MINI PROJECT

(2021-22)

"SOCO"

Project Report

Institute of Engineering & Technology

Submitted By -

Vartika Saxena (191500896)

Sneha Banga (191500813)

Shubh Purwar (191500795)

Rishabh Garg (191500649)

Under the Supervision Of

Mr. Amir Khan

Technical Trainer

Department of Computer Engineering & Applications

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Declaration

I/we hereby declare that the work which is being presented in the Bachelor of technology. Project **"SOCO"**, in partial fulfillment of the requirements forthe award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my/our own work carried under the supervision of **Mr. Amir Khan, Technical Trainer, Dept. of CEA,GLA University.**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign: *Vartika Saxena*

Name of Candidate: Vartika Saxena

University Roll No.:191500896

Sign: *Sneha Banga*

Name of Candidate: Sneha Banga

University Roll No.:191500813

Sign: *Shubh Purwar*

Name of Candidate: Shubh Purwar

University Roll No.:191500795

Sign: *Rishabh Garg*

Name of Candidate: Rishabh Garg

University Roll No.:191500649

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Certificate

This is to certify that the project entitled "SOCO", carried out in Mini Project – I Lab, is a bonafide work by Vartika Saxena, Sneha Banga, Shubh Purwar and Rishabh Garg and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (ComputerScience & Engineering).

Signature of Supervisor:

Name of Supervisor: Mr. Amir Khan

Date:

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

ACKNOWLEDGEMENT

Presenting the ascribed project paper report in this very simple and official form, we would like to place my deep gratitude to GLA University for providing us the instructor Mr Amir Khan, our technical trainer and supervisor.

He has been helping us since Day 1 in this project. He provided us with the roadmap, the basic guidelines explaining on how to work on the project. He has been conducting regular meeting to check the progress of the project and providing us with the resources related to the project. Without his help, we wouldn't have been able to complete this project.

And at last but not the least we would like to thank our dear parents for helping us to grab this opportunity to get trained and also my colleagues who helped me find resources during the training.

Thanking You

Sign:*Vartika Saxena*

Name of Candidate: *Vartika Saxena*

University Roll No.:191500896

Sign: *Sneha Banga*

Name of Candidate: *Sneha Banga*

University Roll No.:191500813

Sign:*Shubh Purwar*

Name of Candidate: *Shubh Purwar*

University Roll No.:191500795

Sign: *Rishabh Garg*

Name of Candidate: *Rishabh Garg*

University Roll No.:191500649

ABSTRACT

In this project, we are creating an React Native application, basically a Music Player App which we have named SOCO. This application will provide us a platform to access the music we want to listen at the ease of our fingertips. All the users will be having their separate accounts on this app which will be connected to their email id. Any music that the user wishes to listen will be entered by him in the search box which works on the basis of queries input. The query may be a song name to which the music is related and will also work on specific keywords input. Apart from searching the music online, the user can save the music he/she likes in the favourites bar. The app is suitable in the present scenario as we are so much busy in our life that we don't have time for our mental health so we can listen to our favourite beats which will calm us. On the profile of the user, one can easily listen the music he/she wants . The app will be completely efficient and transparent to the reviews of the people on the songs and its quality. To get more details about the songs one can click on the song's card and get further grave details which is present in the home page. This app will be using The React Native Track Player for providing all the songs. The app also has a complete User Interface attached to the firebase a perfect login system with email id and password and a forget password too.

React Native app ecosystem is diverse and is changing people's life all over the world. React Native Application works perfectly on both Android and IOS.React Native users are expected to increase because of the advance changes of the operating system and the way it deals with issues and compatibility with other mobile devices. Furthermore designing solutions for the problems that we may face in future is essential. Like this application definitely stands the need of students at any time at their fingertips without any barrier of place.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER-1

INTRODUCTION

1.1 CONTEXT

This React Native Application “SOCO” has been submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering at GLA University, Mathura supervised by Mr. Amir Khan. This project has been completed approximately one month and has been executed in modules, meetings have been organised to check the progress of the work and for instructions and guidelines.

1.2 MOTIVATION

In the recent years, we have realized the importance of listening music and how important it is for our mental health . Music is a great source for relaxing and calming after a long hectic schedule .Due to stress and anxiety we all are suffering from different diseases so Music helps us to improve our health.

