A Personalized Travel Planning and tracking app

Description

A project built using the Android Compose UI toolkit. It demonstrates how to create a simple travel app using the Compose libraries. It also features a personalized feed of recommended accommodations based on the locations.

Our Personalized Travel Planning and Tracking App is designed to revolutionize the way you organize, manage, and cherish your travel experiences. This all-in-one solution caters to travelers who seek seamless planning, real-time tracking, and detailed post-trip summaries.

Our app combines personalized recommendations, robust tracking features, and user-friendly design to ensure that travelers of all types have everything they need at their fingertips. Whether you're planning a solo adventure, a family vacation, or a business trip, this app adapts to your unique travel needs.

Our Personalized Travel Planning and Tracking App is designed to revolutionize the way you organize, manage, and cherish your travel experiences. This all-in-one solution caters to travelers who seek seamless planning, real-time tracking, and detailed post-trip summaries. Our

User Data

```
Package com.example.travelapp

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? )
```

User DAO Interface

```kotlin
package com.example.travelapp

import androidx.room.*

@Dao
interface UserDao {

 @Query("SELECT * FROM user_table WHERE email = :email")
suspend fun getUserByEmail(email: String): User?

 @Insert(onConflict = OnConflictStrategy.REPLACE)
suspend fun insertUser(user: User)

 @Update
suspend fun updateUser(user: User)

 @Delete
suspend fun deleteUser(user: User)
}
```

User Database Class

```kotlin
package com.example.travelapp
```

```kotlin
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

            }
```

```
    }
  }
}
```

User Database Helper Class

```kotlin
import android.annotation.SuppressLint

import android.content.ContentValues

iimport android.content.Context

import android.database.Cursor

import     android.database.sqlite.SQLiteDatabase

import android.database.sqlite.SQLiteOpenHelper


class UserDatabaseHelper(context: Context) :

SQLiteOpenHelper(context, DATABASE_NAME, null, DATABASE_VERSION) {


 companion object {

    private const val DATABASE_VERSION = 1

    private const val DATABASE_NAME = "UserDatabase.db"


    private const val TABLE_NAME = "user_table"

    private const val COLUMN_ID = "id"

    private const val COLUMN_FIRST_NAME = "first_name"

    private const val COLUMN_LAST_NAME = "last_name"

    private const val COLUMN_EMAIL = "email"
```

```kotlin
        private const val COLUMN_PASSWORD = "password"
    }

    override fun onCreate(db: SQLiteDatabase?) {
        val createTable = "CREATE TABLE $TABLE_NAME (" +
                "$COLUMN_ID INTEGER PRIMARY KEY AUTOINCREMENT, " +
                "$COLUMN_FIRST_NAME TEXT, " +
                "$COLUMN_LAST_NAME TEXT, " +
                "$COLUMN_EMAIL TEXT, " +
                "$COLUMN_PASSWORD TEXT" +
                ")"

        db?.execSQL(createTable)
    }

    override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {
        db?.execSQL("DROP TABLE IF EXISTS $TABLE_NAME")
        onCreate(db)
    }

    fun insertUser(user: User) {
        val db = writableDatabase
        val values = ContentValues()
        values.put(COLUMN_FIRST_NAME, user.firstName)
        values.put(COLUMN_LAST_NAME, user.lastName)
```

```kotlin
        values.put(COLUMN_EMAIL, user.email)

        values.put(COLUMN_PASSWORD, user.password)

        db.insert(TABLE_NAME, null, values)

        db.close()

    }


    @SuppressLint("Range")

    fun getUserByUsername(username: String): User? {

        val db = readableDatabase

        val cursor: Cursor = db.rawQuery("SELECT * FROM $TABLE_NAME WHERE
$COLUMN_FIRST_NAME = ?", arrayOf(username))

        var user: User? = null

        if (cursor.moveToFirst()) {

            user = User(

                id = cursor.getInt(cursor.getColumnIndex(COLUMN_ID)),

                firstName = cursor.getString(cursor.getColumnIndex(COLUMN_FIRST_NAME)),

                lastName = cursor.getString(cursor.getColumnIndex(COLUMN_LAST_NAME)),

                email = cursor.getString(cursor.getColumnIndex(COLUMN_EMAIL)),

                password = cursor.getString(cursor.getColumnIndex(COLUMN_PASSWORD)),

            )

        } cursor.close() db.close() return user



    }

    @SuppressLint("Range")
```

```kotlin
fun getUserById(id: Int): User? {

    val db = readableDatabase

    val cursor: Cursor = db.rawQuery("SELECT * FROM $TABLE_NAME WHERE $COLUMN_ID =
?", arrayOf(id.toString()))

    var user: User? = null

    if (cursor.moveToFirst()) {

        user = User(

            id = cursor.getInt(cursor.getColumnIndex(COLUMN_ID)),

            firstName = cursor.getString(cursor.getColumnIndex(COLUMN_FIRST_NAME)),

            lastName = cursor.getString(cursor.getColumnIndex(COLUMN_LAST_NAME)),

            email = cursor.getString(cursor.getColumnIndex(COLUMN_EMAIL)),

            password = cursor.getString(cursor.getColumnIndex(COLUMN_PASSWORD)),

        )

    } cursor.close() db.close() return user



}


@SuppressLint("Range")

fun getAllUsers(): List<User> {

    val users = mutableListOf<User>()

    val db = readableDatabase

    val cursor: Cursor = db.rawQuery("SELECT * FROM $TABLE_NAME", null)

    if (cursor.moveToFirst()) {

        do {
```

```kotlin
            val user = User(

                id = cursor.getInt(cursor.getColumnIndex(COLUMN_ID)),

                firstName = cursor.getString(cursor.getColumnIndex(COLUMN_FIRST_NAME)),

                lastName = cursor.getString(cursor.getColumnIndex(COLUMN_LAST_NAME)),

                email = cursor.getString(cursor.getColumnIndex(COLUMN_EMAIL)),

                password = cursor.getString(cursor.getColumnIndex(COLUMN_PASSWORD)),

            )

            users.add(user)

        } while (cursor.moveToNext())

    } cursor.close() db.close() return users




    }
}
}
```

