A PERSONAL FINANCE MANAGEMENT APP

INTRODUCTION :

A personal finance management app is a digital tool that helps individuals track their income, expenses, investments, and savings. These apps provide users with an overview of their financial status, helping them make informed decisions about how to manage their money.

Personal finance management apps typically allow users to link their bank accounts, credit cards, and other financial accounts to the app, providing them with real-time updates on their financial transactions. Users can also set budgets, track their spending habits, and receive alerts when they are close to reaching their budget limits or have upcoming bill payments due.

Some personal finance management apps also offer investment tracking and planning tools, enabling users to monitor their investment portfolios and make informed decisions about buying or selling assets. Additionally, these apps can help users save money by providing tips and recommendations on how to cut costs and reduce unnecessary expenses.

Overall, personal finance management apps are a convenient and effective way for individuals to take control of their finances and make informed decisions about their money

1.1 OVERVIEW:

A finance tracking app can be a great tool to help people keep track of their expenses and manage their finances. Developing a finance tracking app using Kotlin in Android Studio can be a great project for developers who are interested in mobile app development.

Here is a general overview of the process of developing a finance tracking app using Kotlin in Android Studio:

Define the features: The first step is to define the features of the app. Some of the features that can be included in a finance tracking app are: adding and categorizing expenses, setting up a budget, generating expense reports, syncing data across devices, and providing alerts and reminders.

Design the user interface: Once the features have been defined, the next step is to design the user interface. This involves creating the visual elements of the app, such as buttons, text fields, and menus, and organizing them in a way that is easy for users to navigate.

Implement the logic: After the user interface has been designed, the next step is to implement the logic of the app. This involves writing code to handle user inputs and interactions, store and retrieve data, and perform calculations and other operations as needed.

Test and debug: Once the app has been implemented, it needs to be thoroughly tested and any bugs or errors need to be fixed. This can involve running the app on a variety of devices, testing different scenarios, and getting feedback from users.

Deploy the app: Finally, the app can be deployed to the Google Play Store or other app stores for users to download and use.

Overall, developing a finance tracking app using Kotlin in Android Studio can be a challenging but rewarding project for developers. By following best practices in software development and design, developers can create an app that helps people manage their finances more effectively.

1.2 PURPOSE:

The purpose of a finance tracking app project using Kotlin is to create a mobile application that helps users keep track of their personal finances. Kotlin is a programming language that is well-suited for mobile app development, and it offers many benefits such as better code readability, less code verbosity, and improved type safety.

With a finance tracking app, users can easily input and track their income and expenses, create budgets, set financial goals, and view reports and visualizations of their financial data. The app can also provide alerts and reminders to help users stay on top of their finances.

By using a finance tracking app, users can gain better insights into their spending habits, identify areas where they can save money, and work towards achieving their financial goals. Additionally, the app can help users avoid overspending, prevent overdrafts and missed payments, and reduce financial stress.

Overall, a finance tracking app project using Kotlin can be a useful tool for individuals looking to take control of their finances and achieve financial stability.

2. PROBLEM DEFINITION AND DESIGN THINKING

Problem definition:

The problem we are addressing with the finance management tracking app project is that many individuals struggle with managing their personal finances. This can lead to overspending, missed payments, and financial stress. While there are many finance management tools available, they may not be user-friendly, lack customization options, or require a paid subscription. Therefore, there is a need for a user-friendly and customizable finance management tracking app that helps individuals stay on top of their finances.

Design thinking:

Design thinking is a human-centered approach to problem-solving that involves empathy, creativity, and iterative prototyping. Using design thinking principles, we can create a finance management tracking app that addresses the needs of our target audience. Here are the five stages of design thinking:

Empathize: In this stage, we need to understand the needs and pain points of our target audience. We can conduct user interviews, surveys, and focus groups to gather insights into their financial management challenges.

Define: Based on our research, we can define the problem we are trying to solve and identify the key features and functionalities that our app should have.

