
<SESlay>

<Museos>
Use-Case Specification

Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

[To customize auto

matic fields in Microsoft Word (which display a gray background when selected), select File>Properties and replace the Title, Subject and Company fields with the appropriate information for this document. After closing the dialog, automatic fields may be updated throughout the document by selecting Edit>Select All (or Ctrl-A) and pressing F9, or simply click on the field and press F9. This must be done separately for Headers and Footers. Alt-F9 will toggle between displaying the field names and the field contents. See Word help for more information on working with fields.]

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Revision History

Date	Version	Description	Author
10/11/2023	1.0	2.1, 2.13, 2.14, 2.15	Nguyen Thien Tho
10/11/2023	1.0	2.2, 2.3, 2.4, 2.5, 2.6, 2.7	Lam Thanh Ngoc
10/11/2023	1.0	2.8, 2.9, 2.10, 2.11, 2.12	Tran Nguyen An Phong

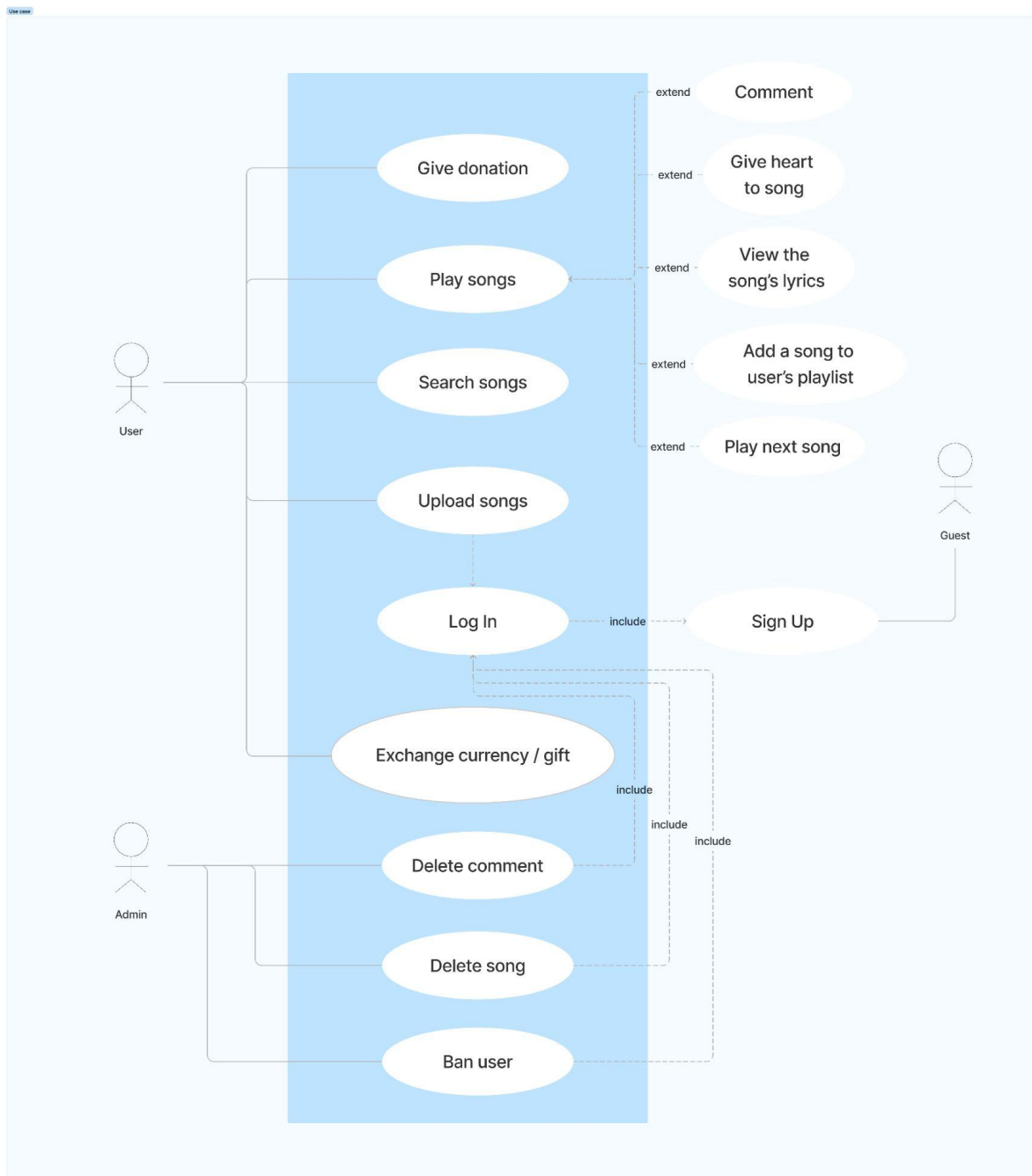
<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Table of Contents

