MONEY MATTERS: A PERSONAL FINANCE MANAGEMENT APP

ABSTRACT:

The Money Matters app is an Android-based personal finance management application designed to help users monitor their income, expenses, and budgets effectively. It provides a user-friendly interface for budgeting, transaction tracking, and financial insights, making it simpler for individuals to manage personal finances and achieve financial goals. Developed in Kotlin with the Room Database, Money Matters employs the MVVM architecture to maintain a clear separation of concerns, ensuring code maintainability and scalability. This report outlines the app's development process, system requirements, features, and future scope for enhancements.

INTRODUCTION:

Managing personal finances is crucial for achieving financial stability and meeting goals. The Money Matters app was developed to provide an accessible tool for individuals to track their spending, manage budgets, and set financial goals in a streamlined, efficient manner. This app serves as a comprehensive finance manager with an emphasis on ease of use, data security, and real-time financial tracking.

SYSTEM REQUIREMENTS:

- Operating System: Android-7 or later
- Ram: 1Gb (Minimum)
- Rom:20Mb (Minimum)
- Internet connection

TOOLS USED:

- Android Studio
- Firebase
- Kotlin

PROPOSED SYSTEM:

The Money Matters app is designed to help users track and manage their finances effortlessly. Key features include:

- **Expense Tracking**: Users can log daily expenses across multiple categories (food, rent, utilities, etc.).
- **Account Management**: Users can link multiple accounts (cash, bank accounts, credit cards) and view consolidated balances.
- **Budget Setting**: Allows users to set monthly/weekly budgets for each category and provides insights into their spending habits.
- **Progress Monitoring**: Offers visual indicators (charts, graphs) to show the progress of spending against the budget in real time.
- **Financial Overview**: Provides a summary of total income, expenditures, savings, and outstanding debts.
- **Notifications and Alerts**: Sends reminders for due dates on bills and alerts when nearing budget limits.

WORKING OF THE APP:

- **User Registration**: Users create an account using their email or third-party services.
- **Adding Accounts**: Users can input details about their financial accounts (e.g., bank, cash, credit card) to monitor their financial inflow and outflow.
- **Expense Tracking**: Users log their expenses daily by selecting a category, entering an amount, and optionally adding notes.
- **Budget Setting**: Users set budgets for each category, such as groceries, travel, or entertainment.
- **Progress Display**: Users can view graphical representations of their budget usage and account balances.
- **Alerts and Notifications**: The app sends notifications if users are close to exceeding their budget.
- **Data Sync and Backup**: Syncs data to the cloud (using Firebase) for data persistence and easy access from multiple devices.

VISION:

The vision for the Money Matters app is to empower individuals with a tool that simplifies personal finance management, making it easier for users to achieve financial control, track expenses, and stay within budgets. This app aims to provide not only a record of past transactions but also insights and analytics that help users make informed financial decisions.

CODE IMPLEMENTATION(sample code) :

LoginActivity

```
package com.example.expensestracker


import android.content.Context

import android.content.Intent

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.Image

import androidx.compose.foundation.layout.*

import androidx.compose.material.*

import androidx.compose.runtime.*

import androidx.compose.ui.Alignment

import androidx.compose.ui.Modifier

import androidx.compose.ui.graphics.Color

import androidx.compose.ui.layout.ContentScale

import androidx.compose.ui.res.painterResource

import androidx.compose.ui.text.font.FontFamily

import androidx.compose.ui.text.font.FontWeight
```

```
class LoginActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(this)
        setContent {
            ExpensesTrackerTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    LoginScreen(this, databaseHelper)
                }
            }
        }
    }
}
@Composable
fun LoginScreen(context: Context, databaseHelper: UserDatabaseHelper) {


    Image(
        painterResource(id = R.drawable.img_1), contentDescription = "",
        alpha =0.3F,
        contentScale = ContentScale.FillHeight,


        )
```

