**Project Report**

**Spotify Clone – Web Music Player**

**👤 Student Details**

* **Name:** Rajan
* **Project Title:** Spotify Clone (Web Music Player)
* **Submission Date:** April 2025
* **Technology Stack:** HTML, CSS, JavaScript
* **Code Editor:** Visual Studio Code
* **Hosted On:** Localhost (Live Server)
* **Submission Type:** Web App

## 📌 ****Project Overview****

This project is a front-end Spotify Clone that simulates a fully functional online music player using core web technologies: HTML, CSS, and JavaScript. It allows users to explore music albums, play songs, pause, download tracks, like/save favorites, and search both within and across playlists. The project is built entirely from scratch, without any libraries or frameworks, showcasing a real-world example of media playback, UI interaction, and localStorage integration.

**🎯 Objectives**

* Design a clean, responsive, and user-friendly music interface
* Implement folder-based dynamic music album rendering
* Enable song control through Play, Pause, Next, Previous buttons
* Add per-song Like/Save and Download features
* Implement local search for songs
* Add advanced UX elements like toast notifications and icon sync
* Sync main player controls with individual song controls

## 🧰 ****Tools & Technologies Used****

## 

## 🗂️ ****Folder Structure****

## 

**✅ Core Features Implemented**

**🎵 Music Playback**

* Per-song play/pause
* Main global control (Play, Pause, Next, Previous)
* Seekbar and volume slider
* Real-time time updates and progress bar animation

**❤️ Like/Save Songs**

* Heart icon toggles liked status
* Liked songs saved in localStorage
* Heart icon turns sky-blue for liked tracks

**⬇️ Download Songs**

* Every song has a direct download icon
* Clicking it downloads the .mp3 file without extra steps

### 🔍 Search System

* **Local Search**: Filters only current playlist
* Results update live as you type

**📁 Dynamic Album Rendering**

* Cards auto-generated from folder structure
* Album info (cover image, title, description) from info.json

**🔄 Icon Syncing**

* Main play/pause icon stays in sync with individual songs
* Small play icon toggles when a song starts or pauses

**🔔 Toast Notifications**

* Shows: "Now Playing: [Song Name]" in the corner
* Smooth appearance and auto-hide after 3 seconds

## 🎨 ****UI & UX Design****

* Left sidebar with collapsible menu (hamburger)
* Responsive layout for desktop/tablet
* Song listing and playback bar clearly structured
* Icons for all actions (play, download, like, volume)
* Minimalist green and black theme (Spotify-style)

## 📈 ****Challenges Faced****

* Syncing main and small play buttons correctly
* Preventing unintended playback when clicking like/download
* Ensuring localStorage doesn’t overwrite data across folders
* Rendering all folders dynamically with fetch() and innerHTML
* Keeping UX smooth during playback and seeking
* Maintaining Cards folders to contain their individual songs

**Working Details**

**Playlist Loading:**

Songs are loaded from multiple folders using JavaScript's fetch() and then rendered dynamically into the DOM.

