



INFORME TECNICO CINE SUCRE

SERVICIO NACIONAL DE APRENDIZAJE
SENA



[FECHA]

[NOMBRE DE LA COMPAÑÍA]

[Dirección de la compañía]

Contenido

Informe Técnico: Cine Sucre	2
1. Introducción.....	2
2. Título del Proyecto	2
3. Resumen	2
4. Objetivos	3
4.1 Objetivo General	3
4.2 Objetivos Específicos	3
5. Alcance del Proyecto	3
6. Análisis del Problema.....	4
6.2 Justificación	4
6.3 Usuarios Involucrados	4
7. Diseño del Aplicativo Web	4
7.1 Arquitectura del Sistema	4
7.2 Diseño de Interfaces	4
7.3 Flujo de Navegación.....	5
8. Desarrollo e Implementación	5
8.1 Tecnologías Utilizadas.....	5
8.2 Estructura del Proyecto.....	5
8.3 Seguridad y Autenticación.....	6
8.4 Implementación de Funcionalidades	6
9. Pruebas y Validación.....	6
9.1 Pruebas de Funcionalidad	6
9.2 Pruebas de Interfaz	6
9.3 Pruebas de Seguridad	6
10. Despliegue del Aplicativo.....	6
10.1 Configuración del Servidor	6
10.2 Instalación y Ejecución	7
11. Conclusiones y Recomendaciones	7
11.1 Resultados Obtenidos.....	7
11.2 Limitaciones	7
11.3 Mejoras Futuras	7
12. Bibliografía y Referencias	7

Informe Técnico: Cine Sucre

1. Introducción

Este documento describe el desarrollo de un aplicativo web para la gestión de una plataforma de streaming de películas llamada **Cine Sucre**. El sistema está diseñado para gestionar clientes, empleados, películas, planes, categorías y facturas, con una interfaz moderna y responsiva. El proyecto utiliza tecnologías como **Flask** para el backend, **MySQL** para la base de datos y **Tailwind CSS 4** para el frontend, garantizando una experiencia de usuario atractiva y funcional.

2. Título del Proyecto

Cine Sucre

3. Resumen

El aplicativo web **Cine Sucre** permite la gestión eficiente de usuarios, películas, planes, facturas y categorías, proporcionando una interfaz moderna y atractiva.

- **Problema resuelto:** Gestión centralizada de recursos en una plataforma de streaming, incluyendo películas, usuarios, planes y facturas.
- **Usuarios principales:**
 - **Clientes:** Interesados en adquirir planes de streaming y explorar el catálogo de películas.
 - **Empleados:** Encargados de gestionar ventas y clientes.
 - **Administradores:** Responsables de la gestión integral del sistema.
- **Funcionalidades principales:**
 - CRUD para gestión de clientes, empleados, películas, planes, categorías y facturas.

- Autenticación de usuarios (login, registro y recuperación de contraseña).
- Visualización de películas y planes disponibles.
- Paginación en las tablas de datos para mejorar la usabilidad.

4. Objetivos

4.1 Objetivo General

Desarrollar un aplicativo web que facilite la gestión integral de una plataforma de streaming de películas, ofreciendo una interfaz moderna y funcionalidades robustas para administrar recursos.

4.2 Objetivos Específicos

- Diseñar una interfaz responsiva y atractiva utilizando **Tailwind CSS 4**.
- Implementar funcionalidades CRUD para la gestión de clientes, empleados, películas, planes, categorías y facturas.
- Integrar un sistema de autenticación seguro utilizando **Flask-Login**.
- Garantizar una experiencia de usuario moderna y eficiente.
- Implementar paginación en las tablas de datos para mejorar la usabilidad.

5. Alcance del Proyecto

Incluye:

- Registro, login y recuperación de cuenta.
- Gestión de clientes, empleados, películas, planes, categorías y facturas.
- Interfaz responsiva y moderna.
- Paginación en las tablas de datos.

No incluye:

- Integración con pasarelas de pago.
- Aplicaciones móviles nativas.

6. Análisis del Problema

6.1 Planteamiento del Problema

Cine Sucre es una empresa ficticia que necesita una plataforma web para gestionar su catálogo de películas, planes de streaming y usuarios. Actualmente, carece de una herramienta centralizada para la gestión de estos recursos, lo que dificulta la organización y escalabilidad del negocio.

6.2 Justificación

El aplicativo web optimiza los procesos de gestión, mejora la experiencia del usuario y aumenta la eficiencia operativa. Además, proporciona una interfaz moderna y responsiva que facilita la interacción con el sistema.