Login Activity.Kt with Database

package com.example.travelapp

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

```kotlin
import androidx.compose.foundation.background

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.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import androidx.core.content.ContextCompat


class LoginActivity : ComponentActivity() {

private lateinit var databaseHelper: UserDatabaseHelper

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)

        databaseHelper = UserDatabaseHelper(this)

        setContent {

            LoginScreen(this, databaseHelper)

        }
```

```kotlin
    }



} @Composable fun LoginScreen(context: Context,

databaseHelper: UserDatabaseHelper) {

var username by remember { mutableStateOf("") }

var password by remember { mutableStateOf("") }

var error by remember { mutableStateOf("") }


Column(

    modifier = Modifier.fillMaxSize().background(Color.White),

    horizontalAlignment = Alignment.CenterHorizontally,

    verticalArrangement = Arrangement.Center

) {


    Image(painterResource(id = R.drawable.trav), contentDescription = "")


    Text(

        fontSize = 36.sp,

        fontWeight = FontWeight.ExtraBold,

        fontFamily = FontFamily.Cursive,

        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") },

    visualTransformation = PasswordVisualTransformation(),

    modifier = Modifier.padding(10.dp)

        .width(280.dp)

)


if (error.isNotEmpty()) {

    Text(

        text = error,

        color = MaterialTheme.colors.error,

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

    )

}
```

```kotlin
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")
```

```kotlin
    }
    Row  {

      TextButton(onClick = {context.startActivity(

        Intent(

          context,

          RegisterActivity::class.java

        )

      )}

      )

      { Text(text = "Register") }

      TextButton(onClick = {

      })


      {

        Spacer(modifier = Modifier.width(60.dp))

        Text(text = "Forget password?")

      }

    }
  }
```

```kotlin
package com.example.travelapp

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.background

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.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import androidx.core.content.ContextCompat
```

```kotlin
class RegisterActivity : ComponentActivity() {

private lateinit var databaseHelper: UserDatabaseHelper

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)

        databaseHelper = UserDatabaseHelper(this)

        setContent {

            RegistrationScreen(this, databaseHelper)

        }

    }

}


@Composable

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


 var username by remember { mutableStateOf("") }

var password by remember { mutableStateOf("") }

var email by remember { mutableStateOf("") }

    va

r error by remember { mutableStateOf("") }

@Composable

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


 var username by remember { mutableStateOf("") }

var password by remember { mutableStateOf("") }

var email by remember { mutableStateOf("") }
```

```kotlin
var error by remember { mutableStateOf("") }


Column(

    modifier = Modifier.fillMaxSize().background(Color.White),

    horizontalAlignment = Alignment.CenterHorizontally,

    verticalArrangement = Arrangement.Center

) {


    Image(painterResource(id = R.drawable.tra), contentDescription = "")


    Text(

        fontSize = 36.sp,

        fontWeight = FontWeight.ExtraBold,

        fontFamily = FontFamily.Cursive,

        text = "Register"

    )


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

    TextField(

        value = username,

        onValueChange = { username = it },

        label = { Text("Username") },

        modifier = Modifier

            .padding(10.dp)

            .width(280.dp)
```

```kotlin
    )

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

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

