PROJECT REQUIREMENTS

Overview

"You work for Rebmem Engineering that produces solutions for external clients. They wish to produce their own media player. They have the capability to produce the electronic hardware but need an interface to allow for the sale of the device as a portable music player."

Objectives

1. "Design, configure, test and install a Music player that meets business requirements."
2. "Roll out within the agreed time frame of a maximum of 5 days."
3. "Build an interface that allows for continuous improvement."

BUSINESS REQUIREMENTS

Analysis

From reviewing the project requirements, I have gathered the below assumptions:

1. Rebmem have the capacity to engineer hardware, therefore software is not necessary to distribute via an app store platform such as Apple App Store, Google Play, etc.
2. The need for an 'interface' alludes to the use of buttons, menus and further UI functionality built into the software, therefore the hardware may not accommodate for buttons outside of basic functions (on/off/volume control).
3. Reviewing the current market for music players, and with the above assumptions, I can garner that the hardware may look similar to the example shown [1.1] with touch screen functionality.
4. 'Music player' should include the most fundamental playback controls including play, pause, shuffle and loop.
5. 'Wish to produce their own music player' could suggest that future development of the product would allow for the playback of other media types, including video files.

Example music player
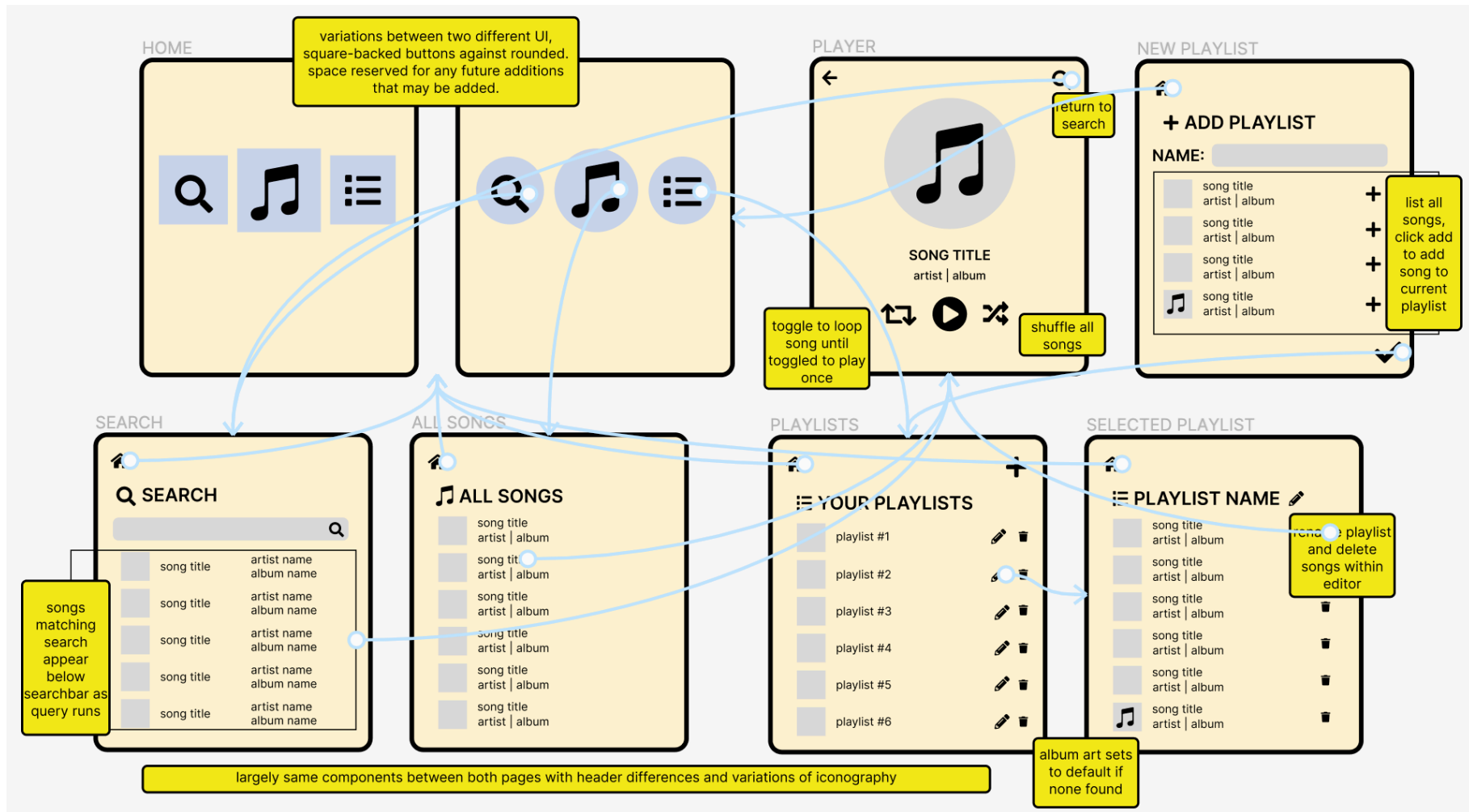
FUNCTIONAL REQUIREMENTS