MainActivity

```
package com.example.expensestracker


import android.annotation.SuppressLint
import android.content.Intent
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.*
import androidx.compose.material.*
import androidx.compose.runtime.*
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier


class MainActivity : ComponentActivity() {
    @SuppressLint("UnusedMaterialScaffoldPaddingParameter")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            Scaffold(
                // in scaffold we are specifying top bar.
                bottomBar = {
                    // inside top bar we are specifying
                    // background color.
                    BottomAppBar(backgroundColor = Color(0xFFadbef4),
                        modifier = Modifier.height(80.dp),
                        // along with that we are specifying
                        // title for our top bar.
                        content = {

                            Spacer(modifier = Modifier.width(15.dp))

                            Button(
                                onClick                                                   =
{startActivity(Intent(applicationContext,AddExpensesActivity::class.java))},
                                colors = ButtonDefaults.buttonColors(backgroundColor = Color.White),
                                modifier = Modifier.size(height = 55.dp, width = 110.dp)
                            )
                            {
                                Text(
                                    text = "Add Expenses", color = Color.Black, fontSize = 14.sp,
                                    textAlign = TextAlign.Center
                                )
                            }

                            Spacer(modifier = Modifier.width(15.dp))

                            Button(
                                onClick = {
                                    startActivity(
                                        Intent(
                                            applicationContext,
                                            SetLimitActivity::class.java
                                        )
                                    )
                                }
```

RegisterActivity

```
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.layout.ContentScale
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.text.font.FontFamily
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.input.PasswordVisualTransformation
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import androidx.core.content.ContextCompat
import com.example.expensestracker.ui.theme.ExpensesTrackerTheme

class RegisterActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(this)
        setContent {
            ExpensesTrackerTheme {
                // A surface container using the 'background' color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {

                    RegistrationScreen(this,databaseHelper)
                }
            }
        }
    }
}

@Composable
fun RegistrationScreen(context: Context, databaseHelper: UserDatabaseHelper) {

    Image(
        painterResource(id = R.drawable.img_1), contentDescription = "",
        alpha =0.3F,
        contentScale = ContentScale.FillHeight,

        )

    var username by remember { mutableStateOf("") }
    var password by remember { mutableStateOf("") }
    var email by remember { mutableStateOf("") }
    var error by remember { mutableStateOf("") }

    Column(
        modifier = Modifier.fillMaxSize(),
        horizontalAlignment = Alignment.CenterHorizontally,
        verticalArrangement = Arrangement.Center
    )
```

SetLimitActivity

```
import androidx.compose.material.*
import androidx.compose.runtime.*
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.style.TextAlign
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import com.example.expensestracker.ui.theme.ExpensesTrackerTheme

class SetLimitActivity : ComponentActivity() {
    private lateinit var expenseDatabaseHelper: ExpenseDatabaseHelper
    @SuppressLint("UnusedMaterialScaffoldPaddingParameter")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        expenseDatabaseHelper = ExpenseDatabaseHelper(this)
        setContent {
            Scaffold(
                // in scaffold we are specifying top bar.
                bottomBar = {
                    // inside top bar we are specifying
                    // background color.
                    BottomAppBar(backgroundColor = Color(0xFFadbef4),
modifier = Modifier.height(80.dp),
// along with that we are specifying
// title for our top bar.
content = {

  Spacer(modifier = Modifier.width(15.dp))

  Button(
    onClick = {
      startActivity(
        Intent(
          applicationContext,
          AddExpensesActivity::class.java
        )
      )
    },
    colors = ButtonDefaults.buttonColors(backgroundColor = Color.White),
    modifier = Modifier.size(height = 55.dp, width = 110.dp)
  )
  {
    Text(
      text = "Add Expenses", color = Color.Black, fontSize = 14.sp,
      textAlign = TextAlign.Center
    )
  }

  Spacer(modifier = Modifier.width(15.dp))
```

```
                            Button(
                                onClick = {
                                    startActivity(
                                        Intent(
                                            applicationContext,
                                            SetLimitActivity::class.java
                                        )
                                    )
                                }
```

