STUDENT BUDGETING APP

Submitted to the
Department of Master of Computer Applications in partial fulfilment of the requirements for the Mini Project in

Mobile Application Development - MCAE52

by

Komal S Kallanagoudar (1MS22MC016)

Pramod M T (1MS22MC028)

Under the Guidance of

Ms. Sushitha S

Assistant Professor

Department of MCA

MSRIT

Ramaiah Institute of Technology

(Autonomous Institute, Affiliated to VTU)

Bangalore – 54

2022-2024

RAMAIAH
Institute of Technology

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project entitled "**STUDENT BUDGETING APP**" is carried out by

Student Name	USN
1) KOMAL S KALLANAGOUDAR	1MS22MC016
2) PRAMOD M T	1MS22MC028

students of 3rd semester, in partial fulfillment for the Mini Project in Mobile Application Development- MCAE52, during the academic year 2022-2024.

Guide **Head of the Department**

(Ms. Sushitha S) **(Dr. Monica R. Mundada)**

Name of Examiners **Signature with Date**

1.

2.

DECLARATION

I hereby declare that the project report entitled "STUDENT BUDGETING SYSTEM" based on study undertaken by me, towards the partial fulfilment for the Mini Project (Mobile Application Development-MCAE52) carried out during the 3rd semester, has been compiled purely from the academic point of view and is, therefore, presented in a true and sincere academic spirit. Contents of this report are based on my original study and findings in relation there to are neither copied nor manipulated from other reports or similar documents, either in part or in full, and it has not been submitted earlier to any University/College/Academic institution for the award of any Degree/Diploma/Fellowship or similar titles or prizes and that the work has not been published in any specific or popular magazines.

Place: Bangalore
Date:

Komal S K
1MS22MC016

Pramod M T
1MS22MC028

Mini-Project Report Evaluation

Subject: Mobile Application Development
Subject Code: MCAE52 **Maximum Mark: 10**

	Project Report(3 mark)	Description of Concepts and Technical details (4 mark)	Conclusion and Discussion (3 mark)	Score
Excellent	• Project report is according to the specified format • There are no grammatical errors fond **(3 mark)**	• Complete explanation of the key concepts • Strong description of the technical requirements of the project **(4 mark)**	• Results are presented in very appropriate manner • Project work is well summarized and concluded • Future extensions in the project are well specified **(3 mark)**	
Good	• Project report is according to the specified format • Minor grammatical errors **(2 mark)**	• Complete explanation of the key concepts • In-sufficient description of the technical requirements of the project **(3 mark)**	• Results are presented in good manner • Project work summary and conclusion not very appropriate • Future extensions in the project are specified **(2 mark)**	
Poor	• Project report not prepared according to the specified format • Major grammatical errors **(1 mark)**	• Inapproiate explanation of the key concepts • Poor description of the technical requirements of the project **(2 mark)**	• Results are not presented properly • Project work is not summarized and concluded • Future extensions in the project are not specified **(1 mark)**	

Mini-Project Report Evaluation

Subject: Mobile Application Development
Subject Code: MCAE52 **Maximum Mark: 10**

	Project Report(3 mark)	Description of Concepts and Technical details (4 mark)	Conclusion and Discussion (3 mark)	Score
Excellent	• Project report is according to the specified format • There are no grammatical errors fond **(3 mark)**	• Complete explanation of the key concepts • Strong description of the technical requirements of the project **(4 mark)**	• Results are presented in very appropriate manner • Project work is well summarized and concluded • Future extensions in the project are well specified **(3 mark)**	
Good	• Project report is according to the specified format • Minor grammatical errors **(2 mark)**	• Complete explanation of the key concepts • In-sufficient description of the technical requirements of the project **(3 mark)**	• Results are presented in good manner • Project work summary and conclusion not very appropriate • Future extensions in the project are specified **(2 mark)**	
Poor	• Project report not prepared according to the specified format • Major grammatical errors **(1 mark)**	• Inapproiate explanation of the key concepts • Poor description of the technical requirements of the project **(2 mark)**	• Results are not presented properly • Project work is not summarized and concluded • Future extensions in the project are not specified **(1 mark)**	

ABSTRACT

Key Components of a Student Budgeting System Application are Budget Planning, Expense Tracking, Savings and Investments. The app enables students to create comprehensive budget by listing all income sources and it helps students allocate their resources effectively. Tracking expenses is a crucial aspect of managing a student budget. Students can log their daily or monthly expenses, categorize them and monitor their spending patterns. The application can provide information and tools for students to explore expenses and savings and also empowers students to take control of their finances, make informed decisions, and develop responsible money management skills.

