

🎵 Aplicación de Tour Musical 🎵

🚀 Descripción del Problema

Los amantes de la música a menudo se enfrentan a desafíos al tratar de descubrir y asistir a eventos y conciertos musicales. La información sobre estos eventos puede estar dispersa y ser difícil de encontrar. Además, la planificación de la ruta y el horario para asistir a varios eventos puede ser compleja y consumir mucho tiempo.

🎯 Objetivo del Proyecto

Desarrollar una aplicación de escritorio que permita a los usuarios descubrir eventos musicales, planificar rutas y compartir su experiencia de manera eficiente y agradable.

👛 Estudio del Negocio

Basándonos en nuestra comprensión de las necesidades del cliente, hemos identificado las siguientes características clave para la aplicación:

Características Principales

1. **Índice de Eventos** 📄: Lista de eventos musicales con detalles esenciales como el nombre del evento, el artista, el género musical y la ubicación.
2. **Búsqueda y Filtrado de Eventos** 🔍: Permite a los usuarios buscar eventos por nombre, género y artista, y filtrar eventos por ubicación y horario.
3. **Mapa y Planificado de Rutas** 🗺️: Visualización de la ubicación de los eventos en un mapa y una herramienta para planificar rutas para asistir a varios eventos.
4. **Reseñas y Calificaciones** ⭐: Los usuarios pueden escribir reseñas y calificar los eventos a los que han asistido, proporcionando valiosos comentarios a otros usuarios.
5. **Historial de Eventos Asistidos** 📖: Los usuarios pueden ver un registro de los eventos a los que han asistido en el pasado.
6. **Señalador de Ánimo del Comentario** 😊/😞: Los usuarios pueden indicar si su comentario sobre la experiencia es positivo o negativo.
7. **Experiencia, Información que Puede Contener Spoilers** 🕵️: Los usuarios pueden compartir detalles sobre su experiencia en el evento.

Características Opcionales

1. **Compartir en Redes Sociales** 🚀: Permite a los usuarios compartir eventos y ubicaciones en sus redes sociales.
2. **Destacado de Eventos Próximos** 🕒: Resalta los eventos que están a punto de comenzar.

Diseño UI

Para la interfaz de usuario, buscamos un diseño limpio y moderno que se alinee con la estética de los eventos musicales.

Tipografía

Para la tipografía, proponemos usar una combinación de fuentes Sans Serif para un aspecto moderno y limpio. Podemos usar **'Roboto'** para los títulos y **'Open Sans'** para el texto del cuerpo. Estas son legibles y se ven bien en una variedad de tamaños de pantalla.

Paleta de Colores



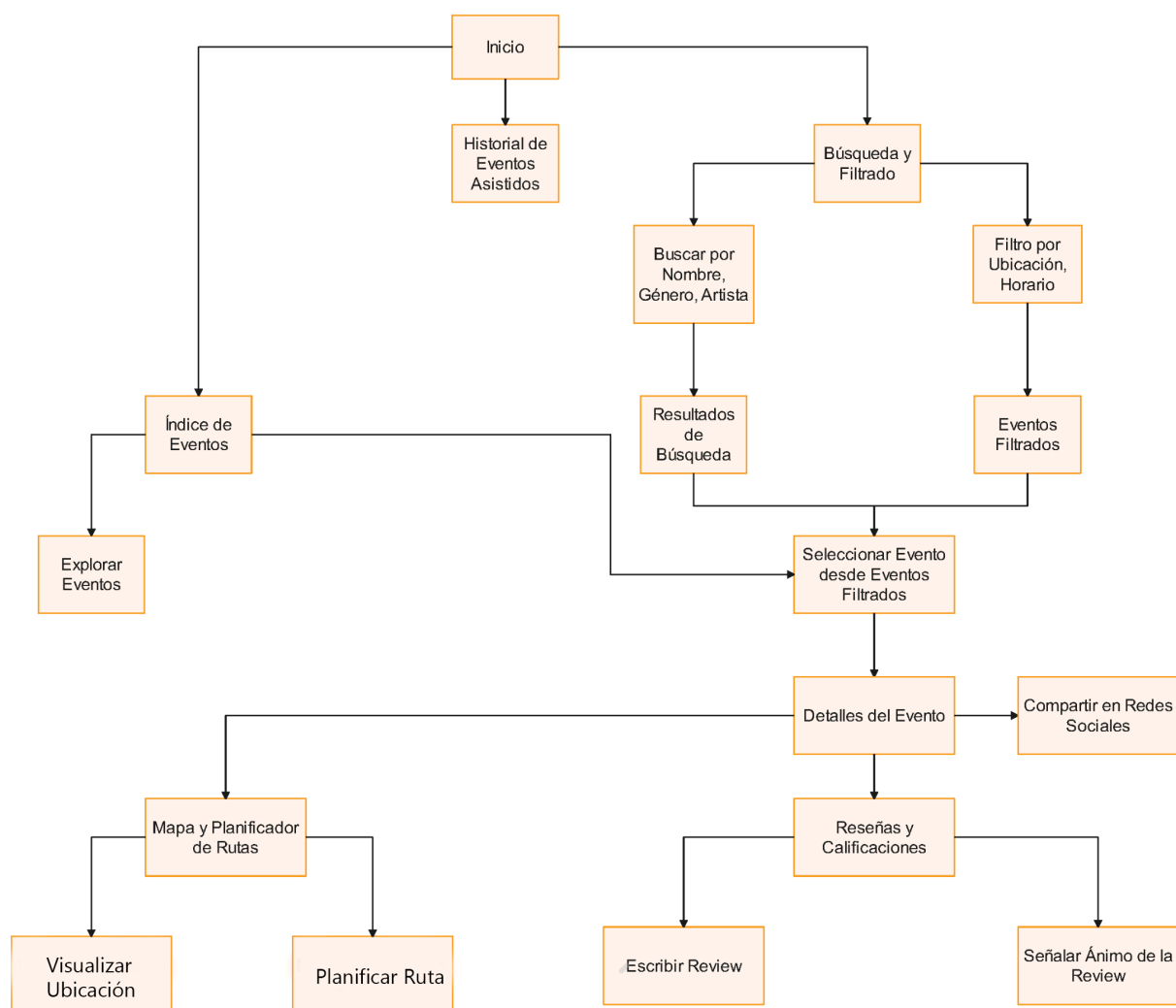
Nuestra paleta de colores se inspira en el mundo de la música y busca equilibrar la energía y la calma. Proponemos los siguientes colores:

- ☐ **Púrpura Oscuro (#2F242C)** 🎵: Este color se utilizará para los títulos y el texto importante. El púrpura oscuro es un color relajante que también proporciona un buen contraste con el fondo.
- ☐ **Gris Claro (#E5E5E5)** 🎵: Este color se utilizará para el texto del cuerpo y los fondos de las tarjetas de eventos (si se utilizan). El gris claro proporciona alto contraste con el púrpura oscuro, lo que facilita la lectura.
- ☐ **Amarillo (#E6D884)** 🎵: Este color se utilizará para resaltar información importante y botones interactivos. El amarillo es un color energético que atrae la atención del usuario.
- ☐ **Verde Claro (#A1A892)** 🎵: Este color se utilizará para los bordes y los elementos de la interfaz de usuario secundarios. El verde claro proporciona un contraste suave con el blanco y el azul oscuro.

Se ofrece la posibilidad de elegir una paleta alternativa para tener una imagen más diferenciada utilizando sitios como:

- <http://colormind.io/>
- <https://coolers.co/>
- <https://colorhunt.co/>
- <https://realtimecolors.com/>

Flujo de Usuario



La pantalla principal contará con tres botones que separan las secciones más importantes de la aplicación. Estos son, el índice de eventos, la herramienta de búsqueda y filtrado de eventos, y el historial de eventos.

La segunda pantalla con opciones es la de los detalles del evento. Aquí el usuario tiene la posibilidad de utilizar mapas para ubicar visualmente los eventos, y por otro lado las reviews en donde se puede calificar un evento dejando un comentario y el estado de ánimo respecto al evento asistido.

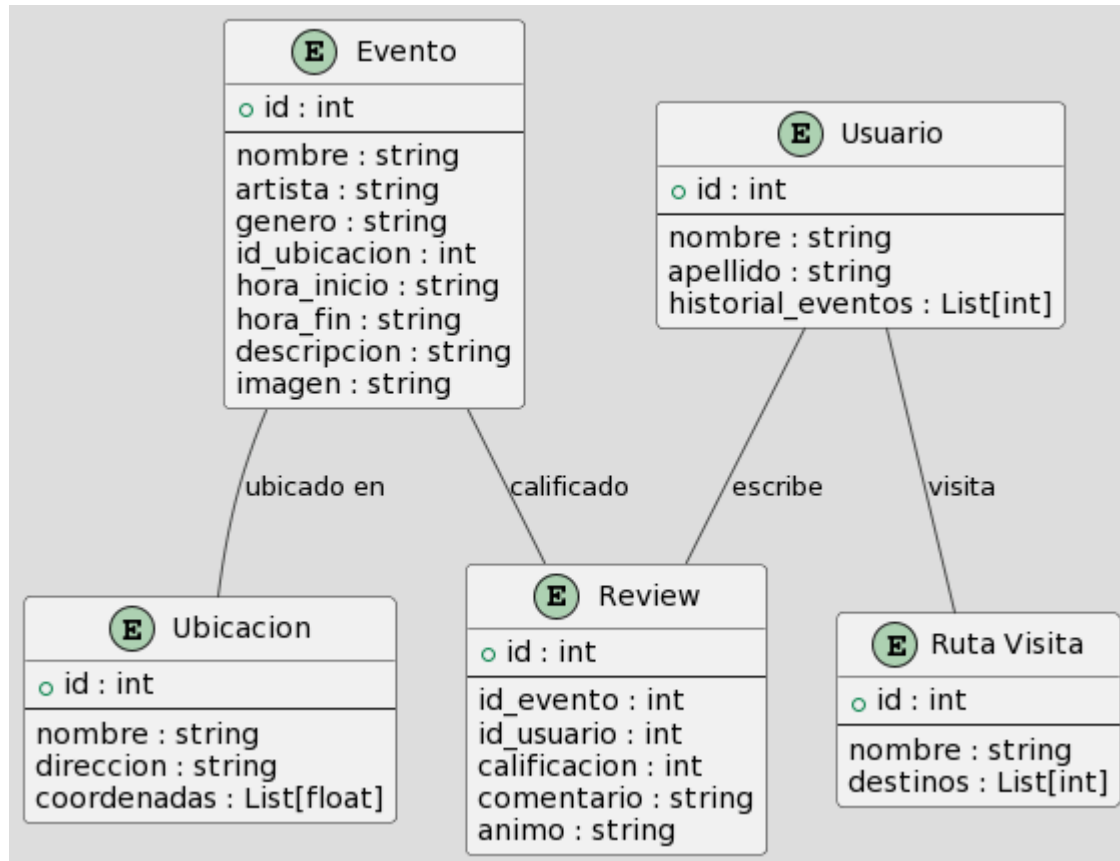
El cliente propondrá aplicaciones de funcionamiento similar para utilizar como guía para el diseño de la interfaz gráfica de la aplicación.