**Playback Control:**

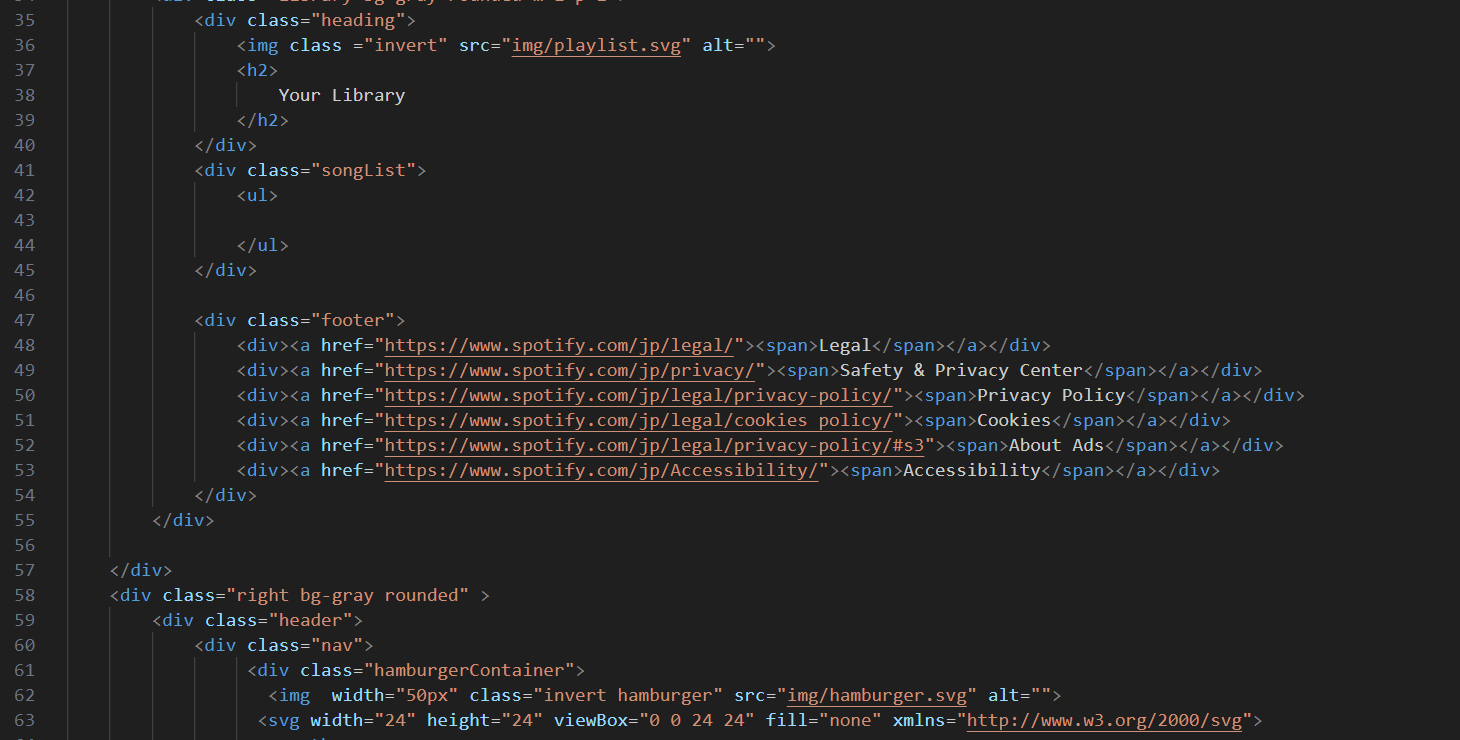
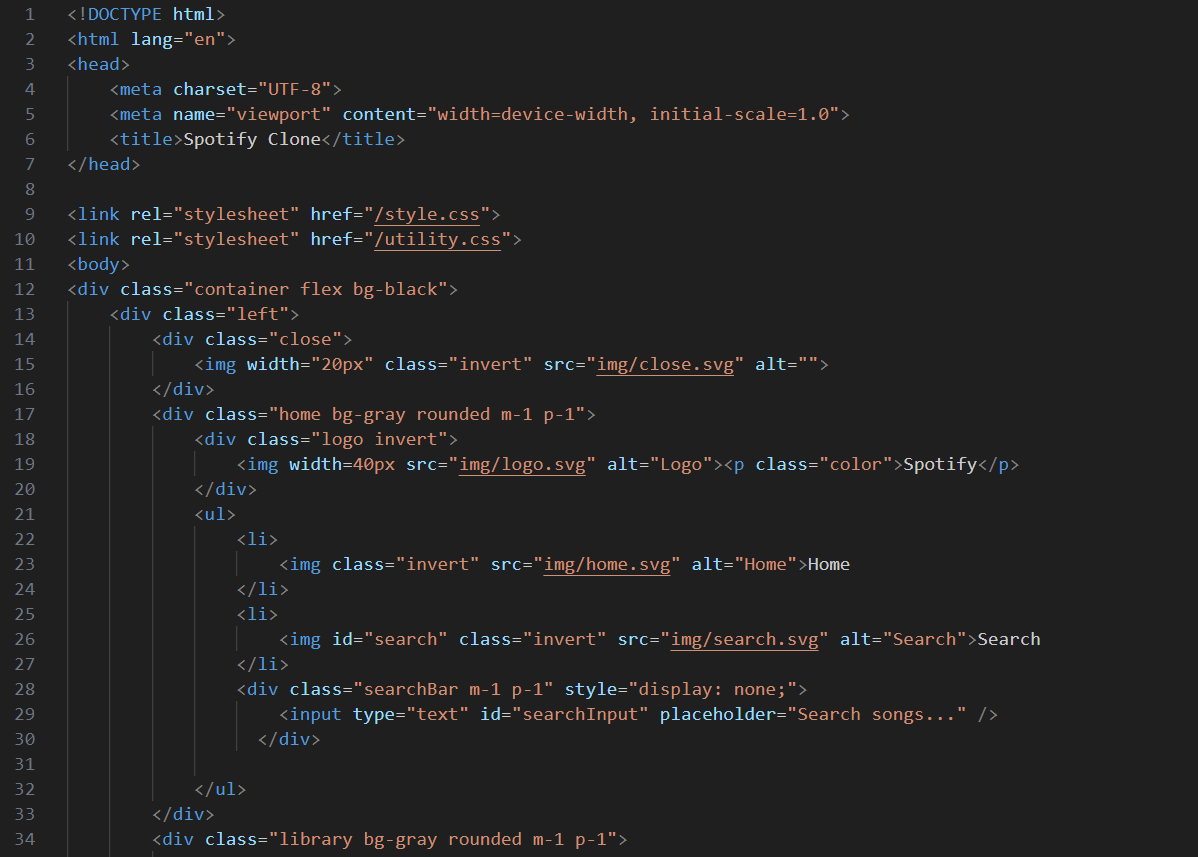
The Audio() API is used to manage song playback, including play/pause, seek, and volume control.