It can also reduce heart rate, lower blood pressure, decrease cortisol (stress hormone) levels and increase serotonin and endorphin levels in the blood. It elevates mood. Music can boost the brain's production of the hormone dopamine. This increased dopamine production helps relieve feelings of anxiety and depression.

This would be an excellent effort to provide everyone with free access to songs.

1.3 OBJECTIVE

Nowadays, due to deepening roots of internet throughout the world the indulgement of peoples in technical things like softwares, sites, audio content, and video content mainly in youtube increases drastically. So the IT sector of every country has a boost in their work.

Apart from all this brain boggling things humans need relaxation as simple as possible. Music which is one of the top of the relaxation tool, providing with a **simple modern UI, having different modes to listen , making playlist functions, providing sleep mode and featured with latest trending music all in one** platform now on **"SOCO".**

Soco stands for "**Sound Companion**". A music player for all. Including age groups senior or junior citizen, student or teacher etc.

1.4 EXISTING SYSTEM

In the present scenario, we are dealing with the manual searching of songs from thousands of other websites present in Google. With the help of this application we are able to find a place where we can easily find our favourite songs with the help of keywords. As this idea as already implemented here are the some snap how our application will look.

As soon as the user enters the app, there will be landing page containing the name of the app and then there will be a login /signup page .Initially there will be search bar as shown in the image below. Then on the basis of certain keywords the app will fetch the results and the songs will be displayed as shown in the second screenshot. In this we will add a feature to bookmarks the song or add to favourites as this will be helpful when the user try the use the same song again.

(a) **(b)**

Figure-1: Existing System

1.5 SOURCES

The source of our project (including all the project work, documentations and presentations will is available at the following

https://github.com/Nitro2000/SOCO_Synopsis

CHAPTER -2

SOFTWARE REQUIREMENT ANALYSIS

2.1 IMPACT OF MUSIC ON DAILY LIFE

There have been many studies attempting to figure out just how music affects the human mind. For example, why do different people like different kinds of music, what parts of the brain are activated and if said parts are affected more or less by different kinds of music, and others.

Some of these questions have remained unanswered, and might stay that way for a while. One thing, however, is for certain: music does affect our daily lives.Just look at the simple facts. Most people listen to music. If someone doesn't, they're sometimes viewed in a strange way. How could you not listen to music?

Music affects our emotions. When we listen to sad songs, we tend to feel a decline in mood. When we listen to happy songs, we feel happier. Upbeat songs with energetic riffs and fast-paced rhythms (such as those we hear at sporting events) tend to make us excited and pumped up.With all this in mind, I sent out a survey to the students of Basehor-Linwood High School, asking some simple questions about their music taste and how music makes them feel. Studying these results show some interesting facts.

When asked about their listening habits, mixed results were found in accordance to the amount of time spent listening to music on a daily basis.

About 22.2 percent of people said that they listen to music between one to two hours everyday, where another 22.2 percent said they listen at least five hours a day.

The category of two to three hours a day sees about 18.4 percent of people in the school, and three to four hours meets a close second to that, at 16.5 percent.

Only 11 percent of people listen to less than an hour's worth of music every day, and even less listen to four to five hours a day; about 9.5 percent.

It seems that there isn't really a happy medium. Either people listen to music a little, or they listen to music all the time. Music takes different standpoints in different people's lives, and it matters more or less to one person than it does another.

A majority of people listen to music in the car, as well as at home; about 90 percent of all those studied for each. Around 71 percent of people here in the school also listen in their classrooms. Both the hallway and the lunchroom receive substantially less; about 37 percent and 25 percent.It seems that music helps us concentrate and study as well. Out of those studied, 88.5 percent of people said that they listen to music when they study, work on homework, and other activities such as that. That leaves on 11.5 percent of people who don't.

It's no surprise that most people (69 percent) listen to pop music. Pop literally stands for popular. 55.2 percent of all people attending BLHS listen to rock and rap. It's also not surprising to hear that 46.6% of people listen to alternative and indie music. Over half of our students listen to country, at 52.3 percent.

2.2 PROBLEM STATEMENT

The main objective of the Music Player is to manage the details of Music, Performer, Album, Customer, Album Type. It manages all the information about Music, Track, Album Type, Music. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Music, Performer, Track, Album. It tracks all the details about the Album, Customer, Album Type.

