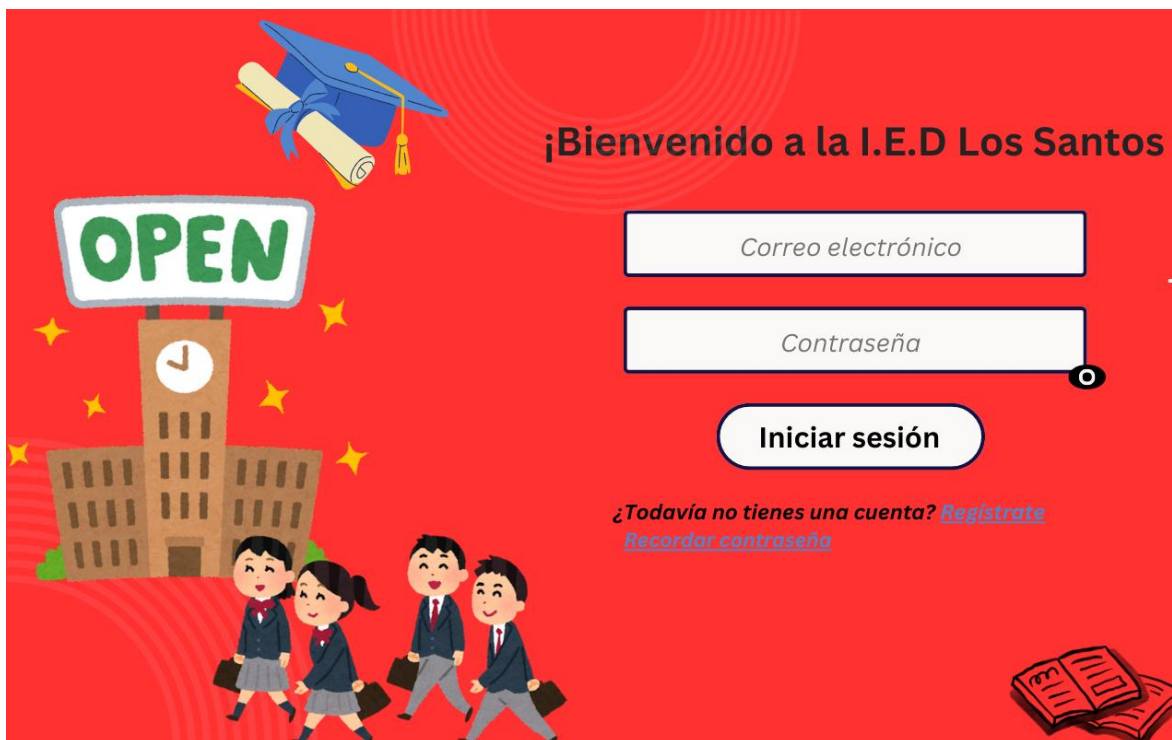
3.1 Prototipos

Ilustración 3 prototipo pantalla inicial



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 4 prototipo inicio de sesión



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 5 prototipo de registro

INGRESA LOS SIGUIENTES DATOS

Primer Nombre	Correo Electronico
Segundo Nombre	N. Documento
Primer Apellido	Fecha de Nacimiento
Segundo Apellido	Contraseña

Registrar

The form is set against a red background. On the left, there is a vertical column of white circles. At the bottom left is a small schoolhouse icon. At the bottom right is a cartoon owl wearing a graduation cap and holding a book. A leafy branch is in the top right corner.

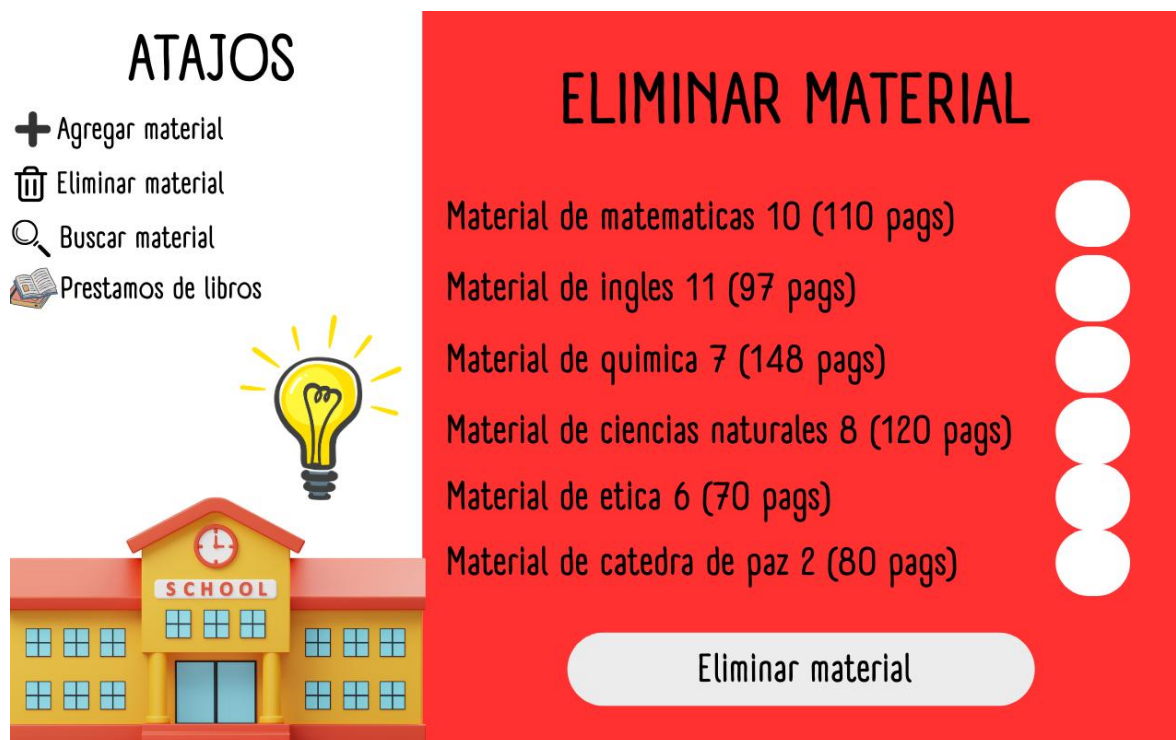
Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 6 Prototipo préstamo de libros



