

Universidad San Carlos de Guatemala  
Facultad de ingeniería.  
Ingeniería en ciencias y sistemas  
Seminario de sistemas 1

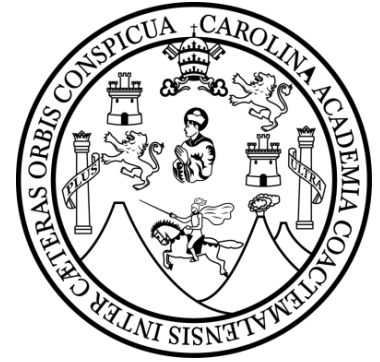
Sección A

Ing. Manuel Fernando López Fernández  
Aux. Elmer Gustavo Sánchez García

Sección B

Ing. Ludwin Federico Altan Sac  
Aux. Ronald Alexandro Castro Pérez

Segundo Semestre 2023



# Proyecto 1

## SoundStream

### Objetivos

#### General

- Implementar los servicios de seguridad y desarrollo de AWS

#### ● Específicos

- Crear un entorno de desarrollo para una aplicación web
- Implementar una arquitectura personalizada
- Integrar servicios y APIs de AWS

### Descripción

Amazon Web Services (AWS) es una plataforma de servicios de nube que ofrece potencia de cómputo, almacenamiento de bases de datos, entrega de contenido y otra funcionalidad para ayudar a las empresas a escalar y crecer.

Se desarrollará una aplicación web inspirada en los servicios de multimedia, en donde se tendrán dos diferentes tipos de usuarios, usuario administrador que podrá subir canciones y agregarlas a álbumes, y un usuario suscriptor que puede reproducir canciones, crear playlist y agregar canciones, realizar búsquedas, y estadísticas de su cuenta.

Los servicios de AWS que se utilizaran para este proyecto son:

- IAM
- EC2
- Load Balancer

- S3
- RDS

## Sitio web

SoundStream es una plataforma de streaming de música cuya intención es promover el arte en el territorio guatemalteco, prometiendo una experiencia robusta, confiable y amigable para sus clientes. Es una plataforma completamente en la nube, diseñada para ser utilizada en cualquier navegador Web. El sistema permite a los clientes escuchar sus canciones favoritas, crear y reproducir playlists personalizadas y escuchar la radio.

A continuación, se describen las secciones que contendrá el sitio web:

### Login

Para la autenticación se realizará por medio de las credenciales: correo electrónico y contraseña.

- Se debe tener un usuario administrador por defecto.

### Registro

Para registrar un nuevo usuario obligatoriamente se pedirán los datos:

- Nombres
- Apellidos
- Foto del usuario
- Correo electrónico

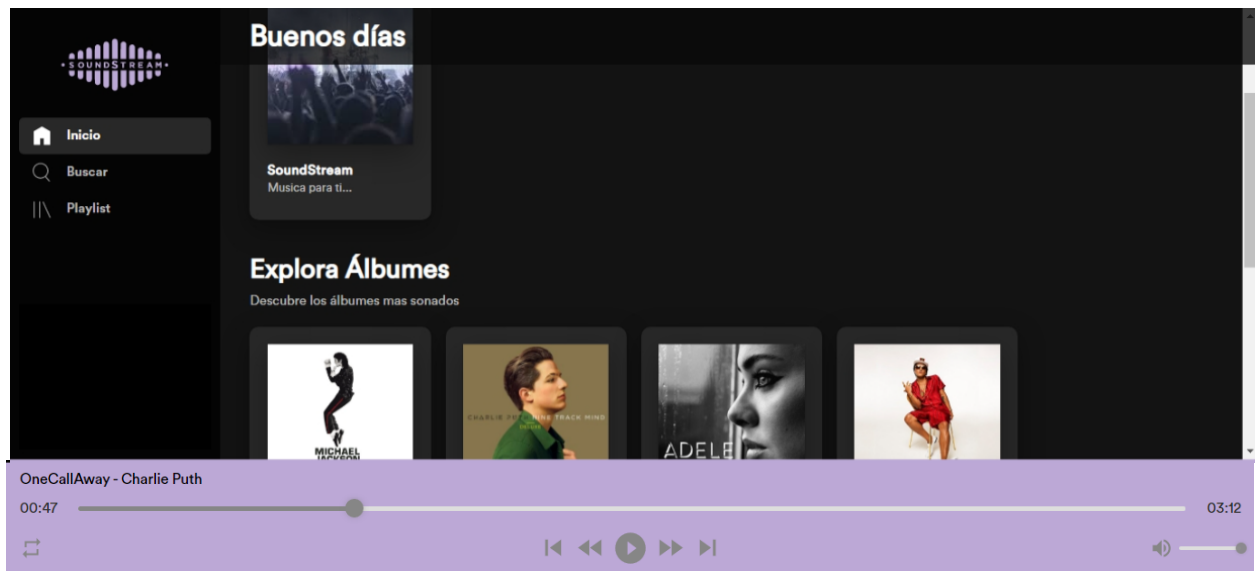
- Contraseña
- Confirmar contraseña
- Fecha nacimiento

La foto del usuario será obligatoria, esta se realizará eligiendo una imagen del ordenador..