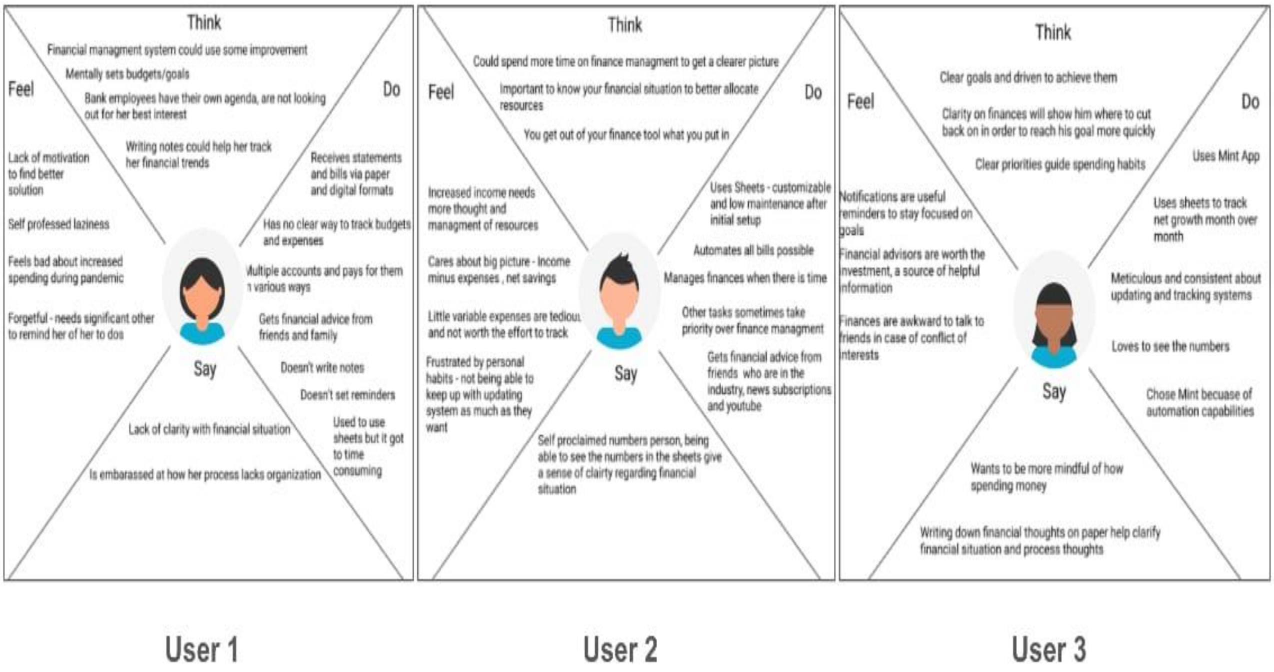
Ideate: This stage involves brainstorming and generating ideas for our app's design and features. We can use techniques such as mind mapping, sketching, and prototyping to explore different options.

Prototype: In this stage, we create a basic prototype of our app's user interface and test it with a small group of users. This helps us identify areas where we need to improve the design and functionality.

Test: Based on feedback from our prototype testing, we can refine and improve our app's design and features. We can conduct additional testing and iterate until we have a final product that meets the needs of our target audience.

By following the design thinking process, we can create a finance management tracking app that is user-friendly, customizable, and addresses the pain points of our target audience. The app can help individuals stay on top of their finances, reduce financial stress, and work towards achieving their financial goals.

2.1 EMPATHY MAP



Think

Financial managment system could use some improvement

Mentally sets budgets/goals

Bank employees have their own agenda, are not looking out for her best interest

Writing notes could help her track her financial trends

Feel

Lack of motivation to find better solution

Self professed laziness

Feels bad about increased spending during pandemic

Forgetful - needs significant other to remind her of her to dos

Do

Receives statements and bills via paper and digital formats

Has no clear way to track budgets and expenses

Multiple accounts and pays for them in various ways

Gets financial advice from friends and family

Doesn't write notes

Doesn't set reminders

Used to use sheets but it got to time consuming

Say

Lack of clarity with financial situation

Is embarassed at how her process lacks organization

Think

Could spend more time on finance managment to get a clearer picture

Important to know your financial situation to better allocate resources

You get out of your finance tool what you put in

Feel

Increased income needs more thought and managment of resources

Cares about big picture - Income minus expenses , net savings

Little variable expenses are tedious and not worth the effort to track

Frustrated by personal habits - not being able to keep up with updating system as much as they want

Do

Uses Sheets - customizable and low maintenance after initial setup

Automates all bills possible

Manages finances when there is time

Other tasks sometimes take priority over finance managment

Gets financial advice from friends who are in the industry, news subscriptions and youtube

Say

Self proclaimed numbers person, being able to see the numbers in the sheets give a sense of clairty regarding financial situation

Think

Clear goals and driven to achieve them

Clarity on finances will show him where to cut back on in order to reach his goal more quickly