    if (error.isNotEmpty()) {
        Text(
```

```kotlin
                text = error,

                color = MaterialTheme.colors.error,

                modifier =

Modifier.padding(vertical = 16.dp)

            )

        }

Button(

        onClick = {

            if (username.isNotEmpty() && password.isNotEmpty() && email.isNotEmpty()) {

                val user = User(

                    id = null,

                    firstName = username,

                    lastName = null,

                    email = email,

                    password = password

                ) databaseHelper.insertUser(user) error = "User registered

                successfully" // Start LoginActivity using the current context

                context.startActivity(


                    Intent(

                        context,

                        LoginActivity::class.java

                    )

                )
```

```kotlin
        } else {

            error = "Please fill all fields"

        }

    },

    modifier = Modifier.padding(top = 16.dp)

) {

    Text(text = "Register")

} Spacer(modifier = Modifier.width(10.dp))

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



Row() {

    Text(

        modifier = Modifier.padding(top = 14.dp), text = "Have an account?"

    ) TextButton(onClick = {

        context.startActivity(

            Intent(

                context,

                LoginActivity::class.java

            )

        )

    })
```

```
            {

                Spacer(modifier = Modifier.width(10.dp))

                Text(text = "Log in")

            }

        }

    }

}  private fun

startLoginActivity(context:

Context) {
    val intent = Intent(context, LoginActivity::class.java)

    ContextCompat.startActivity(context, intent, null)

}
```

Main Activity.kt File

package com.example.travelapp

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.clickable

import androidx.compose.foundation.layout.*

import androidx.compose.foundation.rememberScrollState

import androidx.compose.foundation.verticalScroll

import androidx.compose.material.Card

```kotlin
import androidx.compose.material.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Alignment

import androidx.compose.ui.Modifier

import androidx.compose.ui.draw.scale

import androidx.compose.ui.graphics.Color

import androidx.compose.ui.res.painterResource

import androidx.compose.ui.res.stringResourceimport y

   }


@Composable
   fun TravelApp(context: Context) {
      Column(
         modifier = Modifier
            .padding(20.dp)
            .verticalScroll(rememberScrollState())


      ) {

         Text(
            fontSize = 40.sp,
            color = Color(android.graphics.Color.rgb(120, 40, 251)),
            fontFamily = FontFamily.Cursive,
            text = "Wanderlust Travel"
```

```kotlin
)

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


// 01
Card(
    modifier = Modifier
        .fillMaxWidth()
        .height(250.dp)
        .clickable {
        context.startActivity(
            Intent(context, BaliActivity::class.java)


        )
        } ,
    elevation = 8.dp
)
{
    Column(
        horizontalAlignment = Alignment.CenterHorizontally
    ) {
Image(
        painterResource(id = R.drawable.bali), contentDescription = "",
        modifier = Modifier
        .height(150.dp)
```

```kotlin
            .scale(scaleX = 1.2F, scaleY = 1F)
        )


        Text(
            text = stringResource(id = R.string.place_1),

            fontSize = 18.sp

        )



        Text(
            text = stringResource(id = R.string.description),

            fontWeight = FontWeight.Light,

            fontSize = 16.sp,

            textAlign = TextAlign.Center,

        )


        Text(
            text = stringResource(id = R.string.plan), color = Color.Gray,

            fontSize = 16.sp

        )
    }
}


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

```kotlin
//02

Card(

    modifier = Modifier

        .fillMaxWidth()