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 7 Prototipo eliminar material



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 8 Prototipo buscar material



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 9 Prototipo recuperar contraseña



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 10 Prototipo recuperar contraseña prt 2

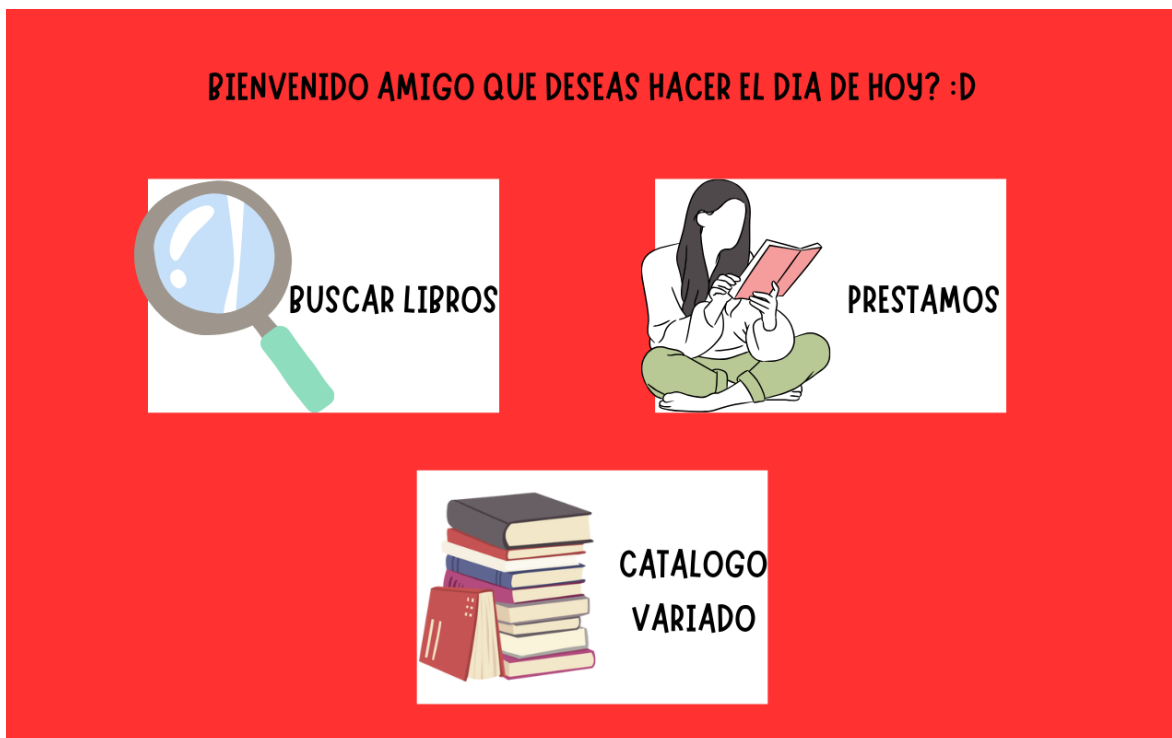


Ingresa nueva contraseña



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 11 Prototipo pantalla inicial usuario



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 12 Prototipo pantalla inicial administrador



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

3.2 Lenguaje de programación

Mi proyecto es una plataforma web la cual es de una biblioteca escolar que presta libros, al ser algo web los lenguajes que utilizare por el momento serán HTML, CSS y JAVASCRIPT, estos 3 me sirven para poder realizar la plataforma de forma correcta y cómoda.

3.3 Gestor de base de datos

La base de datos que utilizare en este proyecto será MySQL debido a su amplia adopción, fiabilidad y facilidad de uso. Es un gestor de bases de datos relacional que se integra perfectamente con Java a través de JDBC.

3.4 Sistema de versionamiento

El sistema que utilizare será GitHub, este es sencillo de entender y me ayuda a poder subir mi proyecto a la nube, el cual si le pasa algo a mi computadora lograre acceder fácilmente a todos mis avances.

