

# Release Summary

Product Name: Groovo

Team Name: Groovo

Date: 12/1/25

## Key user stories and acceptance criteria:

### 1. Authentication and Protected Access

**User Story:** As a user, I want to sign up and log in so that I can access my personalized music profile and features.

#### Acceptance Criteria

1. A new user can create an account with valid credentials (e.g. email/username + password) and is redirected to the logged-in experience.
2. Attempts to sign up with missing fields, invalid email, or mismatched passwords show clear validation errors and do not create an account.
3. A registered user can log in with correct credentials and is redirected to the main app (home/feed).
4. Logging in with invalid credentials shows a clear “invalid email or password” error and does not authenticate the user.
5. When not authenticated and trying to access a protected route (e.g. profile, library, feed), the user is redirected to the login page or shown a clear prompt to sign in.
6. When authenticated, the user can access protected routes without being redirected back to login.

### 2. Profile Viewing and Editing

**User Story:** As a user, I want to view and edit my profile (username and bio) so that I can customize how I appear to others.

#### Acceptance Criteria

1. From the header/navigation, an authenticated user can navigate to **/profile** and see their profile page.
2. The profile page displays the user’s username, profile image (or default avatar), basic stats (e.g. number of reviews, saved albums), and sections like Top 5 and My Reviews.
3. Clicking an **“Edit Profile”** button switches username and bio into editable fields and shows **Save** and **Cancel** actions.
4. On **Save**, valid changes to username and/or bio are persisted and shown on the profile page, even after page reload or re-login.

5. On **Cancel**, any unsaved changes are discarded and the original username/bio remains visible.
6. When viewing another user's profile (via shared link or feed), that user's public information and content are displayed (username, Top 5, reviews) without exposing edit controls.

### 3. Global Search and Album Discovery

**User Story:** As a user, I want to search for music so that I can quickly find albums and navigate to detailed information.

#### Acceptance Criteria

1. From the global search bar, typing a non-empty query triggers a search that returns a list of matching albums (and users if supported).
2. For album results, each result shows at least album title, primary artist name, and cover art.
3. Clearing the search input (deleting all characters) clears the current search results list.
4. Pressing **ESC** (or equivalent close action) dismisses the active search UI state without breaking the page.
5. Clicking an album result navigates to that album's detail page, showing information for the selected album.

### 4. Album Detail Page & Review Functionality

**User Story:** As a user, I want to view detailed information about an album and write a review so that I can share my thoughts.

#### Acceptance Criteria

1. From any album card (search results, artist page, feed), clicking the album navigates to a dedicated album page (e.g. /album/[id]).
2. The album page displays album title, artist name(s), cover image, and key metadata (e.g. release date, track list or basic info).
3. When authenticated, the album page shows a review form with rating input and text area.
4. Submitting a valid review (rating + non-empty text) creates a review that appears in the album's review section, displaying the author's username, rating, text, and timestamp.
5. Refreshing the album page shows the same review, confirming it is persisted in the backend.
6. A user can see their own existing review for that album and in their profile's "My Reviews" section.
7. When not authenticated, the review form returns an error message and prompts user that they must be logged in to create a review

## 5. Library: Saving and Managing Albums

**User Story:** As a user, I want to save albums to my library so that I can maintain a personal list of favorite or tracked albums.

### Acceptance Criteria

1. On an album detail page and/or album card, an **"Add to Library"** control is visible for authenticated users.
2. Clicking **"Add to Library"** saves the album to the user's library, updates the button state (e.g. "In Library" / checkmark), and shows a confirmation message.
3. Navigating to the Library (or library section in the profile) displays all saved albums as cards showing album title, artist, and cover.
4. Clicking **"Remove from Library"** (on album page) removes the album from the user's library and updates both UI and backend; the removed album no longer appears on library page
5. When logged out, attempting to add an album to the library does not succeed and should, in the ideal behavior, prompt the user to log in.

## 6. Social Feed and Following

**User Story:** As a user, I want to follow other users and see their activity in a dedicated feed so that I can keep up with reviews from people I care about.

### Acceptance Criteria

1. An authenticated user can navigate to the main feed section (e.g. on homepage) showing a **Global** and **Following** toggle.
2. In **Global** mode, the feed displays recent reviews/activity from many users (not limited to followed users), each card showing reviewer username, album info, rating, and a snippet of the review.
3. In **Following** mode, the feed shows only activity from users the current user follows; if no users are followed, a clear empty-state message is displayed.
4. From another user's profile, clicking **Follow** creates a follow relationship; the button state changes to **Following** and a confirmation message is shown.
5. Clicking **Unfollow** removes the following relationship and updates the button state accordingly.
6. After following a user, that user's new reviews appear in the **Following** feed; after unfollowing, their reviews no longer appear there.
7. Clicking a username in the feed navigates to that user's profile page.

