

1.INTRODUCTION

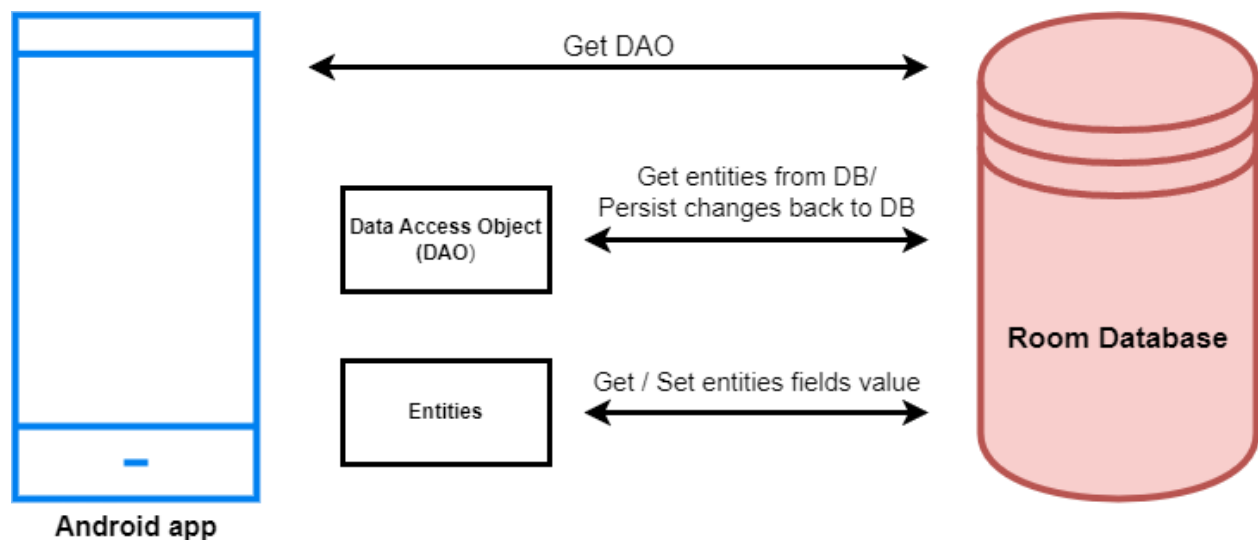
PROJECT OVERVIEW

Money Matters: A Personal Finance Management App

Project Description:

The app allows user to keep track of their expenses and accounts, and provides an overview of their financial status. Users can set a budget for various expenses and view their progress towards it.

Architecture:



Learning Outcomes :

By end of this project:

- You'll be able to work on Android studio and build an app.
- You'll be able to integrate the database accordingly.

Project Workflow:

- Users register into the application.
- After registration , user logins into the application.
- User enters into the main page

Note:

To complete the project you need to finish up the tasks listed below:

Tasks:

- 1.Required initial steps
- 2.Creating a new project.
- 3.Adding required dependencies.
- 4.Creating the database classes.
- 5.Building application UI and connecting to database.
- 6.Using AndroidManifest.xml
- 7.Running the application.

1.2 Purpose:

Achieving financial goals: Personal financial management helps individuals set and achieve financial goals, such as saving for a down payment on a house, paying off debt, or saving for retirement.