ID #	FUNCTIONAL REQUIREMENT DETAILS
1.0 Audio Player	
1.1	Play an audio file.
1.2	Control the playback (pause / play) of an audio file.
1.4	Shuffle a list of audio files and play one at random.
1.5	Repeat an audio file once or continuously.
2.0 Playlist Management	
2.1	Create a playlist.
2.2	Edit the name of a playlist.
2.3	Search for a song based on a search term.
2.4	Sort any search results by song or album title.
3.0 Power Saving Functionality	
3.1	Switch to idle mode after 30 seconds of no interaction.
3.2	Allow music to be played during idle mode.

NON-FUNCTIONAL REQUIREMENTS

ID #	NON-FUNCTIONAL REQUIREMENT DETAILS
1	Edit and delete playlists.
2	Allow the user to skip or rewind an audio file.
3	Carousel home menu to add continuous improvements.
4	Song focus tray to control current music when in different menus.
5	Choose from a selection of pre-set royalty free wallpapers.
6	Built-in help menu with basic user guide.
7	Swipe to edit and delete.
8	Show the user remaining device battery life and current time.
9	'Favourite' songs into a global, default playlist

I SCOPED THESE NON-FUNCTIONAL REQUIREMENTS IN THE HOPES THAT THEY COULD BE ADDED IF TIME PERMITS OR IN FUTURE ITERATIONS OF THE PROJECT.

USER INTERFACE DIAGRAM

HOME

variations between two different UI, square-backed buttons against rounded. space reserved for any future additions that may be added.
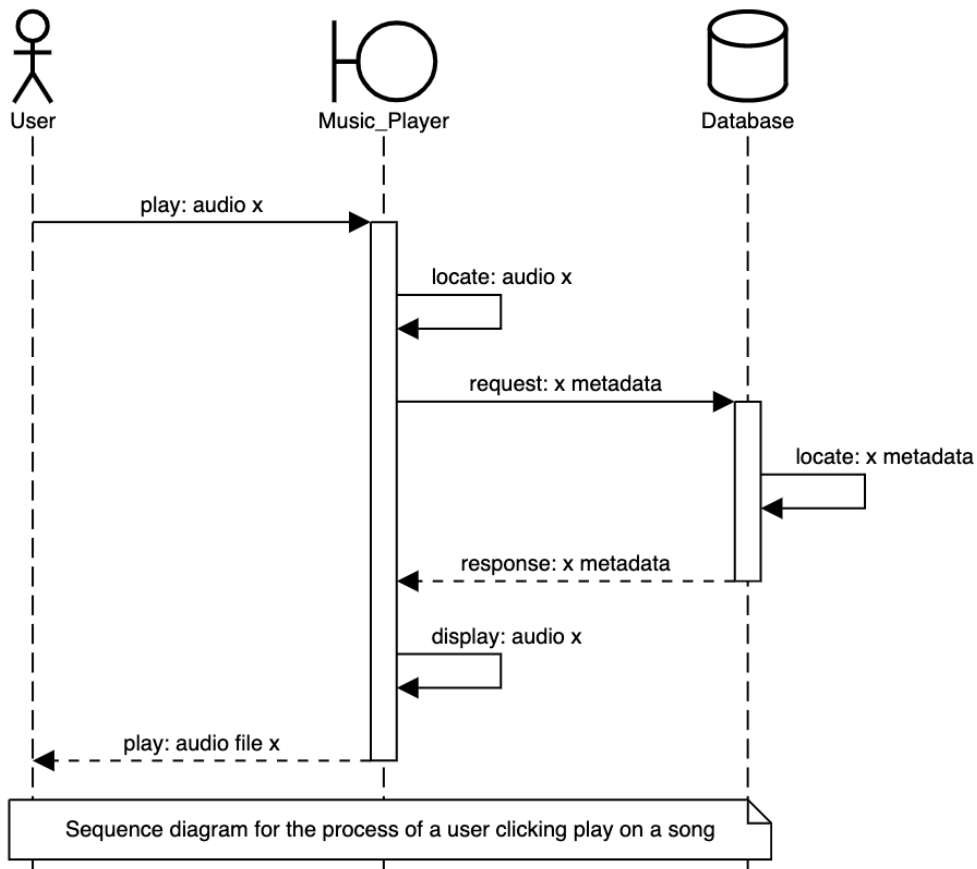
PLAYER