ViewRecordsActivity

```
import androidx.compose.material.*
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.style.TextAlign
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import com.example.expensestracker.ui.theme.ExpensesTrackerTheme

class ViewRecordsActivity : ComponentActivity() {
    private lateinit var itemsDatabaseHelper: ItemsDatabaseHelper
    @SuppressLint("UnusedMaterialScaffoldPaddingParameter", "SuspiciousIndentation")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        itemsDatabaseHelper = ItemsDatabaseHelper(this)
        setContent {
            Scaffold(
                // in scaffold we are specifying top bar.
                bottomBar = {
                    // inside top bar we are specifying
                    // background color.
                    BottomAppBar(backgroundColor = Color(0xFFadbef4),
                        modifier = Modifier.height(80.dp),
                        // along with that we are specifying
                        // title for our top bar.
                        content = {

                            Spacer(modifier = Modifier.width(15.dp))

                            Button(
                                onClick = {
                                    startActivity(
                                        Intent(
                                            applicationContext,
                                            AddExpensesActivity::class.java
                                        )
                                    )
                                },
                                colors = ButtonDefaults.buttonColors(backgroundColor = Color.White),
                                modifier = Modifier.size(height = 55.dp, width = 110.dp)
                            )
```

AddExpensesActivity

```
import androidx.compose.material.*
import androidx.compose.runtime.*
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.platform.LocalContext
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.style.TextAlign
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp

class AddExpensesActivity : ComponentActivity() {
  private lateinit var itemsDatabaseHelper: ItemsDatabaseHelper
  private lateinit var expenseDatabaseHelper: ExpenseDatabaseHelper
  @SuppressLint("UnusedMaterialScaffoldPaddingParameter")
  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    itemsDatabaseHelper = ItemsDatabaseHelper(this)
    expenseDatabaseHelper = ExpenseDatabaseHelper(this)
    setContent {
      Scaffold(
        // in scaffold we are specifying top bar.
        bottomBar = {
          // inside top bar we are specifying
          // background color.
          BottomAppBar(backgroundColor = Color(0xFFadbef4),
            modifier = Modifier.height(80.dp),
            // along with that we are specifying
            // title for our top bar.
            content = {
```

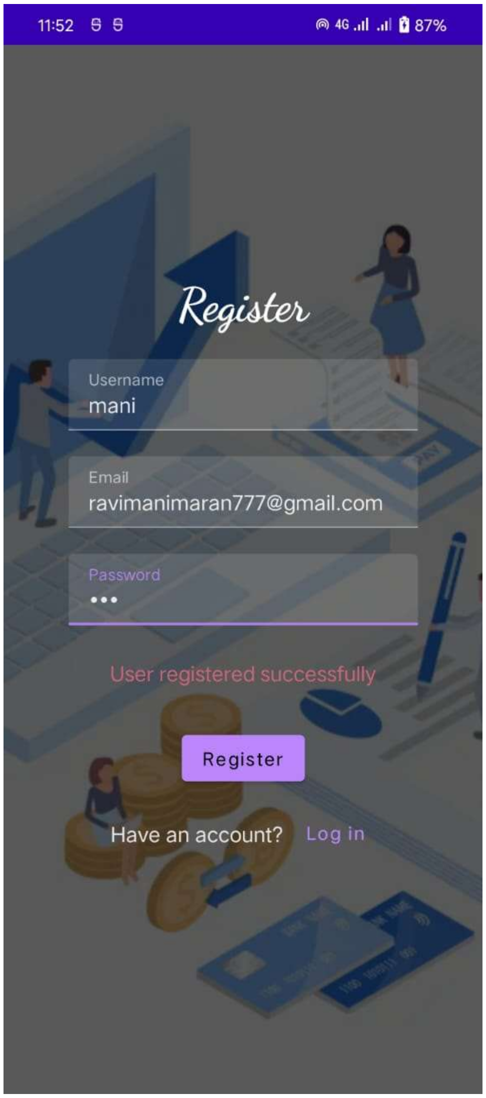
```
                            Spacer(modifier = Modifier.width(15.dp))

                            Button(
                                onClick =
{startActivity(Intent(applicationContext,AddExpensesActivity::class.java))},
                                colors = ButtonDefaults.buttonColors(backgroundColor = Color.White),
                                modifier = Modifier.size(height = 55.dp, width = 110.dp)
                            )
                            {
                                Text(
                                    text = "Add Expenses", color = Color.Black, fontSize = 14.sp,
                                    textAlign = TextAlign.Center
                                )
                            }

                            Spacer(modifier = Modifier.width(15.dp))

                            Button(
                                onClick = {
                                    startActivity(
                                        Intent(
                                            applicationContext,
                                            SetLimitActivity::class.java
                                        )
                                    )
                                }
```

