Group 05 - AlgoRhythm

CANARY MUSIC STREAMING MUSIC
Use-Case Specification

Version 1.2

Revision History

Date	Version	Description	Author
09/11/2024	1.0	Depict the diagram and write the use-case related to the user	Trần Anh Minh
10/11/2024	1.1	Write the use-case of uploading songs, reporting songs, solving requests and managing accounts and songs	Trần Anh Minh
14/11/2024	1.2	Minor change in use-case diagram and use-case description	Trần Anh Minh

Table of Contents

1. Use-case Model **4**

2. Use-case Specifications **5**

2.1 Use-case: Sign Up 5

2.2 Use-case: Sign In 6

2.3 Use-case: Account Authentication - Abstract 6

2.4 Use-case: Find songs 7

2.5 Use-case: Listen to songs 7

2.6 Use-case: Star songs 7

2.7 Use-case: Create playlists 8

2.8 Use-case: Report songs 8

2.9 Use-case: Upload songs 9

2.10 Use-case: Manage Accounts 9

2.11 Use-case: Manage Songs 10

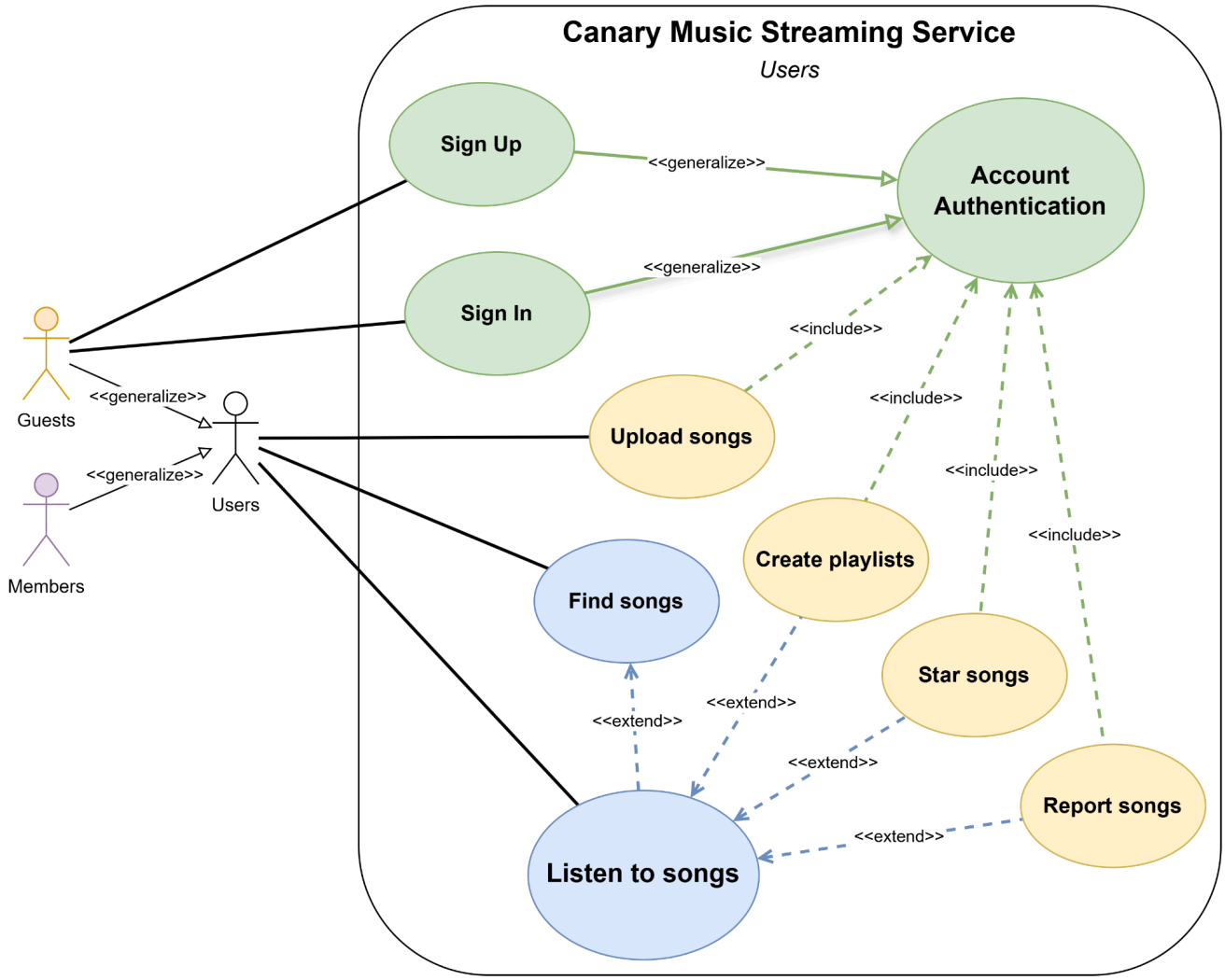
2.12 Use-case: Resolve Requests 10

1. Use-case Model

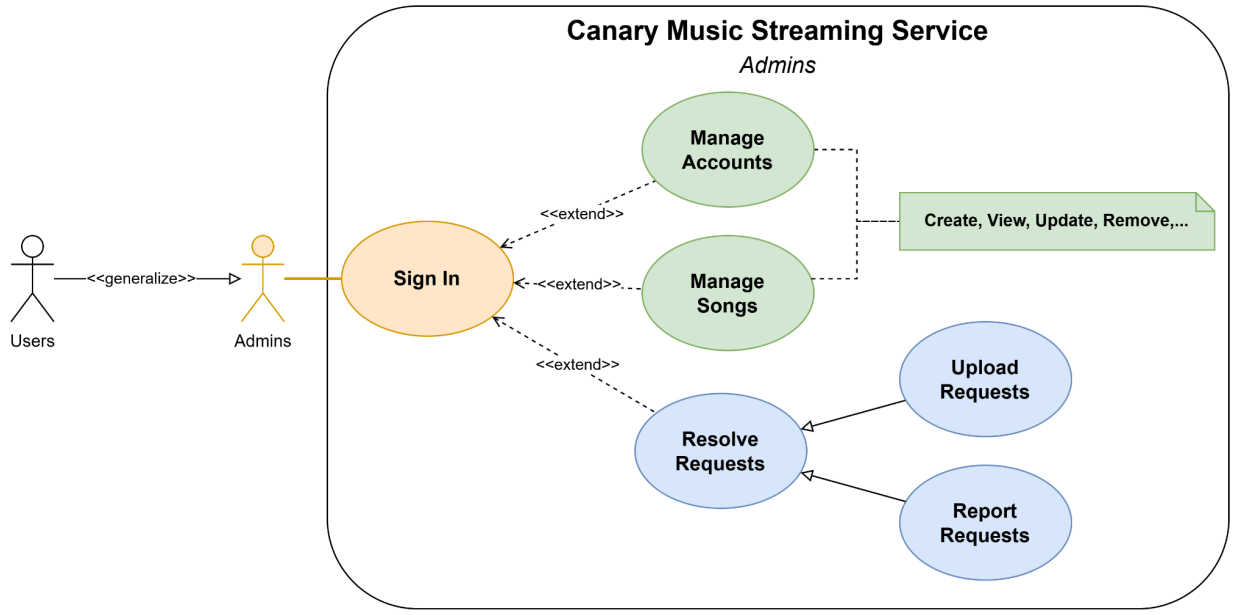
- ***List of Actors***

No.	*Actors*	*Descriptions*
1	Guests	Are users that have not signed in to the website. They can sign up, find songs, and listen to them but cannot star them or add them to playlists.
2	Members	Are the users who have signed in to the service inheriting all the functionalities of the guest users. Furthermore, they now can star songs, create playlists, and even upload songs to the community.
3	Admins	They will manage the issues relating to song reports and upload pending requests. They also have access to the songs and accounts database.

- ***Users***



- ***Admins***



2. Use-case Specifications

2.1 Use-case: Sign Up

Use case Name	Sign Up for the website account
Brief description	As guest users, we want to sign up to create a new account in the system to use the extra functionality and store the data.
Actors	Guests
Basic Flow	1. The guest navigates to the **"Sign Up"** page from the website's **main page** or via a relevant link. 2. The system will require the following information: o Username o Password o Email o Phone number 3. The system would check the validity: o The username isn't already taken. o The email isn't already taken. 4. The data will be stored in the database. 5. A notification will be displayed to notify the user. 6. Navigate back to the **main page**.
Alternative Flows	**The username has been taken.** - The system will notify the user to enter a different username. **The email has been taken.** - The system will notify the user to enter a different email.