return to search

NEW PLAYLIST

+ ADD PLAYLIST

NAME:

song title
artist | album +

song title
artist | album +

song title
artist | album +

♫ song title
artist | album +

list all songs, click add to add song to current playlist

SONG TITLE
artist | album

toggle to loop song until toggled to play once

shuffle all songs

SEARCH

Q SEARCH

songs matching search appear below searchbar as query runs

	song title	artist name album name
	song title	artist name album name
	song title	artist name album name
	song title	artist name album name
	song title	artist name album name

ALL SONGS

♫ ALL SONGS

song title
artist | album

song title
artist | album

song title
artist | album

song title
artist | album

song title
artist | album

song title
artist | album

PLAYLISTS

☰ YOUR PLAYLISTS

playlist #1

playlist #2

playlist #3

playlist #4

playlist #5

playlist #6

SELECTED PLAYLIST

☰ PLAYLIST NAME ✎

song title
artist | album

song title
artist | album

song title
artist | album

song title
artist | album

song title
artist | album

♫ song title
artist | album

rename playlist and delete songs within editor

album art sets to default if none found

largely same components between both pages with header differences and variations of iconography

FLOW DIAGRAM

SEQUENCE DIAGRAMS

Song Request Sequence

User — Music_Player — Database

play: audio x
locate: audio x
request: x metadata
locate: x metadata
response: x metadata
display: audio x
play: audio file x

Sequence diagram for the process of a user clicking play on a song

Music Playlist Sequence

User — Music_Player — Database

add audio x to existing playlist "y"
request: locate playlist "y"
locate: playlist "y"
response: confirm playlist "y"
request: add audio x to playlist "y"
action: add audio x to playlist "y"
response: confirm action
display: audio x in playlist "y"
confirm: added audio x to playlist "y"

