
ÇANKAYA UNIVERSITY

# Project Report
# For

***MyFM: Adaptive music player with personalized playlist generation***

*(Version 1.0)*

***Prepared By:***

***Group ID:*** **202019**

***Orkun Güneri-201611025***

***Erenalp Kesici-201811406***

***Necati Anıl Gökkaya-201511025***

***2020 - 2021***

# Contents

## 1.Literature Review


### Abstract

*In recent years the demand for personalization of recommendations have increased. This is because it's far easier to find something that you're looking for, recommended to you then have to search through the entire application. This is why we wanted to make this project about personalization of the recommendations to the user. We selected the music genre because it is a common occurrence that while listening to certain songs most people would enjoy to listen to similar songs that are in close proximity to the song that they are listening to right now. In this paper, we are trying to explain the processes that are involved in this.*



*Keywords: personalization of the recommendations, music genre, songs*



### Özet

Son yıllarda tavsiyelerin kişiselleştirilmesine olan talep artmıştır. Bunun nedeni, aradığınız bir şeyin size dogru olarak tavsiye edilmesi, kendinizin aramanizdan cok daha kolaydir ve her zaman akliniza bile gelmiyebilir. Bu yüzden bu projenin amacini, tavsiyelerin kullanıcıya göre kişiselleştirilmesi için yapmak istedik. Müzik türünü seçtik çünkü belirli şarkıları dinlerken çoğu insanın şu anda dinledikleri şarkıya çok yakın benzer şarkıları dinlemekten zevk alması yaygın bir durumdur. Bu yazıda, bununla ilgili süreçleri açıklamaya çalışıyoruz.



*Anahtar Kelimeler:* tavsiyelerin kişiselleştirilmesi, müzik türü, şarkılar

## 1.1 Introduction

MyFm is an application that adapts to the songs that the user listens to and create playlists based upon that information. This application will be exclusively available on Android systems due to the fact that we are using Android Studio, as our choice of development platform. The purpose of this project is mostly to keep the user interested in using the application by making sure that the accurate recommendations are made. This goal is achieved by considering the meta fields, previous songs that user listened to and so on.

## 1.2 Background

### 1.2.1 Personalized Playlist Generation

A personalized playlist is a list of songs that have been adapted to the user's preferred music style. Generation of these playlists requires certain information be collected and processed to output desirable playlists [1].

### 1.2.2 Mobile Application Development

Nowadays, there are a lot of options to choose from when it comes to development of mobile applications. In our project we decided Android Studio as it fit our needs the best. One of the reasons is that it allows for independent building, testing and debugging of the applications.

### 1.2.3 Android Studio

Android Studio is the official development environment for the operating system of Google, Android, built on JetBrains's IntelliJ IDEA software and was specifically designed for Android development. It provides visualization of an Android phone and allows for creation of applications on the platform [3].

## 1.3 Similar Applications

Other applications such as Spotify, a well-known music player, has a friendly user interface that allows for streaming music, playlist creation and also provides similar type of recommendation systems. For example, it looks at the length of a song that a user has listened to it for and make recommendations based upon that and more [2].

## 1.4 Conclusion

In the entertainment world of today, it is very useful for the user to be able receive accurate recommendations. While there are other applications that do similar recommendations, we would like to add different functionalities to this system and make it as unique as possible.

## 1.5 References


[1] [Online]. Available: https://hellofuture.orange.com/en/music-recommendation-algorithms-what-influence-do-they-have-on-what-users-listen-to/.

[2] [Online]. Available: https://ericboam.medium.com/i-decoded-the-spotify-recommendation-algorithm-heres-what-i-found-4b0f3654035b.

[3] [Online]. Available: https://en.wikipedia.org/wiki/Android_Studio.

# 2. Software Requirements Specification

## 2.1 Introduction

### 2.1.1 Purpose

The main objective of this document is to explain the adaptive music player with personalized playlist generation called MyFm. This music player aims to generate personalized playlists for the user after examining a pattern in the user's music taste. This application is for people that would like to be able to listen to songs that are in close proximity to what they have listened to before[1].