3.5 Conectividad a base de datos

Será la tan conocida JDBC es la API estándar de Java para la conexión a bases de datos. Permite la ejecución de sentencias SQL desde el código Java, garantizando una conexión segura y eficiente

3.6 Herramienta de prototipado

La herramienta de prototipado que eh destinado al proyecto es canva, esta la llevo utilizando mucho tiempo y ya tengo mucha confianza con esta, si en algún momento debo cambiarla lo hare sin problema, pero por mientras seria canva.

3.7 Herramienta de diagramas

Estoy utilizando Dia, esta me la recomendó el instructor y se me ha hecho muy cómoda a la hora de usar, así que día será la seleccionada.

3.8 Metodología

Para este trabajo eh seleccionado la metodología ágil, esta me permite poder modificar el proyecto sin tener que empezar desde cero, esto me ahorra mucho trabajo ya que al ser principiante comento muchos errores, pero poder solucionarlos rápidamente me ayuda demasiado.

3.9 Módulos del software realizados

Tabla 21 Historia de usuario Iniciar sesión

Identificador:	HU-01
Nombre:	Iniciar sesión
Actores:	Administrador - cliente
Objetivo:	Iniciar sesión en el sistema
Descripción:	Como cliente quiero que dejen el sistema de inicio de sesión como esta ya que me parece muy bueno
Caso de prueba:	El inicio de sesión es un éxito
Puntos estimados de esfuerzo	El administrador calcula que el tiempo mínimo para iniciar sesión es de 30 segundos, ya que solo es poner el correo y la contraseña
observaciones	Este sistema hará que la persona pueda ingresar a la plataforma
Criterios de aceptación	Para ser aceptado debe permitir que la persona ingrese a la plataforma

Nota: tabla hecha por Geffry Santiago Leon Jimenez

Tabla 22 Caso de uso inicio de sesión

Identificador:	CU-01
Nombre:	Iniciar sesión
Actores:	Administrador-usuario
Objetivo:	Añadir un usuario al sistema
Descripción:	<ol style="list-style-type: none"> 1 el usuario administrador entra a la pagina 2 el usuario administrador ingresa el usuario y la contraseña 3 el usuario administrador le da a iniciar sesión
Caso de prueba:	El usuario se ha añadido correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 13 Código inicio

```
<!--
Document   : inicio
Created on : 11/09/2025, 8:39:38 p.m.
Author    : santi
-->

<!!page contentType="text/html" pageEncoding="UTF-8"%>
<!-- en esta parte configuraremos el idioma de la pagina, que reciba caracteres especiales, que sea responsivo para moviles -->
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inicio de sesión - I.E.D Los Santos</title>
  <style>
    /* en body se configuro el margen, tipo de fuente, color de la pagina, color de las fuentes, display y tamaño */
    body {
      margin: 0;
      font-family: Arial, Helvetica, sans-serif;
      background-color: rgb(220, 22, 22);
      color: #000;
      display: flex;
      height: 100vh;
    }
    /* en este apartado se configura la parte izquierda de la pantalla donde ira la imagen */
    .left {
      flex: 1;
      display: flex;
      justify-content: center;
      align-items: flex-start;
      padding-top: 20vh;
      padding-left: 20px;
      padding-right: 20px;
      padding-bottom: 20px;
    }
  </style>
</head>
<body>
```

Nota: Ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 14 Código Inicio

```
    /* aqui se configura la imagen caminando.png */
    .left-image {
      max-width: 100%;
      max-height: 100%;
      object-fit: contain;
    }
    /* se configura la parte derecha de la pantalla donde van los botones, el titulo y los links */
    .right {
      flex: 1;
      display: flex;
      flex-direction: column;
      justify-content: center;
      padding: 40px;
      animation: aparecer 1s ease-out forwards;
      opacity: 0;
      transform: translateY(30px);
    }
    /* en h1 se define el tamaño de la fuente */
    h1 {
      font-size: 40px;
      margin-bottom: 35px;
    }
    /* en el form se configura el display y la direccion la cual sea en columna para una mejor organizacion */
    form {
      display: flex;
      flex-direction: column;
    }
    /* en este apartado se configuran los botones de email y password, tamaño, margen, borde, tamaño de la fuente */
    input[type="email"],
    input[type="password"] {
      padding: 16px;
      margin-bottom: 30px;
      border: 2px solid #00004c;
      border-radius: 12px;
      font-size: 18px;
    }
  </body>
</html>
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 15 Código inicio

