

# Functional Test Plan

**\*This document highlights what steps must be used when testing the application and what steps have already been used to test its functionalities.**

## Test Execution Recording

**Document for each test case:**

- Test ID
- Date/Time executed
- Tester name
- Pass/Fail status (and detailed explanation if it failed)
- Any deviations from steps
- Screenshots of issues
- Notes/Comments

## Pre-Verification Process

Before starting each test case, ensure that a backup of the current state of the database is taken. After performing each modification step, a verification process will be executed to ensure the changes were applied correctly to the database

## 1. User Account Test Cases

### Test Case UA-001: User Registration

**Steps:**

1. Navigate to homepage 2.

Click "Register" button

3. Fill in registration form:

- Username: "testuser123"

- Email: "test@example.com"

- Password: "TestPass123!"
- Confirm Password: "TestPass123!"
- 4. Click Register button
- 5. Verify successful registration message
- 6. Verify redirect to login page
- 7. Verify confirmation email received

#### **Pre-Verification Process**

Before executing, ensure the database contains no existing entries for the test user "testuser123".

#### **Expected Results:**

- **Registration success message displayed:** No longer than **1 seconds**.
- **Redirect to login page:** Should take **2 seconds** in the worst case.
- **Confirmation email received:** Email should be delivered within **5 minutes**.

After running this test, verify that the user has been added to the database and check if all the data (username, email, etc.) is stored correctly.

### **Test Case UA-002: User Login**

#### **Steps:**

1. Navigate to login page
2. Enter credentials:
  - Username: "testuser123"
  - Password: "TestPass123!"
3. Click Login button
4. Verify successful login
5. Verify redirect to homepage
6. Verify user-specific elements visible (sections like: My Profile, My Playlists, Events)

#### **Pre-Verification Process**

Ensure that the user "testuser123" exists in the database with the correct credentials.

**Expected Results:**

- **Successful login message:** No longer than **1.5 seconds**.
- **Redirect to homepage:** Should take no more than **2 seconds**.
- **User-specific sections visible:** Should appear within **1-2 seconds**.

**Test Case UA-003: User Logout****Steps:**

1. Click logout button
2. Verify redirect to login page
3. Verify protected routes inaccessible (anything related to the Admin panel should be inaccessible)
4. Verify session terminated

**Pre-Verification Process**

Ensure that a session is active before logging out.

**Expected Results:**

- **Redirect to login page:** Should occur in **2 seconds**.
- **Protected routes inaccessible:** Validation should take no longer than **2 seconds**.
- **Session termination:** Confirm session expiry within **2-3 seconds**.

**2. Artist Management Test Cases****Test Case AM-001: Artist Creation**

**Precondition:** User is logged in with admin privileges

**Steps:**

1. Navigate to application homepage
2. Click on Admin Panel in the navigation menu
3. Select "Manage Artists" section 4. Click "Create New Artist" button
5. Fill in required fields:
  - Artist Name: "Test Artist"
  - Biography: "This is a test artist biography"
  - Photo URL: "https://example.com/artist.jpg"
6. Click Create button
7. Verify artist appears in the artists list
8. Click on the created artist
9. Verify all details match input data
10. Navigate to homepage
11. Use search bar to find created artist
12. Click on search result
13. Verify artist details page displays correctly

**Pre-Verification Process**

Ensure that no other Artist named "Test Artist" exists in the database.

**Expected Results:**

- **Artist successfully created:** Should take no more than **1 seconds**.
- **Artist visible in admin panel:** Should appear within **2 seconds**.
- **Artist searchable:** Search results should refresh in **2 seconds**.
- **All details correctly displayed:** Page load time should be under **2.5 seconds**.
- **Photo loads properly:** Should load within **2 seconds**.

### **Test Case AM-002: Artist Edit**

#### **Steps:**

1. Navigate to Admin Panel
2. Select "Manage Artists"
3. Find test artist in list
4. Click Edit button
5. Modify biography text
6. Click Save
7. Verify changes appear in list view
8. Verify changes appear in detail view
9. Verify changes appear in search results

#### **Expected Results:**

- **Changes visible in list view:** No longer than **2 seconds**.
- **Changes visible in detail view:** Should be updated within **2 seconds**.
- **Changes appear in search results:** Search should reflect the updated artist within **2 seconds**.

## **3. Event Management Test Cases**

### **Test Case EM-001: Event Creation**

**Precondition:** At least one artist and genre exist in system and user is logged in with admin privileges

#### **Steps:**

1. Navigate to Admin Panel
2. Select "Manage Events"
3. Click "Create New Event"
4. Fill in required fields:
  - Name: "Test Concert Event"

- Description: "This is a test concert event"
- Date: [Current date + 1 month]
- Time: "19:00"
- Location: "Test Venue, City"
- Event Type: Select "Concert" from dropdown
- Artist: Select previously created artist
- Genre: Select available genre

5. Click Create
6. Verify event appears in events list
7. Click on created event
8. Verify all details match input data
9. Search for event by name
10. Verify search results show correct event

#### **Pre-Verification Process**

Ensure that at least one artist and one genre exist in the database before creating the event.