2.3 HARDWARE AND SOFTWARE REQUIREMENTS

Software Requirement

- Technology Implemented : React-Native
- Language Used : JavaScript
- Database : FireBase

• User Interface Design : Native Base

• Web Browser : Google Chrome/Firefox

Hardware Requirements

• Processor : intel i3

• Operating System : Windows, Mac or Linux • RAM : 4+ GB

• Hardware Devices : Computer System

• Hard disk : 64GB

2.4 MODULES AND FUNCTIONALITIES

- **Home Page**: The first screen with which the user interacts will be this screen containing the logo and the app name .This will disappear within 5 seconds after the app is displayed.

- **Login Page**: This page is for those users who have already registered themselves on the app and have a username and a password. There is also a way on this page for the new users to register themselves which will take them to the registration page.

- **Registration Page:** This is page is solely designed for the new users of the app who are willing to register themselves. This page takes input of the various details of the user and stores it in the database, later helping the user to login into the account with credentials they have provided.

- **Forget Password Page:** This page comes into picture when one of the user forgets the login credentials. In this case this page asks for the email-id with which the user has already registered. The app will check if there is any entry in its database

with the id and if there a mail will be sent to the same id for recovering the credentials and notification will be given to the user.

- **Navigation Drawer**: This is the most important part of the application that provides interactivity within the app as it connects the various activities together like it is a side bar on which the profile, the dashboard, the favourites section, the FAQ section ,the About page of the page are linked and on clicking on each you can visit the pages.
- **Dashboard Page:** This is the page displayed for every user after entering the app successfully. It contains the search bar where the user can search the song according to the wish as well as some of the songs are suggested with the genres recently searched or the most popular one.

- **Favourites Page:** Initially the page is empty, but when the users search for a song and like it then one can add it to the favourites section. This place is a user's personal space to store any song he likes or want to mark.

- **Feedback Page:** This page is for users where they can give their feedback about our app and they can also check out other's feedbacks as well.

- **Logout page**: Then is this last panel for the users to sign out from the account. As soon as the users sign out they are brought back to the login page.

2.5 SOCO ON REACT NATIVE APPLICATION

Nowadays, due to deepening roots of internet throughout the world the indulgement of peoples in technical things like softwares, sites, audio content, and video content mainly in youtube increases drastically. So the IT sector of every country has a boost in their work.

- Apart from all this brain boggling things humans need relaxation as simple as possible. Music which is one of the top of the relaxation tool, providing with a **simple modern UI, having different modes to listen , making playlist functions, providing sleep mode and featured with latest trending music all in one** platform now on **"SOCO".**

- Soco stands for "**Sound Companion**". A music player for all. Including age groups senior or junior citizen, student or teacher etc.

CHAPTER-4
TECHNOLOGY USED

REACT NATIVE

React Native is a JavaScript framework for writing real, natively rendering mobile applications for iOS and Android. It's based on React, Facebook's JavaScript library for building user interfaces, but instead of targeting the browser, it targets mobile platforms. In other words: web developers can now write mobile applications that look and feel truly "native," all from the comfort of a JavaScript library that we already know and love. Plus, because most of the code you write can be shared between platforms, React Native makes it easy to simultaneously develop for both Android and iOS.

Similar to React for the Web, React Native applications are written using a mixture of JavaScript and XML-esque markup, known as JSX. Then, under the hood, the React Native "bridge" invokes the native rendering APIs in Objective-C (for iOS) or Java (for Android). Thus, your application will render using real mobile UI components, not webviews, and will look and feel like any other mobile application. React Native also exposes JavaScript interfaces for platform APIs, so your React Native apps can access platform features like the phone camera, or the user's location.

React Native currently supports both iOS and Android, and has the potential to expand to future platforms as well. In this book, we'll cover both iOS and Android. The vast majority of the code we write will be cross-platform. And yes: you can really use React Native to build production-ready mobile applications! Some anecdota: Facebook, Palantir, and TaskRabbit are already using it in production for user-facing applications.

Advantages of React Native

The fact that React Native actually renders using its host platform's standard rendering APIs enables it to stand out from most existing methods of cross-platform application development, like Cordova or Ionic. Existing methods of writing mobile applications using combinations of JavaScript, HTML, and CSS typically render using webviews. While this approach can work, it also comes with drawbacks, especially around performance. Additionally, they do not usually have access to the host platform's set of native UI elements. When these frameworks do try to mimic native UI elements, the results usually "feel" just a little off; reverse-engineering all the fine details of things like animations takes an enormous amount of effort, and they can quickly become out of date.

