

**AMITY UNIVERSITY**

-------------------Maharashtra-------------------

**Project Report on**

# Banking Application

Submitted to

**Amity Institute of Information Technology**

**In partial fulfillment of the requirements for the award of the degree of**

BCA

**Submitted To Submitted By**

Dr. Preeti Gupta Prathmesh Patil

A71004819033

BCA SEM 6

### DECLARATION BY STUDENT

I **Prathmesh Patil** student of BCA hereby declare that the Project titled **“Banking Application”** which is submitted by me to Dr. Preeti Gupta, Amity Institute of Information Technology, Amity University Maharashtra, Mumbai, in partial fulfillment of requirement for the award of the degree of BCA, has not been previously formed the basis for the award of any degree, diploma or other similar title or recognition.

The Author attests that permission has been obtained for the use of any copy righted material appearing in the Dissertation / Project report other than brief excerpts requiring only proper acknowledgement in scholarly writing and all such use is acknowledged.

Place: - Mumbai

Prathmesh Patil

A71004819033

Logo

Description automatically generated

#### GUIDE CERTIFICATE

I hereby certify that the Seminar Report by **Prathmesh Patil**, student of **BCA Sem 6 A71004819033 Banking Application** which is submitted to **Amity Institute of Information Technology, Amity University Maharashtra**, Mumbai in partial fulfillment of requirement for the award of the degree of **BCA** is an original contribution with existing knowledge and faithful record of work carried out by him under my guidance and supervision and to the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Mumbai Dr. Preeti Gupta

Date: Associate Professor

Amity Institute of Information Technology

AUM, Mumbai

Logo

Description automatically generated

#### ABSTRACT

The traditional method of preserving a user's details in a bank was to input and record them. Every time the user has to make a transaction, he or she must go to the bank and complete the relevant steps, which may not always be viable. It might be a difficult task for both users and lenders. The project provides a practical grasp of the banking system and the actions carried out by various supply chain roles.

### ACKNOWLEDGEMENT

It is high privilege for me to express my deep sense of gratitude to those entire faculty Members who helped me in the completion of the project, specially my internal guide Dr. Preeti Gupta who was always there at hour of need.

My special thanks to all other faculty members, Batch mate & seniors of Amity Institute of Information Technology, Amity University,Maharashtra for helping me in the completion of project work and its report submission.

**Prathmesh Patil**

A71004819033

### TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Contents** | **Page No.** |
| 1 | DECLARATION BY STUDENT | 1 |
| 2 | GUIDE CERTIFICATE | 2 |
| 3 | ABSTRACT | 3 |
| 4 | ACKNOWLEDGEMENT | 4 |
| 5 | LIST OF FIGURES | 6 |
| Chapter-1 | Introduction | 7 |
|  | 1.2 Motivation |  |
|  | 1.3 Objectives |  |
| Chapter 2 | Proposed System | 8 |
| Chapter 3 | System Analysis | 9 |
| Chapter 4 | Information Gathering | 10 |
| Chapter 5 | Architecture | 11-13 |
| Chapter 6 | Implementation | 14-15 |
| Chapter 7 | Testing | 16-23 |
| Chapter 8 | Conclusion and Future Scope | 24 |
|  | References | 25 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Title of the Table** | **Page No.** |
| 5.1 | Data flow diagram | 13 |
| 5.2 | Activity diagram | 14 |
| 7.1 | Login screen | 16 |
| 7.2 | Providing credential | 17 |
| 7.3 | Register screen | 17 |
| 7.4 | Log-in screen | 18 |
| 7.5 | Home screen | 18 |
| 7.6 | View Users screen | 19 |
| 7.7 | User details | 19 |
| 7.8 | Sending money | 20 |
| 7.9 | Checking balance | 20 |
| 7.10 | Entering sufficient amount | 21 |
| 7.11 | Selecting user | 21 |
| 7.12 | Successful transaction | 22 |
| 7.13 | View transaction record | 22 |
| 7.14 | Cancelling transaction | 23 |
| 7.15 | Cancelled transaction record | 23 |

**CHAPTER 1:- INTRODUCTION**

* 1. **Motivation:-**

Looking forward to other banking applications, the idea of the project is come through by seeing how other applications works when comes to transaction between user to user and maintaining records of those transaction. People nowadays are like more comfortable with going hand-to-hand cash method only for which they have too keep a record of it. Banking apps like Google pay , Phone pay, Paytm and others are having this main feature of easy simple transactions and keeping records of every transactions of users which have bring my attention and becomes the main motive of this project.

