Sustainable Tourism Guide

BACHELOR OF

TECHNOLOGY IN

COMPUTER SCIENCE AND ENGINEERING

BY

Y. Anusha	**(22501A05J7)**
U. Harshita	**(22501A05I6)**
Y. Srija	**(22501A05J4)**
Y. Abigna	**(22501A05J6)**

Under the Guidance of

Mr. Michael Sadgun Rao Kona,

Assistant Professor

PRASAD V POTLURI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(Permanently affiliated to JNTU: Kakinada, Approved by AICTE)

(An NBA & NAAC A++ accredited and ISO 9001:2015 certified institution)

Kanuru, Vijayawada-520007

2024-25

PRASAD V POTLURI

SIDDHARTHA INSTITUTE OF TECHNOLOGY

(Permanently affiliated to JNTU :: Kakinada, Approved by AICTE)

(An NBA & NAAC A++ accredited and ISO 9001:2015 certified institution)

Kanuru, Vijayawada – 520007

CERTIFICATE

This is to certify that the project report title **"Sustainable Tourism Guide"** is the bonafide work of **Y. Anusha (22501A05J7), U. Harshita (22501A05I6), Y. Srija (22501A05J4), Y. Abigna (22501A05J6)** in partial fulfilment of completing the Academic project in Mobile App Development (20SA8651) during the academic year 2024-25.

Signature of the Incharge **Signature of the HOD**

INDEX

1.1. SDG JUSTIFICATION REPORT

SDG Mapped: SDG 12 – Responsible Consumption and Production

1.1.1 How This Project Supports SDG 12

The **Sustainable Tourism Guide** mobile application directly contributes to achieving **SDG 12** by promoting eco-friendly travel choices, encouraging responsible tourism practices, and fostering environmental conservation. The project aligns with **SDG Target 12.b**, which focuses on developing sustainable tourism practices that create jobs and promote local culture and products. Key contributions of this project to SDG 12 are as follows:

- **Promoting Sustainable Travel**: The app connects tourists with environmentally responsible accommodations and activities, ensuring their choices align with sustainability principles.
- **Reducing Environmental Impact**: By encouraging eco-conscious tourism, the platform helps reduce pollution, resource depletion, and habitat destruction caused by unsustainable travel practices.
- **Encouraging Responsible Tourism Management**: The application enables tourism managers to list eco-friendly accommodations and activities, supporting businesses that prioritize sustainability.
- **Enhancing Awareness and Education**: The app provides tourists with information on eco- certifications, sustainable tourism tips, and updates on responsible travel practices, fostering awareness and education.
- **Scalability for Greater Impact**: The system is designed for future enhancements, including AI-driven recommendations, partnerships with sustainability certification organizations, and expansion to additional destinations.

By integrating these sustainable tourism strategies, the **Sustainable Tourism Guide** contributes to global efforts in promoting responsible consumption and production patterns, ultimately supporting a more eco-friendly and sustainable future for the tourism industry.

1. ABSTRACT

The **Sustainable Tourism Guide** is a mobile application designed to promote eco-friendly travel practices by connecting tourists with sustainable accommodations and activities. It serves as a structured platform where users can explore destinations, discover responsible travel options, and engage in environmentally conscious tourism.

The app features two user roles: **Tourists** and **Managers**. Tourists can search for locations and view curated lists of eco-friendly activities and accommodations, while Managers can list sustainable tourism options, providing details to attract responsible travelers. The platform ensures a seamless experience with an intuitive interface for browsing, registering, and managing travel plans.

Developed using **Java in Android Studio**, the app delivers a robust and user-friendly experience. The backend efficiently handles user authentication and data management, ensuring smooth interactions between tourists and service providers. By encouraging sustainable choices and responsible travel behavior, the **Sustainable Tourism Guide** fosters environmental conservation and supports communities that prioritize eco-friendly tourism.

This application enhances awareness about sustainable travel, simplifies trip planning, and contributes to a greener future by bridging the gap between eco-conscious travelers and responsible tourism providers.

2. INTRODUCTION

Tourism plays a crucial role in global economic development but often comes at the cost of environmental degradation. Unsustainable travel practices contribute to pollution, habitat destruction, and excessive resource consumption. To address these challenges, the **Sustainable Tourism Guide** is designed as a mobile application that promotes eco-friendly travel and responsible tourism.

The **Sustainable Tourism Guide** serves as a structured platform that connects travelers with environmentally responsible accommodations and activities. The application enables users to explore destinations while ensuring their travel choices align with sustainability principles. By offering a categorized and well-organized listing of eco-conscious tourism options, the app helps tourists make informed decisions that minimize their environmental impact.

The app features two primary user roles: **Tourists** and **Managers**. Tourists can search for locations and explore a variety of sustainable travel options, while Managers can list eco-friendly accommodations and activities. Through real-time updates and user-friendly navigation, the app simplifies the process of discovering and participating in responsible tourism experiences.