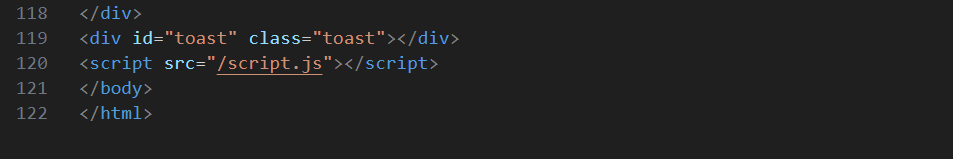
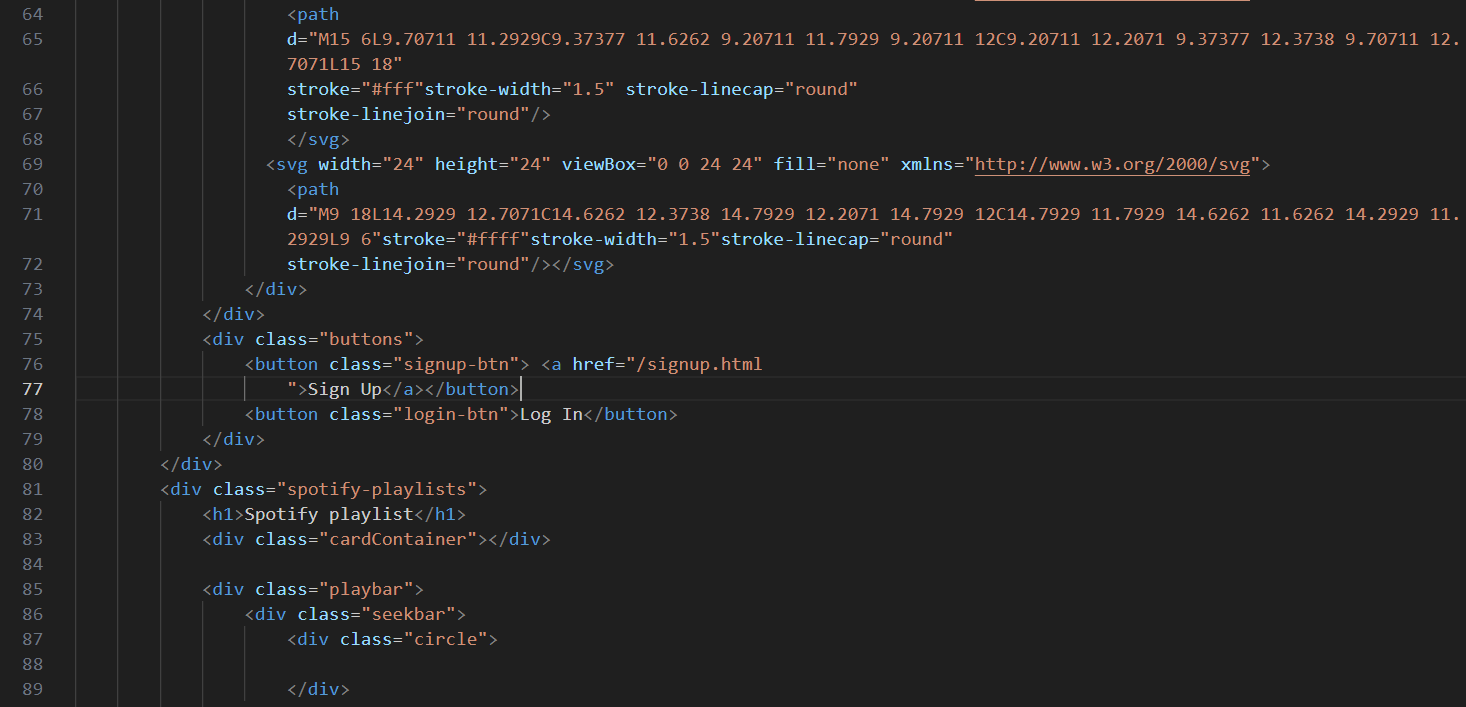
**Responsive Design:**