### 2.1.2 Scope of the Project

MyFm is a music player that allows for the generation of personalized playlists. The aim of this application is to satisfy the user with songs that they would enjoy listening to by finding the right recommendations.

### 2.1.3 Glossary

| | |
|---|---|
| UI | User Interface |
| SRS | Software Requirements Specification |
| OS | Operating System |

### 2.1.4 References

[1] [Online]. Available: https://en.wikipedia.org/wiki/Media_player_software. [Accessed 3 12 2020].

### 2.1.5 Overview of the Document

The second part of this SRS document explains the functionalities of the MyFM: Adaptive music player with personalized playlist generation.

The third part introduces the requirements that come with using this application.

## 2.2 Overall Description

### 2.2.1 Product Perspective

MyFm is a self-contained, music player that will provide the user with personal playlists that will continually adapt to user's listening history. It will look at many fields including metadata etc. in achieving its goal.

#### *2.2.1.1 Development Methodology*

For developing the project, we chose Android Studio as our development platform, because this platform fits our goals for this project and makes it easy to maintain its functions.

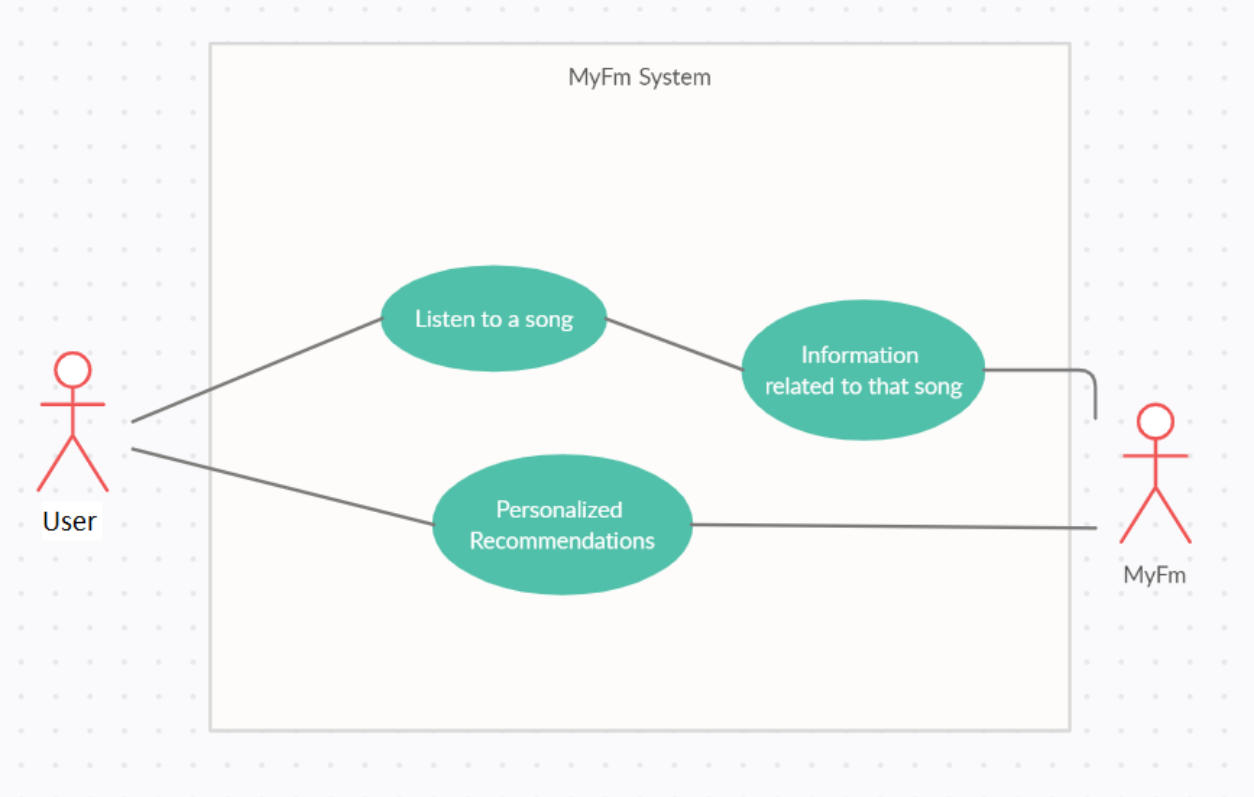
#### *2.2.1.2 User Interface*

One of the most crucial part of MyFm project is its ease of use with the correct placements of certain UI fields (play/pause buttons, previous/next buttons etc.), its goal is to be made as user friendly as possible.