## 7. Review Details Experience

**User Story:** As a user, I want to open a dedicated view for a specific review so that I can see the full content, likes, and comments.

### Acceptance Criteria

1. From a feed or social activity card, a **“Go to review”** action opens a dedicated review view (a dedicated /review/[id] page) showing that review.
2. The review details view displays album spotlight (cover, title, artist), reviewer info (username, avatar), rating, full review text, date, and current like count.
3. The like count and comment count shown in the review details view match those in the main feed card for the same review.
4. The view includes a comments section where authenticated users can read existing comments and (if implemented) add new comments.
5. The review details view includes a navigation action (e.g. “View album details”) which navigates to the associated album page and shows the same album referenced by the review.

## 8. Artist Pages and Discography

**User Story:** As a user, I want to view an artist’s page with their albums so that I can explore their discography.

### Acceptance Criteria

1. On an album page, clicking the artist’s name navigates to an artist detail page (e.g. /artist/[id]).
2. The artist page displays an artist hero section (image, name, follower/genre info where available) and a **“View on Spotify”** link that opens the correct Spotify artist page.
3. The artist page shows a grid of that artist’s albums; each card includes album cover, title, and basic metadata.
4. Clicking an album card on the artist page navigates to the corresponding album detail page and displays the correct album.
5. A Back or navigation control on the artist page returns the user to the previous context (e.g. the album page they came from or search results).

## 9. Top Albums of the Week (Billboard Integration)

**User Story:** As a user, I want to see a “Top Albums This Week” section on the homepage so that I can discover currently popular albums.

### Acceptance Criteria

1. The homepage displays a **Top Albums This Week** section with up to 5 album cards, each showing album cover, title, artist, and a rank label (e.g. #1–#5).
2. The albums in this section are derived from Billboard 200 data and enriched with Spotify metadata (covers, links).
3. Clicking an album card navigates to that album’s detail page within the app.
4. The set of top albums is stable across page reloads for a given period (e.g. week) and does not randomly change on every refresh.
5. If the top albums data cannot be loaded (e.g. scraper/error), a user-friendly fallback message is shown instead of a broken section.

## 10. Upcoming Events (Concerts Integration)

**User Story:** As a user, I want to see upcoming concerts for artists I've reviewed or followed so that I can check if there is a show near me.

### Acceptance Criteria

1. An authenticated user can navigate to an **Upcoming Events** page via the main navigation.
2. The events page shows a list of upcoming events for artists that the user has followed or reviewed, where such data exists in Ticketmaster.
3. Each event card displays artist name, event title, date, venue/city, and a link to "Buy Tickets" or "View on Ticketmaster."
4. Clicking the ticket link opens the corresponding event page on Ticketmaster in a new tab.
5. If no events are found for the user's artists, the page shows a clear "No upcoming events found for your artists" message.
6. If the Ticketmaster API fails, a user-friendly error message is displayed and the rest of the application remains usable.

## Known Problems:

This section documents the major bugs, omissions, design shortcuts, and known limitations observed during the final testing of the Groovo application. Many of these issues are referenced in the Test Report and were discovered across Sprints 1–4.

### 1. Authentication & User Account Issues

1. **Google OAuth sign-in does not allow custom usernames.**
  - a. Users who register via Google cannot edit or set a unique username after signup.
2. **Anonymous reviews bug.**
  - a. Reviews created from the album page occasionally show the reviewer as "anonymous" instead of displaying the correct authenticated user.
3. **Shared review links allow access but block interactions without feedback.**
  - a. When opening a shared review link while logged out, attempts to comment, like, or write a review fail silently with no login prompt.
4. **Missing authentication guard for certain actions.**
  - a. When logged out, actions such as adding albums to the library or posting reviews do nothing and do not inform the user that login is required.

### 2. Profile & Library Issues

1. **Incorrect album count on other users' profiles.**
  - a. Viewing another user's profile always shows **0 albums** in the profile header, even when that user has saved albums.
2. **Incorrect empty-state messaging on other users' profiles.**

- a. If another user has written no reviews, the page shows *"Write your first review from an album page,"* implying the viewer can write for them.
- 3. **Library state persists across users due to the "useLibrary" cache bug.**
  - a. LocalStorage is not cleared on logout.
  - b. The library does not refetch when a new user logs in.
  - c. Example:
    - i. User A saves SZA
    - ii. Logout
    - iii. User B logs in
    - iv. User B sees SZA in their library & concert feed.