Student Budgeting System app is an interesting concept as it helps to organize budget and save money. Instead of maintaining and calculating Expenses on paper, this system will help store details of expenses. Student can see where they spend their money and it helps to cut down unnecessary expenses. The main objective of this app is to reduce the burden on group of students leaving together in a home/flat and spending the money on food, home accessories, paying rent, paying bills and other expenses.

TABLE OF CONTENTS

1. INTRODUCTION

Key Components of a Student Budgeting System Application are Budget Planning, Expense Tracking, Savings and Investments. The app enables students to create comprehensive budget by listing all income sources and it helps students allocate their resources effectively. Tracking expenses is a crucial aspect of managing a student budget. Students can log their daily or monthly expenses, categorize them and monitor their spending patterns. The application can provide information and tools for students to explore expenses and savings and also empowers students to take control of their finances, make informed decisions, and develop responsible money management skills.

Student Budgeting System app is an interesting concept as it helps to organize budget and save money. Instead of maintaining and calculating Expenses on paper, this system will help store details of expenses. Student can see where they spend their money and it helps to cut down unnecessary expenses. The main objective of this app is to reduce the burden on group of students leaving together in a home/flat and spending the money on food, home accessories, paying rent, paying bills and other expenses.

1.1. Problem Definition

Managing finances as a student can be challenging, especially when you're juggling tuition fees, textbooks, rent, groceries, and social activities. A Student Budgeting System android application is a valuable tool that helps students plan, track, and optimize their finances during their academic journey. This app is designed to provide students with the necessary tools and strategies to make informed financial decisions and achieve their financial goals while pursuing education.

2. FUNCTIONAL REQUIREMENTS

2.1. Software Requirements

- User Authentication: Users should be able to create an account or log in securely using email, social media accounts, or other authentication methods.
- Profile Setup: Users should be able to set up their profile with basic information such as name, email, and student status.
- Expense Tracking: Users should be able to track their expenses by categorizing them (e.g., food, transportation, entertainment) and adding details like date, amount, and description
- Data Security: The app should prioritize user data security, employing encryption techniques to protect sensitive information such as financial transactions and personal details.
- User Authentication with Firebase Authentication: Utilize Firebase Authentication to allow users to sign up, log in, and manage their accounts securely.
- Real-time Database with Firebase Realtime Database or Firestore: Implement Firebase Realtime Database or Firestore to store and sync user data in real-time, including expense transactions, budget information, income sources, and user profiles.

2.2. Hardware Requirements

- Operating System: Android Studio runs on Windows, macOS, and Linux. Ensure that your hardware supports the operating system you intend to use.
- Processor: A multi-core processor with a clock speed of at least 2 GHz is recommended for faster compilation and emulation.
- Memory (RAM): Android Studio can be memory-intensive, especially when running alongside other development tools. A minimum of 8 GB RAM is recommended, but 16 GB or more is preferable for smoother performance, especially when dealing with larger projects.
- Storage: Android Studio itself doesn't require much disk space, but you'll need ample space for project files, SDKs, emulators, and other development tools. At least 4 GB of free disk space is recommended, with SSDs being preferable for faster read/write speeds.

3. Modules Descriptions

User Registration and Login:

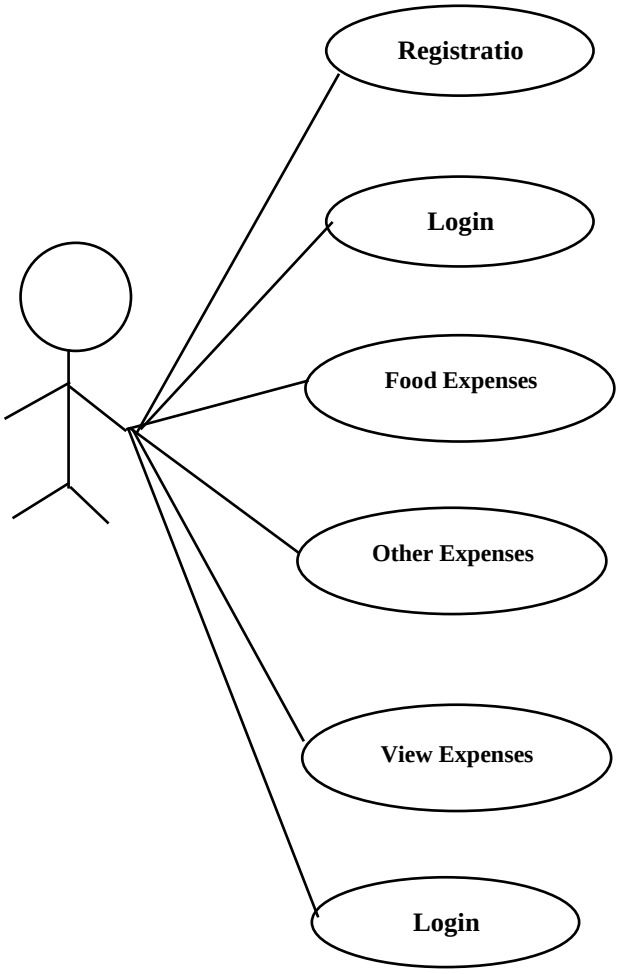
Allows users to create new accounts by providing necessary information. Enables existing users to securely log in to the system using their credentials. Profile Creation and Editing Permits users to create their profiles upon registration. Provides functionality to edit profile details such as name, email, and contact information. Password Management Offers options for users to reset their passwords if forgotten, following a secure verification process. Allows users to change their passwords for security purposes. User Roles and Permissions Management Implements role-based access control, distinguishing between different user roles such as admin and regular user. Admins have additional privileges, such as managing other users and system settings.

Expense Tracking Module:

This module facilitates the tracking of user expenses and categorizes them based on predefined or custom categories. It encompasses the following features like Adding, Editing, and Deleting Expenses. Allows users to input new expenses, edit existing ones, and remove unnecessary entries from their expense history. Categorizing Expenses Provides users with the capability to categorize expenses according to predefined categories such as groceries, transportation, or entertainment. Offers the option to create custom categories to suit individual user needs. Automatic Expense Categorization Utilizes algorithms or user-defined rules to automatically categorize expenses based on transaction details, simplifying the process for users. Tracking of Recurring Expenses enables users to track expenses that occur regularly, such as monthly bills or subscriptions and offers functionalities to set up recurring expense entries for automatic inclusion in expense records.

4. DESIGN MODEL

Use Case Diagram



A use case diagram for a student budgeting app illustrates the various interactions between users (actors) and the system to achieve specific goals or tasks.

Actors:

User: Represents the primary actor interacting with the system, typically a student using the budgeting app.

Use Cases:

Login: Represents the user's action of logging into the system using their credentials.
Register: Represents the user's action of creating a new account within the system.
Manage Profile: Allows the user to view and update their profile information, such as name, email, and contact details.
Track Expenses: Represents the user's ability to record and categorize their expenses.