1. Use-case Model	4
2. Use-Case Specifications	5
2.1 Use-Case: Search songs	5
2.2 Use-Case: Play the song	5
2.3 Use-Case: Give a comment to the song	6
2.4 Use-Case: Give heart to the song	6
2.5 Use-Case: View the song's lyrics	7
2.6 Use-Case: Add a song to the user's playlist	7
2.7 Use-Case: Play the next song	8
2.8 Use-Case: Delete comment	8
2.9 Use-Case: Delete song	9
2.10 Use-Case: Ban user	9
2.11 Use-Case: Exchange currency/gift	10
2.12 Use-Case: Donation	11
2.13 Use-Case: Sign up	11
2.14 Use-Case: Log in	12
2.15 Use-Case: Upload songs	13

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

1. Use-case Model



<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mm/yy>
<document identifier>	

2. Use-Case Specifications

2.1 Use-Case: Search songs

Use-case name	Search Songs.
Brief description	This use case outlines how a user can search for songs on the music platform.
Actors	User.
Basic flow	<p>Step 1: At the homepage, the user enters keywords in the Search field for songs.</p> <p>Step 2: User clicks on the Search button to initiate the search for songs.</p> <p>Step 3: The system displays the songs found based on the entered keywords.</p> <p>Step 4: Users can preview or select a specific song from the search results.</p>
Alternative flow	<p>Alternative flow: User cannot find products searched</p> <ol style="list-style-type: none"> From #3 of the basic flow, the system notifies the user that no songs were found. User can enter different keywords and continue from step #2 in the basic flow.
Pre-condition	The user is on the Homepage.
Post-condition	The user successfully finds and interacts with the searched songs.

2.2 Use-Case: Play the song

Use-case name	Play the song.
Brief description	This use case describes how the listeners can play the song.
Actors	User.
Basic flow	<p>Step 1: On many pages, the actor types the song's name in the "Search" bar to find the song.</p> <p>Step 2: System shows songs that have the same name as the result.</p> <p>Step 3: Actor clicks on the song.</p> <p>Step 4: System plays the song for the actor.</p>
Alternative flow	<p>Alternative flow 1: The song can't be found</p> <ol style="list-style-type: none"> From #1 of the basic flow, the actor searches for another nam Continue #2 in the basic flow <p>Alternative flow 2: The song is being played</p> <p>System opens the "playing song" screen again.</p>

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mm/yy>
<document identifier>	

Pre-condition	Actor has to log in to listen to the song.
Post-condition	The song that the actor finds is played.

2.3 Use-Case: Give a comment to the song

Use-case name	Give a comment to the song.
Brief description	This use case describes how the listeners can give comments to the playing song.
Actors	User.
Basic flow	<p>Step 1: At the “playing song” screen, the actor clicks the “three dot” symbol above the progress bar.</p> <p>Step 2: When the heart is filled , the song is added to the actor's “favorite songs”.</p> <p>Step 3: Actor clicks on the comment function.</p> <p>Step 4: System shows the “comment” screen.</p> <p>Step 5: Actor writes the comment.</p>
Alternative flow	<p>Alternative flow: The comment is inappropriate</p> <p>The comment is deleted and a warning message is sent to the commenter.</p>
Pre-condition	Actor goes to the Song page.
Post-condition	Actor successfully leaves the comment of the song.

2.4 Use-Case: Give heart to the song

Use-case name	Give heart to the song.
Brief description	This use case describes how the listeners can give heart to the playing song.
Actors	User.
Basic flow	<p>Step 1: At the Song page, the actor clicks the Heart symbol above the progress bar.</p> <p>Step 2: System shows the detail box including a comment function.</p>
Alternative flow	<p>Alternative flow: Heart is already given</p> <p>Heart is removed when the actor clicks on a filled heart.</p>

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Pre-condition	Actor goes to the Song page.
Post-condition	Actor successfully added the song to “favorite songs” playlist.

2.5 Use-Case: View the song’s lyrics

Use-case name	View the song’s lyrics.
Brief description	This use case describes how the listeners can view the song’s lyrics.
Actors	User.
Basic flow	Step 1: At the Song page, the song’s lyrics are shown next to the song's poster.
Alternative flow	Alternative flow: Artist doesn’t upload the lyrics The lyrics are not shown.
Pre-condition	Actor goes to the Song page.
Post-condition	Actors can follow the song with the lyrics.