Clear priorities guide spending habits

Feel

Notifications are useful reminders to stay focused on goals

Financial advisors are worth the investment, a source of helpful information

Finances are awkward to talk to friends in case of conflict of interests

Do

Uses Mint App

Uses sheets to track net growth month over month

Meticulous and consistent about updating and tracking systems

Loves to see the numbers

Chose Mint becuase of automation capabilities

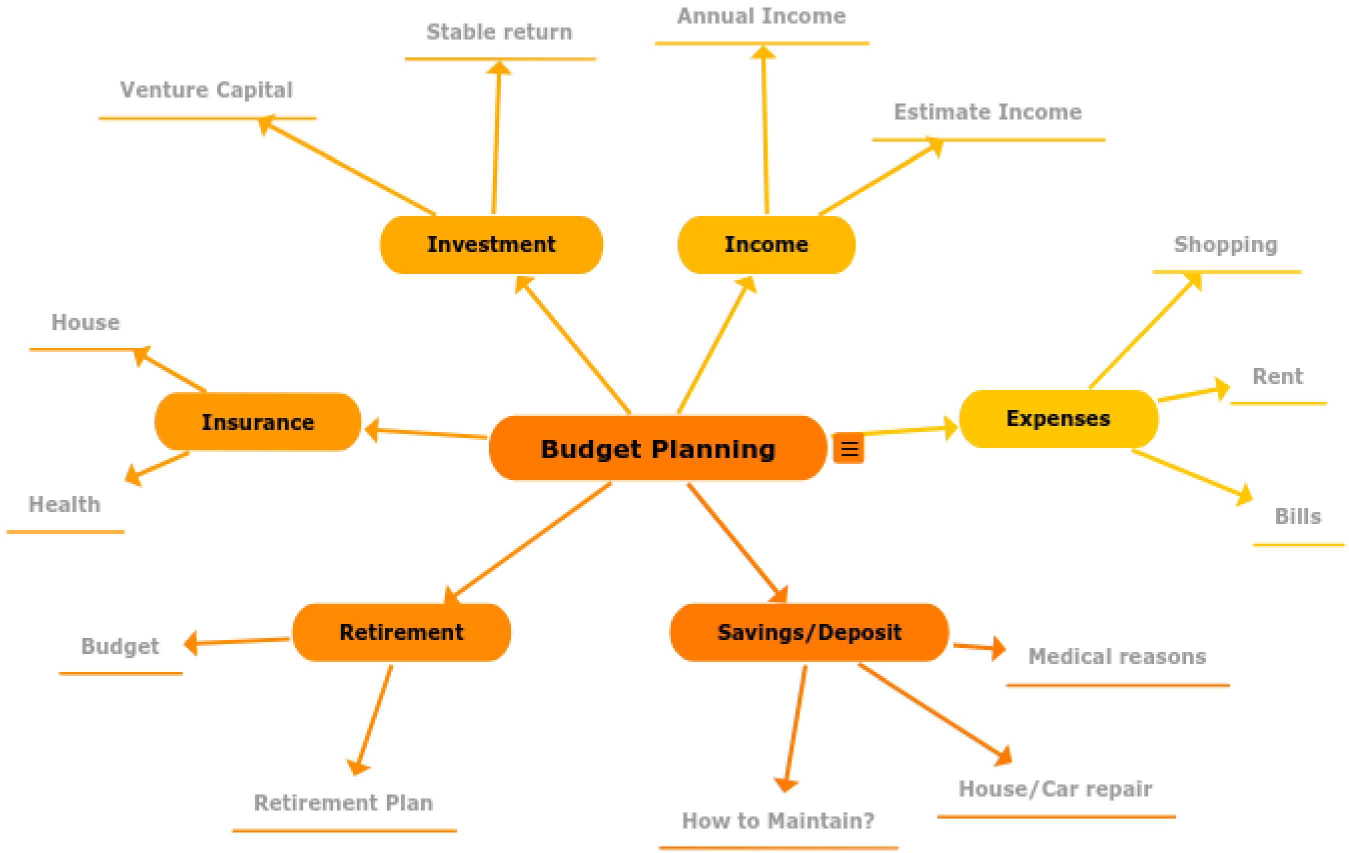
Say

Wants to be more mindful of how spending money

Writing down financial thoughts on paper help clarify financial situation and process thoughts