5. IMPLEMENTATION

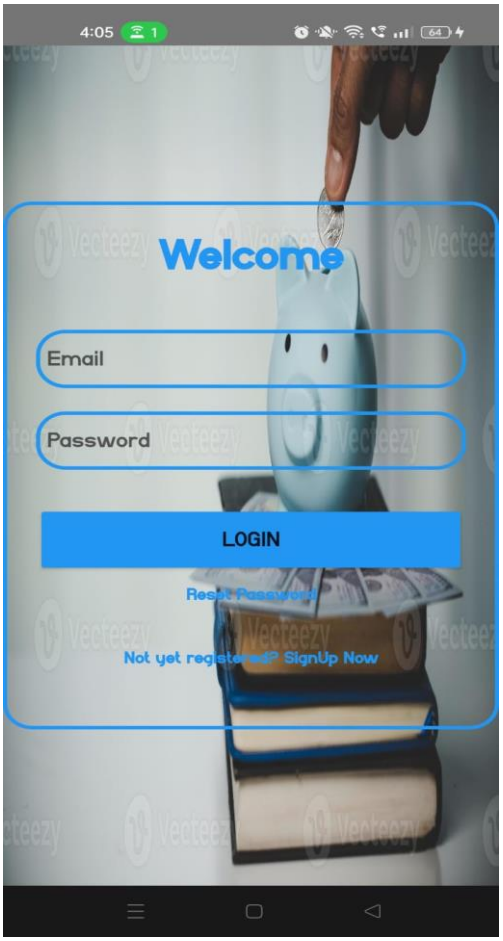
MainActivity.java

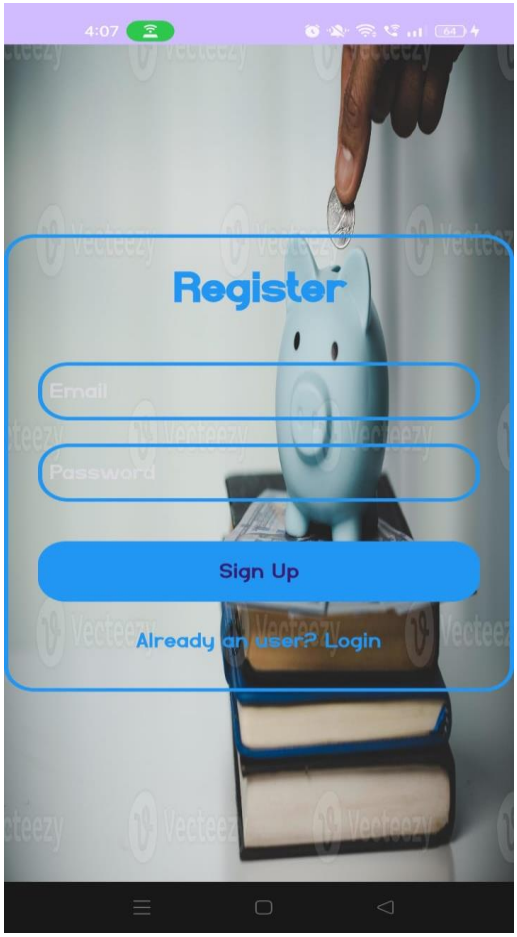
```
2 usages
private FirebaseAuth rAuth;
@SuppressLint("MissingInflatedId")
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    rAuth = FirebaseAuth.getInstance();
    rDialog = new ProgressDialog( context: this);
    login();
}
1 usage
private void login() {
    lemail = findViewById(R.id.lemail);
    lpassword = findViewById(R.id.lpassword);
    Button loginbtn = (Button) findViewById(R.id.llogin);
    TextView signin = (TextView) findViewById(R.id.lsignup);
    TextView reset = (TextView)findViewById(R.id.reset);

    loginbtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String email = lemail.getText().toString().trim();
            String password = lpassword.getText().toString().trim();

            if (TextUtils.isEmpty(email)) {
                lemail.setError("Email Requried");
                return;
            }
            if (TextUtils.isEmpty(password)) {
                lemail.setError("Password Requried");
                return;
```

Activity_main.xml

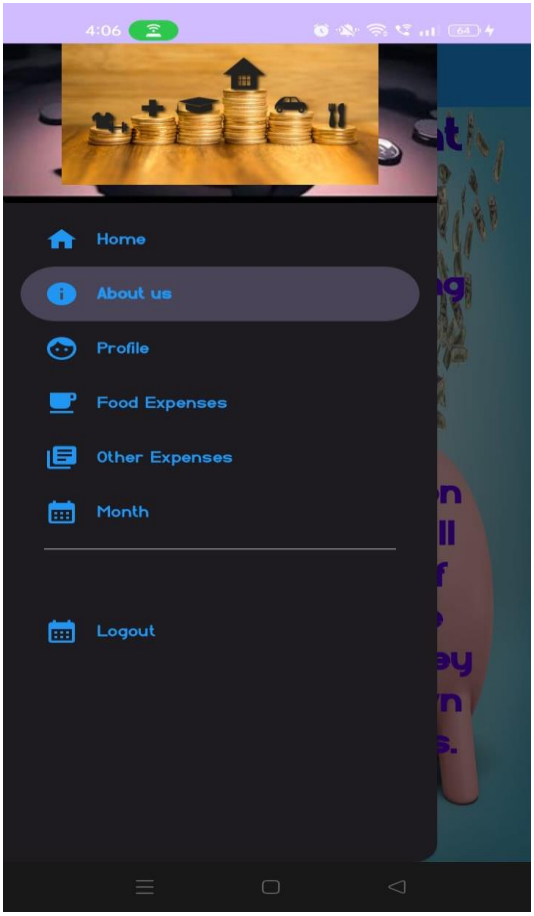
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:background="@drawable/last"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/custom_edit"
        android:orientation="vertical"
        android:gravity="center"
        android:padding="24dp">