6.3 Usuarios Involucrados

- **Administradores:** Gestionan el sistema y usuarios.
- **Empleados:** Manejan ventas y clientes.
- **Clientes:** Consultan películas y adquieren planes de streaming.

7. Diseño del Aplicativo Web

7.1 Arquitectura del Sistema

El sistema sigue el modelo **MVC (Modelo-Vista-Controlador)**:

- **Modelo:** Lógica de negocio y base de datos (Flask-SQLAlchemy).
- **Vista:** Interfaz web desarrollada con HTML, CSS, JavaScript y Tailwind CSS 4.
- **Controlador:** Manejo de solicitudes y lógica de aplicación (Flask).

7.2 Diseño de Interfaces

Se diseñaron prototipos de las pantallas principales, incluyendo:

- Inicio de sesión.
- Panel de administración.
- Gestión de clientes, empleados, películas, planes, categorías y facturas.

7.3 Flujo de Navegación

Se creó un diagrama de navegación que muestra la interacción entre las distintas páginas del aplicativo, garantizando una experiencia de usuario intuitiva.

8. Desarrollo e Implementación

8.1 Tecnologías Utilizadas

- **Backend:** Flask (Python).
- **Frontend:** HTML, CSS, JavaScript, Tailwind CSS 4.
- **Base de Datos:** MySQL (Flask-SQLAlchemy).
- **Control de Versiones:** Git/GitHub.

8.2 Estructura del Proyecto

El proyecto está organizado de la siguiente manera:

Copy

/CineSucre

| — __pycache__/

| — .venv/

| — node_modules/

| — static/

| | — img/

| | — js/

| | — input.css

| | — output.css

| — templates/

| — .gitignore

| — app.py

| — bd.py

| — cinesucre.sql

- | — config.py
- | — package-lock.json
- | — package.json
- | — README.md
- | — requirements.txt

8.3 Seguridad y Autenticación

- **Autenticación:** Implementada con **Flask-Login**.
- **Protección:** Contra inyecciones SQL y ataques CSRF.

8.4 Implementación de Funcionalidades

- **CRUD:** Para clientes, empleados, películas, planes, categorías y facturas.
- **Autenticación:** Login, registro y recuperación de contraseña.
- **Paginación:** Implementada en todas las tablas de datos.

9. Pruebas y Validación

9.1 Pruebas de Funcionalidad

- **Login:** Verificación de acceso solo para usuarios registrados.
- **CRUDs:** Pruebas de creación, lectura, actualización y eliminación de datos.
- **Paginación:** Verificación del correcto funcionamiento en las tablas.

9.2 Pruebas de Interfaz

- **Usabilidad:** Evaluación de la interfaz en distintos dispositivos (responsividad).
- **Experiencia de usuario:** Revisión de la navegación y accesibilidad.

9.3 Pruebas de Seguridad

- **Inyecciones SQL:** Verificación de la protección contra ataques.
- **Accesos no autorizados:** Pruebas de rutas protegidas.

10. Despliegue del Aplicativo

10.1 Configuración del Servidor

- **Plataformas:** Heroku, AWS o un VPS.

- **Pasos:** Configuración del servidor, instalación de dependencias y despliegue.

10.2 Instalación y Ejecución

- **Instalación:** Ejecución de `pip install -r requirements.txt` para instalar dependencias.
- **Ejecución:** Uso de `flask run` para iniciar la aplicación.

11. Conclusiones y Recomendaciones

11.1 Resultados Obtenidos

El aplicativo cumple con los objetivos planteados, ofreciendo una interfaz moderna y funcionalidades robustas para la gestión de recursos.

11.2 Limitaciones

- **Integración con pagos:** No se implementó pasarelas de pago.
- **Aplicaciones móviles:** No se desarrollaron apps nativas.

11.3 Mejoras Futuras

- Integración con pasarelas de pago.
- Desarrollo de aplicaciones móviles nativas.
- Implementación de notificaciones en tiempo real.

12. Bibliografía y Referencias

- Documentación oficial de **Flask**: <https://flask.palletsprojects.com/>
- Documentación oficial de **Tailwind CSS**: <https://tailwindcss.com/>
- Documentación oficial de **MySQL**: <https://dev.mysql.com/doc/>
- Documentación oficial de **Flask-SQLAlchemy**: <https://flask-sqlalchemy.palletsprojects.com/>
- Tutoriales y guías de desarrollo web.