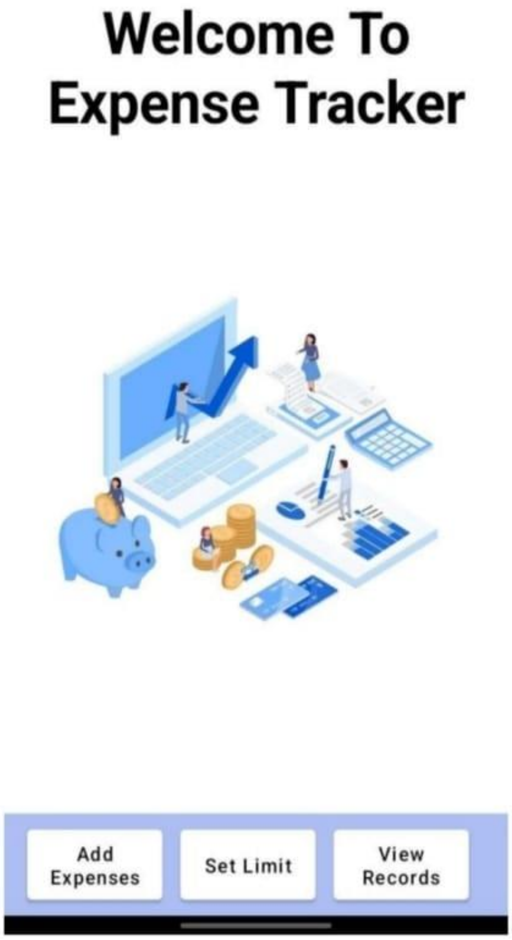
2.2

2.2IDEATION AND BRAINSTORMING MAP



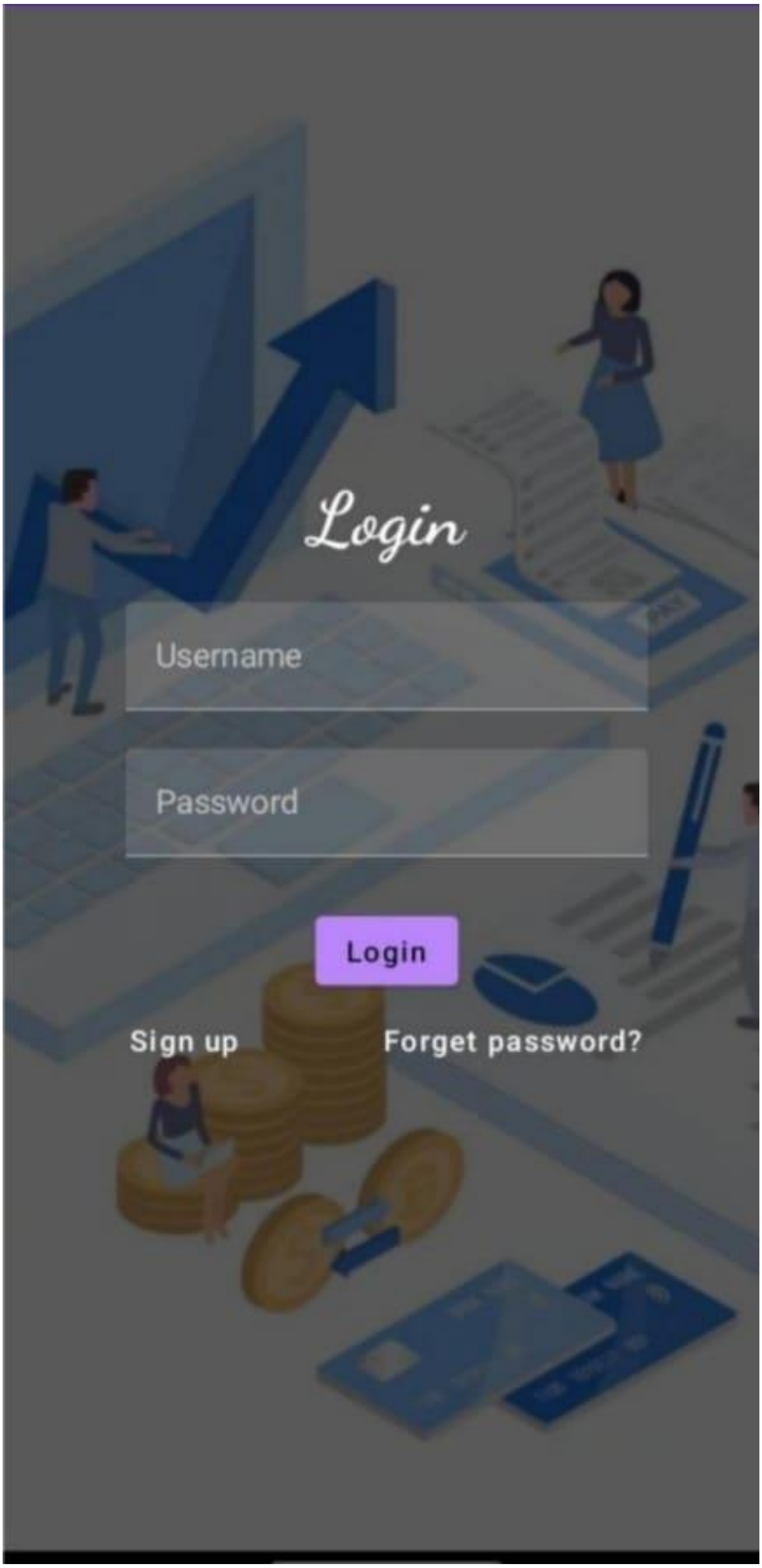
RESULT

WELCOME PAGE

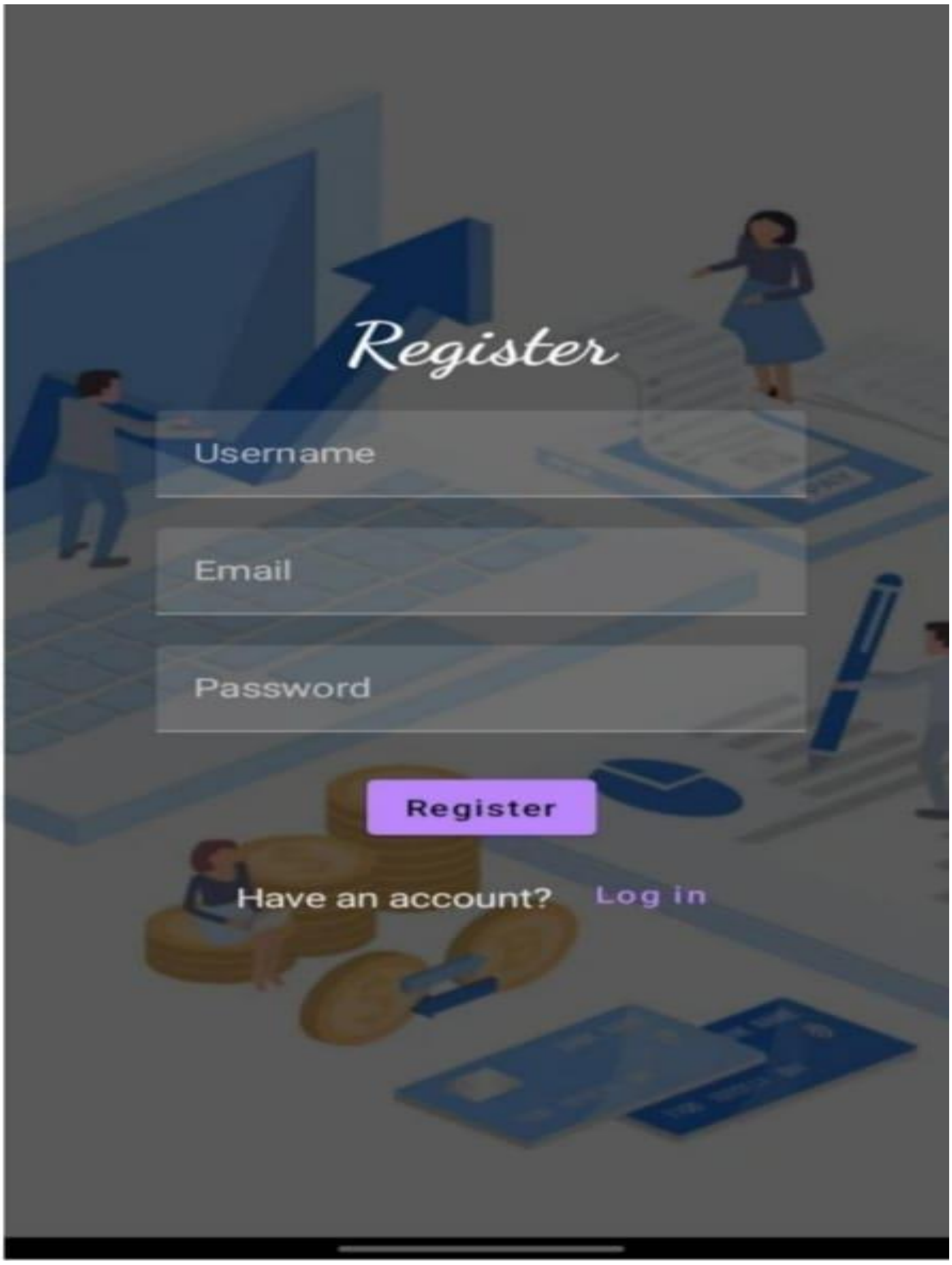


Welcome To Expense Tracker

LOGIN PAGE



REGISTER PAGE



Register

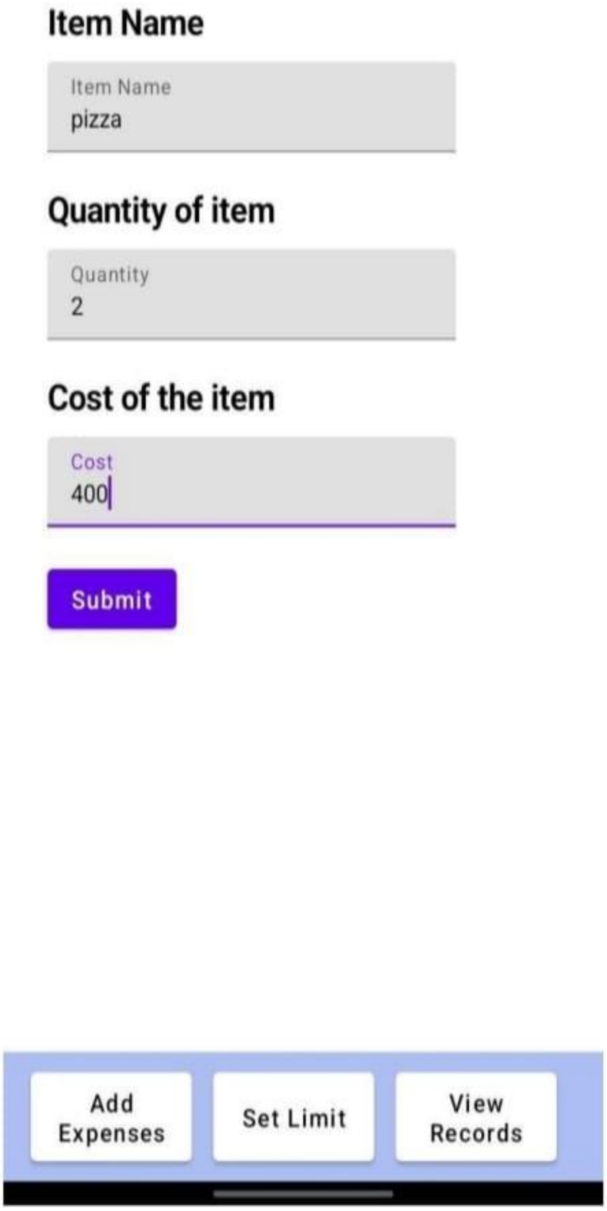
Username

Email

Password

Have an account? Log in

ADD EXPENSES PAGE



Item Name

Item Name
pizza

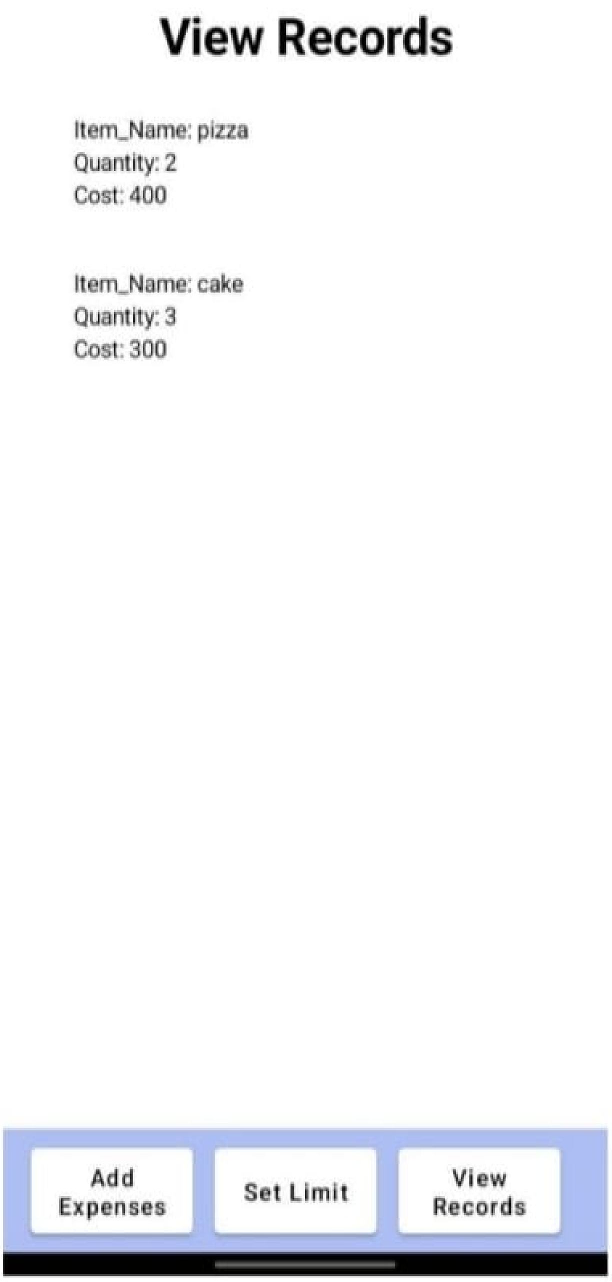
Quantity of item

Quantity
2

Cost of the item

Cost
400

VIEW RECORDS PAGE



View Records

Item_Name: pizza
Quantity: 2
Cost: 400

Item_Name: cake
Quantity: 3
Cost: 300

ADVANTAGES AND DISADVANTAGES

Advantages of finance management app:

Improved financial organization: Finance management apps help users to better manage their finances by tracking expenses and income, categorizing transactions, and creating budgets.

Convenient tracking: With a finance management app, users can track their expenses and income in real-time, which can help them make better financial decisions and avoid overspending.

Saving time: By automatically categorizing transactions and generating financial reports, finance management apps can save users a significant amount of time compared to manual methods of tracking expenses.

Accessibility: Finance management apps can be accessed from anywhere with an internet connection, allowing users to manage their finances on the go.