OUTPUT:

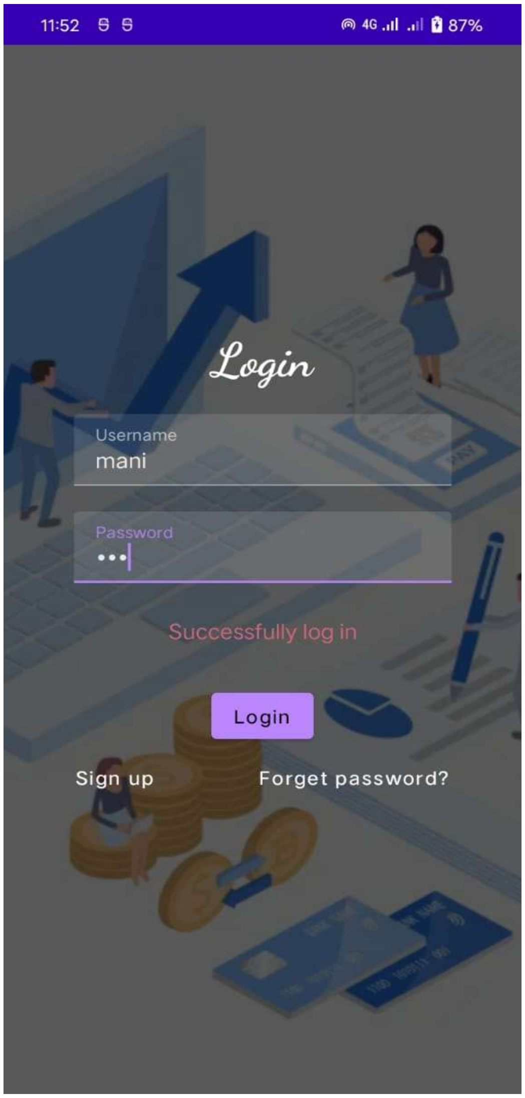


Username
mani

Email
ravimanimaran777@gmail.com

Password
•••

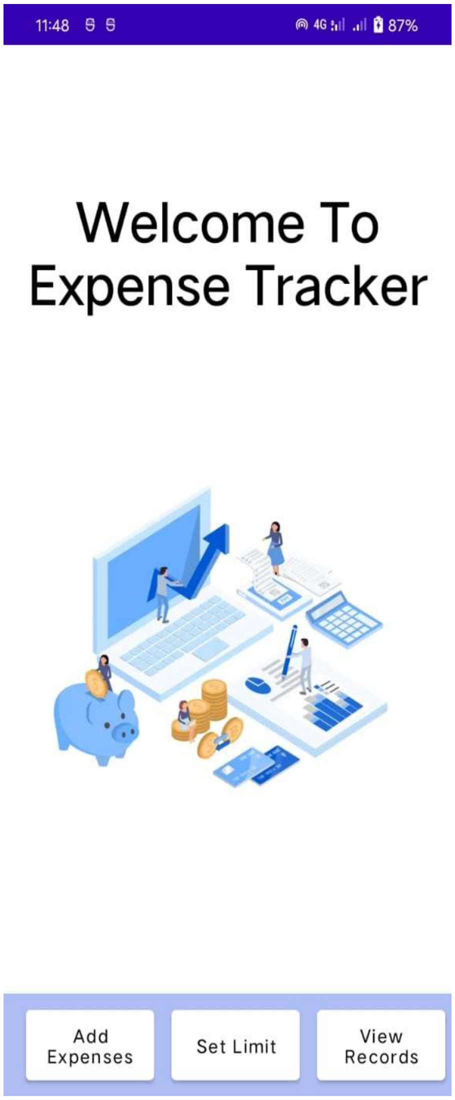
User registered successfully



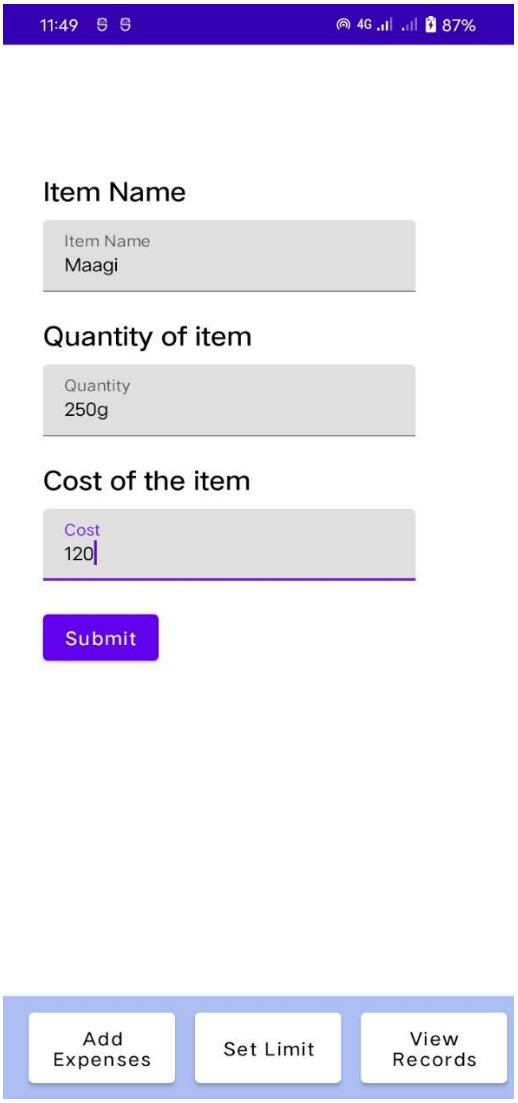
Username
mani

Password
•••

Successfully log in



Welcome To Expense Tracker



Item Name

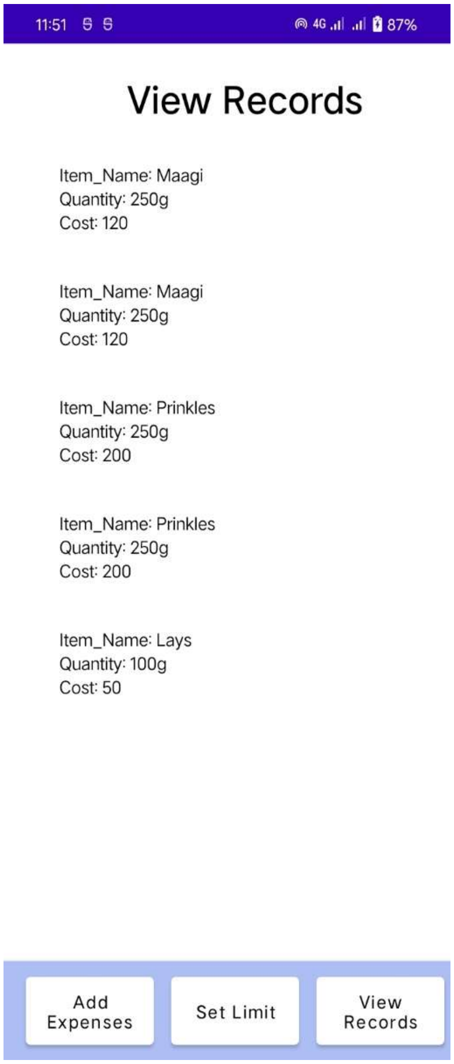
Item Name: Maagi

Quantity of item

Quantity: 250g

Cost of the item

Cost: 120



View Records

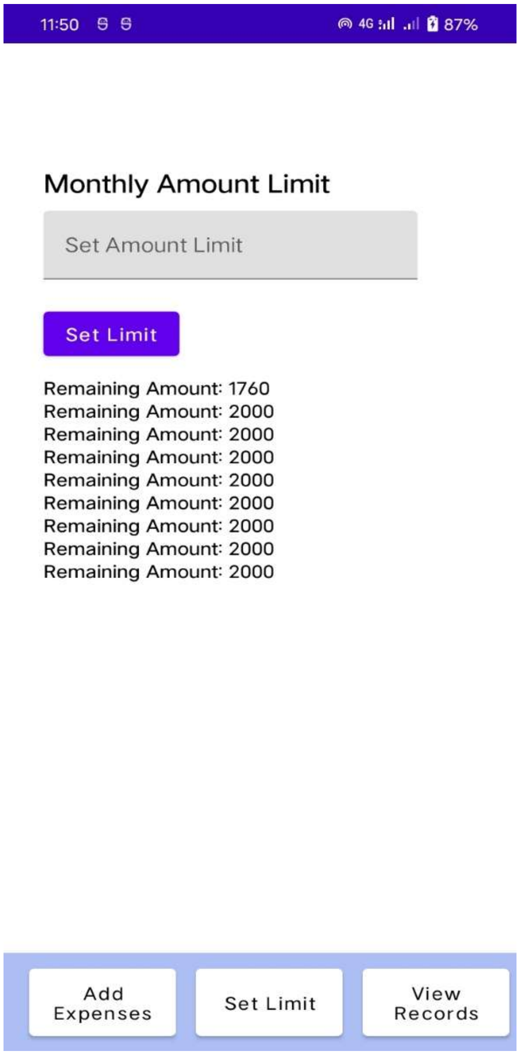
Item_Name: Maagi
Quantity: 250g
Cost: 120

Item_Name: Maagi
Quantity: 250g
Cost: 120

Item_Name: Prinkles
Quantity: 250g
Cost: 200

Item_Name: Prinkles
Quantity: 250g
Cost: 200

Item_Name: Lays
Quantity: 100g
Cost: 50



Monthly Amount Limit

```
Remaining Amount: 1760
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
Remaining Amount: 2000
```

CHALLENGES AND SOLUTIONS:

Accurate Data Categorization