#### **Expected Results:**

- **Event successfully created:** No longer than **1 seconds**.
- **Event visible in events list:** Should appear in **2 seconds**.
- **Event details displayed correctly:** Should match the input data within **2.5 seconds**.
- **Event searchable:** Results should appear in **2 seconds**.

#### **Test Case EM-002: Event Edit**

##### **Steps:**

1. Navigate to Admin Panel
2. Select "Manage Events"
3. Find test event in list
4. Click Edit button

5. Modify event title text
6. Click Save
7. Verify changes appear in list view
8. Verify changes appear in detail view
9. Verify changes appear in search results

**Expected Results:**

- **Changes visible in list view:** Should be visible within **2 seconds**.
- **Changes visible in detail view:** Update should be within **2.5 seconds**.
- **Changes appear in search results:** No longer than **2 seconds**.

## **4. Genre Management Test Cases**

### **Test Case GM-001: Genre Creation**

**Precondition:** User is logged in with admin privileges

**Steps:**

1. Navigate to Admin Panel
2. Select "Manage Genres" 3. Click "Create New Genre"
4. Fill in required fields:
  - Name: "Test Genre"
  - Description: "This is a test genre description that explains the musical style"
5. Click Create
6. Verify genre appears in the genres list
7. Click on the created genre
8. Verify all details match input data
9. Navigate to artist creation form
10. Verify genre appears in genre dropdown
11. Navigate to song creation form
12. Verify genre appears in genre dropdown

### **Pre-Verification Process**

Ensure that no genre exists with the name "Test Genre."

### **Expected Results:**

- **Genre successfully created:** Should complete in no more than **1 seconds**.
- **Genre visible in admin panel:** Visible within **2 seconds**.
- **Genre available in dropdowns:** Available within **1 second**.
- **All details correctly displayed:** Display should be updated within **2.5 seconds**.

### **Test Case GM-002: Genre Edit**

#### **Steps:**

1. Navigate to Admin Panel
2. Select "Manage Genres"
3. Find test genre in list
4. Click Edit button
5. Modify description text
6. Click Save
7. Verify changes appear in list view
8. Verify changes appear in detail view
9. Verify changes reflect in song/artist creation dropdowns

#### **Expected Results:**

- **Changes visible in list view:** No more than **2 seconds**.
- **Changes visible in detail view:** Within **2.5 seconds**.
- **Changes reflect in dropdowns:** Update should occur within **1 second**.



## 5. Song Management Test Cases

### Test Case SM-001: Song Creation

**Precondition:** At least one artist and genre exist in system and user is logged in with admin privileges

#### Steps:

1. Navigate to Admin Panel
2. Select "Manage Songs" section 3. Click "Create New Song" button
4. Fill in required fields:
  - Title: "Test Song"
  - Artist: Select from dropdown (use previously created artist)
  - Genre: Select from dropdown (use previously created genre)
  - Mood: Select from dropdown
  - Duration: "03:45:00"
  - Release Date: Select current date from calendar
5. Click Create button
6. Verify song appears in the songs list
7. Click on the created song
8. Verify all details match input data
9. Navigate to homepage
10. Use search bar to find created song
11. Click on search result
12. Verify song details page displays correctly

#### Pre-Verification Process

Ensure no existing song with the title "Test Song" exists.

#### Expected Results:

- **Song successfully created:** Should complete within **3 seconds**.
- **Song visible in admin panel:** Should appear within **2 seconds**.

- **Song searchable:** Results should show within **2 seconds**.
- **All details correctly displayed:** Display update should be within **2 seconds**.
- **Duration formatted correctly:** Formatting should happen in under **1 seconds**.
- **Release date displayed correctly:** Display time should be under **1 seconds**.

## Test Case SM-002: Song Edit

### Steps:

1. Navigate to Admin Panel
2. Select "Manage Songs"
3. Find test song in list
4. Click Edit button
5. Modify fields:
  - Update title
  - Change mood from dropdown
  - Adjust duration
6. Click Save
7. Verify changes appear in list view
8. Verify changes appear in detail view
9. Verify changes appear in search results

### Expected Results:

- **Changes visible in list view:** No longer than **2 seconds**.
- **Changes visible in detail view:** Should reflect within **2.5 seconds**.
- **Changes appear in search results:** Update should be visible in **2 seconds**.