Increased awareness: By providing users with detailed information about their spending habits and financial health, finance management apps can help increase financial literacy and awareness.

Disadvantages of finance management app:

Security risks: Finance management apps deal with sensitive financial information, making them an attractive target for hackers and identity thieves. Users must ensure that they use secure passwords and keep their login credentials safe.

Technical issues: Finance management apps may experience technical issues such as bugs, glitches, or crashes that can negatively affect the user experience.

Reliance on technology: Users may become too reliant on the finance management app and neglect other important aspects of financial management, such as budgeting and saving.

Privacy concerns: Finance management apps may collect and store personal information, including financial data, which can be sold to third-party advertisers or used for targeted marketing.

Cost: While many finance management apps are free, some may require a subscription or charge a fee for certain features, which may be a barrier for some users.

APPLICATIONS

Finance tracking management apps can have a variety of applications, including:

Personal finance management: Individuals can use finance tracking management apps to monitor their income, expenses, and savings. By tracking their finances, they can make informed decisions about their spending habits and improve their financial health.

Business finance management: Finance tracking management apps can be used by small business owners to track expenses, revenue, and profitability. By monitoring their finances, business owners can make informed decisions about their operations and investments.

Investment management: Finance tracking management apps can help investors monitor their portfolio and track the performance of their investments. By analyzing investment performance, investors can make informed decisions about buying, holding, or selling their assets.

Expense management: Companies can use finance tracking management apps to monitor and control employee expenses. By tracking employee expenses, companies can ensure compliance with expense policies and reduce unnecessary spending.

Budgeting: Finance tracking management apps can be used to set up and manage budgets. By creating a budget and tracking expenses against it, individuals and businesses can stay on top of their spending and avoid overspending.

Tax management: Finance tracking management apps can help individuals and businesses prepare for tax season by tracking deductible expenses and generating reports for tax purposes.

Overall, finance tracking management apps can be a useful tool for anyone looking to improve their financial health, whether for personal or business purposes

CONCLUSION

In conclusion, finance tracking management apps developed using Kotlin can be a useful tool for personal and business financial management. These apps can help users track expenses, income, savings, and investments, and provide valuable insights into their financial health. By monitoring their finances, users can make informed decisions about their spending and saving, and stay on top of their financial goals. While there are some potential disadvantages, such as security risks and privacy concerns, these can be mitigated through careful use and adherence to best practices. Overall, Kotlin-based finance tracking management apps can provide a convenient, accessible, and powerful solution for anyone looking to improve their financial management.

FUTURE SCOPE

The future scope of finance tracking management apps developed using Kotlin is quite promising, as the demand for mobile-based financial management solutions continues to grow. Some of the potential future developments and enhancements for these apps may include:

Integration with emerging technologies: Finance tracking management apps developed using Kotlin may integrate with emerging technologies such as blockchain, artificial intelligence, and machine learning to improve security, automation, and personalization.

Enhanced user experience: Future Kotlin-based finance tracking management apps may incorporate more intuitive and user-friendly interfaces, as well as features such as gamification and social sharing to enhance user engagement.

Predictive analytics: Advanced analytics and predictive modeling may be used to provide users with customized financial advice and recommendations based on their financial behavior and goals.

Smart automation: Kotlin-based finance tracking management apps may incorporate smart automation features, such as automatically categorizing expenses and generating customized budgets and investment recommendations.

Seamless integration with financial institutions: Integration with banks, credit card companies, and other financial institutions may be improved to provide users with more accurate and up-to-date financial information, as well as faster and more seamless transactions.

Overall, the future scope of finance tracking management apps developed using Kotlin is exciting, as developers continue to explore new technologies and strategies to enhance the user experience and provide valuable financial management solutions for individuals and businesses alike

APPENDIX

Here are some additional resources that may be helpful for those interested in finance tracking management apps developed using Kotlin:

Kotlin official documentation: The official documentation for Kotlin provides a comprehensive guide to the language and its features, including how to develop Android apps using Kotlin.

Android Studio: Android Studio is an integrated development environment (IDE) that developers can use to develop Android apps, including finance tracking management apps using Kotlin.

GitHub: GitHub is a platform that allows developers to collaborate on projects and share code. There are many finance tracking management app projects on GitHub that use Kotlin, which can serve as a valuable resource for developers.

Stack Overflow: Stack Overflow is a popular online community where developers can ask and answer technical questions. It can be a helpful resource for developers working on finance tracking management apps using Kotlin.

Udacity: Udacity is an online education platform that offers courses in a variety of topics, including Android development using Kotlin. These courses can be helpful for developers who are new to Kotlin or Android development.

Kotlin for Android Developers book: This book by Antonio Leiva provides an in-depth guide to developing Android apps using Kotlin, including how to build finance tracking management apps.

Overall, there are many resources available for those interested in developing finance tracking management apps using Kotlin. By leveraging these resources, developers can create powerful and effective apps that help users manage their finances and achieve their financial goals

SOURCE CODE:

LOGIN ACTIVITY

```
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

import androidx.compose.ui.text.input.PasswordVisualTransformation
import androidx.compose.ui.text.input.VisualTransformation
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 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,


        )


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


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

```
Text(
    fontSize = 36.sp,
    fontWeight = FontWeight.ExtraBold,
    fontFamily = FontFamily.Cursive,
    color = Color.White,
    text = "Login"
)
Spacer(modifier = Modifier.height(10.dp))

TextField(
    value = username,
    onValueChange = { username = it },
    label = { Text("Username") },
    modifier = Modifier.padding(10.dp)
        .width(280.dp)
)

TextField(
    value = password,
    onValueChange = { password = it },
    label = { Text("Password") },
    modifier = Modifier.padding(10.dp)
        .width(280.dp),
    visualTransformation = PasswordVisualTransformation()


)


if (error.isNotEmpty()) {
    Text(
        text = error,
        color = MaterialTheme.colors.error,
        modifier = Modifier.padding(vertical = 16.dp)
    )
}


Button(
    onClick = {
        if (username.isNotEmpty() && password.isNotEmpty()) {
            val user = databaseHelper.getUserByUsername(username)
            if (user != null && user.password == password) {
                error = "Successfully log in"
                context.startActivity(
                    Intent(
                        context,
                        MainActivity::class.java
                    )
                )
                //onLoginSuccess()
            }
            else {
                error =  "Invalid username or password"
            }

        } else {
            error = "Please fill all fields"
        }
    },
    modifier = Modifier.padding(top = 16.dp)
) {
    Text(text = "Login")
}
Row {
    TextButton(onClick = {context.startActivity(
        Intent(
            context,
                    RegisterActivity::class.java
                )
            )}
            )
            { Text(color = Color.White,text = "Sign up") }
            TextButton(onClick = {
            })

            {
                Spacer(modifier = Modifier.width(60.dp))
                Text(color = Color.White,text = "Forget password?")
            }
        }
    }
}
private fun startMainPage(context: Context) {
    val intent = Intent(context, MainActivity::class.java)
    ContextCompat.startActivity(context, intent, null)
}
```

USER

```
package com.example.expensestracker


import androidx.room.ColumnInfo

import androidx.room.Entity

import androidx.room.PrimaryKey


@Entity(tableName = "user_table")

data class User(

    @PrimaryKey(autoGenerate = true) val id: Int?,

    @ColumnInfo(name = "first_name") val firstName: String?,

    @ColumnInfo(name = "last_name") val lastName: String?,

    @ColumnInfo(name = "email") val email: String?,

    @ColumnInfo(name = "password") val password: String?,


    )
```