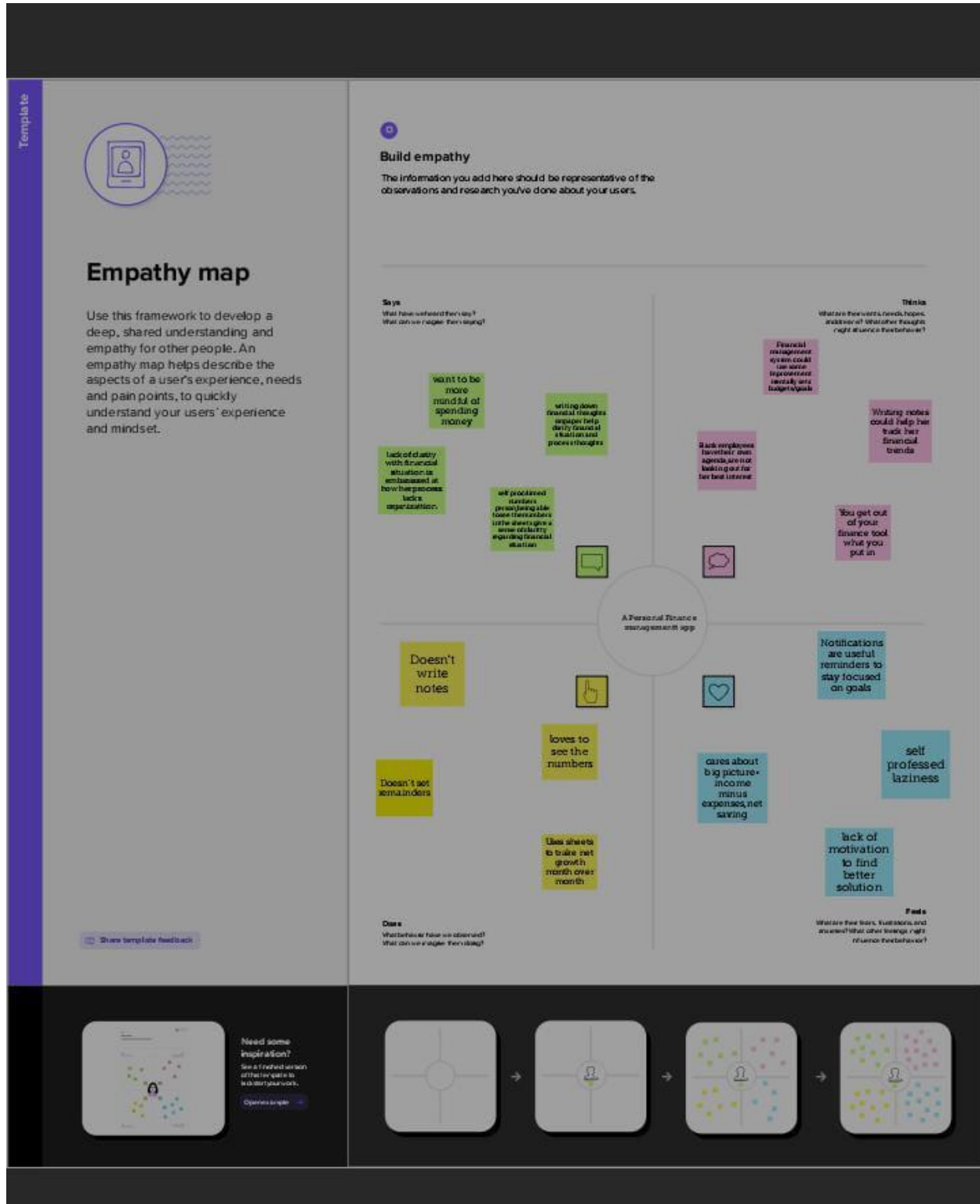
Managing expenses: Keeping track of expenses and creating a budget can help individuals avoid overspending and living beyond their means.

Building wealth: Personal financial management involves investing in assets such as stocks, bonds, and real estate to build long-term wealth.