* 1. **Objective:-**

The objective of this project is to develop a mobile (android) application where user have to go through a authenticate login process, user can do transactions to other users, this mobile application will maintain the records of those transactions which are successful and also those transactions which are cancelled, user can view other users profile. This project will majorly focusing on the transactions of the users.

**CHAPTER 2:-PROPOSED SYSTEM**

The proposed system will provide following features:

* Every time user uses this app he/she has to go with login/sign-up procedure, if user is new to application then he/she have to registered themselves first.
* It provides user where he/she can view other users profile.
* It gives user a freedom of choice where user can select other users during the transactions.
* It provides a user to cancel the transaction, if user by mistakenly select the wrong user to transfer money.
* It keeps well maintained records of every time when a transaction happens and also maintain records of those transactions which are cancelled.

**CHAPTER 3:-SYSTEM ANALYSIS**

In the system analysis of the project, this project consist two analysis :

1. Functional analysis :-

When user access this application he/she will discover UI buttons, each buttons consist of some functions which definitely going to be useful for user by just clicking on it. The list of buttons are as follows,

* Login button :- This button will help user to get logged-in the application and directed to the home screen of the application.
* Sign-up button :- This button will help user to get registered to the application.
* View user profile button:- This button will help user to discover other users of this app and can view some details of the other users.
* Make payment button :- This button will help user to do transactions with other users.
* Recent transaction button :- This button will give user a details of transaction which users has done in the past.
* Logout button :- This button will help user to get log out from his account and will direct to the login screen.
* Exit button :- This button will help user to get log out from his account and get the application closed.

1. Non functional analysis :-

Keeping that in the mind that whenever the user access this application it may possible that the user will face some kind of non function activities such as crashing of the application, some of featured function may not work properly or UI lagging. For this kind of problems I have worked on the solution which as follows,

* Keeping simple and basic UI for the user to avoid getting lagged while accessing the application.
* At a time one activity will be performing as per how user access the featured function of the application which not only helping to avoid the crashing of the application but also get to know if the features of the application works properly or not.

**CHAPTER 4:-INFORMATION GATHERING**

As in this section the information which I have gather about my project which is a banking application which have a basic objective and this banking application is build as a android application which will be available on android phones. Keeping in the mind that user maybe using a low specification phones. Some of the users also using phones of lower memory ram, or maybe the android version is also be lowest version. Therefore looking to the facts I have made the app which would be easily run on android version 7.0 Nougat with minimum 1.00 GB ram. Looking to the app crashing issues or UI lagging of the app in the phones having low specification I have also taken care of it and made this project.

**CHAPTER 5:-ARCHITECTURE**

Coming to the architecture section, this section defines the architecture of the project in details where it shows how project structure is and working of it. Below is the level 1 data flow diagram (DFD) which shows how data passing through the entire project.

Diagram, schematic

Description automatically generated

* 1. *Data flow diagram*

As in the fig (1.1), As it is a level 1 Data Flow Diagram (DFD) it shows how the data passing through processes which are much important in the project. The user have to provide credentials to the login system process where user will be seen if he/she is authenticate it or not then the user will be directed to the home screen interaction where user have to select the processes like View users, Make payment, Recent transaction. Each process has it’s own functionality which will provide result to the user.