        .height(250.dp)

        .clickable {

        context.startActivity(

            Intent(context, ParisActivity::class.java)


        )

        } ,

    elevation = 8.dp

)

{

    Column(

        horizontalAlignment = Alignment.CenterHorizontally

) {Image(

        painterResource(id = R.drawable.bali), contentDescription = "",

        modifier = Modifier

        .height(150.dp)

        .scale(scaleX = 1.2F, scaleY = 1F)

    )


    Text(
```

```kotlin
            text = stringResource(id = R.string.place_1),

            fontSize = 18.sp

        )Text(

            text = stringResource(id = R.string.description),

            fontWeight = FontWeight.Light,

            fontSize = 16.sp,

            textAlign = TextAlign.Center,

        )


        Text(

            text = stringResource(id = R.string.plan), color = Color.Gray,

            fontSize = 16.sp

        )

    }

}


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



//02

Card(

    modifier = Modifier

        .fillMaxWidth()

        .height(250.dp)

        .clickable {
```

```kotlin
        context.startActivity(

            Intent(context, ParisActivity::class.java)


        )

    }, elevation = 8.dp
)
{

    Column(

        horizontalAlignment = Alignment.CenterHorizontally

    ) {

        Image(

            painterResource(id = R.drawable.paris), contentDescription = "",

            modifier = Modifier

            .height(150.dp)

            .scale(scaleX = 1.2F, scaleY = 1F)

        )


        Text(

            text = stringResource(id = R.string.place_2),

            fontSize = 18.sp

        )



        Text(

            text = stringResource(id = R.string.description),
```

```kotlin
            fontWeight = FontWeight.Light,

            fontSize = 16.sp,

            textAlign = TextAlign.Center,

        )



    Text(

        text = stringResource(id = R.string.plan), color = Color.Gray,

        fontSize = 16.sp

    )

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


//03
Card(

    modifier = Modifier

        .fillMaxWidth()

        .height(250.dp)

        .clickable {

        context.startActivity(

            Intent(context, SingaporeActivity::class.java)


        )

        } ,

    elevation = 8.dp

)
```

```kotlin
{
    Column(
        horizontalAlignment = Alignment.CenterHorizontally
    ) {
        Image(
            painterResource(id = R.drawable.singapore), contentDescription = "",
            modifier = Modifier
                .height(150.dp)
                .scale(scaleX = 1.2F, scaleY = 1F)
        )

        Text(
            text = stringResource(id = R.string.place_3),
            fontSize =
18.sp
        )Text(
            text = stringResource(id = R.string.description),
            fontWeight = FontWeight.Light,
            fontSize = 16.sp,
            textAlign = TextAlign.Center,
        )

        Text(
            text = stringResource(id = R.string.plan), color = Color.Gray,
            fontSize = 16.sp
```

```
        )

      }

    }


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

  }

 }

}
```

Bali Activity.Kt File

```kotlin
package com.example.travelapp


import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.Image

import androidx.compose.foundation.background

import androidx.compose.foundation.layout.*

import androidx.compose.foundation.rememberScrollState

import androidx.compose.foundation.verticalScroll

import androidx.compose.material.MaterialTheme

import androidx.compose.material.Surface

import androidx.compose.material.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Modifier

import androidx.compose.ui.draw.scale
```

```kotlin
import androidx.compose.ui.graphics.Color

import androidx.compose.ui.res.painterResource

import androidx.compose.ui.res.stringResource

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

import androidx.compose.ui.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import com.example.travelapp.

ui.theme.TravelAppTheme

class BaliActivity : ComponentActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)

        setContent {

            TravelAppTheme {

                // A surface container using the 'background' color from the theme

                Surface(

                    modifier = Modifier.fillMaxSize(),

                    color = MaterialTheme.colors.background

                ) {

                    PlaceOne()

                }

            }

        }

    }

}
```

```kotlin
@Composable

fun PlaceOne() {

Column(modifier = Modifier.background(color = Color.White)

    .padding(20.dp)

    .verticalScroll(rememberScrollState())

  ) {

    Text(

      fontSize = 40.sp,

      color = Color(android.graphics.Color.rgb(120, 40, 251)),

      fontFamily = FontFamily.Cursive,

      text = stringResource(id = R.string.place_1),

    )