Se propone la siguiente estructura de proyecto:

```
/raíz del proyecto
  /views
    archivos.py que definen la interfaz con Tkinter
  /models
    archivos.py que definen cada uno de los modelos
  /data
    archivos.json con la información almacenada
  /assets
    recursos gráficos como imágenes, fuentes, íconos, etc.
  main.py (Archivo principal que iniciará la aplicación)
```

Requerimientos de Modelo

Los modelos principales propuestos a crear para almacenar los datos son los siguientes.



Evento

- **id** (int): El identificador único del evento.
- **nombre** (str): El nombre del evento.
- **artista** (str): El artista o banda que presenta el evento.
- **genero** (str): El género musical del evento.
- **id_ubicacion** (int): El identificador del lugar donde se lleva a cabo el evento.
- **hora_inicio** (str: datetime ISO 8601): La hora en que comienza el evento, almacenada como un string en formato ISO 8601. Ejemplo: 2023-07-04T09:00:00.
- **hora_fin** (str: datetime ISO 8601): La hora en que finaliza el evento.
- **descripcion** (str): Una breve descripción del evento.
- **imagen** (str): URL de la imagen del evento. Es posible utilizar implementaciones alternativas con imágenes locales.

Ruta Visita

- **id** (int): El identificador único de la ruta de visita.
- **nombre** (str): El nombre de la ruta de visita.
- **destinos** (List[int]): Los identificadores de los destinos musicales que forman parte de la ruta de visita.

Ubicación

- **id** (int): El identificador único de la ubicación.
- **nombre** (str): El nombre de la ubicación.
- **direccion** (str): La dirección de la ubicación.
- **coordenadas** (List[float]): Las coordenadas geográficas de la ubicación, almacenadas como una lista de números flotantes.

Usuario

- **id** (int): El identificador único del usuario.
- **nombre** (str): El nombre del usuario.
- **apellido** (str): El apellido del usuario.
- **historial_eventos** (List[int]): Una lista de IDs de eventos a los que ha asistido el usuario.

Review

- **id** (int): El identificador único de la review.
- **id_evento** (int): El ID del evento que se está calificando.
- **id_usuario** (int): El ID del usuario que escribió la review.
- **calificacion** (int): La calificación del evento por parte del usuario, en una escala de 1 a 5.
- **comentario** (str): El comentario textual sobre el evento.
- **animo** (str): El ánimo del comentario, puede ser 'Positivo' o 'Negativo'.

Tecnologías a Utilizar

- Archivos **JSON** para almacenar datos
- **Tkinter** para interfaz gráfica de escritorio
- **CustomTkinter** para aplicar estilo a la interfaz
- **TkinterMapView** para integrar mapas a la aplicación

Metodología de Trabajo

El proyecto tendrá un máximo de 4 integrantes que trabajarán en el proyecto.

Las tareas a implementar están disponibles en una plantilla de **Notion** la cual se debe duplicar y asignar tareas a cada integrante.

Utilizaremos un tablero Kanban para realizar un seguimiento del avance de proyectos. El profesor dedicará un espacio de la clase a consultar con los grupos uno por uno en cuanto al avance.

Los estudiantes tienen la posibilidad de realizar sus reuniones en el servidor oficial de Academia CIMNE-IBER. Allí también tendrán el espacio de foros para publicar consultas de manera asincrónica para ser respondidas por los profesores.

Entrega del Trabajo

El proyecto estará alojado en GitHub en un repositorio. Allí se tendrá registros de envío de código por cada uno de los integrantes del equipo.

Según disponibilidad de tiempo, la presentación del proyecto se realizará en la última semana de julio y la semana de agosto (última semana de cursada).

El cumplimiento de las tareas mínimas supondrá la aprobación del curso dado que no hay un desafío sobre el contenido del Módulo 5.