```
        outline: none;
    }
    /* en button se configura el boton de accion, el de iniciar sesion */
    button {
        padding: 14px;
        background-color: white;
        border: 2px solid #00004c;
        border-radius: 20px;
        font-size: 12px;
        font-weight: bold;
        cursor: pointer;
        transition: background-color 0.3s;
    }
    /* el hover se configura el color que se pone el boton cuando le pasan el cursor por encima */
    button:hover {
        background-color: #ddd;
    }
    /* en link se configuran los links que te redireccionan a recuperar contraseña o registrarte */
    .links {
        margin-top: 15px;
        font-size: 18px;
    }
    /* en links a se les define el color y el margen que tendran */
    .links a {
        color: #004bff;
        text-decoration: none;
        margin-left: 5px;
    }
    /* este permite poder subrayar los links */
    .links a:hover {
        text-decoration: underline;
    }
    /* Esta parte del código define una animación de aparición suave y un divisor decorativo superior con una forma SVG de color
    @keyframes aparecer {
        from {
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 16 Código inicio

```
        opacity: 0;
        transform: translateY(30px);
    }
    to {
        opacity: 1;
        transform: translateY(0);
    }
}
.custom-shape-divider-top-1751419049 {
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    overflow: hidden;
    line-height: 0;
}
.custom-shape-divider-top-1751419049 svg {
    position: relative;
    display: block;
    width: calc(100% + 1.3px);
    height: 183px;
}
.custom-shape-divider-top-1751419049 .shape-fill {
    fill: #b30f0f;
}
</style>
</head>
<body>
    <!-- en esta parte podremos colocar la imagen caminando.png -->
    <div class="left">
        
    </div>
    <!-- en esta parte de la division colocaremos el titulo -->
    <div class="right">
        <h1>Bienvenido a la plataforma, inicia sesión para continuar!</h1>
    </div>
</body>
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 17 Código inicio

```
<!-- en esta parte de la division colocaremos donde y con que metodo se enviaran los datos, ademas de que añadiremos un formulario que r
<form action="InicioServlet" method="POST">
  <input type="email" name="email" placeholder="Correo electrónico" required>
  <input type="password" name="contraseña" placeholder="Contraseña" required>
  <button type="submit">Iniciar sesión</button>
</form>

<!-- esta parte nos da el mensaje de error que genera si no ponen los datos de manera correcta -->
<% String mensaje = (String) request.getAttribute("mensaje");
if (mensaje != null) { %>
<div style="color: white; font-weight: bold;">
  <%= mensaje %>
</div>
<% } %>

<!-- en esta division podremos colocar link que te direccionen a otras partes, esto se hace con el href -->
<div class="links">
  ¿Todavía no tienes cuenta?
  <a href="registro.jsp">Registrarse</a><br>
  <a href="#">Recordar Contraseña</a>
</div>

<!-- esta parte termina de configurar la animacion de aparicion -->
<div class="custom-shape-divider-top-1751419049">
  <svg data-name="Layer 1" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 1200 120" preserveAspectRatio="none">
    <path d="M321.39,56.44c58-10.79,114.16-30.13,172-41.86,82.39-16.72,168.19-17.73,250.45-.39c823.78,31,906.67,72,985.66,92.
      83c70.05,18.48,146.53,26.09,214.34,3V0H0V27.35A600.21,600.21,0,0,0,0,0,0,0,321.39,56.44z" class="shape-fill"></path>
  </svg>
</div>
</body>
</html>
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 18 Servlet Inicio

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this template
 */
package Servlets;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

//esto indica que el servlet accede con la ruta /InicioServlet//
@WebServlet("/InicioServlet")
public class InicioServlet extends HttpServlet {

    //conexion con la base de datos//
    private static final String DB_URL = "jdbc:mysql://localhost:3306/proyecto_software";
    private static final String DB_USER = "root";
    private static final String DB_PASS = "2007";

    //maneja las peticiones POST ya que en inicio.jsp se manda con method=POST//
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        //aquí se captura los datos del formulario//
        String email = request.getParameter("email");
        String contraseña = request.getParameter("contraseña");
    }
}
```

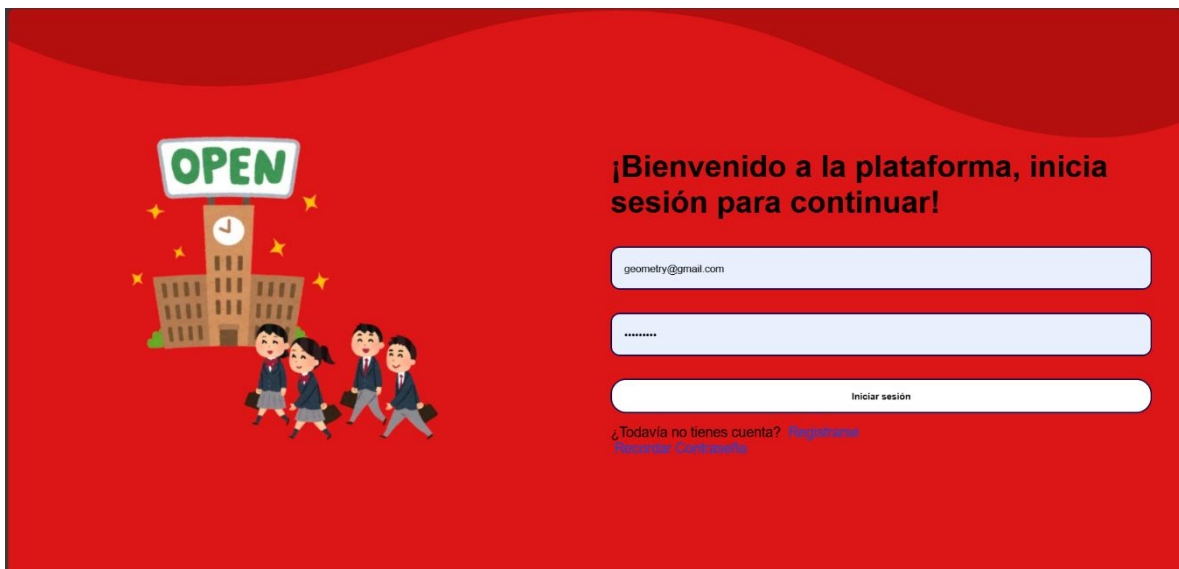
Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 19 Servlet inicio