In contrast, React Native actually translates your markup to real, native UI elements, leveraging existing means of rendering views on whatever platform you are working with. Additionally, React works separately from the main UI thread, so your application can maintain high performance without sacrificing capability. The update cycle in React Native is the same as in React: when props or state change, React Native re-renders the views. The major difference between React Native and React in the browser is that React Native does this by leveraging the UI libraries of its host platform, rather than using HTML and CSS markup.

For developers accustomed to working on the Web with React, this means you can write mobile apps with the performance and look and feel of a native application, while using familiar tools. React Native also represents an improvement over normal mobile development in two other areas: the developer experience and cross-platform development potential.

4.1 TOOLS AND LANGUAGES

Tools used to build the Android App are:-

- **Visual Studio Code**: Visual Studio Code is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git
- **Software Development Kit (SDK)**: Android Studio requires a collection of libraries and data therefore SDK is mandatory..

4.2 BASIC TERMINOLOGY

- **Layout:** Layout is the parent of view. It arranges all the views in a proper manner on the screen.

- **Activity**: An activity can be referred as your device's screen which you see. User can place UI elements in any order in the created window of user's choice.

- **View**: A view is an UI which occupies rectangular area on the screen to draw and handle user events.

- **Emulator**: An emulator is an Android virtual device through which you can select the target Android version or platform to run and test your developed application.

- **Manifest file**: Manifest file acts as a metadata for every application. This file contains all the essential information about the application like app icon, app name, launcher activity, and required permissions etc.

- **API:** Short for Application Programming Interface. APIs are functions that developers can call on to access specific features by calling upon programs, code, and services that others have written. For example, if a developer wants to draw a button on the screen, she can insert a small bit of code that says "draw this kind of button, with this color and size and style, at this location" instead of dozens of lines of code that tells the graphics processor, in detail, exactly how to draw a button. If the application wants your location, it can use the location API to "get the device's location" and let Google's code handle the rest, instead of requiring the developer to build an entire location service from scratch just for her own app. There are thousands of APIs in Android, covering everything from drawing interface elements, to the cameras, to location access, to accessing storage, to 3D graphics (see: OpenGL ES) and much more.

- **Intent:** Intents are an essential part of the Android ecosystem. They are used to express an action to be performed. Intents allow you to interact with components from the same applications as well as with components contributed by other applications. It can be classified into implicit and explicit intents.

- **Implicit intent:** It does not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it.
- **Explicit Intent:** It specifies the component to start by name. You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start.

- **APK**: Short for "Android application package." The extension used in Android app installation files (e.g., app.apk). Similar in nature to an EXE file on Windows.

- **SDK:** Short for "Software Development Kit." As it pertains to Android, the SDK is a set of tools such as code libraries, a debugger, and a handset emulator that can be run on Windows, Mac, or Linux to facilitate the creation of Android apps by developers. While the SDK is generally intended for use by developers, end users can install the software on their home computer to execute ADB and Fast boot commands.

- **Action Bar**: The action bar is an important design element, usually at the top of each screen in an app that provides a consistent familiar look between Android apps. It is used to provide better user interaction and experience by supporting easy navigation through tabs and drop-down lists.

- **Navigation bar**: Android Navigation Drawer is a sliding left menu that is used to display the important links in the application. Navigation drawer makes it easy to navigate to and fro between those links. It's not visible by default and it needs to opened either by sliding from left or clicking its icon in the Action Bar.

- **Fragment**: A Fragment represents a behavior or a portion of user interface in a Fragment Activity. You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities.

- **Firebase** is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Firebase is categorized as a NoSQL database program,

which stores data in JSON-like documents. Firebase has three core services: a real-time database, user authentication and hosting. With the Firebase iOS SDK, you can use these services to create apps without writing any server code.