Pre-conditions	The user does not have an account in the system.
Post-conditions	An account will be created and stored in the database. The user is automatically signed in.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 4-5 seconds.
Related Flow	None

2.2 Use-case: Sign In

Use case Name	Sign Up for the website account
Brief description	As guests and we have an account, we can sign back in using an email and password.
Actors	Guests
Basic Flow	1. The guest navigates to the **"Sign In"** page from the website's **main page** or via a relevant link. 2. The system will require the following information: ○ Email ○ Password 3. The system would validate the credentials: ○ The email is found inside the stored data. ○ The password matches the corresponding data. 4. Upon the successful validation, the system logged the user in. 5. Navigate back to the **main page**.
Alternative Flows	**The email has not been found.** - The system will notify the user that the account has not been created yet. **The corresponding data is mismatched.** - The system will notify the user that the password is incorrect for the current email.
Pre-conditions	The user has an account in the system but has not signed in yet.
Post-conditions	The user is successfully signed in to the system. The system ensures the user is logged in until they log out or quit the session.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 4-5 seconds.
Related Flow	None

2.3 Use-case: Account Authentication - *Abstract*

Use case Name	Authenticate the user account
Brief description	The system checks if the current user session has been signed in or not.
Actors	/
Basic Flow	The system checks the session state before enabling other functionalities.
Alternative Flows	**The session has not been signed in.** - The system will ask the user to sign in to have access to the functionality.
Pre-conditions	The user wants to use certain features, such as 'star songs' or 'upload songs.'

Post-conditions	The user will be able to use those mentioned features.
Non-functional Requirements	- The process should not take longer than 4-5 seconds.
Related Flow	***Sign Up, Sign In***

2.4 Use-case: Find songs

Use case Name	Find songs
Brief description	A user wants to browse through the list of songs via searching and filtering.
Actors	Guests, Members
Basic Flow	1. The users navigate the search bar located at the top-most section. 2. Users enter the name of the song or artist's name and hit 'enter'. 3. A new page will show up showing the search results. 4. Users can additionally filter out the songs by genre, tags, etc.
Alternative Flows	**No song has been found.** - The page will display that no matched result was found.
Pre-conditions	The user wants to search for the song.
Post-conditions	The system shows results on the screen for the user to choose from.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 4-5 seconds. - The result should match the searched songs.
Related Flow	None

2.5 Use-case: Listen to songs

Use case Name	Listen to songs
Brief description	The users can play the songs via the music player.
Actors	Guests, Members
Basic Flow	1. On the **main page,** users can choose the songs via: - The recommended songs, which are shown up on the **main page**. - Songs that are listed as the search results. 2. Simply hit the song to play it.
Alternative Flows	**No sound is being played.** - The queue is empty, waiting for the user to choose the song.
Pre-conditions	The user wants to play the song to listen to.
Post-conditions	The system plays the songs and adds the song to the playing queue.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 2–3 seconds. - Songs should be played smoothly without interruption.
Related Flow	***Find songs***

2.6 Use-case: Star songs

Use case Name	Star songs
Brief description	The users can star the songs for ease of access via the **Library page**.

Actors	Members
Basic Flow	1. Locate the song via search or through recommendation. 2. Hit the ☆ that is shown next to the song name. 3. The starred items will be shown up as a shortcut in the **Library page**.
Alternative Flows	**The songs have been starred.** - The songs will be unstarred and removed from the **Library page**. **Users have not signed in yet.** - The system will notify the users to sign in to use this feature.
Pre-conditions	Logged-in users want to have certain songs to access more quickly.
Post-conditions	The system will add the songs to the **Library** for ease of accessibility.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 2–3 seconds.
Related Flow	***Listen to songs, Account Authentication***

2.7 Use-case: Create playlists

Use case Name	Create playlists
Brief description	The users can gather songs up and create their own playlists.
Actors	Members
Basic Flow	1. Upon playing the song, user can hit the ⋮ 2. The user now can choose to add the song to certain playlists: - Playlist can be created newly. - Existed playlists. 3. The user can access it through the **Library page**.
Alternative Flows	**Remove the song from playlists.** - When accessing the songs inside a playlist, the user can choose to remove them from the playlist. **Users have not signed in yet.** - The system will notify the users to sign in to use this feature.
Pre-conditions	Logged-in user, have certain song to add to playlist.
Post-conditions	The system will add the songs to the chosen playlist.
Non-functional Requirements	- The system must handle up to 1000 concurrent requests. - The process should not take longer than 2–3 seconds.
Related Flow	***Listen to songs, Account Authentication***