2.6 Use-Case: Add a song to the user’s playlist

Use-case name	Add a song to the user's playlist.
Brief description	This use case describes how the listeners can add a song to the playlist.
Actors	User.
Basic flow	Step 1: At the Song page, the actor clicks the Three Dot symbol above the progress bar. Step 2: Actor selects the Add To Playlist button. Step 3: System shows all the playlists of the actor. Step 4: Actor chooses which playlist for the song. Step 5: When the playlist is selected, a message will appear announcing the song is successfully added.
Alternative flow	Alternative flow: No playlist is created <ol style="list-style-type: none"> From #3 of the basic flow, the actor adds a new playlist. Enter the name of the playlist and press enter or click OK.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Pre-condition	Actor goes to the Song page.
Post-condition	Actors can add the song to the playlist.

2.7 Use-Case: Play the next song

Use-case name	Play the next song.
Brief description	This use case describes how the listeners can play the next song.
Actors	User.
Basic flow	Step 1: At the Song page, click the Next button or click the Next Song box in the bottom right corner. Step 2: System shows the next song from the queue.
Alternative flow	Alternative flow: There is no song from the queue The song is randomly chosen from the system's library.
Pre-condition	Actor goes to the Song page.
Post-condition	Actors can play the next song from the queue or randomly.

2.8 Use-Case: Delete comment

Use-case name	Delete comments.
Brief description	This use case describes how the admin can delete a comment that he/she finds unfit with the community standard.
Actors	Admin.
Basic flow	Step 1: Admin click on the delete comment button. Step 2: A pop-up appears to ask for action confirmation. Step 3: The admin clicks Confirm on the pop-up. Step 4: The system deletes the user's comment.
Alternative flow	Alternative flow: Cancel confirmation process 1. When the pop-up appears, instead of clicking Confirm the admin choose Cancel . 2. The pop-up closes. There is no transaction happening.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Pre-condition	Admin hover to the delete comment button next to the user's comment.
Post-condition	Post-condition: The admin gets a notification of successful system transaction and the comment is deleted from the database

2.9 Use-Case: Delete song

Use-case name	Delete Song
Brief description	This use case describes how the admin deletes a user's song.
Actors	Admin.
Basic flow	Step 1: Admin click on the delete song button next to each song. Step 2: A pop-up appears to ask for action confirmation. Step 3: The admin clicks Confirm on the pop-up. Step 4: The system deletes the user's song.
Alternative flow	Alternative flow: Cancel confirmation process 1. When the pop -up appears, instead of clicking Confirm the admin choose Cancel . 2. The pop-up closes. There is no transaction happening.
Pre-condition	Admin goes to the user's profile page and a list of the user's songs show up.
Post-condition	Post-condition: The admin gets a notification of successful system transaction and the song is deleted from the database.

2.10 Use-Case: Ban user

Use-case name	Ban users.
Brief description	This use case describes how the admin bans a user.
Actors	Admin.
Basic flow	Step 1: The admin clicks on the “...” button on the user profile picture. Step 2: A list of options shows up, one is Ban user . The admin chooses Ban user . Step 3: A pop-up appears to ask for action confirmation. Step 4: The admin clicks Confirm on the pop-up.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mm/yy>
<document identifier>	

	Step 5: The system bans the user.
Alternative flow	Alternative flow: Cancel confirmation process <ol style="list-style-type: none"> 1. When the pop up appears, instead of clicking Confirm the admin choose Cancel. 2. The pop-up closes. There is no transaction happening.
Pre-condition	Admin go to the user's profile page.
Post-condition	Post-condition: The admin gets a notification of successful system transaction and the user is banned.

2.11 Use-Case: Exchange currency/gift

Use-case name	Exchange currency/gift
Brief description	This use case describes how users can exchange from normal currency to the platform's gift currency and vice versa.
Actors	User.
Basic flow	Step 1: The user hovers on their profile picture at the top right section of the home page. Step 2: A list of options shows up, one is Payment . Users choose this option. Step 3: A pop-up shows up, asking the user if they want to make a normal currency => gift transaction or a gift => normal currency transaction. The user chooses as they want. Step 4: The pop-up changes into a number form that the user can type in. Step 5: The user enters the amount they want to transfer and hits the Confirm button. Step 6: The pop-up closes and the transaction begins.
Alternative flow	Alternative flow 1: The User doesn't choose any option and click Cancel when the pop-up shows up. <ol style="list-style-type: none"> 1. The User clicks the Cancel button. 2. The pop-up closes and there is no transaction happening. Alternative flow 2: The User cancels after choosing the currency – gift option they want. <ol style="list-style-type: none"> 3. The Users click the Cancel button. 4. The pop-up closes and there is no transaction happening.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Pre-condition	Users go to the home page.
Post-condition	Post-condition: The system processes the transaction and the user gets the result of the transaction. If user account balance will be adjusted.