CSS media queries ensure that the layout adjusts for desktop and mobile views. The sidebar collapses into a hamburger menu on smaller screens.

**Screenshots**

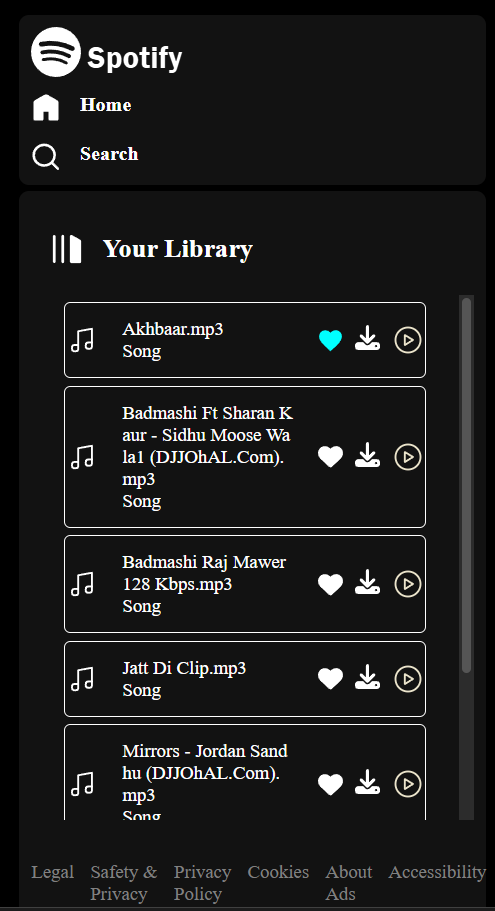
**Code Screenshot**

****

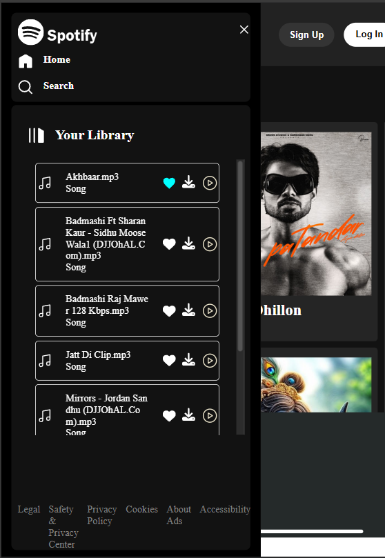
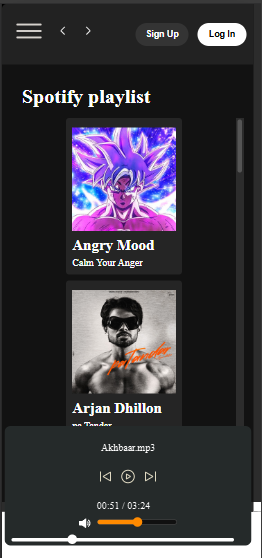
****

**Spotify Screenshot**

**Homepage with playlist**

****

**Sidebar open and close**

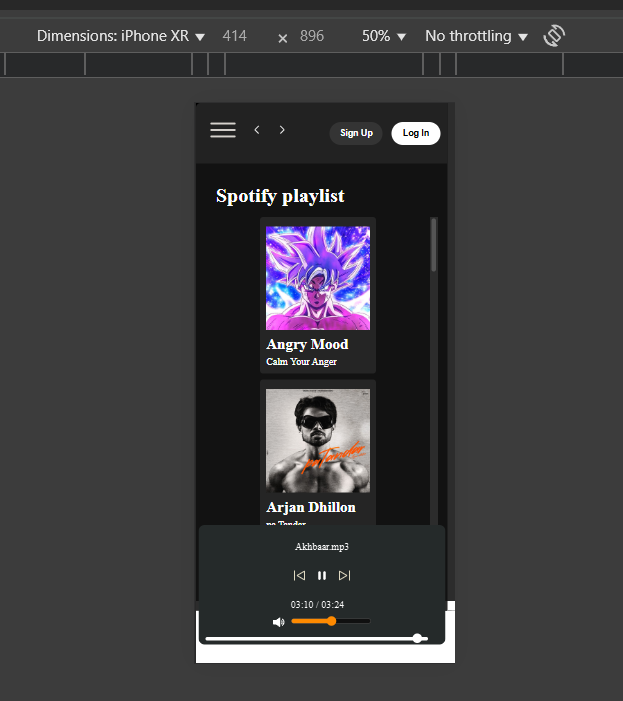
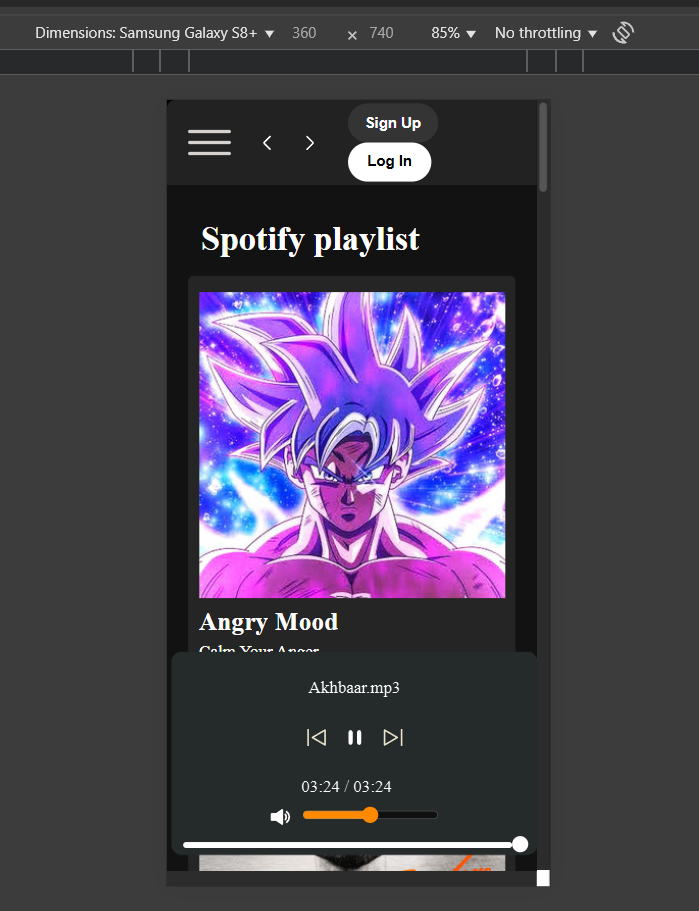
****