For the better understanding of this processes where the user data is passed fig (1.2) Activity diagram will show you how this processes works in details. The activity diagram is used to demonstrate the flow of control within the system rather than the implementation. It models the concurrent and sequential activities. The activity diagram helps in envisioning the workflow from one activity to another. It put emphasis on the condition of flow and the order in which it occurs.

Diagram

Description automatically generated

* 1. *Activity diagram*

In the figure (1.2), the activity diagram has start point of opening of the app after that the user has to do login with login credentials, if user is new then registration needs to be done after that user have to login with the registered credentials which save in the database. After login the user gets to select processes as there are five processes which are view user profile, make payment , recent transaction , logout and exit. The view user process will direct the user to select which user profile he/she want to view details.

In the make payment process the user have to select the user then have to enter the amount there will check if user have enough amount balanced in his/her account if not then user will get message and if yes then user will get to select user then there will also check if user wants to cancel the transaction or not if not the transaction process will complete. Then the recent transaction process will invoke as their has been a transaction so record must be maintain. After coming to the logout process it will direct the user to the giving login credential and last the exit process will complete the activity diagram which will end the processes.

**CHAPTER 6:-IMPLEMENTATION**

Keeping in mind that the project would be building from the scratch and for that below is the list of project’s software and hardware requirement which as follows,

1. Software requirement :-

* Android Studio Bumblebee 2021.1.1
* Java 17.0.2 LTS
* SQLite Database
* Microsoft Windows 10 Pro

1. Hardware requirement :-

* Intel(R) Core(TM) i3-9100F CPU @ 3.60GHz, 3600 Mhz, 4 Core(s), 4 Logical Processor(s)
* 8.00 GB Ram
* 1 TB HDD storage

Considering the above list of project’s software requirement the platform which I have used is android studio bumblebee 2021.1.1, for making the project the programming language which is used is java which is commonly used to build api’s, apps and much more. As the project is based on banking application system there is a necessary to use the database to manage data for that SQLite database is used. SQLite is an open-source relational database i.e. used to perform database operations on android devices such as storing, manipulating or retrieving persistent data from the database.

Till now the software and hardware requirement of the project is done, now how the project works will actual explain in detail. So the project starts with the backend part where first I have created a packaged called Data then I have created two java classes transaction and user under the Data packaged. In both the classes I have declared some private static variables which will be use in further also both the classes have getters and setters method [1] which will return some values. After this I have created a packaged called DB where five classes (Loginhelper, TransactionContract, TransactionHelper, UserContract) [2] are created which having the databases functions and methods are going to be having most of the times lot of the uses of it.

This will help to save data in the databases and use it in further requirement. With help of cursors [4] in those contracts it help not only to read the data but also helps to fetch those data from the databases. After the DB package I have created ListAdapters package in which I have created three classes (SendToUserAdapter, TransacationHistoryAdapter, UserListAdapter) those three classes contain array adapter [4] which are used to populate data and feed to user in the form of list where user can view and select other user those array adapters are used with recycler view [5]. It helps to create list dynamically with array adapter. After this I have created UI package where there are list of activities are created which are most important in the project each activities are linked with other activities behind which each activity has it’s own functionality. Each activity consist java class and xml file where layout of the app are design and for accessing those design it linked with java classes where main functions are implemented.

**CHAPTER 7:-TESTING**

Graphical user interface, text, application, email

Description automatically generated

*7.1 Login screen*

In the figure (7.1) this is the login screen when user opens the apps.

Graphical user interface, text, application, email

Description automatically generated

*7.2 Providing Credential*

In figure (7.2), if user provide unregistered invalid credential it will popup with message.

Graphical user interface, text, application, email

Description automatically generated

*7.3 Register screen*

In figure (7.3), when user wants to register then user have to provide details which will be stored in the database

Graphical user interface, text, application, email

Description automatically generated

*7.4 Log-in screen*

In figure (7.4) after user register himself then user have to login with the registered details which provided.

Diagram

Description automatically generated

*7.5 Home screen*

In figure (7.5), when user gets logged-in he/she will direct to the home screen. This the user interaction screen where user can use this various options.

