A PERSONALISED TRAVEL PLANNING AND TRACKING APP

ABSTRACT:

This app offers a seamless travel experience by combining personalized itinerary planning with real-time tracking and social engagement. Leveraging AI and data-driven insights, it tailors travel recommendations to individual preferences, ensuring a unique journey for every user.

Key functionalities include an intuitive itinerary builder, budget management tools, and real-time navigation. Users can log and track their experiences through travel journals, expense trackers, and health monitors. The app fosters community through shared itineraries, reviews, and social challenges, enhancing the overall travel experience.

INTRODUCTION:

Traveling is one of the most enriching experiences, yet planning and managing a trip can often be overwhelming. From creating itineraries to staying within budget, and ensuring all activities go smoothly, travelers face numerous challenges. A personalized travel planning and tracking app aims to simplify this process, offering a one-stop solution tailored to individual preferences and needs.

This app leverages advanced technologies, such as artificial intelligence and realtime data integration, to provide users with personalized travel recommendations, efficient itinerary planning, and seamless tracking features. Whether you're an adventurer, a cultural explorer, or a leisure traveler, the app adapts to your unique travel style, ensuring a more enjoyable and stress-free experience. By combining convenience, personalization, and social connectivity, this app redefines how we plan, experience, and share our journeys

DOMAIN:

The domain of personalized travel planning and tracking focuses on developing digital solutions that enhance the travel experience through tailored planning, real-time management, and

post-trip reflection. This domain intersects various fields, including tourism, technology, data analytics, and user experience design.

In the domain of travel technology, personalized travel planning and tracking apps serve as dynamic tools for enhancing user experiences. These apps leverage AI and data analytics to offer tailored recommendations for destinations, activities, and accommodations based on individual preferences. They also facilitate real-time itinerary management and location tracking, ensuring a seamless travel experience. By integrating features like offline maps, expense tracking, and social sharing, these apps cater to both solo travelers and group trips. Their role extends beyond convenience, providing users with unique insights and efficient planning tools to optimize their journeys.

SYSTEM REQUIREMENTS:

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

TOOLS USED:

- Android Studio
- Firebase
- Kotlin

PROPOSED SYSTEM:

Wanderlust is a personalized travel planning and tracking app that leverages Android's Jetpack Compose toolkit to provide a seamless, responsive user interface. The core features include:

- **Personalized Accommodations Feed**: The app uses location-based recommendations to suggest accommodations, curated based on user preferences and travel history.
- **Itinerary Planning and Management**: Users can create, edit, and manage their travel itineraries.
- **Travel Tracking**: Allows users to track their past and current travels, including check-ins, photos, and notes.

- **Explore Destinations**: Recommends popular places, attractions, and experiences based on the user's selected location.
- **Booking Integration** (Optional): Links to external booking platforms for flights, hotels, and transportation.

WORKING OF THE SYSTEM:

1. **User Onboarding and Setup**:

   New users sign up and fill out a brief survey about travel preferences, including favored types of accommodations, activity interests, and desired trip frequencies.

2. **Personalized Recommendations**:
   - Upon location selection, the app fetches relevant data (using APIs such as Google Places or similar) to provide recommendations for accommodations and experiences.
   - Users receive suggestions based on previous trips, preferred types of stays, and other travel preferences.
3. **Itinerary Planning and Management**:
   - Users can manually enter details for each segment of their travel itinerary or select from recommended options.
   - The app allows users to view their itinerary in calendar or list formats and set reminders for bookings and events.
4. **Travel Tracking**:
   - The app uses GPS to record check-ins and save photos or notes associated with visited places.
   - Users can look back on previous trips and access their travel history.

CODE IMPLEMENTATION(sample code):

LoginActivity

```
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)
        }
    }
}
@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)
```

```
    )
```

MainActivity

```
import androidx.compose.material.Card
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.stringResource
import androidx.compose.ui.text.font.FontFamily
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 MainActivity : ComponentActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContent {
         TravelApp(this)
      }
   }


   @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"
        )

        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)
                .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,
        )
```

ParisActivity

```
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
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)
            .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),
        )
        Image(
            painterResource(id = R.drawable.paris), contentDescription = "",
            modifier = Modifier
                .padding(16.dp)
                .fillMaxWidth()
                .height(200.dp)
                .scale(scaleX = 1.2F, scaleY = 1F)
        )
```

BaliActivity

```
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
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()
                }
            }
        }
```

```
    }
}

@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),
        )
        Image(
            painterResource(id = R.drawable.bali), contentDescription = "",
            modifier = Modifier
                .padding(16.dp)
                .fillMaxWidth()
                .height(200.dp)
                .scale(scaleX = 1.2F, scaleY = 1F)
        )
```