    I  m  a  g  e  (

      painterResource(id = R.drawable.bali), contentDescription = "",

      modifier = Modifier

        .padding(16.dp)

        .fillMaxWidth()

        .height(200.dp)

        .scale(scaleX = 1.2F, scaleY = 1F)

)

T

e      color=Color.Black,

x      text = "Day 1: Arrival and Relaxation\n" +

t  (
```

"Arrive in Bali and check into your hotel or accommodation.\n" +

"Spend the day relaxing and getting acclimated to the island.\n" +

"If you have time, explore the nearby area or head to the beach.\n" +

"\n" +

"Day 2: Ubud Tour\n" +

"Start your day early and head to Ubud, a cultural and artistic hub in Bali.\n" +

"Visit the Monkey Forest and the Ubud Palace.\n" +

"Take a tour of the Tegalalang Rice Terrace, a beautiful UNESCO World Heritage Site.\n" +

"End your day with a traditional Balinese dance performance.\n" +

"\n" +

"Day 3: Temple Hopping\n" +

"Visit some of Bali's most famous temples, such as Tanah Lot and Uluwatu.\n" +

"Take in the stunning views of the ocean and cli s.\n" +

"Enjoy a sunset dinner at one of the many restaurants near the temples.\n" +

"\n" +

"Day 4: Waterfalls and Beaches\n" +

"Take a day trip to Bali's beautiful waterfalls, such as Tegenungan or Gitgit.\n" +

"Spend the afternoon at one of Bali's world-renowned beaches, like Seminyak or Nusa Dua.\n" +

"\n" +

"Day 5: Island Hopping\n" +

```kotlin
            "Take a day trip to one of Bali's neighboring islands, such as Nusa Lembongan or
Gili Islands.\n" +

            "Snorkel or scuba dive in the clear waters and relax on the beach.\n" +

            "\n" +

            "Day 6: Cultural Activities\n" +

            "Visit a traditional Balinese village and learn about the island.\n" +

        "\n" +

            "Day 7: Departure\n" +

            "Explore the surrounding area and take in the stunning sunset views.\n" +

            "Have dinner at a local restaurant before returning to your accommodation."

    )


    }
}
```

Paris Activity.Kt File

```kotlin
package com.example.travelapp


import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.Image

import androidx.compose.foundation.background

import androidx.compose.foundation.layout.*

importandroidx.compose.foundation.rememberScrollState
```

```kotlin
import androidx.compose.ui.unit.sp

import com.example.travelapp.ui.theme.TravelAppTheme


class ParisActivity : ComponentActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)

        setContent {

TravelAppTheme {

            // A surface container using the 'background' color from the theme

            Surface(

                modifier = Modifier.fillMaxSize(),

                color = MaterialTheme.colors.background

            ) {

                Greeting()

            }

        }

    }

}


@Composable

fun Greeting() {


 Column(

     modifier = Modifier.background(color = Color.White)
```

```kotlin
        .padding(20.dp)

        .verticalScroll(rememberScrollState())

) {

    Text(

        fontSize = 40.sp,

        color = Color(android.graphics.Color.rgb(120, 40, 251)),

        fontFamily = FontFamily.Cursive,

        text = stringResource(id = R.string.place_2),

    )

    I  m  a  g  e  (

        painterResource(id = R.drawable.paris), contentDescription = "",

        modifier = Modifier

            .padding(16.dp)

            .fillMaxWidth()

            .height(200.dp)

            .scale(scaleX = 1.2F, scaleY = 1F)

