

TITTLE : A PERSONAL FINANCE MANAGEMENT APP

TEAM MEMBERS:

YOGESHRHAJA. R -812022104090

SAFIK BASHA. S -812022104065

YOGESH. K -812022104089

DINESH. B -812022104302

DESCRIPTION :

Our app is a user-friendly, all-in-one financial tool designed to help individuals take control of their personal finances effortlessly. From budgeting and expense tracking to savings goals and investment insights, this app provides users with everything they need to make smarter financial decisions.

Code:

```
Package com.example.expensetracker
```

```
Import android.annotation.SuppressLint
```

```
Import android.content.Context
```

```
Import android.content.Intent
```

```
Import android.os.Bundle
```

```
Import android.widget.Toast
```

```
Import androidx.activity.ComponentActivity
```

```
Import androidx.activity.compose.setContent
```

```
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.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
```

```
    @SuppressWarnings("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(background-color = Color(0xFFadbf4),
```

```
                        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
        )
    }
}

)
}

) {
    AddExpenses(this, itemsDatabaseHelper, expenseDatabaseHelper)
}
}
}
}
}

```

@SuppressWarnings("Range")

@Composable

```
Fun AddExpenses(context: Context, itemsDatabaseHelper: ItemsDatabaseHelper,  
expenseDatabaseHelper: ExpenseDatabaseHelper) {
```

```
    Column(
```

```
        Modifier = Modifier
```

```
            .padding(top = 100.dp, start = 30.dp)
```

```
            .fillMaxHeight()
```

```
            .fillMaxWidth(),
```

```
        horizontalAlignment = Alignment.Start
```

```
    ) {
```

```
        Val mContext = LocalContext.current
```

```
        Var items by remember { mutableStateOf("") } 
```

```
        Var quantity by remember { mutableStateOf("") } 
```

```
        Var cost by remember { mutableStateOf("") } 
```

```
        Var error by remember { mutableStateOf("") } 
```

```
        Text(text = "Item Name", fontWeight = FontWeight.Bold, fontSize = 20.sp)
```

```
        Spacer(modifier = Modifier.height(10.dp))
```

```
        TextField(value = items, onValueChange = { items = it },
```

```
            Label = { Text(text = "Item Name") })
```

```
        Spacer(modifier = Modifier.height(20.dp))
```

```
        Text(text = "Quantity of item", fontWeight = FontWeight.Bold, fontSize =  
20.sp)
```

```
Spacer(modifier = Modifier.height(10.dp))
```

```
TextField(value = quantity, onValueChange = { quantity = it },
```

```
    Label = { Text(text = "Quantity") })
```

```
Spacer(modifier = Modifier.height(20.dp))
```

```
Text(text = "Cost of the item", fontWeight = FontWeight.Bold, fontSize =  
20.sp)
```

```
Spacer(modifier = Modifier.height(10.dp))
```

```
TextField(value = cost, onValueChange = { cost = it },
```

```
    Label = { Text(text = "Cost") })
```

```
Spacer(modifier = Modifier.height(20.dp))
```

```
If (error.isNotEmpty()) {
```

```
    Text(  
        Text = error,
```

```
        Color = MaterialTheme.colors.error,
```

```
        Modifier = Modifier.padding(vertical = 16.dp)
```

```
    )
```

```
}
```

```
Button(onClick = {
```

```
    If (items.isNotEmpty() && quantity.isNotEmpty() && cost.isNotEmpty())  
{
```



```
Val items = Items(  
    Id = null,  
    itemName = items,  
    quantity = quantity,  
    cost = cost  
)
```

```
Val limit= expenseDatabaseHelper.getExpenseAmount(1)
```

```
Val actualvalue = limit?.minus(cost.toInt())  
  
// Toast.makeText(mContext, actualvalue.toString(),  
Toast.LENGTH_SHORT).show()
```

```
Val expense = Expense(  
    Id = 1,  
    Amount = actualvalue.toString()  
)  
  
If (actualvalue != null) {  
    If (actualvalue < 1) {  
        Toast.makeText(mContext, "Limit Over",  
Toast.LENGTH_SHORT).show()  
    } else {
```

```
        expenseDatabaseHelper.updateExpense(expense)
        itemsDatabaseHelper.insertItems(items)
    }
}

    }
}) {
    Text(text = "Submit")
}

}
}
```

USER DATA BASE:

```
Package com.example.expensetracker
```

```
Import android.content.Context
```

```
Import androidx.room.Database
```

```
Import androidx.room.Room
```

```
Import androidx.room.RoomDatabase
```

```
@Database(entities = [User::class], version = 1)
```

```
Abstract class UserDatabase : RoomDatabase() {
```

```
    Abstract fun userDao(): UserDao
```

```
    Companion object {
```

```
        @Volatile
```

```
        Private var instance: UserDatabase? = null
```

```
    Fun getDatabase(context: Context): UserDatabase {
```

```
        Return instance ?: synchronized(this) {
```

```
            Val newInstance = Room.databaseBuilder(
```

```
                Context.applicationContext,
```

```
                UserDatabase::class.java,
```

```
                “user_database”
```

```
            ).build()
```

```
            Instance = newInstance
```

```
            newInstance
```

```
        }
```

```
    }
```

```
}
```

```
}
```

MAIN ACTIVITY :

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.ExpensesTrackerTheme
```

```
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(background-color = Color(0xFFadbf4),
                        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))

    }
}
```

OUTPUT:

