**PLANIFICACIÓN.**

**1. Definición de los requisitos**

Funcionales:

- Los usuarios deben registrarse.

- Los usuarios pueden buscar canciones por título, artista o género.

- Los usuarios pueden reproducir canciones.

No funcionales:

- La página carga rápidamente (tiempo de carga máximo de 3 segundos).

- El diseño se adapta a dispositivos móviles, tablets y escritorios.

- Seguridad en el manejo de datos personales (contraseñas encriptadas, autenticación).

**2. Análisis del problema**

- Problema: Los usuarios desean una página con más apartados donde puedan descubrir y organizar música sin interrupciones, con la capacidad de personalizar su experiencia.

- Solución: Una página web intuitiva, rápida y accesible donde los usuarios puedan ingresar a apartados más sorprendentes, reproducir música y gestionar sus preferencias musicales en un solo lugar.

**3. Elección de la tecnología**

- Base de datos: MongoDB para almacenar información de usuarios, listas de reproducción y artistas.

- Control de versiones:GitHub para el control del código.

**4. Diseño del sistema**

- Modelo de datos:

- Usuarios: nombre, email, contraseña, listas de reproducción.

- Canciones: título, artista, género

- Listas de reproducción: nombre, canciones incluidas, creador, fecha de creación.

- Artistas: nombre, biografía, canciones, seguidores.

**5. Planificación del desarrollo**

- Fase 1: Configuración del entorno y estructura básica del proyecto (3 días).

- Configurar el repositorio y las herramientas de desarrollo.

- Fase 2: Desarrollo del backend (3 días).

- Configurar las rutas para la autenticación y manejo de usuarios.

- Crear endpoints para la búsqueda de canciones y gestión de listas de reproducción.

- Fase 3: Desarrollo del frontend (1 semana).

- Crear la interfaz de usuario: página de inicio, búsqueda de canciones, perfil de usuario.

- Implementar el reproductor de música.

- Crear la funcionalidad de listas de reproducción.

- Fase 4: Integración y pruebas (2 días).

- Pruebas unitarias para asegurarse de que las funcionalidades están bien implementadas.

- Pruebas de integración para verificar la comunicación entre el frontend y backend.

- Fase 5: Despliegue (2 días).

- Despliegue de la aplicación en un servidor.

**6. Control de versiones y gestión de código**

- Ramas:

- main: versión estable.

- dev: desarrollo activo.

- feature/auth: desarrollo de la autenticación.

- feature/player: implementación del reproductor de música.

- feature/playlist: gestión de listas de reproducción.

**7. Pruebas**

- Unitarias: Probar la autenticación, la creación de listas de reproducción y la reproducción de canciones.

- De integración: Asegurarse de que el frontend (React.js) está correctamente conectado al backend y la base de datos.

**8. Documentación**

- Investigación sobre los géneros, cantantes y canciones .

**9. Revisión y ajustes**

- Revisión semanal: Realizar revisiones del progreso con el equipo, ajustar cronogramas si es necesario.