Developed using **Java in Android Studio**, the **Sustainable Tourism Guide** ensures a smooth and accessible mobile experience. By encouraging sustainable travel habits and bridging the gap between eco-conscious travelers and responsible tourism providers, this application contributes to the promotion of ethical and environmentally friendly tourism practices.

3. OBJECTIVES AND SCOPE OF THE PROJECT

Objectives:

The **Sustainable Tourism Guide** is designed to promote eco-friendly travel by providing a structured platform for discovering and managing sustainable tourism options. The project focuses on the following key objectives:

1. **Promote Sustainable Travel Practices:**
   The app encourages responsible tourism by connecting travelers with environmentally friendly accommodations, activities, and destinations, helping to reduce the ecological footprint of travel.
2. **Enhance Accessibility to Eco-Conscious Tourism Options:**
   By offering a categorized and searchable platform, the app ensures that tourists can easily explore sustainable travel choices based on location, type of activity, and environmental impact.
3. **Facilitate Efficient Tourism Management:**
   Tourism managers can list and manage eco-friendly accommodations and activities, providing accurate details to attract responsible travelers. The app allows for seamless content management and real-time updates.
4. **Improve User Engagement and Awareness:**
   Tourists can receive notifications about new sustainable tourism opportunities, ensuring they stay informed about eco-friendly travel experiences. This helps raise awareness about responsible tourism practices.
5. **Ensure Scalability and Future Expansion:**
   The platform is designed for future enhancements such as AI-driven travel recommendations, user-generated reviews, and integration with sustainability certification organizations. It also allows for expansion to include more destinations and eco-tourism services.

Scope of the Project:

The **Sustainable Tourism Guide** extends beyond a basic travel directory; it serves as a **comprehensive eco-tourism platform** that bridges the gap between environmentally conscious travelers and responsible tourism providers. The app facilitates structured exploration, booking, and management of sustainable travel options, ensuring both accessibility and reliability.

The platform provides a **mobile-first experience**, offering an intuitive and user-friendly interface for travelers and tourism managers. It supports **location-based searches, role-based access, and real-time updates**, making sustainable travel planning more efficient and engaging.

By integrating smart tourism management tools, the **Sustainable Tourism Guide** aims to **redefine travel experiences**, encouraging eco-conscious decisions and contributing to global sustainability efforts.

4. SOFTWARE USED – EXPLANATION

The **Sustainable Tourism Guide** is developed using modern mobile development technologies to ensure a seamless and interactive experience for both tourists and tourism managers. The key technologies used in the development include:

Frontend Technologies:

Java (Android Studio)

- A widely used programming language for building robust and scalable Android applications.
- Provides native performance and direct access to Android APIs for seamless user interactions.
- Supports structured object-oriented programming, making the codebase maintainable and efficient.
- Ensures compatibility with various Android devices, optimizing user experience.

XML (User Interface Design)

- Used for designing intuitive and responsive UI layouts in Android applications.
- Provides a structured approach to defining UI elements, ensuring consistency across screens.
- Supports ConstraintLayout, RecyclerView, and other Android UI components for flexible layouts.

Backend Technologies:

Firebase (Authentication & Database - Optional)

- A cloud-based backend service for authentication and real-time database management.
- Provides seamless user authentication with email/password and Google sign-in integration.
- Supports Firestore for storing user profiles, tourism listings, and user interactions securely.

By integrating these database and storage tools, the Sustainable Tourism Guide ensures a structured, scalable, and efficient data management system that enhances user engagement and simplifies eco-friendly travel planning.

5. PROPOSED MODEL

The **Sustainable Tourism Guide** aims to provide an efficient and user-friendly platform for promoting eco-friendly travel, enabling users to discover and engage with sustainable tourism options. The system is structured into various key components:

Home Page & Navigation:

- The landing page offers an intuitive interface where tourists can search for eco-friendly destinations, accommodations, and activities.
- Simplified navigation ensures smooth user interactions, allowing both tourists and tourism managers to explore listings effortlessly.

Location-Based Sustainable Tourism Listings:

- The platform categorizes sustainable travel options based on location, type of activity, and eco-friendly certifications.
- Users can browse listings, read descriptions, and access detailed information to make responsible travel choices.

Tourism Manager Dashboard:

- Managers can register and list sustainable accommodations or activities by providing details such as name, location, eco-certifications, pricing, and availability.

Real-Time Updates & Notifications:

- Users receive real-time notifications about new listings, updates to existing options, or upcoming sustainable tourism events.
- Push notifications ensure tourists stay informed about limited-time eco-tourism opportunities.