## ***Página principal (Inicio)***

Esta es la página principal de la aplicación web, es la primera en aparecer luego de iniciar sesión. En ella se encuentra:

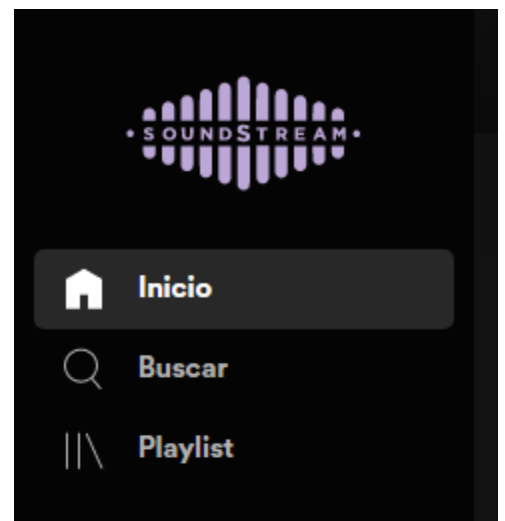
- Barra de navegación.
- Listado de canciones, álbumes, y artistas disponibles.
- Reproductor de música.
  - Será visible en todo momento.



## **Navegación**

Esta debe contar con una sección que servirá para acceder a los diferentes módulos que tiene la aplicación.

1. **Inicio:** Dirige al usuario a la página principal.
2. **Buscar:** Permite realizar una búsqueda por nombre de canción, álbum o artista.
3. **Perfil:** Muestra la información personal del usuario.
4. **Playlist:** El usuario puede gestionar sus playlist.
5. **Favoritos:** Lista las canciones que el usuario le dio 'Me gusta'.
6. **Histórico:** Este es un módulo de reportes.
7. **Radio:** Reproduce de forma aleatoria todas las canciones.



8. **Administrador:** Opción visible solo para usuario con rol administrador, la funcionalidad se detalla más adelante.

## **Perfil**

### **Información del usuario**

En esta sección, el usuario puede ver y modificar sus datos, tales como:

- Nombre
- Apellido
- Foto de perfil
- Correo electrónico
- Se debe ingresar la contraseña correcta para que se guarden los datos modificados.

## **Buscar.**

El usuario puede realizar la búsqueda de álbumes, canciones o artistas por medio de la entrada del usuario.

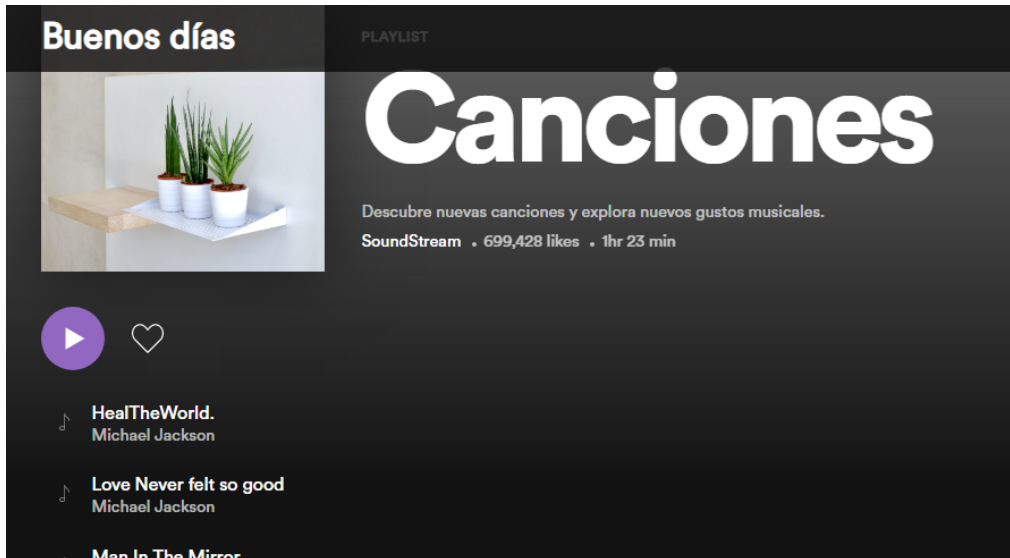
- Se debe mostrar el listado de coincidencias, y poder seleccionarlas.
  - **Canciones**
    - Se mostrará el detalle de canciones
      - Nombre
      - Imagen
      - Duración.
      - Artista
    - Dar “Me gusta” (favoritos)
  - **Álbum**
    - Mostrará la información del álbum y el listado de canciones.
  - **Artista.**
    - Mostrará el listado de canciones que le pertenecen.