### 3. Search & Discover Limitations

- 1. **Global search does not support searching for artists.**
  - a. Only Spotify albums and users appear in global search results.
- 2. **Artist pages cannot be accessed directly from search.**
  - a. Users must go to an album page, then click the artist name.
- 3. **No Last.fm search integration.**
  - a. Only genre endpoints use Last.fm; no artist/album search from Last.fm is implemented.
- 4. **The search bar** lacks a dedicated "clear" button and relies on deleting text manually.
  - a. After clearing results, the previous query persists in search results.

### 4. Album, Review, and Review Details Issues

- 1. **Posting reviews while logged out silently fails.**
  - a. No redirect, no message, no UI feedback.
- 2. **Review detail interactions (like/comment) require login but do not prompt for it.**
  - a. Logged-out users may attempt actions that fail silently.
- 3. **Temporary mismatches in like/comment counts between feed cards and review-details view.**
  - a. Counts update after backend sync.

### 5. Artist Pages

- 1. **Artist search missing entirely.**
  - a. Artist pages are implemented but cannot be reached directly except through an album page.
- 2. **Back navigation may behave inconsistently:**
  - a. This is depending on navigation history (Next.js routing quirk).

### 6. Top Albums This Week (Billboard Integration)

- 1. **Billboard 200 scraper does not parse all entries reliably.**
  - a. Only a portion of the chart is successfully scraped; the app displays the top 5 from the parsed subset instead of the true Billboard Top 5.

2. **Top Albums data is stored in a static JSON file instead of the database.**
  - a. Requires manual updates
  - b. No weekly refresh or cron job.
  - c. Not a fully dynamic feature.

## 7. Upcoming Events (Ticketmaster Integration) Issues

1. **False-positive events due to name-based search.**
  - a. Searching by artist name instead of Ticketmaster's artist ID returns unrelated events.
  - b. Examples include:
    - i. *"Murray State Racers Men's Basketball vs. Drake University Men's Basketball"* appearing under events for the artist **Drake**.
2. **Event availability varies widely.**
  - a. Some major artists show 0 events due to Ticketmaster API limitations.
3. **Events may appear for artists with:**
  - a. Similar names
  - b. Substrings
  - c. Venue references.

## 8. General UI/UX and Structural Limitations

4. **Inconsistent empty-state UI across the app.**
  - a. Some pages (feed, library, reviews) lack loading states, causing layout jumps.
5. **Inconsistent loading skeletons:**
  - a. Some pages (feed, library, reviews) lack loading states, causing layout jumps.

# Product Backlog

## High-Priority Items for Follow-On Development

This backlog outlines the highest-priority enhancements, bug fixes, and new user stories recommended for future development of Groovo. These items address known limitations, missing features, and architectural improvements identified during all sprints.

Priority	Backlog Item (Description)	SPs	Acceptance Criteria	Status
1	Add login-required prompts for posting reviews, liking, commenting, and adding to library	3	All restricted actions properly show login prompts when the user is not authenticated.	To Do
2	Fix "anonymous review" bug	3	All reviews display correct authenticated username; no anonymous posts unless	To Do

			intended.	
3	Allow Google sign-in users to set or edit their username	5	Google-authenticated users can customize their username after account creation.	To Do
4	Fix useLibrary to clear on logout and refetch correctly on user change	5	The library resets on logout and loads correct albums for each new login session.	To Do
5	Implement full artist search support	8	Searching an artist shows artist results with direct navigation to /artist/[id].	To Do
6	Improve Billboard scraper for accurate Top 5 albums	5	Scraper reliably parses all Billboard entries; displayed Top 5 matches Billboard Top 5.	To Do
7	Store “Top Albums This Week” in a database (replace static JSON)	8	Top albums are stored in DB and persist automatically; no manual file updates needed.	To Do
8	Improve Ticketmaster event search using artist IDs (remove false-positive events)	5	Events displayed correspond only to the correct artist; irrelevant events removed.	To Do
9	Improve empty-state messaging across profile, feed, library, and reviews	3	All major sections show correct, user-friendly empty-state messages.	To Do
10	Standardize loading skeletons across major pages	3	Feed, artist pages, library, and reviews share consistent loading UI.	To Do
11	US4: Personalized album recommendations based on user taste	5	Recommendations generated from user taste vector + Spotify recs; displayed on homepage.	To Do
12	US3: Moderator tools for removing comments/posts & managing roles	6	Moderators can soft-delete posts/comments, manage roles, and see moderator-only UI controls.	To Do