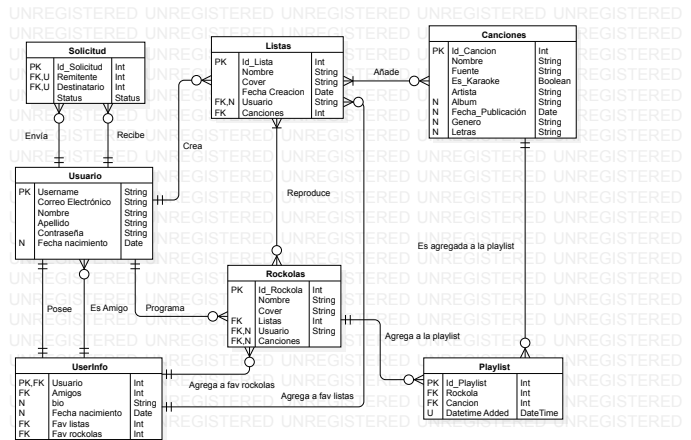


Reporte de Aplicación MusicBoss

Miguel Peña, Ernesto Bastidas

Abstract—Creating a social network about music, where you can share your favorite music with other users and listening to playlist of songs simultaneously is the idea that gave birth to this app: MusicBoss. In this document you will understand how it was created, what tools were used to make it and some things that could be done to improve it.

Abstract—Crear una red social que trate de música, donde puedas compartir tu música favorita con otros usuarios y escuchar listas de reproducción simultáneamente es la idea con la que concebimos nuestra app: MusicBoss. En este documento vas a entender cómo fue creada, qué herramientas fueron usadas para hacerla y algunas cosas que se pueden hacer para mejorarla.



I. INTRODUCCIÓN

Esta aplicación se creó con la intención de que las personas tengan la oportunidad de compartir su música favorita en vivo con otras personas de cualquier parte del mundo, ya que de esa forma pueden conocer gente con sus mismos gustos y escuchar de forma simultánea las mismas canciones. Todo esto lo permite la aplicación ya que tiene las funcionalidades de crear cuenta, crear lista de canciones, crear rockolas con lista de reproducción (las cuales se pueden compartir con cualquier persona) y agregar amigos.

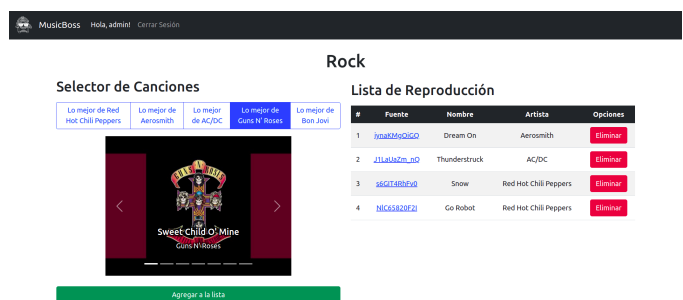
También como una aproximación a lo que sería la versión web de la aplicación, se programó una pantalla de bienvenida básica en la dirección web de la aplicación, para esto usando el gestor de plantillas Jinja 3 también presente en el Framework de Django.

En el momento actual, sólo tiene incorporadas las funciones de agregar canciones a las listas de reproducción de una Rockola, y también la opción de Iniciar Sesión y Registrarse a través de esta interfaz. En un futuro se piensa agregar la posibilidad de escuchar las Rockolas y poder administrar las listas y Rockolas del usuario a través de esta interfaz.