Security & Role-Based Access Control:

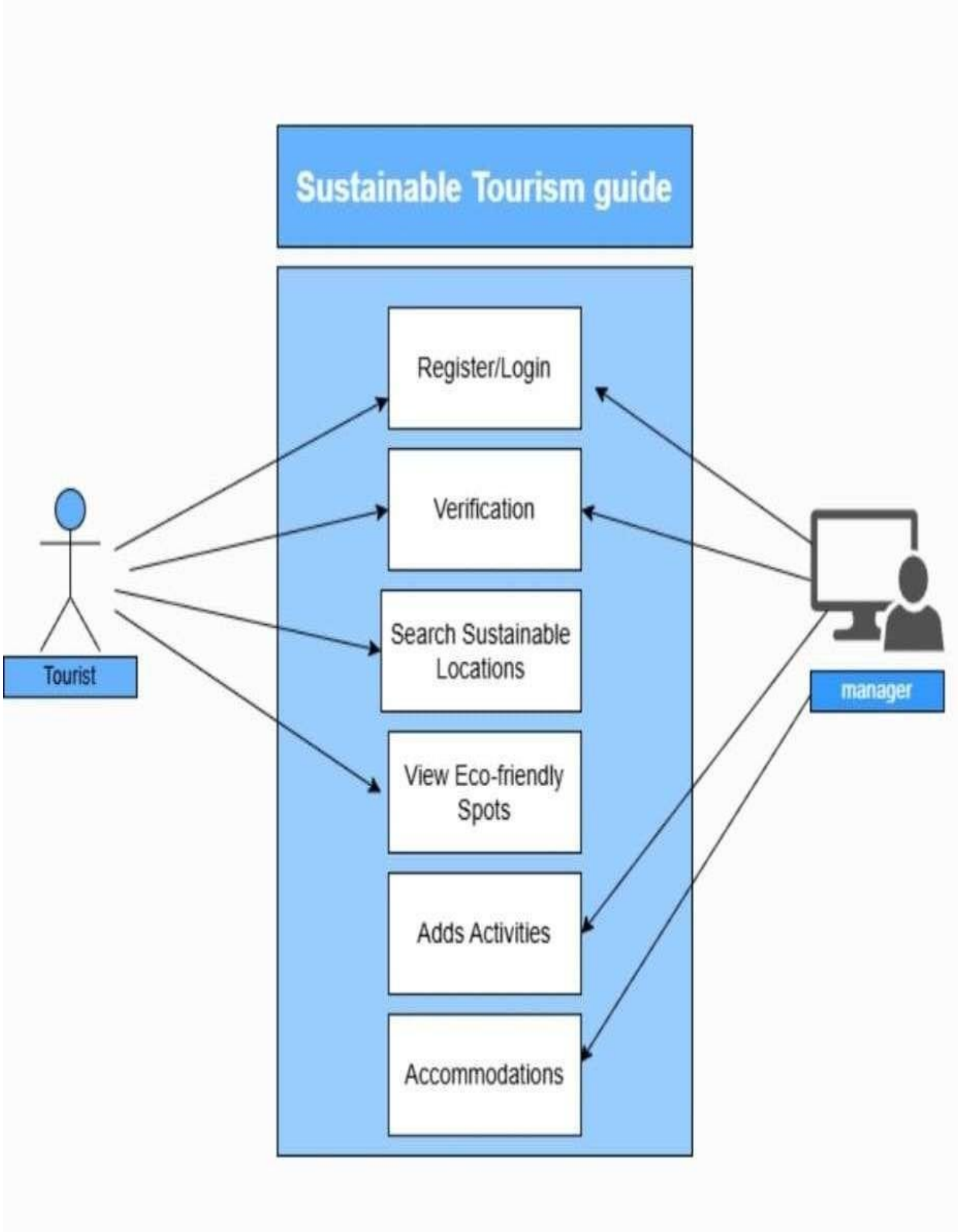
- The system ensures secure access, allowing only verified managers to add and manage tourism listings.
- User authentication mechanisms protect personal data and prevent unauthorized access.

Responsive Design & Scalability:

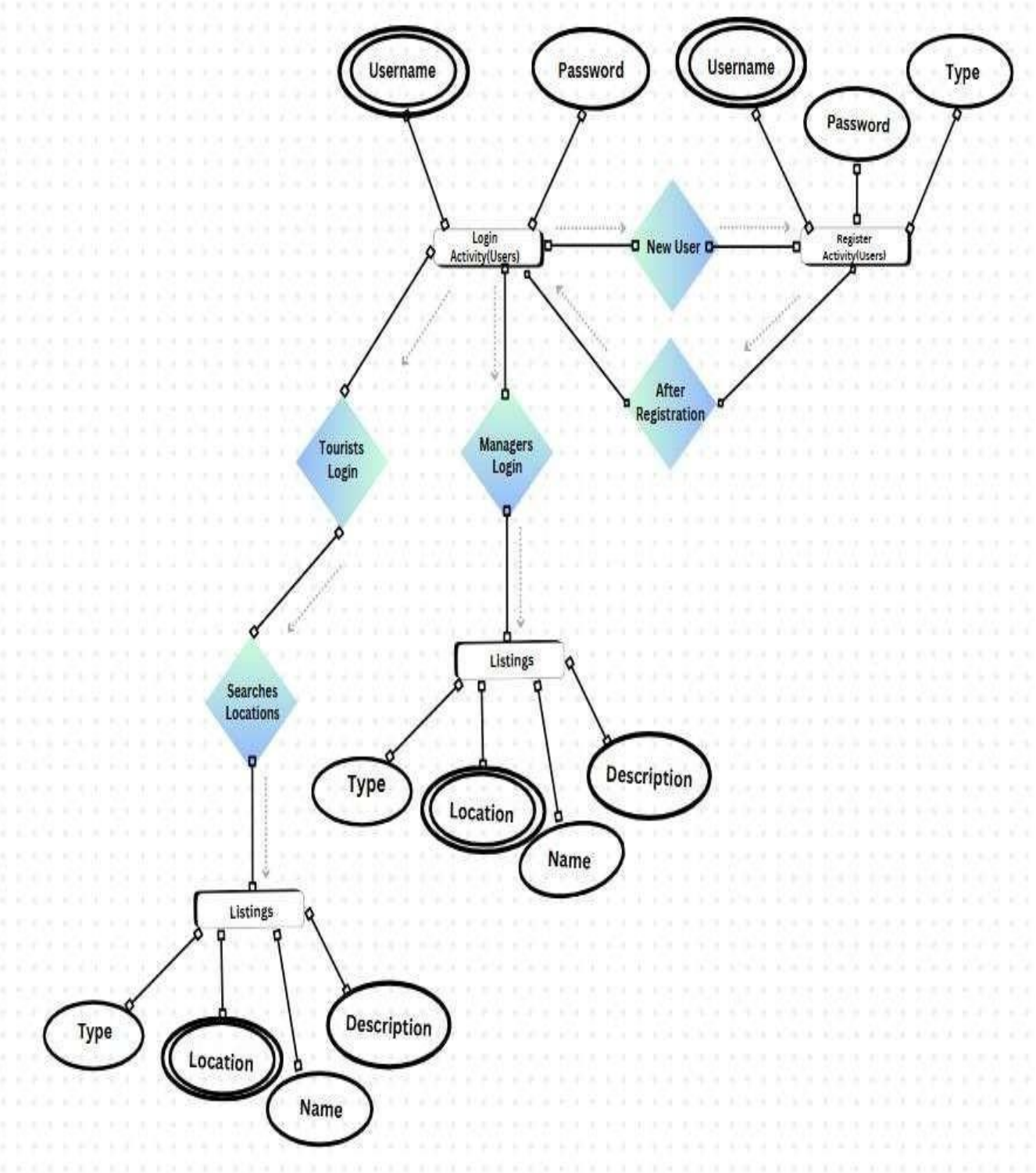
- The app follows a mobile-first approach, ensuring a seamless experience across various devices.
- Future enhancements may include AI-driven travel recommendations, integration with eco-certification agencies, and an interactive map for real-time navigation of sustainable destinations.

By integrating these features, the **Sustainable Tourism Guide** simplifies eco-friendly travel planning, enhances user engagement, and promotes sustainable tourism practices worldwide.

Use Case Diagram:



Entity-Relationship (ER) Diagram:



6. SAMPLE CODE

GitHub repository links:

https://github.com/HARSHITAUPPALAPATI/Sustainable-Tourism-Guide

Folder Structure:

tourism-guide

```
├── app
│   ├── manifests
│   │   ├── AndroidManifest.xml
│
│   ├── java
│   │   ├── com.example.tourism_guide
│   │   │   ├── DatabaseHelper.java
│   │   │   ├── LoginActivity.java
│   │   │   ├── MainActivity.java
│   │   │   ├── ManagerActivity.java
│   │   │   ├── RegisterActivity.java
│   │   │   ├── TouristActivity.java
│
│   ├── res
│   │   ├── drawable
│   │   │   ├── ic_launcher_background.xml
│   │   │   ├── ic_launcher_foreground.xml
│   │   │   ├── image_mad.jpeg
│   │   ├── layout
│   │   │   ├── activity_database_helper.xml
│   │   │   ├── activity_login.xml
│   │   │   ├── activity_main.xml
│   │   │   ├── activity_manager.xml
│   │   │   ├── activity_register.xml
│   │   │   ├── activity_tourist.xml
│   │   ├── mipmap
│   │   ├── values
│   │   ├── xml
│   │   │   ├── backup_rules.xml
│   │   │   ├── data_extraction_rules.xml
```

│
├── Gradle Scripts
│ ├── build.gradle.kts (Project: tourism-guide)
│ ├── build.gradle.kts (Module: app)
│ ├── proguard-rules.pro
│ ├── gradle.properties
│ ├── gradle-wrapper.properties
│ ├── libs.versions.toml
│ ├── local.properties
│ ├── settings.gradle.kts