JSON stands for JavaScript Object Notation. It is an independent data exchange format and is the best alternative for XML. JSON is used for data interchange (posting and retrieving) from the server. Hence knowing the syntax and it's usability is important. JSON is the best alternative for XML and its more readable by human

CHAPTER -5

IMPLEMENTATION AND USER INTERFACE

Creating an app concept design with screen sketches and functional flow diagrams is the best way to communicate your vision to the mobile app developer. Making the concept clear to the developer is probably the most important factor in successful mobile app development. Yet it is one of the most common problems or obstacles in a mobile app development outsourcing project.

No matter what the marketing and profit goals are or if you are outsourcing an app for your personal use, you need to fully design and document the app concept if you expect a programmer to make your vision a reality. Developers are not mind readers and even descriptions given during conversations can be very fleeting or interpreted differently. Fully documenting your concept, therefore, leaves little to chance. The two most important things to do are: A) make a comprehensive description of how the app works and what it does (functionality) and B) create a comprehensive description of what the user sees and does (look and feel).

5.1.1 Step to be followed by the user

1. Firstly, we have to open the application.
2. Then, we have the Login activity which consists of following steps
    - Register : for new User
    - Login: For existing as well as new user
    - Forgot Password: To reset your password
3. We authenticate and store the user information from the Firebase authentication.
4. After that, we made a Drawer layout of our Music app which includes variousfunctionality
    - Profile Fragment: To check the profile and update the database.
    - Dashboard Fragment: Show the book on the genre basis and it is open by default.
    - Favourite: To see the favourites book that you have added or you found interesting.
    - FAQ's Fragment: It comprises all the frequently asked questions.
    - Sign-out Fragment: Remove/logout you from the app.
5. In Dashboard fragment we also include the search bar in which the user can search for the songs of their interest.
6. After that list of song according to your search will appear
7. Select the song according to your choice.
8. Then the description of song will appear. It comprises of the following things:
    - Song Name
    - Singer Name
    - Releasing Date
    - Languages
    - Rating
    - Preview
9. If you want to add the song into favourites then in the description page Add to Favourites option is also given. You can see the song that are added in the favourite into navigation Drawer menu item favourites.
10. Now you can enjoy the listening of your favourite song.

5.2 User Interface

- **Opening Page**

Figure-7: Opening page

- **Register Page**

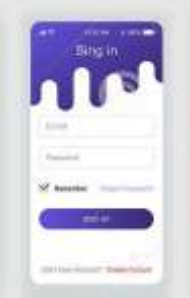


Figure-8: Register Page

- **Login Page**

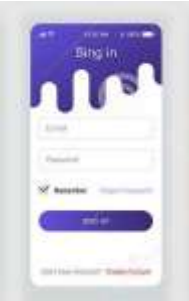


Figure-9: Login Page

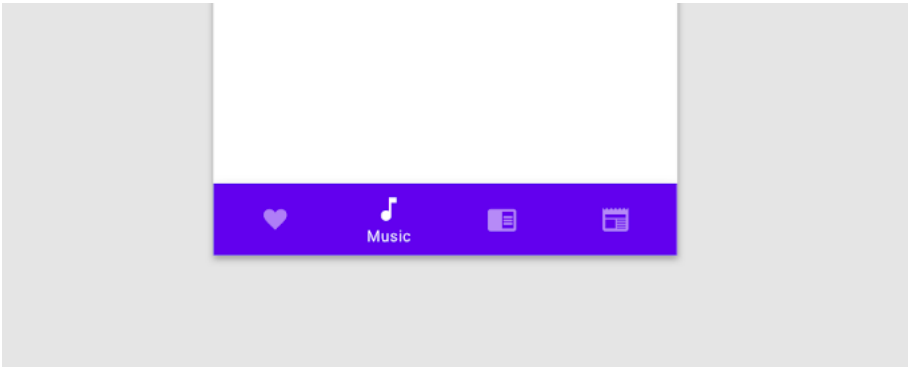
- **Forgot Password**

Figure-10: Forget Password

- **Home Page**

Figure-11: Home Page

- **Down Navigation**



- Song Play Page

CHAPTER - 6

TESTING

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery. It is very important to work the system successfully and achieve high quality of software. Testing include designing a series of test cases that have a high likelihood of finding errors by applying software-testing techniques.

System testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. The system should be checked logically. Validations and cross checks should be there. Avoid duplications of record that cause redundancy of data.