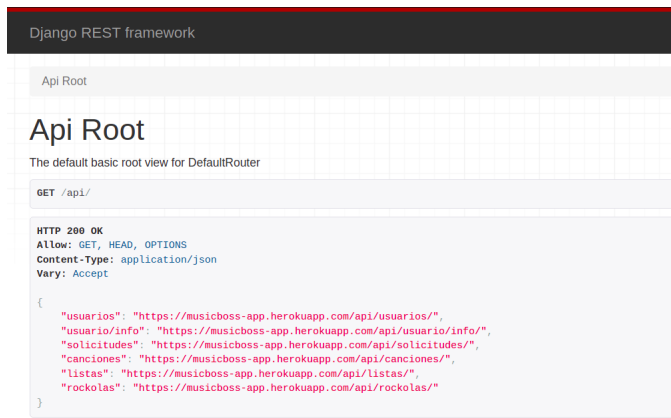
II. BACKEND

El backend consiste en un servidor que ha sido montado usando el Framework Django, escrito en Python, el cual se escogió debido a su flexibilidad y facilidad de programación, y además a su facilidad de despliegue en la nube, más concretamente usando la plataforma de Heroku para hacer esta publicación.

La base de datos usa el ORM propio de Django, por lo que se facilita mucho la conversión entre diferentes formatos de bases de datos durante el desarrollo de la aplicación. Prueba de esto es que, para el desarrollo local, se usó una base de datos SQLite, pero para la versión desplegada en Heroku, fue muy directa la configuración con una de las instancias gratuitas de PostgreSQL manejadas por esta plataforma para el despliegue de aplicaciones. La siguiente figura muestra el diagrama Entidad Relación que se usa para el manejo de los datos de la aplicación.



Para el manejo del API que va a ser consumida por la aplicación móvil, se usó el módulo Django Rest Framework, por medio del cual se publica el punto de entrada a través de peticiones HTTP al servidor. Entre las funciones actuales del API, están el manejo y autenticación de los usuarios, la búsqueda, creación, edición y eliminación de listas, canciones, rockolas, solicitudes de amistad y canciones de una determinada playlist.



III. FRONTEND

El frontend fue hecho con react native, el cual es un framework de javascript para crear aplicaciones reales nativas para iOS y Android, basado en la librería de JavaScript React para la creación de componentes visuales, cambiando el propósito de los mismos para, en lugar de ser ejecutados en navegador, correr directamente sobre las plataformas móviles nativas, en este caso iOS y Andorid.[1]

La aplicación contiene 13 vistas, y cada una realiza una funcionalidad de la aplicación. Esta es la lista de las vistas:

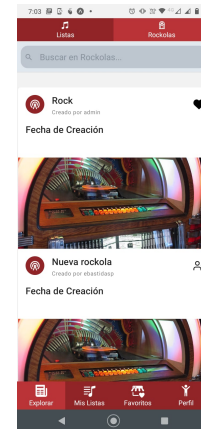
- Login: Esta vista tiene campos para poner el usuario y la contraseña, así como botones para iniciar sesión (login), ir a la vista de registro o ingresar a la aplicación sin hacer uso de una cuenta. (Aunque con restricciones)



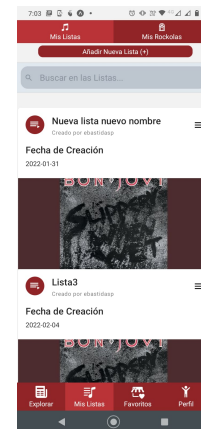
- Registro: Esta vista tiene campos para poner el nombre de usuario, un correo, contraseña y la confirmación de la contraseña, así como botones para registrarse o ir a la vista de login.
- Explorar-listas: Esta vista tiene una barra de búsqueda para filtrar las listas que se ven ahí, así como la lista de todas las listas de canciones que hay en la aplicación. En cada lista hay uno de 3 posibles íconos, un corazón vacío indicando que esa lista no esta en nuestros favoritos, un corazón relleno indicando que esa lista está en nuestros favoritos y un ícono de un usuario con un chulo indicando que esa lista fue creada por uno. Cabe aclarar que si uno ingresa sin cuenta, solo se verá el corazón

vacío.

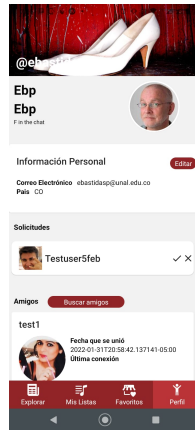
- Explorar-rockolas: Esta vista es prácticamente igual que la de listas, solo que muestra la lista de rockolas en vez de la lista de listas de canciones. Una rockola se diferencia de una lista en que con ella se pueden reproducir canciones, y en ella se puede agregar más de una lista.