### 2.2.2 User Characteristic

The user should know the basics of how to operate smart phones.

### 2.2.3 Product Functions




## 2.3 Requirements Specifications

### 2.3.1 External Interface Requirements

#### *2.3.1.1 User Interface*

The user interface requires to be run on Android platforms.

#### *2.3.1.2 Hardware Interface*

There are no strict hardware requirements other than that it is required for the user to have a smart phone with Android OS installed on it.

#### *2.3.1.3. Software interfaces*

This application will need a minimum of Android version 4.1(Jelly Bean) or higher to be run.

#### *2.3.1.4 Communications interfaces*

Wireless or wired internet connection is required for listening to the songs and using the application.

### 2.3.2 Functional requirements

Some data will need to be collected to fulfill the goal of the application and ensure accurate recommendations delivered to the user.

### 2.3.3 Software System Attributes

#### *2.3.3.1 Portability*

The system was designed and coded in Android Studio thus it will be available exclusively on Android devices.

#### *2.3.3.2 Availability*

The system will be usable as long as there is an active internet connection.

#### *2.3.3.3 Security*

By using a number of secure network protocol techniques, it will be made as secure of an application as possible.

#### *2.3.3.4 Adaptability*

Since some data is being collected when using the application, it will be highly adaptive to user inputs.

#### *2.3.3.5 Usability*

The MyFm system will be easy to use thanks to its heavy emphasis on the simplicity of its UI mechanics.

*2.3.3.6 Safety Requirements*

Listening to music at a high volume for a long time can cause long-term damage to user's ears.

# 3. Software Design Document

## 3.1 INTRODUCTION

### 3.1.1 - Purpose

The purpose of MyFM provides user's to reach their personal music tastes and give them lookalike musics, groups or even different types of musics they might prefer. The Purpose of this document is guiding to users throughout the development phase of the MyFM and let's users to look at producer's poin.This document also fulfills the requirements of the Software Requirements Spesifications.

### 3.1.2 - Glossary

| Term | Definition |
|---|---|
| User | Users of the application |
| Android Studio | Android Studio is a development platform that allows us to work on the Android operating system and develop applications. |

| | |
|---|---|
| Selected Music | Music that applications selected by using Users Data |

### 3.1.3 - Intended Audience

Anybody that likes to listen to music.There is no age limit.

### 3.1.4 - Motivation

As a 3-member senior student, our motivation was our learning appetite for application development, artificial intelligence and machine learning. In this project, we tried to find and satisfy those who like to listen to music as a band and those who want to discover new music.

## 3.2 DESIGN OVERVIEW

### 3.2.1 - System Design Approach

Although we did not meet outside the house, as much as we could and as our other lessons allowed, we connected via discord and talked about our project.

### 3.2.2 - Used Technologies

MyFM made and developed with Android Studio IDE.Our target users are mostly Android users for the beginning.Language is mostly Java.

### 3.2.3 - Architecture Design of Simulation

Users will be able to choose from around 30 different music genres. As a result of the elections, they will choose from the music groups that come across. As a result of the selections, they will reveal your general musical taste and recommend songs accordingly.

#### *3.2.3.1 Options Menu*

Users can reach their Library, Liked, Your Mix, History and Downloads.They will be able to  Stop, Play , Skip songs .

## 3.3 EXTERNAL INTERFACE REQUIREMENTS

### 3.3.1 - Hardware Interfaces

Users only needs a Android Smartphone.

### 3.3.2 - Software Interfaces

A better cellphone would make the experience better.

### 3.3.3 - Communications Interfaces

To listen Music Player player doesn't need internet connection but internet is mandatory to find new music groups ,albums ,songs.

### 3.3.4 - Performance Requirements

For finding new songs and download them Internet speed is important.