2.8 Use-case: Report songs

Use case Name	Report songs
Brief description	The users can send reports of the song they think violates the policy.
Actors	Members
Basic Flow	1. Upon playing the songs, if they can hit the ⋮ to have access to the report feature. 2. The user will be asked to provide the reason why they think it violates the rule. 3. A request will be sent toward admins and wait to be resolved.

Alternative Flows	**Users have not signed in yet.** - The system will notify the users to sign in to use this feature.
Pre-conditions	User logged in.
Post-conditions	The system will queue up the report waiting to be resolved the Admins
Non-functional Requirements	- The process should not take longer than 2–3 seconds. - The report must be resolved within a day.
Related Flow	***Listen to songs, Account Authentication***

2.9 Use-case: Upload songs

Use case Name	Upload songs
Brief description	Users can upload their songs to share them with the community.
Actors	Members
Basic Flow	1. The user navigates the **"Upload"** section located on the navigation bar in the **main page**. 2. Upon clicking it, the system will show a window allowing the user to note down the song's information and upload the corresponding file. ○ The song's name ○ The song's metadata, such as its background, etc. ○ The song file. 3. The artist's name will automatically be marked as the username, alongside the release date, which is the current uploaded date. 4. But the song will not be released to the public just yet. A uploading request will be sent toward admins waiting to be verified. 5. When the song upload request is acceptable, the song can now be found across the platform and will be stored in the database.
Alternative Flows	**The song upload request is denied.** - The user will be notified that the song is denied as it may violate the policy. **Users have not signed in yet.** - The system will notify the users to sign in to use this feature.
Pre-conditions	Logged in users with song detail prepared.
Post-conditions	The song will be stored in the database, and everyone else can experience it.
Non-functional Requirements	- The process should not take longer than 2–3 seconds. - The request should be resolved within a day.
Related Flow	***Account Authentication***

2.10 Use-case: Manage Accounts

Use case Name	Manage Accounts
Brief description	Admins want to change information related to accounts.
Actors	Admins

Basic Flow	1. Upon logging in with the executive account, administrators can access the **"Management"** page. 2. Admins can view and change accounts' information: a. Change the password b. Change the username c. Add new account d. Remove account
Alternative Flows	**Cancel the update process.** - By hitting cancel, no information will be changed.
Pre-conditions	The admins have been logged in with an executive account.
Post-conditions	The account data will be updated accordingly.
Non-functional Requirements	- Only authorized administrators can access and manage accounts.
Related Flow	***Account Authentication***

2.11 Use-case: Manage Songs

Use case Name	Manage songs
Brief description	Admins want to change information related to songs.
Actors	Admins
Basic Flow	3. Upon logging in with the executive account, administrators can access the **"Management"** page. 4. Admins can view and change songs' information: a. Change the name b. Change it tags c. Update the artist's name or release date.
Alternative Flows	**Cancel the update process.** - By hitting cancel, no information will be changed.
Pre-conditions	The admins have been logged in with an executive account.
Post-conditions	The song data will be updated accordingly.
Non-functional Requirements	- Only authorized administrators can access and manage songs.
Related Flow	***Account Authentication***

2.12 Use-case: Resolve Requests

Use case Name	Resolve requests
Brief description	Admins need to resolve the requests sent by other users.
Actors	Admins
Basic Flow	1. Admins can access the request page via the navigation bar. 2. There will be two tabs: a. The report tab contains reports from the users.

	b. The pending upload songs. 3. Admins can mark the request as accepted or denied.
Alternative Flows	**Requests accepted.** - When the admins verify the requests as an accept, the reported song will be removed, or the uploaded song will be able to show up in the community. **Requests denied.** - If the requests are denied, users will be notified as their requests are not accepted by the admins.
Pre-conditions	The admins have been logged in with an executive account.
Post-conditions	The songs will be handled accordingly.
Non-functional Requirements	- Only authorized administrators can resolve the requests.
Related Flow	***Account Authentication***