In other Words, Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

The Android framework includes an integrated testing framework that helps you test all aspects of your application and the SDK tools include tools for setting up and running test applications. Whether you are working in Eclipse with ADT or working from the command line, the SDK tools help you set up and run your tests within an emulator or the device you are targeting.

There are different types of testing some of them are listed below:

6.1 Installation Testing:

There are two types of apps on an Android device i.e., Pre-installed applications and the applications which are installed later by the user.

For both of the above, installation testing is carried out by our teammates. It is ensuring smooth installation of the application without ending up in errors, partial installation etc.

6.2 Unit Testing

It focuses on smallest unit of software design. In this we test an individual unit or groups of inter related units. It is often done by programmer by using sample input and observing its corresponding outputs. In this testing technique we are primarily focuses on

- Loop methods and function is working fine or not.
- Misunderstood or incorrect Arithmetic precedence
- Incorrect Initialization

Unit Testing of the app:

Test cases	**Description**	**Expected Outcome**	**Result**
1	Start Page – Launch Screen	Should display splash screen with animated text	Pass
2	Register Screen	Should display register activity where you need to fill the required details	Pass

3	Login Screen	Should display login screen And ask for your credentials.	Pass
4	Forget Password	Should Receive mail to reset the password	Pass
5	FAQ	Should display the Frequently asked question	Pass
6	View Favorites	Should display the Favourite activity	Pass

7	Profile	Should display the information that you have entered	Pass
8	View Songs	View Home Page	Pass
9	Search bar	Should give the details of the books that you have searched	Pass
10	View Song Description page	Should show the information of the displayed book	Pass

11	Add to favourites	Should add the selected book into the favourites	Pass
12	Clear Favourites	Should remove the selected book from the favourites	Pass
13	Preview	Should display the preview of the book	Pass
14	Logout	Sign out you from the app	Pass

Table 1: Unit Testing of Bookopedia

6.3 User Testing

User testing is the process through which the interface and functions of a website, app, product, or service are tested by real users who perform specific tasks in realistic conditions. The purpose of this process is to evaluate the usability of that website or app and to decide whether the product is ready to be launched for real users.

This app was tested by our team mates and friends who are using different mobile phones (and having different android version) also tested on different emulator to check its performance and it seems to be working fine and users of this app are satisfied with the facilities and performance of the app and like the way how the app is worked.

6.4 Performance Testing

In this type of testing we have checked the performances of our application under some peculiar conditions are checked. Those conditions include:

- Low memory in the device.
- The battery in extremely at a low level.
- Poor/Bad network reception.

Performance is basically tested from 2 ends, application end, and the application server end. Our app is also performing well in this phase of testing as well. And we are getting positive feedback from user of our app.

6.5 Compability Testing

This application was tested and used on different devices like LG G3, Google Nexus 4. The application worked fine and is stable. The application worked fine in portrait mode and there isn't any problem with compatibility.

On all types of testing (that we have performed above) our performing well on our app i.e. Bookopedia.

CHAPTER -7
CONCLUSION

Proposed SOCO App is an React Native application that will allow users to search for songs by name. This application takes in a user input and searches the React Native Tracker with the user input and gets a list of songs based on the users search query. Search result screen will contain a list of songs with following details: Singer of the Song,lyricist,music composer and releasing date. To get the information of the particular song user can click upon the song from the list and then willbe taken to the new tab where description and other information related to the song will be available. Users can also add the song to the favourites.

This application has wide range of scope in the upcoming era. It is impossible to arrange the hard copies of every book so this type of application can reduce the barrier to get knowledge at any place in a cost effective, productive way. For students who are interested in learning online can use this application and keep all the books they want to learn from at one place (in favourites section) and can create their own personal E-library. Even individual book stores can have this system of book apps promoting their brand name as Digital Marketing and can gain number of customers.

REFERENCES

1. **Introduction to Android:**

   http://developer.android.com/guide/index.html.

2. ***Android API:***

   http://developer.android.com/reference/packages.html

3. ***Android User Interfaces:***

   http://developer.android.com/guide/topics/ui/index.html

4. ***Layout:***

   http://developer.android.com/guide/topics/ui/declaring-layout.html

5. ***Android Training:***

   http://developer.android.com/training/index.html.

6. **Internshala:**

   https://trainings.internshala.com/

7. **Android developer Guide:**

   https://developer.android.com/

8. **For rectifying the error :**

   https://stackoverflow.com/