2.12 Use-Case: Donation

Use-case name	Donation.
Brief description	Users give donations to another user.
Actors	User.
Basic flow	Step 1: The User clicks on the “...” button on the user's profile picture. Step 2: A pop-up shows up with a number form for the User to enter the amount of gift they wish to donate. Step 3: Users type the amount and click Confirm . Step 4: The gift is transferred from the user’s balance to target balance. Step 5: The pop-up close.
Alternative flow	Alternative flow: User cancels the donation 1. Users click on the Cancel button and the pop-up closes.
Pre-condition	The user goes to the profile page of the other user that they wish to give a donation to.
Post-condition	Post-condition: The system processes the transaction and the user gets the result of the transaction. The user and artist balance will be adjusted.

2.13 Use-Case: Sign up

Use-case name	Sign Up.
Brief description	This use case describes how a guest can create a new account on the music platform.
Actors	Guest.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

Basic flow	<p>Step 1: Guest navigates to the Sign Up section at the Sign Up page.</p> <p>Step 2: Guest provides required information such as email, username, and password.</p> <p>Step 3: Guest clicks the Sign Up button.</p> <p>Step 4: The system validates the provided information and creates a new user account.</p> <p>Step 5: The system logs in the newly created user automatically and directs them to Homepage.</p>
Alternative flow	<p>Alternative flow: Existing Email/Username, Missing Information, Invalid Password</p> <ol style="list-style-type: none"> 1. The system notifies the guest. 2. Guest is prompted to choose a different email or username and continues from step #3.
Pre-condition	The guest is on the Sign Up page.
Post-condition	The user successfully logs in and directs them to Homepage.

2.14 Use-Case: Log in

Use-case name	Log In.
Brief description	This use case outlines the steps for an existing user to log in to the music platform.
Actors	User.
Basic flow	<p>Step 1: User navigates to the Login section at the Login page.</p> <p>Step 2: User enters their registered email/username and password.</p> <p>Step 3: User clicks the Login button.</p> <p>Step 4: The system verifies the entered credentials.</p> <p>Step 5: If the credentials are valid, the system logs the user in and directs them to Homepage.</p>
Alternative flow	<p>Alternative flow: Incorrect Credentials</p> <ol style="list-style-type: none"> 1. From #5 of the basic flow, if the entered credentials are invalid, the system notifies the user of the incorrect information. 2. User is prompted to re-enter the correct email/username and password and continues from step #3.
Pre-condition	User has a registered account on the music platform and is on the Homepage.
Post-condition	The user successfully logs in and directs them to Homepage.

<Museos>	Version: <1.0>
Use-Case Specification	Date: <dd/mmm/yy>
<document identifier>	

2.15 Use-Case: Upload songs

Use-case name	Upload Songs.
Brief description	This use case describes how a user can upload songs to the music platform.
Actors	User.
Basic flow	<p>Step 1: User navigates to the Upload section of the music platform.</p> <p>Step 2: User selects the option to upload a new song.</p> <p>Step 3: The system prompts the user to provide details about the song, such as title, artist, genre, and upload the file.</p> <p>Step 4: User fills in the required details and uploads the song file.</p> <p>Step 5: The system processes the uploaded song and adds it to the platform's database.</p> <p>Step 6: The system confirms successful song upload to the user.</p>
Alternative flow	<p>Alternative flow 1: Invalid song details or file format</p> <ol style="list-style-type: none"> From #3 of the basic flow, if the user provides invalid details or an unsupported file format, the system notifies the user. User corrects the details or chooses a supported file format and continues from step #4. <p>Alternative flow 2: Duplicate Song Upload</p> <ol style="list-style-type: none"> From #5 of the basic flow, if the uploaded song is found to be a duplicate (already exists on the platform), the system notifies the user. The system provides options for the user, such as updating the existing song details or canceling the upload. User chooses an appropriate action and continues or cancels the upload.
Pre-condition	User is logged into the music platform and is on the Profile page.
Post-condition	The user successfully uploads a new song to the platform.