    ) Text(


        color=Color.Black,

        text = "Day 1: Arrival and Introduction\n" +


            "Check into your accommodation and freshen up\n" +

            "Take a stroll around the neighborhood to get acquainted\n" +

            "Visit the Ei el Tower, preferably in the evening when it is lit up\n" +

            "Have a relaxing dinner at a nearby restaurant\n" +
```

"\n" +

"Day 2: Art and History\n" +

"Visit the Louvre Museum to see some of the world's most famous art pieces\n" +

"Stroll through the Tuileries Garden and the Place de la Concorde\n" +

"Visit the Orsay Museum, which houses a large collection of impressionist art\n" +

"Have dinner at a local French restaurant\n" +

"\n" +

"Day 3: French Culture and Food\n" +

"Visit the Montmartre neighborhood to see the famous Basilique du Sacré-Cœur and Place du Tertre\n" +

"Explore the historic neighborhood of Le Marais\n" +

"Try some delicious French pastries at a local bakery\n" +

"Have dinner at a brasserie to taste some classic French cuisine\n" +

"\n" +

"Day 4: Architecture and Gardens\n" +

"Visit the Palace of Versailles, a UNESCO World Heritage site, and explore its

beautiful gardens\n" +

"Walk along the Champs-Elysées and stop at the Arc de Triomphe\n" +

"Visit the Sainte-Chapelle, a beautiful Gothic chapel with stunning stained-glass windows\n" +

```
"Have dinner at a local restaurant in the 7th arrondissement\n" +

"\n" +

"Day 5: Shopping and Sightseeing\n" +

"Visit the Notre-Dame Cathedral and climb up to the top for a stunning view of the
city\n" +

"Explore the Latin Quarter and visit the Panthéon\n" +

"Go shopping at the famous Galeries Lafayette or Printemps department stores\n"
+

"Have dinner at a local bistro\n" +

"\n" +

"Day 6: Parisian Parks and Museums\n" +

"Visit the Musée Rodin and explore its beautiful gardens\n" +

"Stroll through the Luxembourg Gardens and visit the Luxembourg Palace\n" +

"Visit the Centre Pompidou, a modern art museum in the Marais neighborhood\n" +

"Have dinner at a local restaurant in the Latin Quarter\n" +

"\n" +

"Day 7: River Cruise and Farewell\n" +

"Take a boat cruise along the Seine River to see the city from a di erent
perspective\n" +

"Visit the Musée de l'Orangerie, which houses Monet's famous water lilies
paintings\n" +

"Have a farewell dinner at a Michelin-starred restaurant"

)
```

```
    }
}
```

Singapore Activity.Kt File

package com.example.travelapp

```kotlin
import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.Image

import androidx.compose.foundation.background

import androidx.compose.foundation.layout.*

import androidx.compose.foundation.rememberScrollState

import androidx.compose.foundation.verticalScroll

import androidx.compose.material.MaterialTheme

import androidx.compose.material.Surface

import androidx.compose.material.Text

import androidx.compose.runtime.Composable

import androidx.compose.ui.Modifier

import androidx.compose.ui.draw.scale

import androidx.compose.ui.graphics.Color

import androidx.compose.ui.res.painterResource

import androidx.compose.ui.res.stringResource

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

import androidx.compose.ui.tooling.preview.Preview

import androidx.compose.ui.unit.dp
```

```kotlin
import androidx.compose.ui.unit.sp

import com.example.travelapp.ui.theme.TravelAppTheme


class SingaporeActivity : ComponentActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)

        setContent {

            TravelAppTheme {

                // A surface container using the 'background' color from the theme

                Surface(

                    modifier = Modifier.fillMaxSize(),

                    color = MaterialTheme.colors.background

                ) {

                    Greeting2()

                }

            }

        }

    }

} @Composable fun Greeting2() {




Column(

    modifier = Modifier.background(color = Color.White)

        .padding(20.dp)
```

```kotlin
            .verticalScroll(rememberScrollState())
    ) {
        Text(
            fontSize = 40.sp,
            color = Color(android.graphics.Color.rgb(120, 40, 251)),
            fontFamily = FontFamily.Cursive,
            text = stringResource(id = R.string.place_3),
        )
        Image (
            painterResource(id = R.drawable.singapore), contentDescription = "",
            modifier = Modifier
                .padding(16.dp)
                .fillMaxWidth()
                .height(200.dp)
                .scale(scaleX = 1.2F, scaleY = 1F)
        )
        Text (
            color = Color.Black,
            text = "Day 1:\n" +


                    "Morning: Visit Gardens by the Bay and marvel at the Supertree Grove and the
Flower Dome and Cloud Forest conservatories.\n" +

                    "Afternoon: Explore the Marina Bay Sands complex, which includes a casino, luxury
shopping mall, and observation deck with a stunning view of the city.\n" +