## **Playlist**

En esta pantalla el suscriptor puede hacer uso de playlist para un mejor control de sus canciones.

- Detalle playlist
  - Lista de canciones
- Creación y edición de playlist
  - Nombre (Obligatorio)
  - Descripción
  - Fondo de portada.(Obligatorio)

- Agregar canción.
- Eliminar canción.
- Eliminar playlist
  - Se eliminan todas las canciones que contiene.
- Reproducción de canciones



## ***Favoritos***

En esta pantalla se visualizará el listado de las canciones a las que el usuario le dio 'Me gusta'.

## ***Histórico***

En esta sección aparecerán estadísticas de la cuenta del usuario, la forma de visualizarlo queda a discreción del estudiante.

- Top 5 canciones más reproducidas.
- Top 3 artistas más escuchados.
- Top 5 álbumes más reproducidos.
- Historial de canciones reproducidas

## **Administrador**

Esta pantalla de usuario con rol administrador debe mostrar el acceso a las funcionalidades CRUD de las siguientes entidades.

- Artista
- Álbum
- Canción

### **Artista.**

En esta pantalla se realizan las siguientes acciones.

- Creación
  - Nombre (Obligatorio)
  - Fotografía (Obligatorio)
  - Fecha de nacimiento
- Ver detalle.
- Actualizar
- Eliminar
  - Para confirmar la eliminación de un artista se necesita confirmar la contraseña del usuario administrador.
  - Se debe eliminar todo lo relacionado al artista.

### **Álbum**

En esta pantalla se realizan las siguientes acciones.

- Creación
  - Nombre (Obligatorio)
  - Descripción
  - Imagen de portada. (Obligatorio)
  - Artista (Obligatorio)
- Ver detalle.
  - Se listan las canciones agregadas.
- Agregar canción
  - Se debe mostrar un listado de canciones que pertenezcan al artista y que no estén agregadas a otro álbum.
  - Cada canción puede ser agregada y eliminada del álbum.
- Actualizar
- Eliminar
  - Para confirmar la eliminación de un álbum se necesita confirmar la contraseña del usuario administrador.

## Canción.

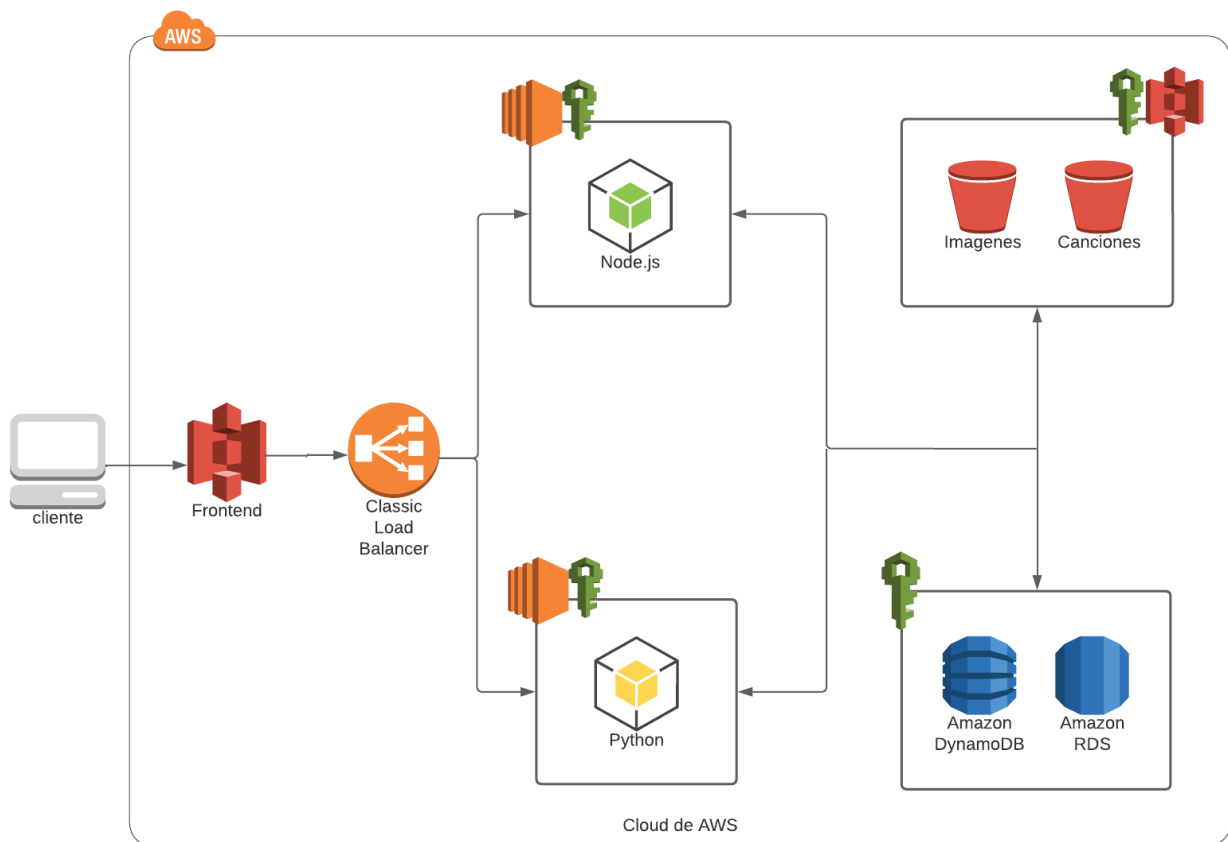
En esta pantalla se realizan las siguientes acciones.

