

**UNIVERSIDAD PERUANA LOS ANDES
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS Y COMPUTACIÓN**



REPORTES

CURSO : DESARROLLO DE APLICACIONES
DOCENTE : RAUL ENRIQUE FERNANDEZ BEJARANO
ESTUDIANTE : MEZA MEDINA JARED JOSUÉ
CICLO : IV
SECCION : A1

**2025
HUANCAYO**

REPORTE DEL PROCESO DE DESARROLLO

1. Introducción

El presente reporte describe de manera detallada el proceso de diseño, desarrollo e implementación de una aplicación web para la gestión de tesis universitarias. El sistema fue desarrollado con el objetivo de automatizar y organizar el proceso académico que involucra a estudiantes, docentes y administradores, desde el registro de tesis hasta la evaluación final.

La aplicación fue construida utilizando tecnologías Java Web (JSP, Servlets, JDBC), una base de datos relacional MySQL Workbench y un diseño basado en el patrón MVC (Modelo–Vista–Controlador).

2. Objetivos del Sistema

Objetivo General

Desarrollar una aplicación web que permita gestionar el proceso completo de tesis universitarias, asegurando un flujo ordenado y seguro de información entre estudiantes, docentes y administradores.

Objetivos Específicos

Gestionar usuarios con distintos roles (Administrador, Docente y Estudiante).

Permitir al administrador registrar tesis y asignar docentes evaluadores.

Facilitar al docente la evaluación de tesis mediante una rúbrica.

Permitir al estudiante visualizar el estado y resultado de su tesis.

Centralizar la información en una base de datos segura y estructurada.

3. Tecnologías Utilizadas

Lenguaje de programación: Java

Tecnologías web: JSP, Servlets

Base de datos: MySQL Workbench

Acceso a datos: JDBC

Servidor: Apache Tomcat

Frontend: HTML, CSS

Arquitectura: Modelo–Vista–Controlador (MVC)

4. Diseño de la Base de Datos

Se diseñó una base de datos relacional llamada tesisdb que contiene las siguientes tablas:

Usuario: almacena credenciales y rol del sistema.

Estudiante: información académica del estudiante.

Docente: información del docente y su especialidad.

Tesis: registro de tesis, estado y asignación de docente.

Evaluación: resultados de la evaluación realizada por el docente.

Se utilizaron claves foráneas para garantizar la integridad referencial entre las tablas.

5. Desarrollo por Roles

5.1 Administrador

El administrador es el rol con mayores privilegios dentro del sistema.

Funciones principales:

Crear cuentas de estudiantes y docentes.

Generar correos automáticamente a partir del código.

Registrar y eliminar tesis.

Asignar docentes evaluadores a tesis.

Visualizar listados generales y resúmenes de procesos.

Durante el desarrollo se implementaron validaciones de sesión para evitar accesos no autorizados.

5.2 Docente

El docente tiene acceso únicamente a las tesis que le han sido asignadas.

Funciones principales:

Visualizar tesis asignadas.

Acceder al archivo PDF de la tesis.

Evaluar la tesis mediante una rúbrica.

Registrar comentarios y nota final.

Se desarrolló un formulario dinámico que calcula automáticamente la nota final a partir de criterios parciales.

5.3 Estudiante

El estudiante tiene un rol pasivo dentro del sistema.

Funciones principales:

Visualizar información de su tesis.

Consultar el estado de evaluación.

Ver la nota final y comentarios del docente.

El estudiante no puede subir ni modificar tesis, garantizando la integridad del proceso.

6. Principales Problemas Encontrados y Soluciones

6.1 Problemas con la Base de Datos

Error: Columnas inexistentes (Error 1054).

Causa: Inconsistencia entre el modelo y la base de datos.

Solución: Revisión completa del esquema y alineación con los DAOs.

6.2 Problemas de Sesión

Error: Redirección constante al login.

Causa: No se validaba correctamente el atributo usuario en sesión.

Solución: Implementación de validaciones en cada JSP y controlador.

6.3 Asignación de Docente no Persistía

Error: idDocenteAsignado aparecía como NULL.

Causa: Error en el método SQL UPDATE.

Solución: Corrección de parámetros y verificación de filas afectadas.

6.4 Problemas de Diseño (CSS)

Error: El menú lateral desaparecía al agregar el logo.

Causa: Uso incorrecto de overflow y position.

Solución: Ajuste del layout y encapsulación del logo en un contenedor.

7. Seguridad y Buenas Prácticas

Uso de sesiones para control de acceso.

Separación de responsabilidades mediante MVC.

Uso de PreparedStatement para evitar inyección SQL.

Validación de roles antes de cada acción.

8. Resultados Obtenidos

El sistema cumple satisfactoriamente con todos los requerimientos planteados:

Flujo completo del proceso de tesis.

Roles correctamente definidos.

Persistencia de datos confiable.

Interfaz clara y funcional.

9. Conclusiones

El desarrollo de la aplicación permitió integrar conocimientos de programación web, bases de datos y arquitectura de software. A pesar de los múltiples errores encontrados durante el proceso, cada problema contribuyó a una mejor comprensión del funcionamiento interno de aplicaciones Java Web.

El sistema desarrollado es escalable, mantenible y cumple con los estándares básicos de una aplicación académica real.

10. Recomendaciones

Implementar cifrado de contraseñas.

Mejorar la interfaz con frameworks modernos.

Añadir reportes en PDF.

Implementar control de versiones.