MainActivity.java

```
package com.example.tourism_guide;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      EdgeToEdge.enable(this);
      setContentView(R.layout.activity_main);
      ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
         Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
         v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
         return insets;
      });
      // Find buttons
      Button btnEnter = findViewById(R.id.btnEnter);
      // Set click listeners
      btnEnter.setOnClickListener(v -> {
         Intent intent = new Intent(MainActivity.this, LoginActivity.class);
         startActivity(intent);
      });
   }
```

RegisterActivity.java

```
package com.example.tourism_guide;
import android.content.ContentValues;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class RegisterActivity extends AppCompatActivity {
    private EditText etUsername, etPassword;
    private RadioGroup rgRole;
    private Button btnRegister;
    private DatabaseHelper dbHelper;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);
        // Initialize UI elements
        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        rgRole = findViewById(R.id.rgRole);
        btnRegister = findViewById(R.id.btnRegister);
        // Initialize database helper
        dbHelper = new DatabaseHelper(this);
        // Register button click listener
        btnRegister.setOnClickListener(v -> registerUser());
    }
    private void registerUser() {
        String username = etUsername.getText().toString().trim();
        String password = etPassword.getText().toString().trim();
        int selectedRoleId = rgRole.getCheckedRadioButtonId();

        // Validate input
        if (username.isEmpty() || password.isEmpty() || selectedRoleId == -1) {
            Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show();
            return;
        }
        String role = ((RadioButton) findViewById(selectedRoleId)).getText().toString();
        // Store user in database
        SQLiteDatabase db = dbHelper.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("username", username);
        values.put("password", password);
```

```
        values.put("role", role);
        long newRowId = db.insert("Users", null, values);
        db.close();
        if (newRowId != -1) {
            Toast.makeText(this, "Registration Successful", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(this, LoginActivity.class)); // Navigate to Login
            finish();
        } else {
            Toast.makeText(this, "Registration Failed", Toast.LENGTH_SHORT).show();
        }
    }
}
```

LoginActivity.java

```
package com.example.tourism_guide;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class LoginActivity extends AppCompatActivity {
    private EditText etUsername, etPassword;
    private Button btnLogin, btnRegister;
    private TextView tvNewUser; // Text prompting new users
    private DatabaseHelper dbHelper;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        // Initialize UI elements
        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        btnLogin = findViewById(R.id.btnLogin);
        btnRegister = findViewById(R.id.btnRegister);
        tvNewUser = findViewById(R.id.tvNewUser); // New user text
        // Initialize database helper
        dbHelper = new DatabaseHelper(this);
        // Login button click listener
        btnLogin.setOnClickListener(v -> loginUser());
        // Register button click listener
        btnRegister.setOnClickListener(v -> {
            Intent intent = new Intent(LoginActivity.this, RegisterActivity.class);
```

```
            startActivity(intent);
        });
    }
    private void loginUser() {
        String username = etUsername.getText().toString().trim();
        String password = etPassword.getText().toString().trim();
        // Validate input
        if (username.isEmpty() || password.isEmpty()) {
            Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show();
            return;
        }
        // Check user credentials
        SQLiteDatabase db = dbHelper.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT role FROM Users WHERE username=? AND password=?", new
String[]{username, password});
        if (cursor.moveToFirst()) {
            String role = cursor.getString(0);
            cursor.close();
            db.close();
            Toast.makeText(this, "Login Successful", Toast.LENGTH_SHORT).show();
            // Redirect based on role
            if ("Tourist".equals(role)) {
                startActivity(new Intent(this, TouristActivity.class));
            } else {
                startActivity(new Intent(this, ManagerActivity.class));
            }
            finish(); // Close login activity after successful login
        } else {
            cursor.close();
            db.close();
            Toast.makeText(this, "Invalid Credentials", Toast.LENGTH_SHORT).show();
        }
    }
}
```

TouristActivity.java

```
package com.example.tourism_guide;
import android.database.Cursor;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.util.ArrayList;
public class TouristActivity extends AppCompatActivity {
```

```
    private EditText etLocation;
    private Button btnSearch;
    private ListView listViewListings;
    private DatabaseHelper dbHelper;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tourist);
        // Initialize UI elements
        etLocation = findViewById(R.id.etLocation);
        btnSearch = findViewById(R.id.btnSearch);
        listViewListings = findViewById(R.id.listViewListings);
        // Initialize database helper
        dbHelper = new DatabaseHelper(this);
        // Search button click listener
        btnSearch.setOnClickListener(v -> searchListings());
    }
    private void searchListings() {
        String location = etLocation.getText().toString().trim();
        if (location.isEmpty()) {
            Toast.makeText(this, "Please enter a location", Toast.LENGTH_SHORT).show();
            return;
        }
        // Retrieve listings from the database
        Cursor cursor = dbHelper.getListingsByLocation(location);
        ArrayList<String> listings = new ArrayList<>();
        if (cursor.moveToFirst()) {
            do {
                String type = cursor.getString(cursor.getColumnIndexOrThrow("type"));
                String name = cursor.getString(cursor.getColumnIndexOrThrow("name"));
                String description = cursor.getString(cursor.getColumnIndexOrThrow("description"));
                listings.add(type + ": " + name + "\nDescription: " + description);
            } while (cursor.moveToNext());
        } else {
            listings.add("No listings found for " + location);
        }
        cursor.close();
        // Display results in ListView
        ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, listings);
        listViewListings.setAdapter(adapter);
    }
}
```

