

SKYROUTE TRAVEL SYSTEM: MODULARIZACIÓN, PERSISTENCIA Y EXPERIENCIA DE USUARIO AMPLIADA

TIPO DE PROYECTO:
Tecnológico

ESPACIO CURRICULAR/MÓDULO:
Módulo programador

OBJETIVO GENERAL:

Desarrollar un prototipo del sistema SkyRoute Travel System, para la gestión eficiente de transacciones de vuelos, implementando modularización, uso de estructuras de datos y conexión con una base de datos MySQL para la gestión persistente de clientes, destinos, ventas y la funcionalidad de arrepentimiento.

PROBLEMÁTICAS/NECESIDADES:

SkyRoute S.R.L. necesita digitalizar su gestión de pasajes para superar problemas de procesos manuales, falta de integración y errores. Esto afecta la eficiencia, la disponibilidad de asientos, la experiencia del cliente y la seguridad de datos.

Un nuevo sistema es crucial para:

- Gestionar clientes y destinos sin errores.
- Controlar ventas de forma eficiente.
- Implementar un "botón de arrepentimiento" legalmente.
- Generar reportes de ventas fácilmente.
- Asegurar los datos con cifrado y autenticación multifactor.

OBJETIVO ESPECÍFICO:

Diseño de Base de Datos (MySQL): Crear el esquema relacional para clientes, destinos y ventas, incluyendo sus relaciones y restricciones (DDL).

Desarrollo Modular en Python: Construir módulos como `gestion_clientes.py` y `config.py` para encapsular la lógica de negocio y las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) con MySQL.

Funcionalidades CRUD Completas: Implementar las operaciones de agregar, listar, modificar y eliminar para clientes, destinos y ventas, usando tanto estructuras de datos en memoria como persistencia en MySQL.

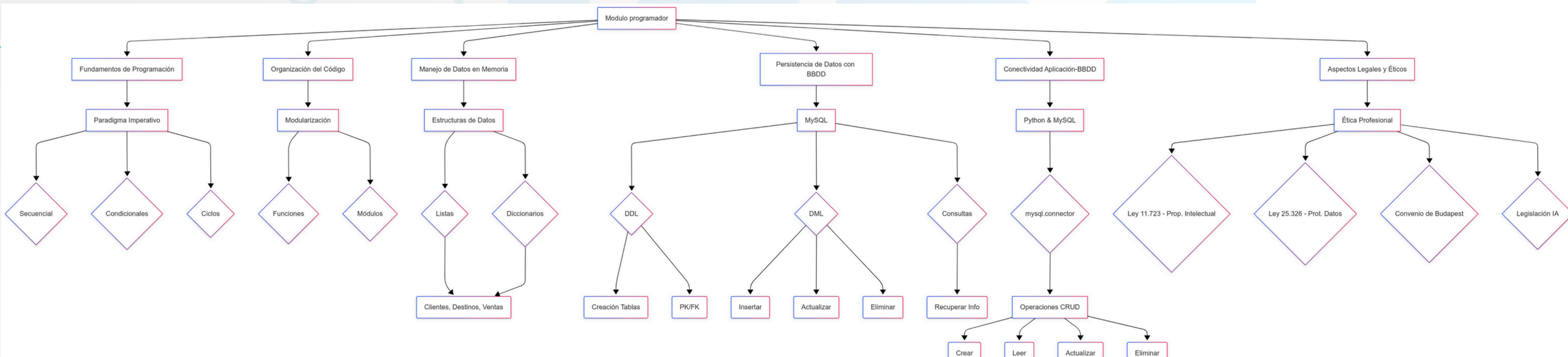
"Botón de Arrepentimiento": Desarrollar la funcionalidad para anular ventas, cumpliendo con la normativa legal.

Menú Interactivo: Crear `main.py` como un punto de entrada con un menú que integre todas las funcionalidades.

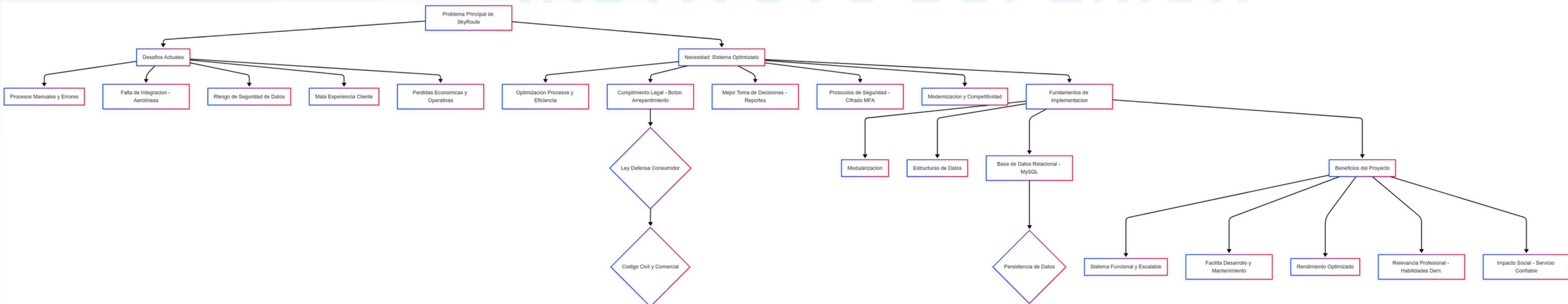
Consultas: Generar sentencias SQL (DML) para insertar datos de prueba y consultas relevantes para análisis.