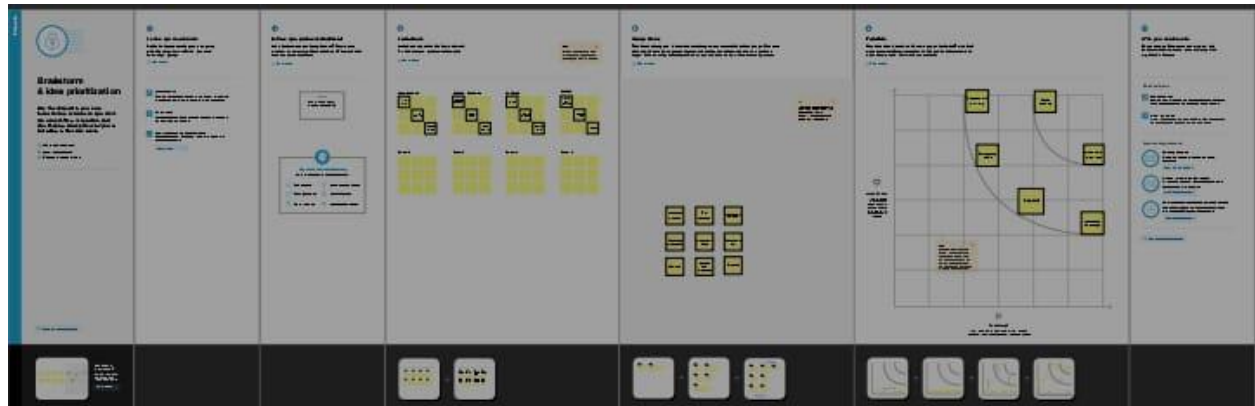
Reducing financial stress: By having a solid understanding of their financial situation and taking steps to improve it, individuals can reduce financial stress and improve their overall well-being.

2 .Problem Definition & Design Thinking.

2.1 Empathy map:



2.2 Ideation & Brainstroming map



3.Result

4:10

Vo
LTE 4G R 42%

Login

Username

Password

Login

Sign up

Forget password?

4:10

Vo
LTE 4G R 42%

Register

Username

Email

Password

Register

Have an account?

Log in

3:54

Vo
LTE 4G R 44%

Welcome To Expense Tracker



Add Expenses

Set Limit

*View
Records*

4. Advantages and Disadvantages

Advantages:

Helps you set financial goals: By managing your personal finances, you can identify your short-term and long-term financial goals, such as saving for a down payment on a home, creating an emergency fund, or planning for retirement.

Helps you stick to a budget: Personal financial management can help you create and stick to a budget that ensures you are living within your means and not overspending.

Reduces financial stress: By having a clear picture of your financial situation and a plan for how to manage your money, you can reduce financial stress and anxiety.

Helps you save money: Through personal financial management, you can identify areas where you can cut back on expenses and save money, such as reducing your utility bills, finding more affordable insurance options, or cooking at home more often.

Helps you avoid debt: By managing your personal finances effectively, you can avoid taking on excessive debt and minimize the amount of interest and fees you pay on existing debt.

Improves your credit score: Personal financial management can help you improve your credit score by making sure you pay your bills on time, keeping your credit utilization low, and monitoring your credit report for errors.

Disadvantages:

Can be time-consuming: Personal financial management can require a significant amount of time and effort, especially when creating a budget, tracking expenses, and monitoring investments.

Can be overwhelming: For some people, personal financial management can be overwhelming, especially if they are dealing with significant debt or financial issues. In some cases, seeking the help of a financial advisor or counselor may be necessary.

May require sacrifice: Managing your personal finances effectively may require you to make sacrifices, such as cutting back on discretionary spending, working extra hours, or delaying major purchases.

Can be impacted by external factors: Even with the most effective personal financial management practices in place, external factors such as job loss, illness, or economic downturns can still impact your financial situation.

Can lead to overanalyzing: Sometimes, the pursuit of perfect financial management can lead to overanalyzing and obsessing over every financial decision, which can be counterproductive and cause unnecessary stress.

5.Application

Mint: Mint is a free online application that allows you to track your income, expenses, and investments in one place. It also provides alerts for bills, credit score tracking, and budget tracking.

Personal Capital: Personal Capital is an online financial management application that offers investment tracking, retirement planning, and budgeting tools. It also provides a comprehensive overview of your financial situation.

YNAB (You Need A Budget): YNAB is a budgeting application that helps you allocate your income toward expenses, savings, and debt payments. It provides a simple yet effective budgeting system and also offers tools for goal setting and tracking.