                    "\n" +

                    "Day 2:\n" +
```

"Morning: Explore the historic district of Chinatown, including the Buddha Tooth Relic Temple and Museum and the Sri Mariamman Temple.\n" +

"Afternoon: Visit the nearby Clarke Quay for lunch and to explore its waterfront restaurants, bars, and shops.\n" +

"\n" +

"Day 3:\n" +

"Morning: Take a tour of the UNESCO-listed Botanic Gardens, one of the world's most famous and significant tropical gardens.\n" +

"Afternoon: Head over to the National Museum of Singapore, which houses a vast collection of hi
storical and cultural artifacts.\n" +

"\n" +

"Day 4:\n" +

"Morning: Visit the Singapore Zoo and admire the wildlife, including orangutans, tigers, and elephants.\n" +

"Afternoon: Head over to Sentosa Island and relax at one of its many beaches or try some of the many attractions such as Universal Studios Singapore or Adventure Cove Waterpark.\n" +

"\n" +

"Day 5:\n" +

"Morning: Go on a nature walk at MacRitchie Reservoir, which o ers hiking trails and stunning views of the city skyline.\n" +

"Afternoon: Visit Little India, a vibrant and colorful neighborhood, and explore the shops, temples, and food stalls.\n" +

"\n" +

```
        "Day 6:\n" +


        "Morning: Explore the trendy neighborhood of Tiong Bahru, known for its hip cafes
and boutiques, as well as its Art Deco architecture.\n" +

        "Afternoon: Visit the National Gallery Singapore, which houses the largest public
collection of modern art in Singapore and Southeast Asia.\n" +

        "\n" +

        "Day 7:\n" +


        "Morning: Take a day trip to the nearby island of Pulau Ubin, where you can rent a "

    )

  }

}
```

Modifying AndroidManifest.Xml

```xml
<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools">


    <application

        android:allowBackup="true"

        android:dataExtractionRules="@xml/data_extraction_rules"

        android:fullBackupContent="@xml/backup_rules"

        android:icon="@mipmap/ic_launcher"

        android:label="@string/app_name"

        android:supportsRtl="true"

        android:theme="@style/Theme.TravelApp"
```

```xml
    tools:targetApi="31">

    <activity
        android:name=".RegisterActivity"

        android:exported="false"

        android:label="RegisterActivity"

        android:theme="@style/Theme.TravelApp" />

    <activity
        android:name=".SingaporeActivity"

        android:exported="false"

        android:label="@string/title_activity_singapore"

        android:theme="@style/Theme.TravelApp" />

    <activity
        android:name=".ParisActivity"

        android:exported="false"

        android:label="@string/title_activity_paris"

        android:theme="@style/Theme.TravelApp" />

    <activity
        android:name=".BaliActivity"

        android:exported="false"

        android:label="@string/title_activity_bali"

        android:theme="@style/Theme.TravelApp" />

    <activity
        android:name=".MainActivity"

        android:exported="true"

        android:label="@string/app_name"
```

```xml
        android:theme="@style/Theme.TravelApp"/>
    <activity
        android:name=".LoginActivity"
        android:exported="true"
        android:label="@string/app_name"
        android:theme="@style/Theme.TravelApp">
        <intent-filter>
            <action
android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
  </application>