ManagerActivity.java

```
package com.example.tourism_guide;
import android.content.ContentValues;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class ManagerActivity extends AppCompatActivity {
    private EditText etLocation, etName, etDescription;
    private RadioGroup rgType;
    private Button btnAddListing;
    private DatabaseHelper dbHelper;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_manager);
        // Initialize UI elements
        etLocation = findViewById(R.id.etLocation);
        etName = findViewById(R.id.etName);
        etDescription = findViewById(R.id.etDescription);
        rgType = findViewById(R.id.rgType);
        btnAddListing = findViewById(R.id.btnAddListing);
        // Initialize database helper
        dbHelper = new DatabaseHelper(this);
        // Button click listener to add a listing
        btnAddListing.setOnClickListener(v -> addListing());
    }
    private void addListing() {
        String location = etLocation.getText().toString().trim();
        String name = etName.getText().toString().trim();
        String description = etDescription.getText().toString().trim();
        int selectedTypeId = rgType.getCheckedRadioButtonId();
        if (location.isEmpty() || name.isEmpty() || description.isEmpty() || selectedTypeId == -1) {
            Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show();
            return;
        }
        // Get selected type (Activity or Accommodation)
        String type = ((RadioButton) findViewById(selectedTypeId)).getText().toString();
        // Insert into database
        boolean success = dbHelper.insertListing(type, location, name, description);
        if (success) {
            Toast.makeText(this, "Listing Added Successfully", Toast.LENGTH_SHORT).show();
            etLocation.setText("");
            etName.setText("");
```

```
            etDescription.setText("");
            rgType.clearCheck();
        } else {
            Toast.makeText(this, "Failed to Add Listing", Toast.LENGTH_SHORT).show();
        }
    }
}
```

DatabaseHelper.java

```
package com.example.tourism_guide;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "tourism_guide.db";
    private static final int DATABASE_VERSION = 1;
    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        // Create Users Table
        db.execSQL("CREATE TABLE Users (id INTEGER PRIMARY KEY AUTOINCREMENT, username
TEXT UNIQUE, password TEXT, role TEXT)");
        // Create Listings Table (for activities and accommodations)
        db.execSQL("CREATE TABLE Listings (id INTEGER PRIMARY KEY AUTOINCREMENT, type TEXT,
location TEXT, name TEXT, description TEXT)");
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS Users");
        db.execSQL("DROP TABLE IF EXISTS Listings");
        onCreate(db);

    // Insert New User
    public boolean insertUser(String username, String password, String role) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("username", username);
        values.put("password", password);
        values.put("role", role);
        long result = db.insert("Users", null, values);
        return result != -1;
    }
    // Authenticate User (Login)
```

```
    public String authenticateUser(String username, String password) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT role FROM Users WHERE username=? AND password=?", new
String[]{username, password});
        if (cursor.moveToFirst()) {
            String role = cursor.getString(0);
            cursor.close();
            return role;
        }
        cursor.close();
        return null;
    }
    // Insert New Listing (Activity or Accommodation)
    public boolean insertListing(String type, String location, String name, String description) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("type", type);
        values.put("location", location);
        values.put("name", name);
        values.put("description", description);
        long result = db.insert("Listings", null, values);
        return result != -1;
    }
    // Retrieve Listings by Location
    public Cursor getListingsByLocation(String location) {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM Listings WHERE location=?", new String[]{location});
    }
}
```

7. RESULT/OUTPUT SCREENS

Figure 1: Home Page: This is the home page which comes when we open the app. The "Enter" button here redirects to the login page for the user to login