### 3.3.5 - Performance

The internet connection you can Access differs the performance.Also performance is based on the phone you have.

## 3.4 – Conclusion

On this documentation we tried to let users to see how we approach to this Project and we handled our problems.Also users are now able to see what it is intended to use in the most appropriate way.

Enjoy Responsibly.

# 4.Test Plan

## 4.1 INTRODUCTION

### 4.1.1 Version Control

| Version No | Description of Changes | Date |
|---|---|---|
| 1.0 | First Version | April 15, 2021 |

### 4.1.2 Overview

In this test plan document the features of an adaptive music player called MyFm will be tested.

### 4.1.3 Scope

This test plan document aims to include the test cases that we are completing currently.

### 4.1.4 Terminology

Acronym Definition

| Acronym | Definition |
|---|---|
| UI | User Interface |

## 4.2 FEATURES TO BE TESTED

### 4.2.1 Login (LG)

Logging into the system by entering username and password fields is required to be able to use the application.

#### 4.2.2 Register (RG)

Users need to be able to added to the system by registering them in order for them to be able to use the system.

#### 4.2.3 Update Account (UA)

When users wish to, they can update their account information.

### 4.3 FEATURES NOT TO BE TESTED

The system will not be tested on devices running operating systems other than Android because of availability.

### 4.4 ITEM PASS/FAIL CRITERIA

#### 4.4.1 Exit Criteria

- 100% of the test cases are executed
- 95% of the test cases passed
- All High and Medium Priority test cases passed

### 4.5 References


- [1] [Online]. Available: https://github.com/CankayaUniversity/ceng-407-408-2020-2021-MyFM-Adaptive-music-player-with-personalized-playlist-generation/wiki/Software-Requirements-Specification.
- [2] "MyFmSDD," [Online]. Available: https://github.com/CankayaUniversity/ceng-407-408-2020-2021-MyFM-Adaptive-music-player-with-personalized-playlist-generation/wiki/Software-Design-Document.

## 4.6 TEST DESIGN SPECIFICATIONS

### 4.6.1 Login (LG)

#### *4.6.1.1 Test Cases*

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| LG.01 | | H | Enter a valid email and password |
| LG.02 | | H | Enter a valid email and wrong/blank password |

### 4.6.2 Register (RG)

#### *4.6.2.1 Test Cases*

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| RG.01 | | H | Enter an existing valid email and password |
| RG.02 | | H | Enter a non-existing/wrong/blank email and password |

### 4.6.3 Update Account (UA)

#### *4.6.3.1 Test Cases*

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| UA.01 | | H | Enter a valid email and password |

### 4.6.4 Add Playlist (AP)

#### *4.6.4.1 Test Cases*

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| AP.01 | | M | Click create and enter a non-empty name |
| AP.02 | | M | Click create and enter an empty name |

### 4.6.5 Search Song (SS)

#### *4.6.5.1 Test Cases*

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| SS.01 | | M | Search by entering a non-empty song/artist |

| TC ID | Requirements | Priority | Scenario Description |
|---|---|---|---|
| SS.02 | | M | Search by entering an empty song/artist |

### 4.7 Detailed Test Cases

#### 4.7.1 LG.01

| TC ID | LG.01 |
|---|---|
| Purpose | Enter a valid email and password |
| Requirements | - |
| Priority | High |
| Estimated Time Needed | 30 seconds |
| Dependency | Registering cases need to be passed |

| TC ID | LG.01 |
| --- | --- |
| Procedure | [A01] Go to the login page<br>[A02] Enter a valid email and password<br>[A03] Press the login button |

#### 4.7.2 LG.02

| TC ID | LG.02 |
| --- | --- |
| Purpose | Enter a valid email and wrong/blank password |
| Requirements | - |
| Priority | High |
| Estimated Time Needed | 30 seconds |
| Dependency | Registering cases need to be passed |

| TC ID | LG.02 |
| --- | --- |
| Procedure | [A01] Go to the login page<br>[A02] Enter a valid email and wrong/blank password<br>[A03] Press the login button |

