

# Phase 4: Functional and Performance testing

Date	13 April 2025
Team ID	SWTID1744391109
Project Title	Music Streaming App
Maximum Marks	10

## 6.1 User Acceptance Testing (UAT)

### Project Overview

- **Project Name:** MERN Music Streaming Platform
- **Project Description:**  
A robust full-stack MERN (MongoDB, Express.js, React.js, Node.js) web application designed for music streaming. It offers database based authentication and MongoDB Atlas integration. The platform allows users to sign up/log in, play/pause/navigate songs, like or dislike tracks, and manage their personal playlists. Admins can upload songs and manage content through a dedicated dashboard. Users can create, delete, and populate playlists with their favorite tracks.
- **Project Version:** 1.0.0
- **Testing Period:** April 13, 2025 – April 15, 2025

### Testing Scope

#### Features and Functionalities to be Tested:

- User authentication and session management
- Song play, pause, and navigation controls
- Playlist creation and deletion
- Adding/removing songs from playlists
- Like/dislike functionality for individual songs

- Admin dashboard for uploading/managing songs
- User dashboard with playlist and favorite song management
- Data consistency between client, backend, and MongoDB Atlas
- Responsive design compatibility across devices and browsers

**User Stories to be Validated:**

- Users can securely authenticate maintain session state
- Users can play, pause, and navigate through songs seamlessly
- Users can create, delete, and manage multiple playlists
- Users can like and dislike songs
- Admins can upload new songs using the dashboard
- Songs can be added to playlists and played from them
- All UI/UX flows match the intended design and usability expectations

**Testing Environment**

- Deployment URL: <https://localhost:5173.com>
- Tech Stack: React.js (frontend), Express.js & Node.js (backend), MongoDB Atlas (database)
- Credentials (if required):

Email: [user123@gmail.com](mailto:user123@gmail.com)  
Password: user123

**Test Cases**

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Pass/Fail
TC-001	Authentication	1. Navigate to login page 2. Enter valid credentials 3. Submit form	User is redirected to dashboard and session is maintained	Authenti cation failed	Fail

TC-002	Play/Pause Music	1. Click on a song 2. Press Play 3. Pause mid-way	Song plays and pauses as expected	Feature was working	Pass
TC-003	Create Playlist	1. Navigate to playlists2. Click "Create Playlist"3. Name and save	Playlist appears in user dashboard	Feature was working	Pass
TC-004	Add Songs to Playlist	1. Open a song2. Click "Add to Playlist"3. Choose playlist	Song is added to the selected Albumt	Feature was working	Pass
TC-005	Like a Song	1. Play a song2. Click on Like icon	Song is marked as liked	Feature was not working	Failed
TC-006	Admin Song Upload	1. Login as admin2. Upload song from dashboard	Song appears in music list	Feature was working	Passed
TC-007	Playlist Deletion	1. Navigate to user playlist2. Delete a playlist	Playlist is removed from UI and database	Feature was working	Passed

### Bug Tracking

Bug ID	Bug Description	Steps to Reproduce	Severity	Status	Additional Feedback
BG-001	Authentication not working	1. Navigate to login page 2. Enter valid credentials 3. Click Login button	High	Open	Login form submits but no user is redirected

Bug ID	Bug Description	Steps to Reproduce	Severity	Status	Additional Feedback
BG-002	Like button not functioning	<ol style="list-style-type: none"> <li>1. Play a song</li> <li>2. Click the Like button</li> <li>3. Refresh page</li> </ol>	Medium	In Progress	Like state not persisting or updating visually
BG-003	Playlist not refreshing after song addition	<ol style="list-style-type: none"> <li>1. Add a song to playlist</li> <li>2. Go back to dashboard</li> <li>3. Playlist doesn't update</li> </ol>	Medium	Open	Requires manual refresh to reflect changes
BG-004	Music playback glitches during quick switching	<ol style="list-style-type: none"> <li>1. Rapidly click different songs in a row</li> </ol>	Low	Open	Overlapping playback or delayed response

**Notes:**

- Ensure that all test cases cover both positive and negative scenarios.
- Encourage testers to provide detailed feedback, including any suggestions for improvement.
- Bug tracking should include details such as severity, status, and steps to reproduce.
- Obtain sign-off from both the project manager and product owner before proceeding with deployment.