```
try {  
  
    Class.forName("com.mysql.cj.jdbc.Driver");  
  
    Connection conn = DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);  
  
    //este apartado consulta si existe un usuario con el email y la contraseña que ingresen en inicio.jsp//  
    String sql = "SELECT * FROM usuario WHERE email=? AND contraseña=?";  
    PreparedStatement stmt = conn.prepareStatement(sql);  
    stmt.setString(1, email);  
    stmt.setString(2, contraseña);  
  
    ResultSet rs = stmt.executeQuery();  
  
    //aquí se define el resultado, si encuentra al usuario lo manda a bienvenido.jsp//  
    if (rs.next()) {  
        request.setAttribute("mensaje", "Inicio de sesión exitoso, bienvenido " + rs.getString("nombre_apellido"));  
        request.getRequestDispatcher("bienvenido.jsp").forward(request, response);  
    } else {  
        request.setAttribute("mensaje", "Correo o contraseña incorrectos");  
        request.getRequestDispatcher("inicio.jsp").forward(request, response);  
    }  
  
    rs.close();  
    stmt.close();  
    conn.close();  
  
    //cuando ocurre un error (la base no se conecta) se muestra el error en inicio.jsp//  
} catch (Exception e) {  
    e.printStackTrace();  
    request.setAttribute("mensaje", "Error en el login: " + e.getMessage());  
    request.getRequestDispatcher("inicio.jsp").forward(request, response);  
}  
}
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 20 Apartado Visual inicio



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 23 Historia de usuario registro

Identificador:	HU-02
Nombre:	Añadir usuario - Registrarse
Actores:	Administrador - cliente
Objetivo:	Añadir un usuario al sistema
Descripción:	Como administrador quiero tener un sistema de un botón donde se permita crear un usuario para poder tener un perfil en el que puedan pedir sus préstamos a partir de las características (nombre, edad, celular, Gmail) y que se genere un mensaje en donde me confirme si la operación salió correctamente para un mejor sistema
Caso de prueba:	El usuario se ha añadido correctamente
Puntos estimados de esfuerzo	El administrador a calculado cuando podría demorar en agregar un usuario, el cual dice que demora 1 minuto
observaciones	Esta función ayudara a que la persona acceda al sistema
Criterios de aceptación	Para ser aceptado se necesita que al añadir el usuario se le dé la notificación al cliente de que su usuario ha sido activado correctamente

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 24 Caso de uso registro

Identificador:	CU-02
Nombre:	Añadir usuario - Registrarse
Actores:	Administrador - cliente
Objetivo:	Añadir un usuario al sistema
Descripción:	<ol style="list-style-type: none"> 1 el usuario administrador entra a la pagina 2 el usuario administrador le da a donde dice registrarse 3 el usuario administrador pone los datos 4 el usuario administrador le da al botón que dice registrase 5 se devuelve el mensaje de que la operación ha sido correcta

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 21 Código registro

```
1 <!--
2 Document : registro
3 Created on : 12/09/2025, 11:09:46 a.m.
4 Author : santi
5 -->
6
7 <!!page contentType="text/html" pageEncoding="UTF-8"%>
8 <!-- en esta parte del codigo se establece el idioma de la pagina, validacion para telefonos, caracteres especiales y el titulo-->
9 <!DOCTYPE html>
10 <html lang="es">
11 <head>
12 <meta charset="UTF-8">
13 <title>Registro de usuario</title>
14 <style>
15 /* en body se define el color de la pagina, tipo de fuente, color de la fuente, display, organizacion, alineacion de los items */
16 body {
17     background-color: red;
18     font-family: Arial, Helvetica, sans-serif;
19     color: black;
20     display: flex;
21     justify-content: center;
22     align-items: center;
23     height: 100vh;
24 }
25 /* la decoracion buho es la configuracion de la imagen buho.png */
26 .decoracion-buho {
27     position: absolute;
28     bottom: 20px;
29     right: 20px;
30     width: 600px;
31     z-index: 10;
32 }
33 /* este bloque es lo que hace aparecer el bloque blanco donde estan los botones */
34 .form-box {
35     background-color: white;
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 22 Código registro