### 4.7.3 RG.01

| TC ID | RG.01 |
| --- | --- |
| Purpose | Enter a valid email and password |
| Requirements | - |
| Priority | High |
| Estimated Time Needed | 1 minute |
| Dependency | - |

| TC ID | RG.01 |
| --- | --- |
| Procedure | [A01] Go to the register page<br>[A02] Enter a valid email and password<br>[A03] Press the register button |

#### 4.7.4 RG.02

| TC ID | RG.02 |
| --- | --- |
| Purpose | Enter a non-existing/wrong/blank email and password |
| Requirements | - |
| Priority | High |
| Estimated Time Needed | 1 minute |
| Dependency | - |

| TC ID | RG.02 |
| --- | --- |
| Procedure | [A01] Go to the register page<br>[A02] Enter a non-existing/wrong/blank email and password<br>[A03] Press the register button |

### 4.7.5 UA.01

| TC ID | UA.01 |
| --- | --- |
| Purpose | Enter a valid email and password |
| Requirements | - |
| Priority | Medium |
| Estimated Time Needed | 1 minute |
| Dependency | - |

| TC ID | UA.01 |
| --- | --- |
| Procedure | [A01] Go to the update account page<br>[A02] Enter a valid email and password<br>[A03] Press the update button |

#### 4.7.6 AP.01

| TC ID | AP.01 |
| --- | --- |
| Purpose | Add a playlist |
| Requirements | - |
| Priority | Medium |
| Estimated Time Needed | 30 seconds |
| Dependency | - |

| TC ID | AP.01 |
| --- | --- |
| Procedure | [A01] Go to the playlist page<br><br>[A02] Enter a non-empty name |

#### 4.7.7 AP.02

| TC ID | AP.02 |
| --- | --- |
| Purpose | Add a playlist |
| Requirements | - |
| Priority | Medium |
| Estimated Time Needed | 30 seconds |
| Dependency | - |
| Procedure | [A01] Go to the playlist page<br><br>[A02] Enter an empty name |

#### 4.7.5 SS.01

| TC ID | SS.01 |
|---|---|
| Purpose | Search for a song |
| Requirements | - |
| Priority | Medium |
| Estimated Time Needed | 1 minute |
| Dependency | - |
| Procedure | [A01] Go to the home page<br>[A02] Enter a non-empty song/artist name<br>[A03] Press the register button |

#### 4.7.6 SS.02

| **TC ID** | **SS.02** |
|---|---|
| Purpose | Search for a song |
| Requirements | - |
| Priority | Medium |
| Estimated Time Needed | 1 minute |
| Dependency | - |
| Procedure | [A01] Go to the home page<br>[A02] Enter a empty song/artist name<br>[A03] Press the register button |

# 5.Test Plan Results

## 5.1 Individual Test Results

| TC ID | Priority | Date Run | Run By | Result | Explanation |
|---|---|---|---|---|---|
| LG.01 | H | 28/05/2021 | Erenalp Kesici | Pass | User logged in |
| LG.02 | H | 28/05/2021 | Erenalp Kesici | Pass | User didn't log in |
| RG.01 | H | 28/05/2021 | Erenalp Kesici | Pass | User registered |
| RG.02 | H | 28/05/2021 | Erenalp Kesici | Pass | User didn't register |
| UA.01 | H | 28/05/2021 | Erenalp Kesici | Pass | User logged in |
| AP.01 | M | 28/05/2021 | Erenalp Kesici | Pass | Created a playlist |

| TC ID | Priority | Date Run | Run By | Result | Explanation |
|---|---|---|---|---|---|
| AP.02 | M | 28/05/2021 | Erenalp Kesici | Pass | Didn't create a playlist |
| SS.01 | M | 28/05/2021 | Erenalp Kesici | Pass | Searched for a song |
| SS.02 | M | 28/05/2021 | Erenalp Kesici | Pass | Didn't search for a song |

## 5.2 Summary of the Test Results

| Priority | Number of TCs | Executed | Passed |
|---|---|---|---|
| H | 5 | 5 | 5 |
| M | 4 | 4 | 4 |

## 5.3 Exit Criteria

| Criteria | Yes or No |
|---|---|
| 100% of the tests executed | Yes |
| 100% of the test cases passed | Yes |
| All high, medium and low priority test cases passed | Yes |