Assignment 1 - Vision and Scope Document

Team [25]

September 7, 2025

1. Project Vision

1.5 Vision Statement

For music enthusiasts who seek hassle-free playlist creation, searching and sharing, with precise control over which parts of songs are heard in a playlist. Our playlist manager supports track start/end markers as metadata and allows building playlists that play only the track parts you want. Unlike other media player vendors, it lets you set start/end markers for each individual track and combine them into quick playlists, skipping the undesired parts of your music library altogether.

1.3 Business Objectives

- Users can create, delete and name their playlists, as well as set the visibility of their playlists to others.
- Users can add start and end markers to individual tracks, allowing them to play only the parts they want, skipping unwanted sections.
- The order of tracks in each playlist remains stable after edits or deletion of the actual sound files.
- Playlists can be tagged, searched and filtered, helping users find the content they want.
- Users can share their playlists via visibility setting and save public playlists made by others.
- No one other than the playlist owner can modify their original playlist.
- A user can make a copy of another user's playlist if they want to add their own modifications without manually rebuilding the said playlist.
- The playlist system should be smooth and reliably support all presented use cases.

2.2 Scope of the Initial Release

- Accounts: creation, deletion, profile customisation, the ability to sign in.
- Playlists: creation, deletion, naming, renaming, visibility setting (public/private).
- Default track catalogue containing full metadata (e.g., artist, song title, album, tags).
- Start/end marker metadata, stored per track and applied within a playlist.
- Track search with result filtering and paginated results.
- Stable ordering rules within a playlist.
- Adding and removing playlists from favourites.
- Playlist copying for personal modificaitons.
- The ability to set or remove a playlist image.
- Ensuring that only playlist owners can delete or modify their playlists.

2. Use Cases

2.1 Use Case 1 – Add tracks to a playlist

Primary Actor: User.

Preconditions: The user must be logged into the system, the user must have a playlist that has already been created.

Success Guarantee: The song that has been selected by the user is added to the playlist.

Main Success Scenario: The user opens a playlist that already exists. He selects the “add song” option. He searches for a song (as described in UC2) and selects it. The system adds the song to the playlist and gives confirmation that the song has successfully been added to the playlist.

Extensions / Alternated Scenarios: If a song has already been added to the playlist, the system could display “this song has already been added to the playlist”.

Miscellaneous / Open issues: The ability to add multiple songs at once.

2.2 Use Case 2 – Search tracks

Primary Actor: User.

Preconditions: None.

Trigger: User selects the “search” option.

Main Success Scenario: The user enters a search query (song name, artist, album, etc). The system displays a list of matching songs. The user can view details of a selected song.

Success Guarantee: The system returns relevant songs that match the search criteria.

Extensions / Alternate Scenarios: If no results are found, the system displays “No songs found.” If the search query is empty, the system prompts the user to enter a valid search term.

2.3 Use Case 3 – Let the user specify start and end times for tracks in a playlist

Primary Actor: User who is a playlist owner.

Preconditions: User is signed in, owns the playlist, the playlist contains at least one track.

Main Success Scenario: The user selects a track inside their playlist. They then provide start and end times (e.g., 00:50 - 01:35). The system checks and confirms these times. The system then saves the provided metadata and confirms the operation. The user sees the updated track in the playlist with correct time markers.

Success Guarantee: The system stores the marker settings for the selected track in the context of the playlist.

Extensions / Alternate Scenarios:

- The provided start/end times are invalid (e.g., the times exceed the track length or are negative).
- The user cancels the operation, and the time markers are not updated.
- The track is removed during the operation, the system detects the change and aborts the operation.

2.4 Brief Use Cases

Brief Use Case 1 (UC4) – Delete playlist

The user is able to permanently delete a playlist.

Brief Use Case 2 (UC5) – Share playlists

The user is able to generate a link to share his playlist with friends, family and the community.

Brief Use Case 3 (UC6) – Save playlists

The user is able to save any public playlist to their favourites, or remove it from favourites.

Brief Use Case 4 (UC7) - Remove song from a playlist

The user can remove a song from his playlist.

Brief Use Case 5 (UC8) – Create Playlist

User can create a playlist.

3. Project Estimation and Prioritization

Lower number = higher priority

Use Case	Time Estimation	Priority
UC1	2 weeks	P1
UC2	2 weeks	P1
UC3	2 weeks	P2

4. Project Plan and Schedule

Inception: Use cases and vision scope

Elaboration: UC2,

Construction: UC1, UC4, UC5, UC8

Transition:

Week	Use Cases	Expected Hours	P. O: (Initials)	Sprint	Consultation
1	None			1	**A1 Presentation**
2	UC1, Android skeleton			1	Model Drafts
3	UC2, UC3		VG	2	**A2 Presentation**
4	UC4, UC5, UC6			2	Dev support
5	UC7, UC8			3	Dev support
6	UC9, UC10			3	**A3 Presentation**

5. Project skeleton

https://github.com/jok10/HBV501G_Hopur25

jok90, jok10 -- Jóhann Kjartansson

Loftur5 -- Loftur Páll Eiríksson

masderni -- Andri Már Sigurðsson

MountResonance -- Valentin Grechikhin