```
36     padding: 100px;
37     border-radius: 20px;
38     width: 400px;
39 }
40 /* el input son los botones donde se mete la informacion, este bloque es la configuracion como tamaño, margen etc */
41 input {
42     width: 100px;
43     padding: 12px;
44     margin: 12px 0;
45     border: 2px solid #ff2d2d;
46     border-radius: 10px;
47 }
48 /* este bloque es la configuracion del boton registrar que hace la ejecucion */
49 button {
50     background-color: #ff2d2d;
51     color: white;
52     padding: 15px;
53     border: none;
54     border-radius: 10px;
55     width: 100%;
56     font-weight: bold;
57     cursor: pointer;
58 }
59 /* el hover es para que cuando el usuario pase el cursor el boton cambie de opacidad */
60 button:hover {
61     opacity: 0.9;
62 }
63 </style>
64 </head>
65 <body>
66 <!-- en esta parte podremos colocar la imagen buho.png -->
67 
68 <!-- en esta division son las cosas que pondremos dentro del cuadro blanco -->
69 <div class="form-box">
70     <h2>INGRESA LOS SIGUIENTES DATOS</h2>
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 23 Código registro

```
71         <!-- este apartado es el mensaje de error que se pone al no poner todos los datos -->
72         <% String mensaje = (String) request.getAttribute("mensaje");
73         if (mensaje != null) { %>
74             <p style="color: red; font-weight: bold;"><%= mensaje %></p>
75         <% } %>
76
77         <!-- este es el cuestionario que recolecta los datos y los manda al servlet -->
78         <form action="RegistroServlet" method="POST">
79             <input type="text" name="nombre_apellido" placeholder="Nombre completo" required>
80             <input type="text" name="telefono" placeholder="Teléfono">
81             <input type="number" name="edad" min="7" max="50" placeholder="Edad" required>
82             <input type="email" name="email" placeholder="Correo electrónico" required>
83             <input type="password" name="contraseña" placeholder="Contraseña" required>
84             <button type="submit">Registrarse</button>
85         </form>
86     </div>
87 </body>
88 </html>
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 24 Servlet registro

```
1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this template
4   */
5  package Servlets;
6
7  import java.io.IOException;
8  import java.sql.Connection;
9  import java.sql.DriverManager;
10 import java.sql.PreparedStatement;
11 import jakarta.servlet.ServletException;
12 import jakarta.servlet.annotation.WebServlet;
13 import jakarta.servlet.http.HttpServlet;
14 import jakarta.servlet.http.HttpServletRequest;
15 import jakarta.servlet.http.HttpServletResponse;
16
17 @WebServlet("/RegistroServlet")
18 public class RegistroServlet extends HttpServlet {
19
20     //conexion con la base de datos//
21     private static final String DB_URL = "jdbc:mysql://localhost:3306/proyecto_software";
22     private static final String DB_USER = "root";
23     private static final String DB_PASS = "2007";
24
25     //maneja las peticiones POST de registro.jsp//
26     @Override
27     protected void doPost(HttpServletRequest request, HttpServletResponse response)
28         throws ServletException, IOException {
29         //captura de datos del formulario//
30         //cada getParameter es un name de de los <input> de registro.jsp//
31         String nombre_apellido = request.getParameter("nombre_apellido");
32         String telefono = request.getParameter("telefono");
33         String edad = request.getParameter("edad");
34         String email = request.getParameter("email");
35         String contraseña = request.getParameter("contraseña");
```

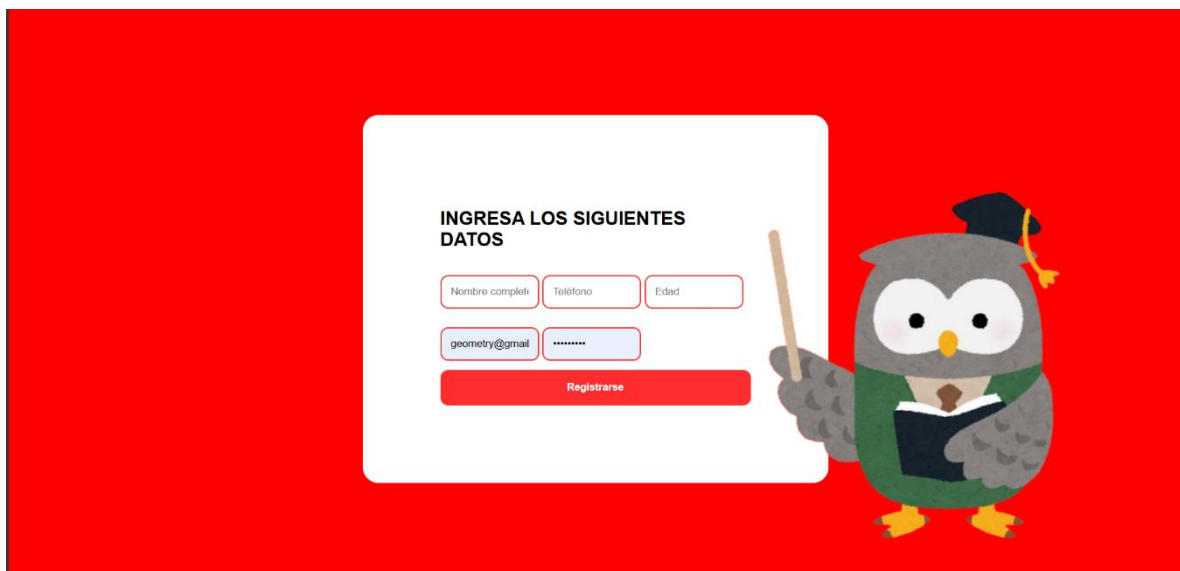
Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 25 Servlet registro