- Creación con los siguientes campos obligatorios
  - Nombre
  - Fotografía
  - Duración.
  - Artista
  - Archivo mp3
- Se solicitará un archivo .mp3 a subir desde el ordenador.
- Ver detalle.
- Actualizar
- Eliminar

NOTA: El usuario administrador tendrá acceso a todas las funcionalidades del usuario suscriptor.

Las canciones pueden ser reproducidas en cualquier lugar de la aplicación donde sean visibles.

## Arquitectura e Implementación



La arquitectura anterior demuestra como funcionarían todos los servicios de AWS internamente en nuestra aplicación.

- **S3:**

- La aplicación está alojada dentro de un bucket de S3 funcionando como un sitio web estático. El nombre de este debe ser **soundStreamSemi1\_g#**
- En otro bucket almacenará todos los archivos mp3 y todas las imágenes tanto de perfil, artista, álbum y de la playlist que se suban dentro de la aplicación, configurado con las políticas públicas para poder ser accedido desde nuestra aplicación. El nombre de este bucket será **multimediaSemi1\_g#** dentro contendrá dos carpetas.
  - **Fotos**
  - **Canciones**

- **EC2:**

- Se tendrán 2 instancias de EC2 donde se montarán 2 servidores con exactamente las mismas rutas de nuestra aplicación ya que esto nos permitirá enlazar nuestro balanceador de carga.
- El lenguaje de los servidores debe de ser en **NODEJS y PYTHON.**
- Asegurarse de habilitar **únicamente** los puertos necesarios para su aplicación en los **Security Groups** de cada instancia.
- El sistema operativo de las instancias queda a discreción del estudiante.

- **Load Balancer:**

- Se configurará 1 balanceador de carga donde estará verificando el estado de los 2 servidores de las EC2.
- Este es el que se tiene que consumir desde la aplicación, ya que como se sabe este redirecciona la solicitud a alguno de los servidores disponibles. Se verificará que es el que se consume desde la aplicación y no los servidores como tal.

- **RDS o DynamoDB:**

- Se utilizará una instancia de RDS o DynamoDB, quedando a discreción del estudiante para la base de datos.
- Las contraseñas de los usuarios deben de estar encriptadas con cualquier método o librería que el estudiante desee.
- Para guardar los archivos e imágenes de un usuario solamente deben de guardar la URL de este donde está almacenado.
- El motor de la base de datos queda a criterio del estudiante.



- **IAM**
  - Se debe de crear los usuarios de IAM que sean necesarios para el manejo y uso de los servicios de aws que lo requieran con su política asociada.

## Manual Técnico

Se necesita que se realice un manual técnico que incluya el siguiente contenido:

- Objetivos del Manual
- Explicación de Arquitectura del proyecto.
- Diagrama Entidad Relación
- Descripción de cada usuario de IAM creado con las políticas asociadas.
- Capturas y descripción de cómo se configuró cada servicio.
- Conclusiones

## Manual de Usuario

Se necesita que se realice un manual de usuario que incluya el siguiente contenido:

- Objetivos del Manual
- Explicación y descripción de la aplicación.
- Pasos con capturas de cómo utilizar la aplicación.

## CONSIDERACIONES:

- Nombre de Repositorio: **Semi1-Grupo#-Proyecto1**
- Repositorio en Github en modo privado y documentado con el formato Markdown (Manual de configuraciones y de Usuario).
- Agregar como colaborador en el repositorio al usuario dependiendo de su Sección de Laboratorio.
  - Sección A - gstavojr
  - Sección B - Ronald-CP
- El proyecto debe ser en Grupos
- Usar los respectivos usuarios de IAM con sus respectivas políticas de acuerdo con el servicio que se está utilizando.
- Cualquier copia total o parcial será reportada a la escuela y se obtendrá una nota de 0.
- Fecha de entrega: 17/09/2023 a las 23:59 por UEDI