- *Challenge*: Users may struggle to categorize expenses accurately.
- *Solution*: Use machine learning to suggest categories based on user history, improving ease and accuracy over time.

Data Privacy and Security

- *Challenge*: Ensuring financial data remains private and secure.
- *Solution*: Use Firebase's authentication and secure database rules, plus end-to-end encryption for added protection.

Real-Time Budget Tracking

- *Challenge*: The app must update budgets instantly as expenses are added.
- *Solution*: Optimize backend processes and use data caching for real-time performance.

User Retention and Engagement

- *Challenge*: Users may lose interest without consistent motivation.
- *Solution*: Include insights, notifications, and gamification elements to encourage active use.

FUTURE ENHANCEMENTS:

- **AI-Driven Insights and Suggestions**: Introduce AI algorithms that analyze spending patterns and offer insights or personalized savings tips.
- **Bill Payment and Reminders**: Add a bill payment feature that enables users to link accounts and pay bills directly through the app.
- **Expense Prediction**: Use historical spending data to predict future expenses and alert users in advance of large, upcoming expenses.
- **Financial Goal Setting**: Allow users to set long-term financial goals (e.g., saving for a vacation) and provide progress tracking towards those goals.

CONCLUSION:

Money Matters is a robust, user-friendly personal finance app developed for Android. It simplifies the process of tracking income and expenses, setting budgets, and monitoring financial goals. Through the use of Kotlin, Room Database, and MVVM architecture, the app offers an efficient and responsive user experience. The app fulfills its objective of helping users take control of their finances and serves as a foundation for further enhancements.