```
38 try {
39     Class.forName("com.mysql.cj.jdbc.Driver");
40     Connection conn = DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);
41
42     //aqui se inserta los datos del formulario, cada ? se reemplazan por los datos del formulario//
43     String sql = "INSERT INTO usuario (nombre_apellido, telefono, edad, email, contraseña, rol) VALUES (?, ?, ?, ?, ?, ?)";
44     PreparedStatement stmt = conn.prepareStatement(sql);
45     stmt.setString(1, nombre_apellido);
46     stmt.setString(2, telefono);
47     stmt.setString(3, edad);
48     stmt.setString(4, email);
49     stmt.setString(5, contraseña);
50     stmt.setString(6, "usuario"); //el rol se establece automaticamente como usuario//
51     //en este apartado se ejecuta y se verifica el proceso//
52     int filas = stmt.executeUpdate();
53
54     //si el numero de filas es mayor a 0, significa que el registro fue exitoso//
55     if (filas > 0) {
56         request.setAttribute("mensaje", "Registro exitoso");
57         request.getRequestDispatcher("inicio.jsp").forward(request, response);
58     } //si no se cumple, aparece un mensaje y te devuelve a registro.jsp//
59     else {
60         request.setAttribute("mensaje", "No se pudo registrar");
61         request.getRequestDispatcher("registro.jsp").forward(request, response);
62     }
63     stmt.close();
64     conn.close();
65     //cuando ocurre un error (la base no se conecta) se muestra el error en registro.jsp//
66 } catch (Exception e) {
67     e.printStackTrace();
68     request.setAttribute("mensaje", "Error: " + e.getMessage());
69     request.getRequestDispatcher("registro.jsp").forward(request, response);
70 }
71 }
72 }
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 26 Apartado visual registrarse



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

[Página Principal](#)

Ilustración 27 Código index

```
2 <!DOCTYPE html>
3 <html lang="es">
4 <!-- en head se configuran que la pagina soporte caracteres especiales, que sea responsivo para moviles-->
5 <head>
6 <meta charset="UTF-8">
7 <meta name="viewport" content="width=device-width, initial-scale=1.0">
8 <title>Pantalla Inicial - I.E.D Los Santos</title>
9 <!-- dentro de style se define el apartado visual de la pagina -->
10 <style>
11 /* en body se define el margen y el tipo de fuente */
12 body {
13     margin: 0;
14     font-family: Arial, sans-serif;
15 }
16 /* en header se configuro la franja roja donde aparece el nombre */
17 .header {
18     background-color: #ff2d2d;
19     padding: 80px;
20     text-align: center;
21     color: black;
22     font-size: 28px;
23     font-weight: bold;
24 }
25 /* Este es el contenedor principal dividido en dos columnas */
26 .container {
27     display: flex;
28     flex-direction: row;
29     height: calc(100vh - 120px);
30     background-color: white;
31 }
32 /* en left muestra la imagen de caminando.png */
33 .left {
34     flex: 1;
35     display: flex;
36     flex-direction: column;
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 28 Código index

```
37     align-items: center;
38     justify-content: center;
39 }
40 /* configuracion de la imagen */
41 .left img {
42     max-width: 80%;
43     margin: 10px;
44 }
45 /* el .right contiene los botones de registro y inicio de sesion */
46 .right {
47     flex: 1;
48     display: flex;
49     flex-direction: column;
50     justify-content: center;
51     align-items: flex-start;
52     padding: 40px;
53     gap: 20px;
54 }
55 /* En button podemos definir el estilo de los botones */
56 .button {
57     display: flex;
58     align-items: center;
59     justify-content: flex-start;
60     width: 300px;
61     padding: 15px 20px;
62     font-size: 18px;
63     font-weight: bold;
64     border: none;
65     border-radius: 50px;
66     cursor: pointer;
67     transition: 0.3s;
68 }
69 /* aqui se define el color del boton blanco con un borde negro */
70 .btn-registrar {
71     background-color: white;
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 29 Código index

```
72     border: 2px solid #000;
73 }
74 /* boton rojo con un texto blanco */
75 .btn-iniciar {
76     background-color: #ff2d2d;
77     color: white;
78 }
79 /* se configura la imagen avatar.png de los botones */
80 .button img {
81     width: 30px;
82     height: 30px;
83     margin-right: 10px;
84     border-radius: 50%;
85     background: #ddd;
86     padding: 3px;
87 }
88 /* cuando el usuario pasa el raton por el boton baja la opacidad */
89 .button:hover {
90     opacity: 0.8;
91 }
92 </style>
93 </head>
94 <body>
95     <!-- en este apartado podemos poner el titulo en la franja roja -->
96     <div class="header">
97         INSTITUCIÓN EDUCATIVA LOS SANTOS<br>
98         DONDE SACAMOS A RELUCIR LAS MENTES MAESTRAS
99     </div>
100     <!-- permite colocar la imagen caminando.png -->
101     <div class="container">
102         <div class="left">
103             
104         </div>
105         <!-- en este apartado podremos colocar las imagenes de los botones -->
106         <div class="right">
```


Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 30 Código index

```
107 <a href="registro.jsp" class="button btn-registrar">
108  REGISTRARSE
109 </a>
110
111 <a href="inicio.jsp" class="button btn-iniciar">
112  INICIAR SESIÓN
113 </a>
114
115 </div>
116 </div>
117 </body>
118 </html>
119
```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 31 Apartado visual index



Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Ilustración 32 Base de datos MySQL workbench

```
1 • create database proyecto_software;
2 • use proyecto_software;
3 #biblioteca los santos
4 • CREATE TABLE usuario (
5     id_usuario INT AUTO_INCREMENT PRIMARY KEY,
6     nombre_apellido VARCHAR (50) NOT NULL,
7     telefono VARCHAR (15),
8     edad INT NOT NULL CHECK (edad >= 7 AND edad <=50),
9     email VARCHAR (100) UNIQUE,
10    contraseña VARCHAR (20) NOT NULL,
11    rol VARCHAR (50) NOT NULL
12 );
13 • CREATE TABLE libros (
14     id_libro INT AUTO_INCREMENT PRIMARY KEY,
15     titulo VARCHAR (50) NOT NULL,
16     autor VARCHAR (50) NOT NULL,
17     año_publicacion YEAR,
18 ✖    genero VARCHAR (50),usuario,
19     disponibilidad BOOLEAN NOT NULL
20 );
```

Nota: ilustración hecha por Geffry Santiago León Jimenez

Ilustración 33 Base de datos MySQL workbench

```

21 • CREATE TABLE empleados (
22     id_empleado INT AUTO_INCREMENT PRIMARY KEY,
23     nombre_apellido VARCHAR (100) NOT NULL,
24     correo_electronico VARCHAR (50) UNIQUE NOT NULL,
25     cargo VARCHAR (50) NOT NULL,
26     telefono VARCHAR (50)
27 );
28 • CREATE TABLE prestamos (
29     id_prestamo INT AUTO_INCREMENT PRIMARY KEY,
30     id_usuario INT NOT NULL,
31     id_empleado INT NOT NULL,
32     id_libro INT NOT NULL,
33     fecha_prestamo DATE NOT NULL,
34     fecha_devolucion_limite DATE,
35     fecha_devolucion_real DATE NOT NULL,
36     estado_prestamo VARCHAR (30) NOT NULL,
37     FOREIGN KEY (id_usuario) REFERENCES usuario (id_usuario),
38     FOREIGN KEY (id_empleado) REFERENCES empleados (id_empleado),
39     FOREIGN KEY (id_libro) REFERENCES libros (id_libro)
40 );

```

Nota: ilustración hecha por Geffry Santiago Leon Jimenez

Tabla 25 Lista de chequeo

Nombre del proveedor/Aprendiz	Geffry Leon		Gmail	Santileonj03@gmail.com
Identificación o nit	1099740435			
Dirección	Calle 15ª # 29-22	Fecha	23/09/2025	
teléfono	3108672387			
Diagramas				
REQUISITOS	CUMPLE			OBSERVACIONES
	SI	NO	N/A	
Diagrama de clases	X			El diagrama de clases fue realizado
Diagrama de casos de uso	X			El diagrama de casos de uso se encuentra realizado

Historia de usuario	X			Las historias de usuarios están hechas
Diseños				
Arquitectura del sistema	X			La arquitectura del sistema se encuentra definida
Prototipos de pantalla	X			Los prototipos están realizados
Mockups de pantalla final		X		los mockups no se han realizado
Base de datos				
Tablas definidas	X			Las tablas fueron definidas
Atributos y tipos de datos	X			Se encuentran definidos
Llaves primarias y llaves foráneas	X			Se encuentran definidas
Codificación				
Índex	X			Se encuentra realizada
Inicio de sesión	X			Se encuentra realizada
Recuperar contraseña		X		No se ha realizado
Registro	X			Se encuentra realizada

Pantalla de bienvenida		X		No se ha hecho
Buscar material		X		No se ha hecho
Añadir material		X		No se ha hecho
Modificar material		X		No se ha hecho
Eliminar material		X		No se ha hecho
Pedir prestamos		X		No se ha hecho
Prestamos		X		No se ha hecho
Validación de roles		X		No se han realizado
Estándares de codificación	X			Los estándares están correctos
Comentarios	X			El código si tiene comentarios
Herramienta de versionamiento				
Repositorio creado en GitHub	X			El repositorio ya está creado
Conexión entre repositorio local y repositorio remoto	X			La conexión se encuentra hecha
Nombre	Geffry Santiago Leon Jimenez			

Nota: tabla hecha por Geffry Santiago Leon Jimenez

4. CONCLUSIONES

Estar armando poco a poco el documento en lo que se registra todos los progresos que he hecho me alegra ya que puedo ver lo que eh avanzado en este aprendizaje, y hacer esta lista me permite poder ver que me falta por realizar dándome motivacion

5. REFERENCIAS BIBLIOGRÁFICAS

Git: <https://git-scm.com/downloads/win>

GitHub: <https://github.com/>

https://zajuna.sena.edu.co/Repositorio/Titulada/institution/SENA/Tecnologia/228118/Contenido/DocArtic/GUI7/Guia_aprendizaje_7.pdf