Documentación del Proyecto: Elaborar un archivo `README.md`

EJE TEMÁTICO/RED DE CONCEPTOS:



FUNDAMENTACIÓN:



ACCIONES:

8. SELECCIÓN DE ACCIONES

OBJETIVO ESPECÍFICO	ACCIONES
1- Diseñar e implementar el esquema de la base de datos relacional (DDL).	- Definir el modelo entidad-relación. - Escribir sentencias DDL para crear tablas (Clientes, Destinos, Ventas). - Establecer claves primarias y foráneas, y restricciones. - Crear el script SQL para la estructura de la base de datos.
2-Desarrollar los módulos Python (gestion_clientes.py, gestion_destinos.py, gestion_ventas.py, conexion_base_datos.py, config.py).	- Configurar <code>config.py</code> para la conexión a la base de datos. - Desarrollar <code>conexion_base_datos.py</code> para manejar la conexión y operaciones básicas con MySQL. - Crear estructuras de carpetas del proyecto. - Implementar la estructura básica de los módulos.
3- Implementar la funcionalidad completa de gestión de clientes, destinos y ventas.	- Desarrollar funciones para agregar, listar, modificar y eliminar clientes, destinos y ventas. - Integrar con la base de datos. - Usar listas/diccionarios para manejo temporal.
4- Implementar el "botón de arrepentimiento".	- Desarrollar la lógica en <code>gestion_ventas.py</code> para anular ventas recientes. - Registrar fecha y hora de anulación en la base de datos. - Definir la escala de tiempo para la anulación
5- Crear el archivo main.py con un menú interactivo.	- Diseñar la interfaz del menú principal. - Importar y llamar a las funciones de los módulos correspondientes. - Implementar el bucle principal del programa.
6- Generar sentencias SQL (DML) para datos de ejemplo y consultas relevantes.	- Escribir sentencias DML para insertar al menos 3 registros por tabla. - Escribir al menos 5 consultas SQL relevantes (SELECT). - Agrupar las sentencias DDL y DML en archivos <code>.sql</code> comentados.
7- Elaborar el archivo README.md.	- Redactar la descripción del proyecto. - Incluir integrantes y usuarios de GitHub. - Detallar instrucciones de ejecución. - Describir el contenido del repositorio.

PRODUCTO FINAL:

El producto final de este proyecto es un sistema de gestión de viajes funcional, modular y persistente en Python, conectado a una base de datos MySQL

README



SkyRoute S.R.L. - Sistema de Gestión de Pasajes Aéreos

Python 3.x Estado Funcional con Base de Datos

Propósito del Sistema

SkyRoute es un sistema de gestión de pasajes aéreos desarrollado para administrar clientes, destinos y ventas de manera eficiente.

En su versión actual, el sistema permite la gestión real de datos mediante la integración con una base de datos MySQL, ofreciendo funcionalidades completas para el registro, consulta y modificación de información relevante. El objetivo es brindar una herramienta práctica y robusta para la administración de una agencia de viajes, facilitando la operatoria diaria y el acceso a reportes y consultas en tiempo real.

- Lógica de negocio implementada.
- Conexión real a base de datos MySQL.
- Código modularizado en funciones y archivos independientes.
- Validaciones y gestión de datos reales.
- Manejo de errores y mensajes al usuario.

INTEGRANTES:

- ROCÍO AYLÉN FISCHER DNI: 40773260
- JULIETA STRADA DNI: 30124731
- SABRINA ELIZABETH MALDONADO DNI: 35579029
- OSCAR PONCE DNI: 33123602
- JUAN CALERO DNI: 43231250

DOCENTES:

CHARLETTI, CARLOS
CONDE, JULIAN
HERNANDEZ, ANAHI

TECNICATURA:

TECNICATURA SUPERIOR EN CIENCIA DE DATOS E INTELIGENCIA ARTIFICIAL

COHORTE: 2025