

Software Engineering Project Report

Topic: Android Music Player

Submitted To: ABH3

Course: CSE327

Section: 4

Submitted By:

Saqib Hasan – 1731792042

Koushiq Das – 1721627642

Zannatul Ferdous Prome - 1621109042

**CONTENTS:**

* INTRODUCTION
* PROBLEM STATEMENT
* SOLUTION
* STAKEHOLDERS
* TOOLS & TECHNOLOGY
* MODELS & METHODS
* STAKEHOLDER
* USER STORIES
* BREAKDOWNS USER STORIES TO TASK
* FUNCTIONAL REQUIREMENTS
* USER REQUIREMENTS
* SEQUENCE DIAGRAM
* USE CASE DIAGRAM
* CLASS DIAGRAM
* DESIGN
* TEST CASES
* RISK ANALYSIS
* PRODUCT BACKLOG
* APPENDIX

**INTRODUCTION**

Android is open source code mobile phone operating system that comes out by Google in November 2007. Its appearance has broken the traditional closed mobile phone operating system. Anyone can modify the mobile phone operating system as well as function according to personal preference, which is also the most attractive merit of Android.

Music player based on Android application is popular in the market at the present. The completing development of Android operating system gives developers a nice platform, which can learn the popular computer technology combining with learned knowledge, and master the latest knowledge, enrich oneself, and enjoy entertainment.

**PROBLEM STATEMENT**

If we don’t use any music player application on android handsets it will be both very difficult and annoying for the users to perform their tasks. Lack of music player applications will guide users to enter in file manager, go to music folder and then choose music and play. Besides, they won’t have any search and filtering options. These steps will create a hustle for the users.

**SOLUTION**

We will design an android application which will enable the users to:

* Sign up and Login to the application
* Check recent playlist
* See the full music list
* Play, Stop, Pause Audio files (mp3)
* Search their preferences
* Create own playlist
* Notification on the home screen

**TOOLS & Technology**

* **JDK**：Java Runtime Environment virtual machine、Java Development Kit(JDK) Installation steps of the developing environment
* **Android Studio: Android Studio** is the official integrated development environment (IDE) for Android application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.
* **Firebase:** **Firebase** is a mobile-backend-as-a-service that provides powerful features for building mobile apps. **Firebase** has three core services: a realtime database, user authentication and hosting. We use the Firebase here for authentication.

**Methods**

* Register for Application
* Sign In
* View Playlist
* PLAY ANY SONG
* Play. Pause, Previous & next
* Seekbar
* Voice Command

**STAKEHOLDER**

* All classes of people.

**USER STORIES**

* As a user I want to create an account of my own in the application so that I can have the privacy of using the app and have my playlist and works saved.
* As a user I want to see the full playlist, play, pause, stop options so that I can have full command over the application usage.
* As a user I want to search music and create my own playlist in the app so that I can have flexibility over my application.

**BREAKDOWNS USER STORIES TO TASK**

* Create simple & friendly user interface.
* Create welcome page
* Create login for users.
* Create help & feedback
* Create music library as home page
* Create recent activity
* Create search option in music library
* Create play, stop, pause option in player screen
* Create shuffle on/off option in player screen
* Create Thumbs up option in player screen
* Create next/previous option in player screen
* Create playlist in music library
* Create Albums in music library
* Create Artists in music library
* Create songs in music library
* Create genres in music library
* Notification on home screen
* Create log out option