</manifest>
```

Login

Username

Password

Successfully log in

Login

Register   Forget password?

Register

Username

Email

Password

Register

Have an account?   Log in

Register

Username

kethsiyal

Email

kethsiyaldj@gmail.com

Password

.........|

Register

Have an account?   Log in

Login

Username

Password

Login

Register   Forget password?
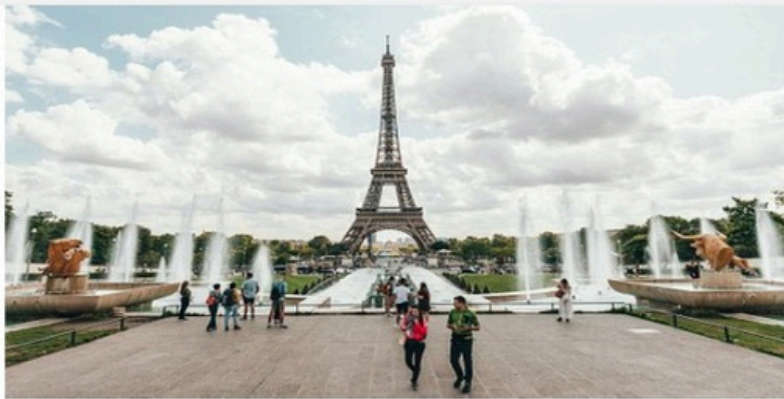
Bali

Super saver pack with less than $10000

7days/2persons

Paris

Super saver pack with less than $10000

7days/2persons

accommodation.
Spend the day relaxing and getting acclimated to
the island.
If you have time, explore the nearby area or head
to the beach.

Day 2: Ubud Tour
Start your day early and head to Ubud, a cultural
and artistic hub in Bali.
Visit the Monkey Forest and the Ubud Palace.
Take a tour of the Tegalalang Rice Terrace, a
beautiful UNESCO World Heritage Site.
End your day with a traditional Balinese dance
performance.

Day 3: Temple Hopping
Visit some of Bali's most famous temples, such as
Tanah Lot and Uluwatu.
Take in the stunning views of the ocean and cliffs.
Enjoy a sunset dinner at one of the many
restaurants near the temples.

Day 4: Waterfalls and Beaches
Take a day trip to Bali's beautiful waterfalls, such
as Tegenungan or Gitgit.
Spend the afternoon at one of Bali's
world-renowned beaches, like Seminyak or Nusa
Dua.

Day 5: Island Hopping
Take a day trip to one of Bali's neighboring islands,
such as Nusa Lembongan or Gili Islands.
Snorkel or scuba dive in the clear waters and relax
on the beach.

Day 6: Cultural Activities
Visit a traditional Balinese village and learn about
the island.

Day 7: Departure
Explore the surrounding area and take in the

Day 1: Arrival and Introduction
Check into your accommodation and freshen up
Take a stroll around the neighborhood to get acquainted
Visit the Eiffel Tower, preferably in the evening when it is lit up
Have a relaxing dinner at a nearby restaurant

Day 2: Art and History
Visit the Louvre Museum to see some of the world's most famous art pieces
Stroll through the Tuileries Garden and the Place de la Concorde
Visit the Orsay Museum, which houses a large collection of impressionist art
Have dinner at a local French restaurant

Day 3: French Culture and Food
Visit the Montmartre neighborhood to see the famous Basilique du Sacré-Cœur and Place du Tertre
Explore the historic neighborhood of Le Marais
Try some delicious French pastries at a local bakery
Have dinner at a brasserie to taste some classic French cuisine

Day 4: Architecture and Gardens
Visit the Palace of Versailles, a UNESCO World Heritage site, and explore its beautiful gardens
Walk along the Champs-Elysées and stop at the Arc de Triomphe
Visit the Sainte-Chapelle, a beautiful Gothic chapel with stunning stained-glass windows
Have dinner at a local restaurant in the 7th arrondissement

Day 5: Shopping and Sightseeing
Visit the Notre-Dame Cathedral and climb up to the top for a stunning view of the city
Explore the Latin Quarter and visit the Panthéon

Day 1:
Morning: Visit Gardens by the Bay and marvel at the Supertree Grove and the Flower Dome and Cloud Forest conservatories.
Afternoon: Explore the Marina Bay Sands complex, which includes a casino, luxury shopping mall, and observation deck with a stunning view of the city.

Day 2:
Morning: Explore the historic district of Chinatown, including the Buddha Tooth Relic Temple and Museum and the Sri Mariamman Temple.
Afternoon: Visit the nearby Clarke Quay for lunch and to explore its waterfront restaurants, bars, and shops.

Day 3:
Morning: Take a tour of the UNESCO-listed Botanic Gardens, one of the world's most famous and significant tropical gardens.
Afternoon: Head over to the National Museum of Singapore, which houses a vast collection of historical and cultural artifacts.

Day 4:
Morning: Visit the Singapore Zoo and admire the wildlife, including orangutans, tigers, and elephants.
Afternoon: Head over to Sentosa Island and relax at one of its many beaches or try some of the many attractions such as Universal Studios Singapore or Adventure Cove Waterpark.

Day 5:
Morning: Go on a nature walk at MacRitchie Reservoir, which offers hiking trails and stunning views of the city skyline.
Afternoon: Visit Little India, a vibrant and colorful neighborhood, and explore the shops, temples, and food stalls.

Thank You