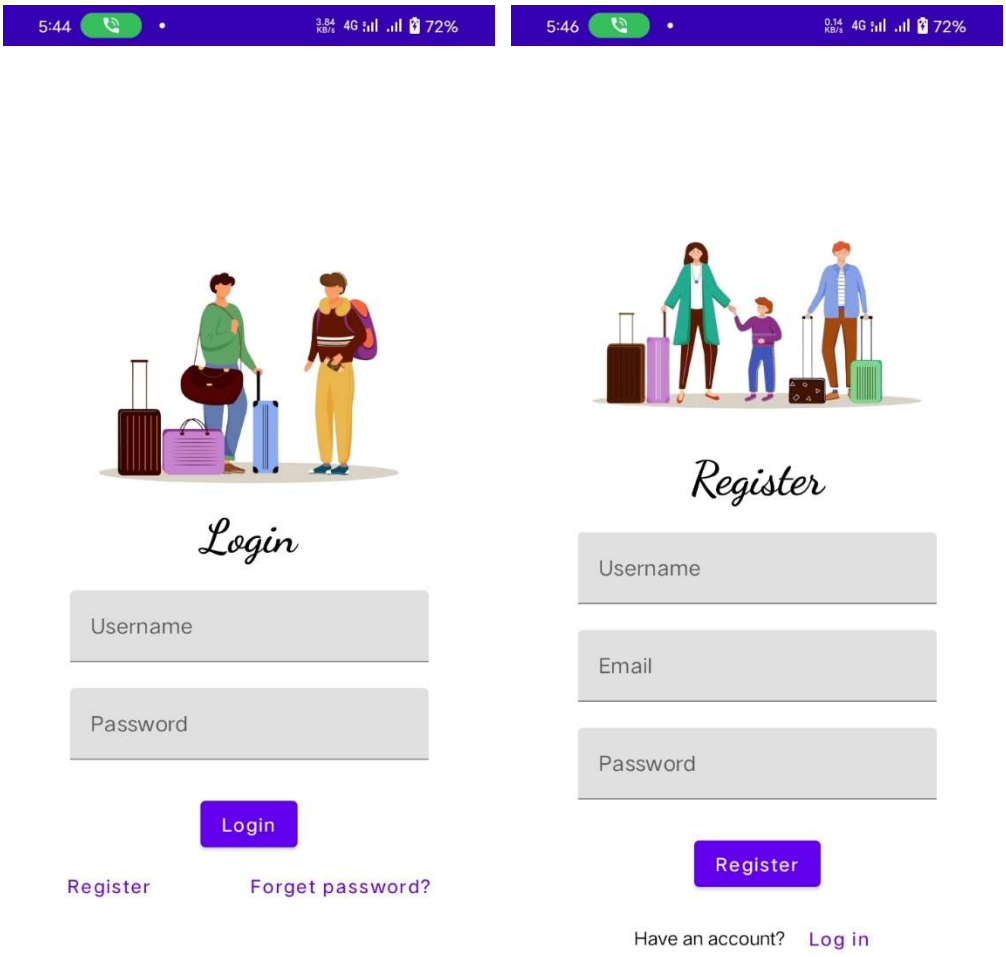
SingaporeActivity

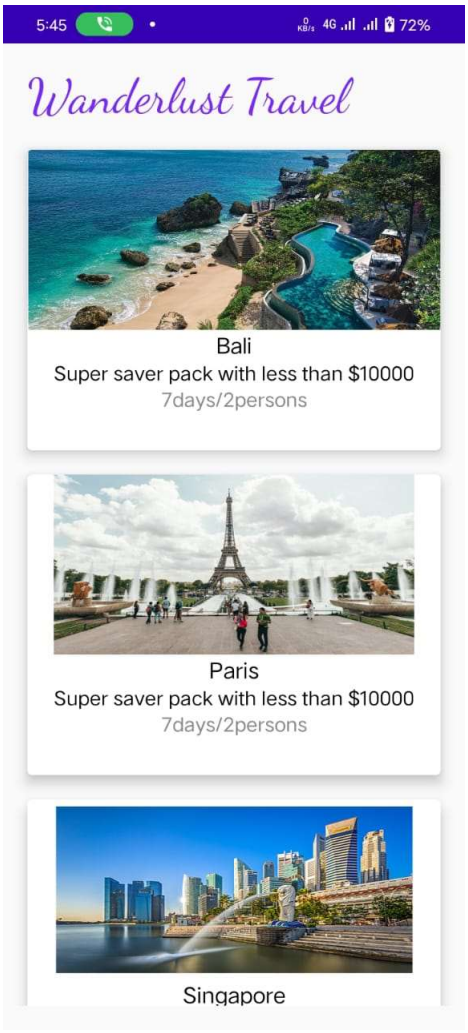
```
import androidx.compose.material.Text
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
```