**Song Playing with active play bar**

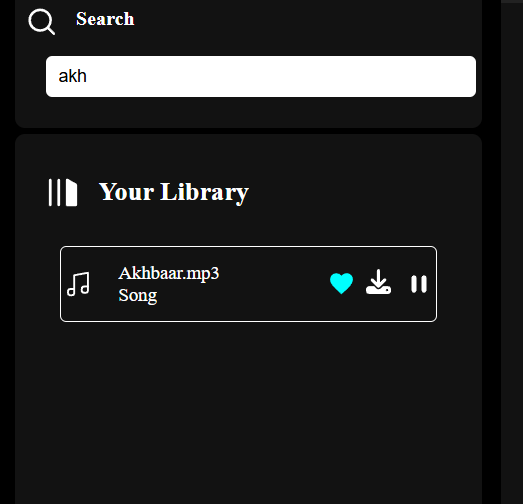
****

**Mobile view responsive design**

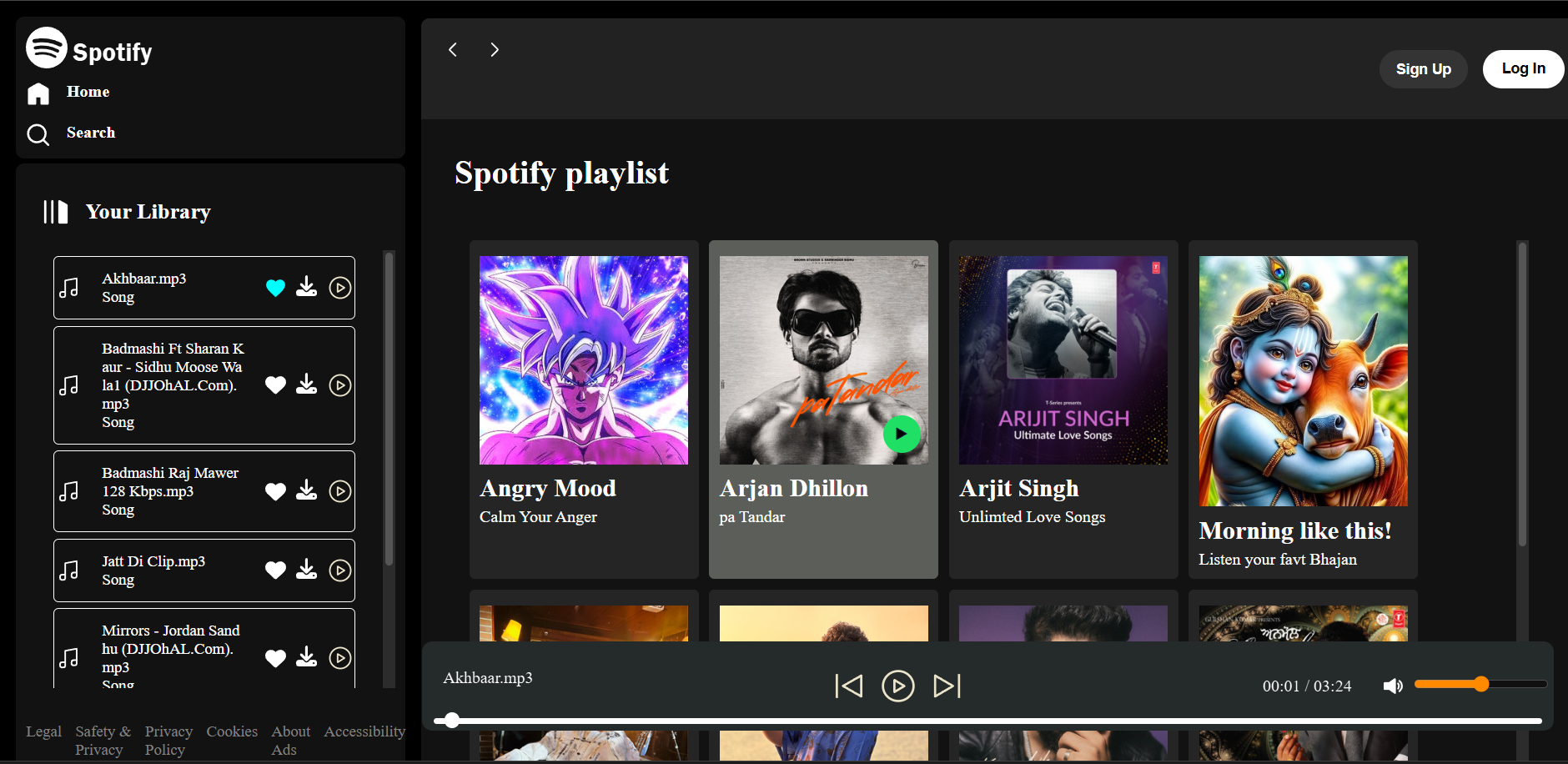
**Iphone XR dimension & Samsung Galaxy S8+**