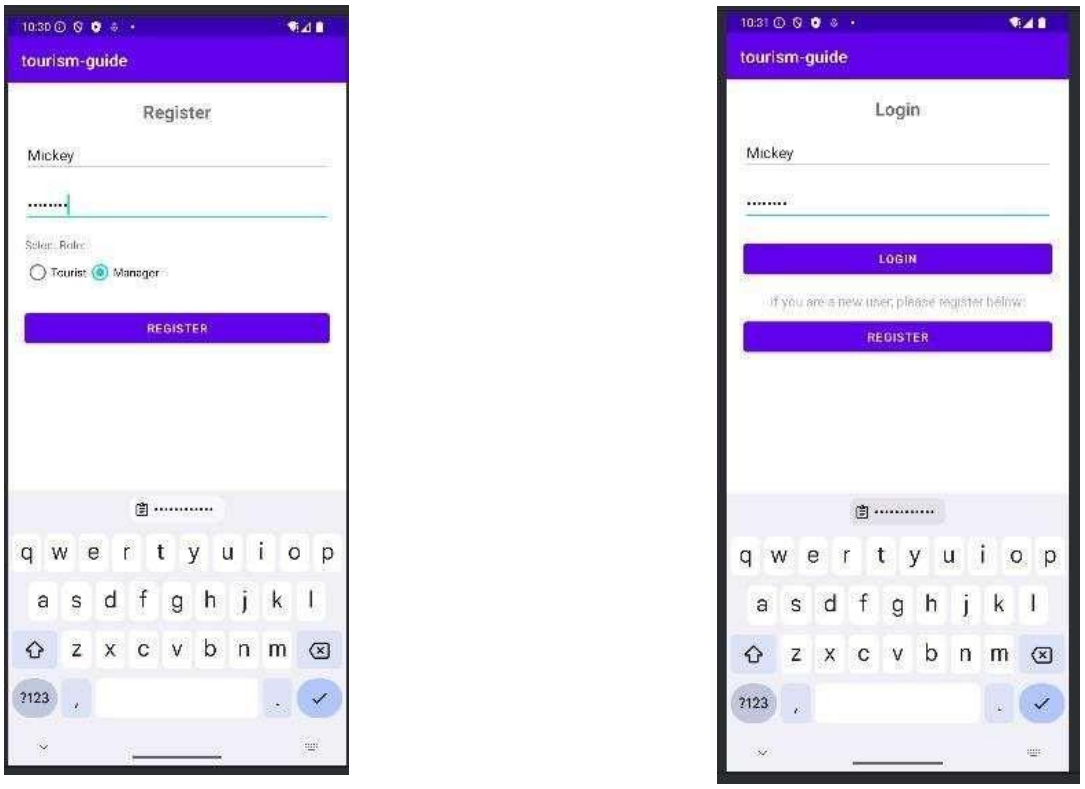


Figure 2: Register and Login- These are register and login pages for managers. The signup page allows users to create an account and use the app.

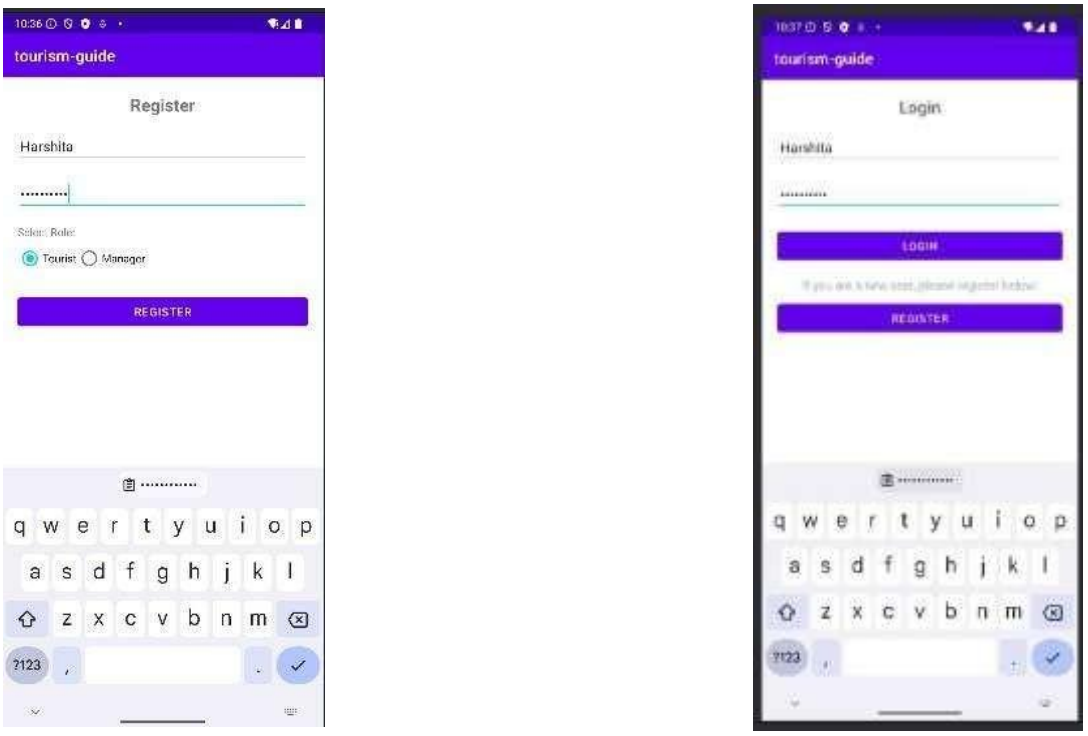
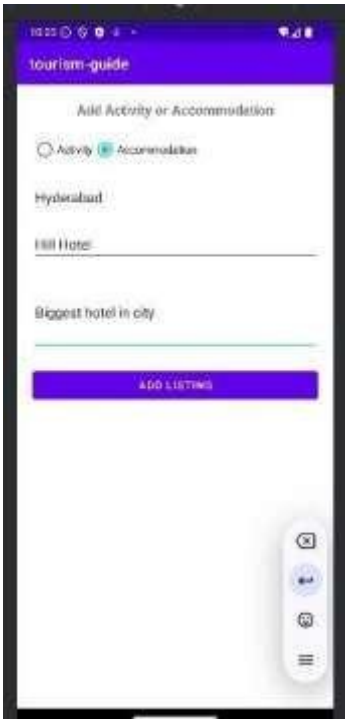