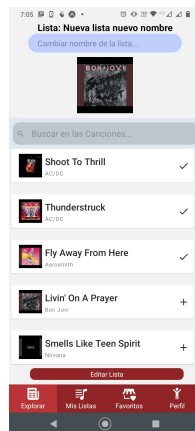
- Mis listas: Contiene una lista de las listas de canciones que fueron creadas por uno. En cada lista hay un menú de opciones que contiene los botones de editar (para ir a la vista de editar lista) y borrar (para eliminar esa lista de la aplicación).



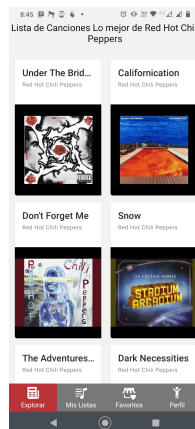
- Mis rockolas: Contiene una lista de las rockolas de canciones que fueron creadas por uno, y en cada rockola está el mismo menú de opciones que en la vista de mis listas.
- Favoritos-listas: Contiene una lista de las listas favoritas de uno.
- Favoritos-rockolas: Contiene una lista de las rockolas favoritas de uno.
- Perfil: Contiene la información de la cuenta de uno, como el nombre, apellidos, biografía, una lista de solicitudes sin responder y una lista de usuarios que se han agregado a amigos.



- Editar lista: Contiene una lista de canciones indicando cuales ya están agregadas a la lista y cuales no. También hay un buscador de las canciones y un espacio que muestra el cover actual de la lista.

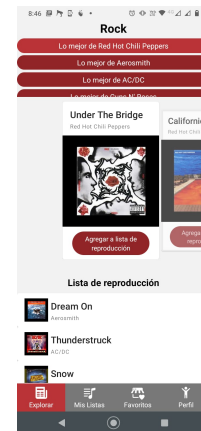


- Buscar amigos: Contiene un buscador para filtrar los usuarios que todavía se pueden agregar a la lista de amigos de uno, así como una lista de esos usuarios.
- Canciones de lista: Muestra las canciones de la lista que se escogió anteriormente.



- Rockola: Contiene una lista de botones que indican todas las listas de canciones que hay en la rockola. Cuando se selecciona uno de esos botones, se muestra en un carrusel la lista de canciones de la lista seleccionada,

con la opción de agregarlas a la lista de reproducción, y al final se encuentra la lista de reproducción, la cual contiene las listas de canciones que se van a reproducir y en cada canción hay un botón con la opción de eliminar la canción seleccionada de la lista de reproducción.



IV. POSIBLES MEJORAS A FUTURO

- Permitir actualizar los favoritos sin que se actualicen las vistas de explorar o favoritos. También poder ver los perfiles de otros usuarios, que la lista de reproducción se reproduzca en vivo para todos los que están escuchando una rockola, añadir la opción de chatear con otras personas o crear rockolas privadas, etc...
- Además, se puede en vez de usar un reproductor interno para cada dispositivo, crear un streaming el cual haga la difusión de la misma música a diferentes dispositivos, para esto podría pensarse en posibles servicios como los que ofrece Firebase
- En el caso del Backend, se podría agregar otras entidades como "Invitación" para que un usuario pueda invitar a una o muchas personas a escuchar su rockola. También se podría con el tiempo ir cambiando los permisos de seguridad de pendiendo de la petición que se haga

V. CONCLUSIONES

- Se puede crear una aplicación que funcione como una red social de música que le permita a las personas compartir sus gustos y escuchar música al mismo tiempo.
- El uso de React Native para la aplicación móvil no s permite fácilmente migrar entre una aplicación Android y una para iOS
- El uso de Django para el Backend dió gran flexibilidad y posibilidad de cambios rápidos a la lógica del negocio

VI. REFERENCIAS

- [1]J. Blanes, "¿Qué es React Native?". Available: <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-react-native.html> [Accessed 6 Feb 2022].