

## **Use Case Document: Web-Based Music Player Tool**

**Domain:** Media & Entertainment

**Use Case Name:** Web-Based Music Player Tool

**Use Case Description:** This use case outlines the functionality of a responsive web-based music player application. The system enables users to log in, browse and stream songs, manage their playlists and favorites, and interact with intuitive controls such as play, pause, and skip. It is designed to operate efficiently across various devices and incorporates best practices for security, code quality, and performance.

**Use Case Goal:** To provide a secure, user-friendly, and responsive platform that allows users to stream, organize, and enjoy music seamlessly.

**Actor:**

- User (Listener)

**Pre-Conditions:**

- The user is registered or has the ability to register an account.
- The user has a stable internet connection.
- The application is accessed via a modern browser on desktop or mobile.

**Post-Conditions:**

- The user is able to log in and perform actions such as streaming music, creating playlists, marking songs as favorites, and navigating tracks.
- All changes (e.g., playlists, favorites) are saved and persist across sessions.

**Main Flow (Basic Path):**

### **1. Initiate Application**

- The user opens the web application on a browser.
- The application automatically adjusts UI elements to match the device (responsive layout).

### **2. User Authentication (Login)**

- The user inputs valid credentials.
- The system verifies the input.
- On success, the user is redirected to the dashboard; otherwise, an error message is displayed.

### 3. Browse Song List

- The application fetches songs via an external or internal API.
- Songs are displayed with metadata (title, artist, album cover, duration).

### 4. Play a Song

- The user selects a song from the displayed list.
- The player fetches the stream and begins playback.
- Playback controls are displayed.

### 5. Pause Song

- The user clicks the pause button.
- The playback is paused and can be resumed with the play button.

### 6. Navigate Tracks

- The user clicks 'Previous' to go to the last played song.
- The system fetches and plays the previous track.

### 7. Create Playlist

- The user clicks on 'Create Playlist'.
- They input a playlist name and add songs.
- The playlist is saved to the user's profile.

### 8. Mark as Favorite

- The user marks a song as favorite using a heart icon.
- The song is added to the 'Favorites' section.

### 9. Logout

- The user logs out securely.
- The session is terminated, and the token is invalidated.

### Alternative Flows:

- **Invalid Login:** Displays an error message; prompts retry.
- **Empty Song List:** Displays a placeholder message.
- **Song Not Found:** Shows an error and suggests similar songs.

- **Playlist Exists:** Shows a warning about duplicate playlist names.
- **Unauthorized Access:** Redirects unauthenticated users to the login page.

#### Exception Handling:

- **Network Issues:** Displays appropriate messages; attempts automatic retry.
- **API Failure:** Fallback to cached content or maintenance page.
- **Playback Error:** Skips to the next available track.
- **Security Threat:** Suspicious activities are logged and sessions invalidated.
- **Possible Test Cases:**

Test Case ID	Description	Input Data	Expected Outcome
TC01	Valid Login	Valid credentials	Dashboard is shown
TC02	Invalid Login	Incorrect password	Error message displayed
TC03	Play Song	Select a song	Song plays
TC04	Pause Song	Click pause	Song pauses
TC05	Previous Song	Click previous	Last song plays
TC06	Create Playlist	Playlist name and songs	Playlist created
TC07	Add Favorite	Click heart icon	Song added to favorites
TC08	Responsive Test	Resize browser	Layout adjusts appropriately
TC09	API Down	API offline	Error page shown
TC10	Unauthorized Access	Unauthenticated visit	Redirect to login
TC11	Duplicate Playlist Name	Existing playlist name	Warning message displayed
TC12	Empty Playlist Creation	No songs selected	Playlist not created; prompt to add songs
TC13	Song Load Failure	Corrupted audio source	Error message shown; fallback suggested
TC14	Mark Already Favorite Song	Click heart again	Song removed from favorites
TC15	Add Playlist with Special Characters	Playlist name with symbols	System handles or restricts appropriately
TC16	Session Timeout	Inactivity timeout reached	Auto logout and redirect to login
TC17	Browser Compatibility	Open in different browsers	Works consistently across supported browsers
TC18	Search Song	Enter song name in search bar	Matching results displayed
TC19	Volume Control	Adjust volume slider	Audio volume changes accordingly

TC20	Mobile Functionality Check	Open on mobile device	Mobile-friendly layout and touch support works
------	-------------------------------	-----------------------	--

**Expected Outcome:**

- Users experience seamless music playback.
- The application behaves consistently across devices.
- User interactions (playlists/favorites) are persistent.
- The system handles errors gracefully.
- Security and performance best practices are maintained.