        <TextView
            android:id="@+id/logintext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Welcome"
            android:textAlignment="center"
            android:textColor="@color/blue_200"
            android:textSize="36sp"
            android:textStyle="bold" />

        <EditText
```

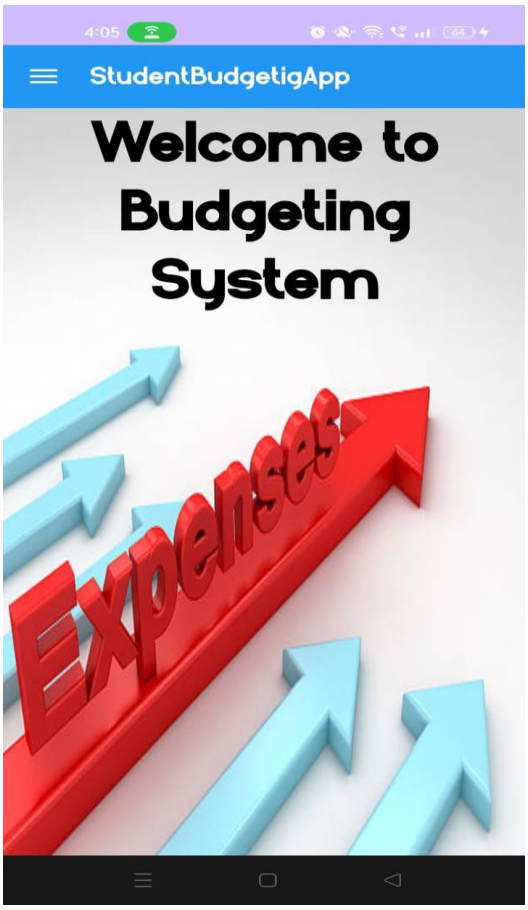
6. User Interfaces/Screenshots with Descriptions





Student login page by entering the name and password you can log in it redirect to home page. If a new user there is option for registration and also there is option for rest password. Registration form requires email, password. Clicking "Sign Up" submits and log you and redirect to the home page.





Navigation bar helps to navigate through the various functionalities available in the app which include enter the food expenses, other expenses and user can view monthly expenditure. When the user credentials are validated, the app redirects to the home page. From here user can navigate to other pages using navigation bar.

7. Conclusion and Scope for Further Enhancement

Conclusion

The Student Budgeting App is a vital resource for students, offering them a structured approach to managing expenses and developing essential financial literacy skills. By utilizing this system, students can allocate resources effectively, achieve academic and personal goals, and maintain financial stability throughout their educational journey. Through features such as expense tracking, budget planning, and goal setting, the system promotes financial responsibility and empowers students to make smart financial choices. Moreover, by instilling disciplined spending habits and providing insights into their financial behaviour, the system prepares students for long-term financial well-being beyond their academic years. Overall, the Student Budgeting App serves as a cornerstone in students' quest for financial independence and success.

Further Enhancement

The scope for future enhancement of the Student Budgeting System is vast, offering opportunities to further empower students in managing their finances effectively. Integration with financial institutions could provide real-time transaction data, while advanced data analysis techniques could offer deeper insights and personalized recommendations. Gamification elements and incentives could incentivize responsible spending, and a comprehensive library of financial education resources could enhance users' financial literacy. Collaborative budgeting features, mobile payment integration, accessibility improvements, and social engagement tools could broaden the system's usability and appeal. By focusing on these areas, the Student Budgeting System can continue to evolve as a versatile and indispensable tool for students navigating their financial journeys.