PocketGuard: PocketGuard is a mobile application that links to your bank accounts and credit cards, providing real-time updates on your balances, bills, and spending. It also

offers budget tracking and savings suggestions.

Quicken: Quicken is a desktop software that allows you to track and manage your finances. It provides budget tracking, investment tracking, and bill reminders, as well as tax preparation tools.

6.Conclusion

However, personal financial management can also be time-consuming and overwhelming, especially if you are dealing with significant financial issues. It is important to be aware of the potential disadvantages of personal financial management and seek help from financial advisors or counselors if necessary.

With the help of personal financial management tools and applications, such as budgeting software or investment tracking apps, you can simplify the process of

managing your finances and achieve greater control over your financial future. Ultimately, effective personal financial management requires discipline, consistency, and a willingness to make necessary sacrifices in order to achieve your financial goals.

7.Future Scope

Artificial intelligence (AI) and machine learning: AI and machine learning can help personalize financial advice, automate budgeting and investment tracking, and provide real-time alerts and recommendations based on spending patterns and financial goals.

Mobile technology: The increasing prevalence of mobile devices provides new opportunities for personal financial management, such as mobile payment options, mobile budget tracking, and mobile investment management.

Cryptocurrencies and blockchain: As cryptocurrencies and blockchain technology continue to gain popularity, personal financial management tools will need to incorporate these assets into their offerings, providing new ways to manage and invest in digital currencies.

Sustainable investing: With the growing interest in sustainable investing, personal financial management tools may need to incorporate environmental, social, and governance (ESG) factors into investment tracking and portfolio management.

Financial wellness: Personal financial management tools may need to expand beyond traditional budgeting and investment tracking, offering resources and support for overall financial wellness, including debt

**management, retirement planning, and
emergency preparedness**

8.Appendix

A.source code

Mainactivity

package com.example.expensetracker

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  
import androidx.compose.ui.graphics.Color  
import  
androidx.compose.ui.res.painterResource  
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.expensetracker.ui.theme.Expen  
sesTrackerTheme  
  
class MainActivity : ComponentActivity() {
```

```
@SuppressWarnings("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

```
        )  
    )  
    },  
    colors =  
    ButtonDefaults.buttonColors(backgroundColor  
= Color.White),  
    modifier =  
    Modifier.size(height = 55.dp, width = 110.dp)  
    )  
    {  
        Text(  
            text = "Set Limit", color =  
            Color.Black, fontSize = 14.sp,  
            textAlign =  
            TextAlign.Center  
        )  
    }
```

```
}
```

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

```
        Button(  
            onClick = {  
                startActivity(  
                    Intent(  
                        applicationContext,
```

```
ViewRecordsActivity::class.java
```

```
        )
```

```
    )
```

```
},
```

```
        colors =
ButtonDefaults.buttonColors(backgroundColor
= Color.White),

        modifier =
Modifier.size(height = 55.dp, width = 110.dp)

    )

    {

        Text(

            text = "View Records",
            color = Color.Black, fontSize = 14.sp,

            textAlign =
TextAlign.Center

        )

    }

}

)
```

```
    ) {  
        MainPage()  
    }  
}  
}
```

@Composable

```
fun MainPage() {  
    Column(  
        modifier =  
        Modifier.padding(20.dp).fillMaxSize(),  
        verticalArrangement =  
        Arrangement.Center,  
        horizontalAlignment =  
        Alignment.CenterHorizontally  
    ) {
```



```
Text(text = "Welcome To Expense  
Tracker", fontSize = 42.sp, fontWeight =  
FontWeight.Bold,
```

```
textAlign = TextAlign.Center)
```

```
Image(painterResource(id =  
R.drawable.img_1), contentDescription = "",  
modifier = Modifier.size(height = 500.dp, width  
= 500.dp))
```

```
}
```

```
}
```

```
LoginActivity
```

```
package com.example.expensetracker
```

```
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.expensetracker.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  
e)  
  
            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)
```

```
}
```