Graphical user interface, application

Description automatically generated

*7.6 View Users screen*

In figure (7.6) when user clicks on the view users it will direct to this screen where user can scroll down and select user and view details of other user which is shown in the below figure (7.7).

Graphical user interface

Description automatically generated

*7.7 User Details*

Graphical user interface, application

Description automatically generated

*7.8 Sending money*

In figure (7.8) when user select transfer money he/she has to enter the amount if the amount is greater than his current balanced and click on the send then it will show error in the form of alert box which is shown in the below figure (7.9).

Graphical user interface, text, application

Description automatically generated

*7.9 Checking balance*

Graphical user interface, text, application

Description automatically generated

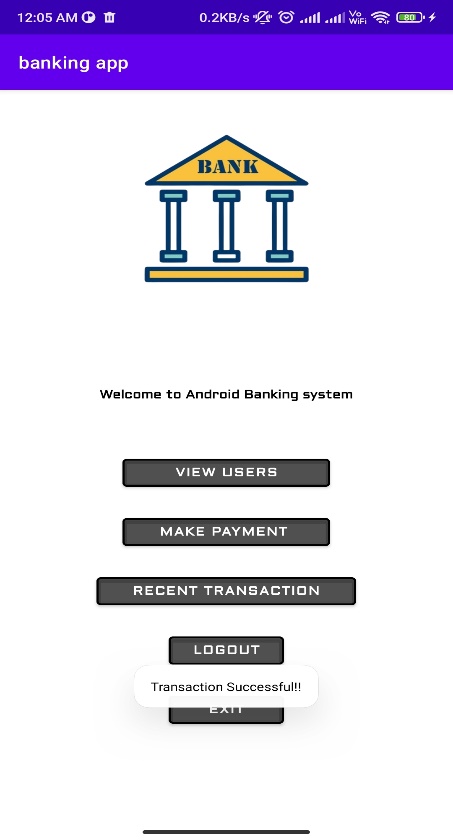
*7.10 Entering sufficient amount*

Graphical user interface, application

Description automatically generated

*7.11 Selecting user*

In figure (7.10) & (7.11) when user enter the sufficient amount and click on the send he/she will direct to the select user screen where user have select the receiver.



*7.12 Successful transaction*

A picture containing background pattern

Description automatically generated

*7.13 Viewing transaction record*

In figure (7.12) & (7.13) when user gets transaction successful the user will get a message and then if user select on recent transaction it will direct to screen where recent transaction details.

Graphical user interface, text, application

Description automatically generated

*7.14 Cancelling transaction*

Graphical user interface, application

Description automatically generated

*7.15 Cancelled transaction record*

In the figure (7.14) & (7.15) when user entered amount and by accidently select wrong user to transfer money, the user can cancel the transaction and the cancelled transaction can also get maintain.

**CHAPTER 8:-CONCLUSION & FUTURE SCOPE**

By seeing the various outputs of the application I can see the app is perfectly working with no lagging UI and doesn’t see any crashing issue of the application. As the application is made for compatible in lower specification android phones the user can easily use this application. As this application is still require more fascinating features where Google pay, Phone Pay, Paytm have. It will take time to get adapt this features. As in the future I can say there will be no longer to use cash everywhere as technology is getting evolve faster it will definitely replace the wallet system and in the replacement of the wallet these online payment apps will be taken and their would be no user remain who will use cash as a payment method and have to keep record of it.

**REFERENCES**

1. <https://www.geeksforgeeks.org/how-to-create-constructor-getter-setter-methods-and-new-activity-in-android-studio-using-shortcuts/>
2. <https://developer.android.com/training/data-storage/sqlite#java>
3. <https://developer.android.com/reference/android/database/Cursor>
4. <https://developer.android.com/reference/android/widget/ArrayAdapter>
5. <https://developer.android.com/reference/androidx/recyclerview/widget/RecyclerView.ViewHolder>