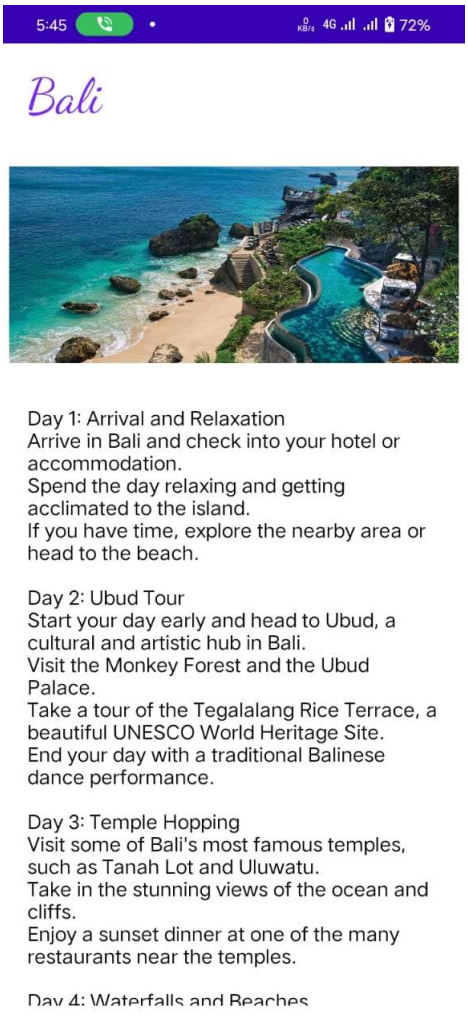
```
import androidx.compose.ui.draw.scale
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.res.stringResource
import androidx.compose.ui.text.font.FontFamily
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
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()
                }
            }
        }
    }
}
```

OUTPUT:


5:44
72%
Login
Username
Password
Login
Register
Forget password?
5:46
72%
Register
Username
Email
Password
Register
Have an account?
Log in


5:45
72%
Wanderlust Travel
Bali
Super saver pack with less than $10000
7days/2persons
Paris
Super saver pack with less than $10000
7days/2persons
Singapore


5:45
72%
Bali
Day 1: Arrival and Relaxation
Arrive in Bali and check into your hotel or 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

5:45 • 0 KB/s 4G 72%

Singapore

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.

5:45 • 15.75 KB/s 4G 72%

Paris

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

FUTURE ENHANCEMENTS:

1. **AI-Powered Recommendations**: Incorporate machine learning algorithms to analyze user behavior and improve personalized recommendations.
2. **Social Sharing and Collaboration**: Enable users to share itineraries or recommendations with friends or collaborate on group travel plans.
3. **Offline Access**: Allow users to download parts of their itinerary and maps for access in areas with limited connectivity.
4. **Expense Tracking**: Integrate an expense tracker that automatically categorizes travel-related expenses to help users manage budgets effectively.
5. **Integration with AR and VR**: Provide virtual previews of recommended accommodations or points of interest.
6. **Advanced Analytics**: Track users' travel patterns and offer insights into their travel habits, such as most-visited types of places or favorite destinations.

CHALLENGES AND SOLUTIONS:

1. **Challenge: Gathering High-Quality Location Data**
    - **Solution**: Use reputable APIs like Google Places, Foursquare, or Booking.com APIs to retrieve high-quality and accurate location data. Consider caching essential data locally to reduce API calls.
2. **Challenge: Personalizing Recommendations**
    - **Solution**: Implement user-specific settings and preferences storage, allowing the app to offer more targeted recommendations. Basic machine learning algorithms, such as collaborative filtering, can be applied to improve suggestions over time.
3. **Challenge: Battery and Data Usage**
    - **Solution**: Optimize GPS usage by turning it off when not required and using lower-power location services when feasible. For data-heavy features, prompt users to switch to Wi-Fi for downloads or offline mode.
4. **Challenge: Ensuring a Smooth Compose UI Experience**
    - **Solution**: Take advantage of Jetpack Compose’s performance benefits but optimize complex layouts and animations to ensure responsiveness, especially on lower-end devices. Conduct thorough testing across devices.

CONCLUSION:

In conclusion, a personalized travel planning and tracking app offers a dynamic, user-centric solution to enhance the travel experience from start to finish. By combining seamless itinerary management, real-time booking capabilities, personalized recommendations, and integrated tracking features, the app ensures that users have everything they need at their fingertips. With a focus on data security, scalability, and crossplatform compatibility, the app is designed to deliver a smooth, reliable experience on a range of devices. By providing essential features such as offline access, real-time notifications, and emergency assistance, the app not only simplifies travel planning but also offers peace of mind throughout the journey. This combination of innovation, convenience, and personalization positions the app as an invaluable tool for modern travelers, ensuring a more enjoyable, efficient, and stress-free travel experience.