Sequence diagram for the process of a user adding a song to a playlist
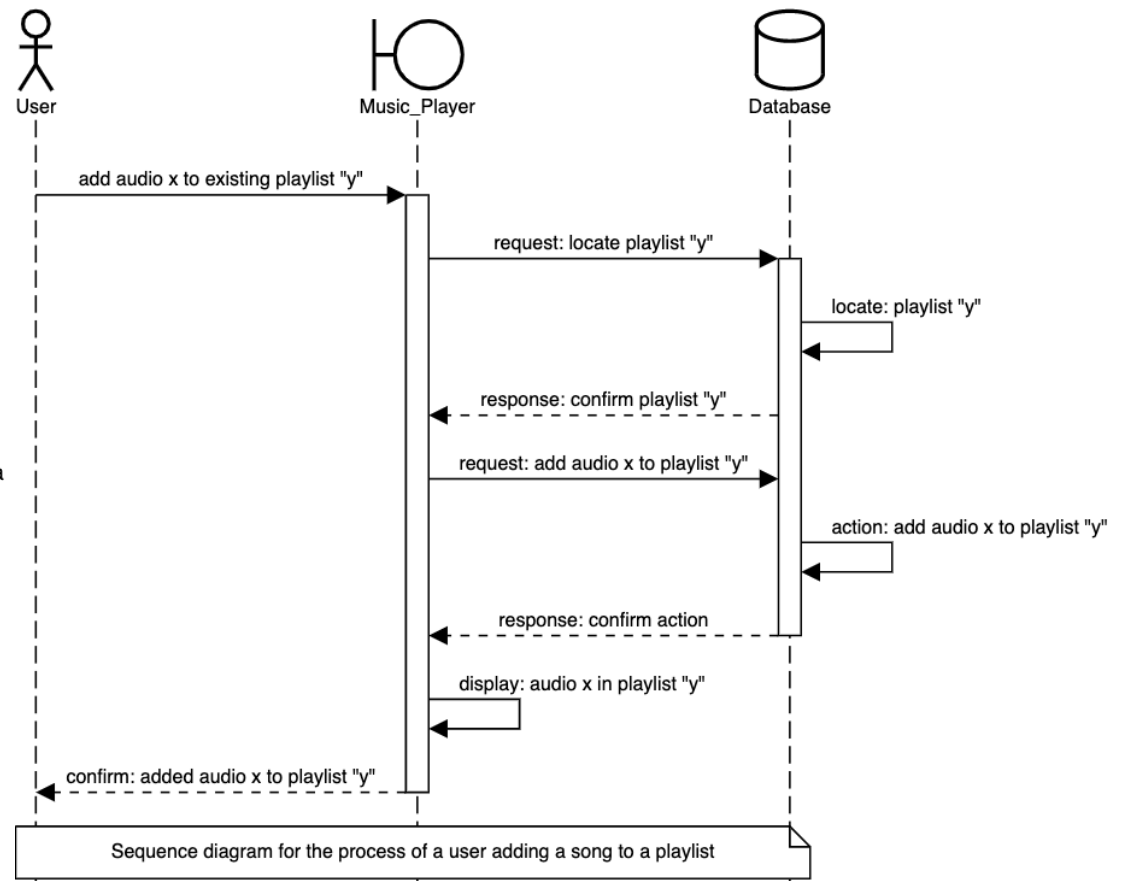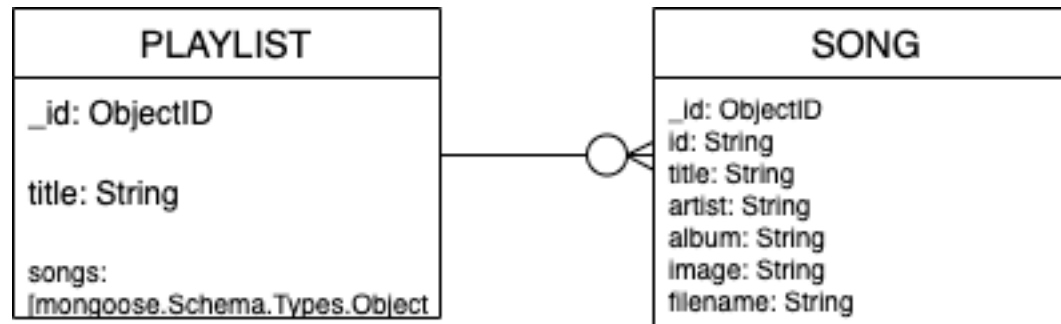
ENTITY RELATIONSHIP DIAGRAM

Each song has an ID generated by MongoD, a title (eg. 'Hyper Beam'), an artist (eg. 'MDK'), an album (eg. 'The Singles'), an image (eg. '/resources/default.png') and a filename (eg. '/resources/hyperbeam_mdk.mp3').