****

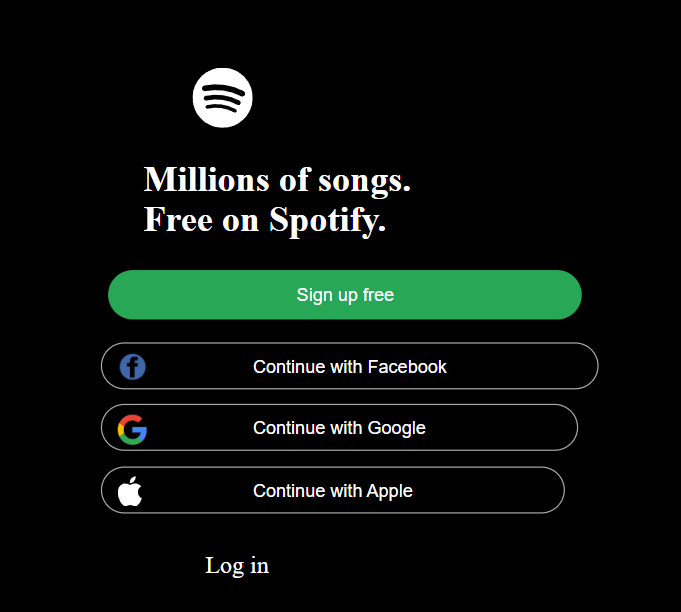
**Search bar icon**

****

**Full screenshot of clone**

****

**Signup Page Screenshot**

****

## 🔥 ****Why This Clone is Better Than the Original Spotify****

1. **100% Control Over Code**  
   Every feature is hand-coded and fully customizable.
2. **No Ads, No Premium**  
   All songs are playable and downloadable — forever free.
3. **Offline-Ready & Lightweight**  
   No login, no server dependencies — runs locally without internet.
4. **Educational Value**  
   Teaches full-stack interaction with real media playback logic.
5. **Expandable for Anything**  
   Turn it into a podcast app, a DJ tool, a study music player — your app, your rules.
6. **Developer Credit**  
   It’s built by me, Rajan — not a corporation. I’m the developer and the user.

## 💡 ****Possible Future Improvements****

* ✅ Liked Songs playlist tab
* ✅ Shuffle and repeat functionality
* ✅ Theme toggle (Dark/Light mode)
* ✅ Allow user authentication and custom playlists
* ✅ Drag-and-drop playlist queue
* ✅ Lyrics or subtitle sync
* ✅ Hosting project online with GitHub Pages or Netlify
* ✅ Add backend integration (Node.js or Firebase)
* ✅ Add lyrics or visual equalizer animations

**Conclusion**

This project provided hands-on experience with web technologies and helped simulate a real-world application like Spotify. It strengthened my understanding of front-end development and user experience design, especially with dynamic content and responsive layouts.