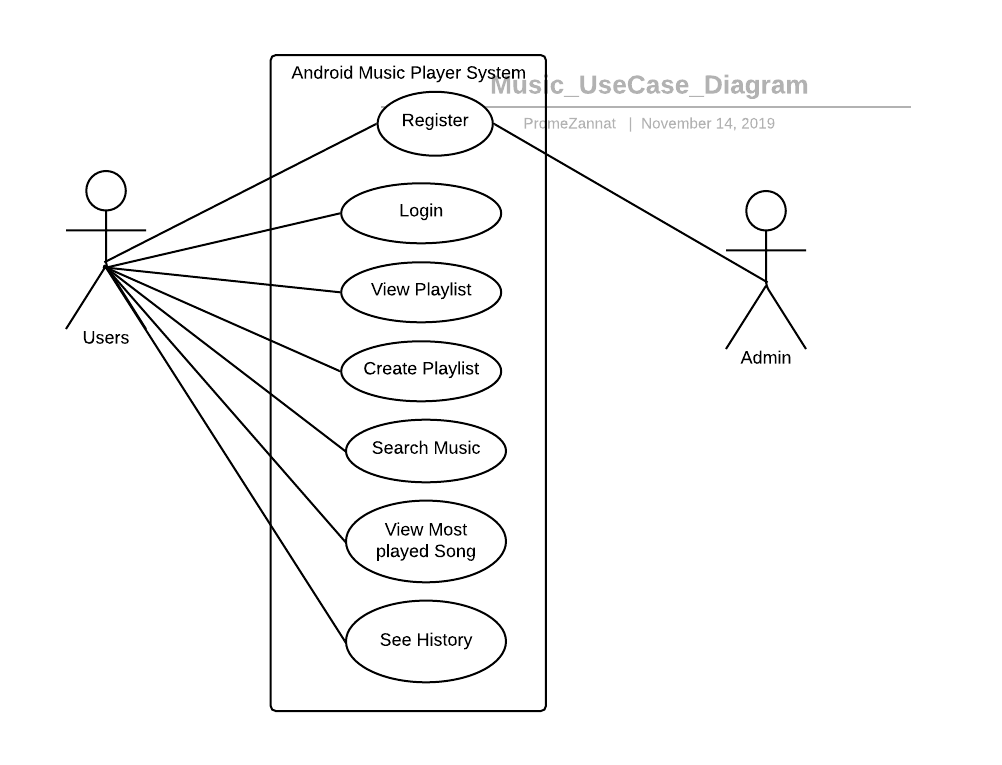
**FUNCTIONAL REQUIREMENTS**

* Android operating system on the Smartphone.
* The target device should be sound enabled.
* Ability to play Audio file.
* Welcome Screen.
* Login, Signup Screen.
* Main Screen.
* Player Screen.

**USER REQUIREMENTS**

|  |  |
| --- | --- |
| Sign Up | User will be able to sign up with their email and password. |
| Sign In | User will be able to sign in with their registered email and password. |
| Playlist | App will be read all the songs from the phone storage and user will be able to view playlist |
| Play Songs | User will be able to play, pause, next, previous music. They also use seekbar also. |
| Voice command | User will be able to control the play, pause next previous button by their voice command. |

**USE CASE DIAGRAM**



**Android Music Player UML Use Case Diagram**

**CLASS DIAGRAM**

|  |
| --- |
| USER |
| -UserId  +String email  -String password |
| -viefInfo()  -add()  -delete() |

|  |
| --- |
| PLAYLIST |
| -PlaylistId  -string song name  -bool next  -bool previous  -bool pause  -string voice |
| -viewSongs()  -viewButton()  -Command() |

**SEQUENCE DIAGRAM**

**RISK ANALYSIS**

|  |  |  |
| --- | --- | --- |
| RISK | AFFECT | DESCRIPTION |
| Schedule | Project | Due to some management problem we have to change the schedule of this project. |
| Gradle File Sync | Project & Product | When we 3 develop the project with different devices and finally connect it together its gradle file sync failed because of different version of android studio. These affect the performance and hamper the schedule. |
| Firebase Authentication | Project & Product | When we use the firebase authentication in the sign in and sign up activity its causes some change in the manifest file. That affect the performance and hamper the schedule. |
| Voice Manipulation | Product |  |
| App Crash | Product |  |

**PRODUCT BACKLOG**

**TEST CASES**

JUnit is the most popular unit[Testing](https://www.guru99.com/software-testing.html)framework in Java. It is explicitly recommended for [Unit Testing](https://www.guru99.com/unit-testing-guide.html). JUnit does not require server for testing web application, which makes the testing process fast.

JUnit framework also allows quick and easy generation of test cases and test data. The **org.Junit** package consist of many interfaces and classes for JUnit Testing such as Test, Assert, After, Before, etc.