Figure 3: Register and Login - These are register and login pages for tourists. The signup page allows users to create an account and use the app.



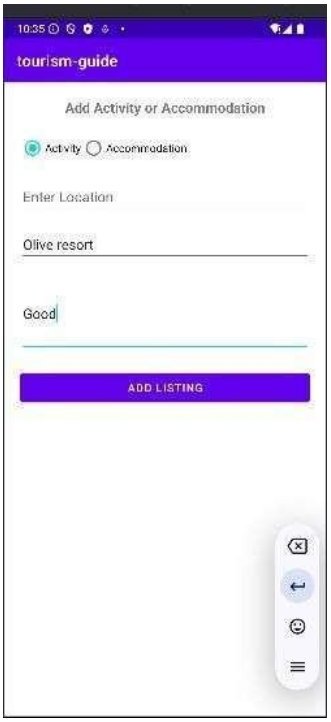


Figure 4: Add Activity or Accommodation: This is the page where the manager adds an activity or an accommodation by specifying the location, its name and description about it. After giving the details and clicking on "ADD LIST" it will store the details into the database

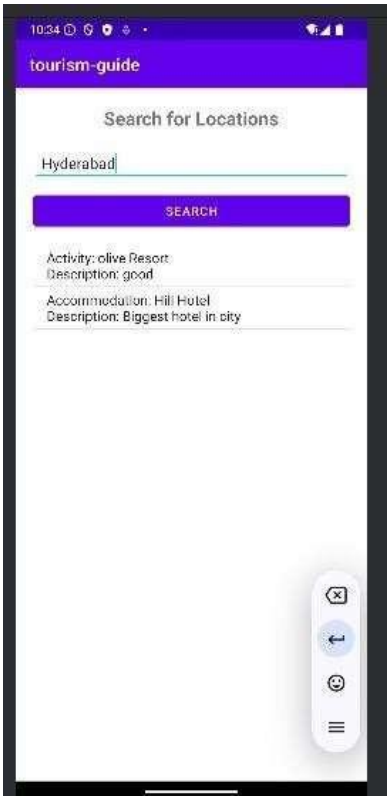


Figure 5: Search for Locations Page - This Page is for the tourists to search for the accommodations and activities in that particular location.

8. CONCLUSION & FUTURE ENHANCEMENTS

The Sustainable Tourism Guide is a user-friendly mobile application designed to promote eco-friendly travel by connecting tourists with sustainable accommodations and activities. By offering an intuitive interface and structured tourism listings, the platform simplifies travel planning while encouraging responsible tourism practices. The system ensures a seamless experience for both tourists and tourism managers, fostering greater awareness of environmentally conscious travel choices.

Developed using Java in Android Studio, the app provides a smooth and scalable experience, ensuring accessibility across a wide range of devices. The integration of location-based exploration and real-time updates enhances user engagement, making sustainable tourism more accessible and impactful.

Future Enhancements:

Future improvements will focus on expanding the app's capabilities to provide a more immersive and efficient experience. Planned enhancements include:

- AI-driven Travel Recommendations – Personalized eco-friendly travel suggestions based on user preferences.
- Interactive Map Integration – Real-time mapping of sustainable destinations and activities.
- User Reviews & Ratings – Allowing travelers to share experiences and rate tourism options.
- Offline Mode Support – Enabling access to saved listings and travel details without an internet connection.
- Integration with Eco-Certification Agencies – Verifying sustainable accommodations and activities for authenticity.

By incorporating these advanced features, the Sustainable Tourism Guide aims to become a comprehensive platform for promoting responsible travel while ensuring a seamless and informative experience for users worldwide.

9. REFERENCES (WEB SITE URLS)

GitHub repository links:

https://github.com/HARSHITAUPPALAPATI/Sustainable-Tourism-Guide

1. Android Studio Docs - https://developer.android.com/develop

2. Firebase - https://firebase.google.com/