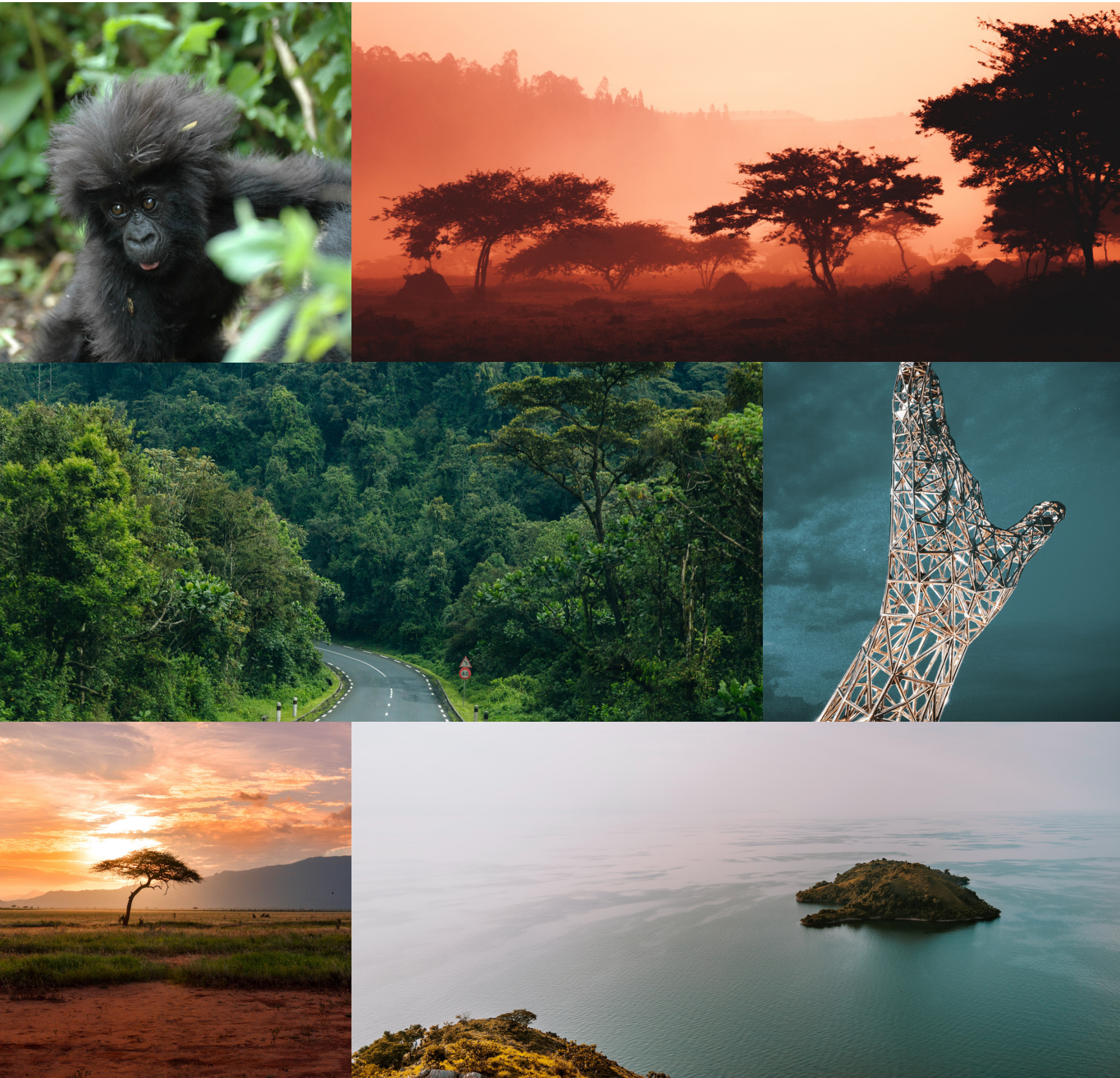# Travel and Tourism Blog Spring Boot MVC App Documentation

Travel and Tourism Blog Spring Boot MVC App Documentation

The project purpose

The purpose of the Travel and Tourism Blog Spring Boot MVC App is to create a web-based platform for users to share and explore travel experiences .The system will allow both administrators and regular users to manage blog posts, including creating, reading, updating, and deleting (CRUD) operations. Additionally, the system will support user management, authentication, authorization, validation, file uploads, and email communication.

II. Project Plan Scope

Goals and Deliverables

Establish a Functional Blogging Platform:
- Goal: Develop a fully functional travel and tourism blogging platform.
- Deliverable: A user-friendly application allowing users to create, edit, and manage blog posts seamlessly.

Implement User Authentication and Authorization:
- Goal: Ensure secure access with user authentication and authorization.
- Deliverable: A robust system that differentiates between administrators and regular users, providing appropriate access levels.

Enable CRUD Operations for Blog Posts:
- Goal: Implement Create, Read, Update, and Delete (CRUD) operations for blog posts.
- Deliverable: A comprehensive system allowing users to interact with their blog content effectively.

Facilitate User Management:
- Goal: Provide administrators with tools to manage user accounts.
- Deliverable: User management functionality, allowing administrators to create, modify, or deactivate user accounts.

Incorporate Secure Authentication Mechanisms:
- Goal: Implement secure authentication measures.
- Deliverable: A secure login system that protects user credentials and ensures data integrity.

Integrate File Upload Functionality:
- Goal: Enable users to upload files, particularly images for blog posts.
- Deliverable: File upload functionality seamlessly integrated into the blogging platform.

Implement Email Communication Features:
- Goal: Allow users to communicate with administrators through emails.
- Deliverable: An integrated email system facilitating direct communication between users and administrators

System Features

1. ***Login and Logout:***
   - *Admin: [Admin credentials]*
   - *User: [User credentials]*
2. ***Blog Management:***
   - *Create, Read, Update, Delete (CRUD) operations on blog posts.*
   - *Upload images for blog posts.*
3. ***User Management:***
   - *Admin can manage users (Create, Read, Update, Delete).*
4. ***Authentication and Authorization:***
   - *Secure login with authentication.*
   - *Admin and user roles with specific permissions.*
5. ***File Upload:***
   - *Upload images for blog posts.*
6. ***Email Communication:***
   - *Users can send emails to the admin.*

Technical Documentation

Architecture:

- **Frontend:** *Thymeleaf, HTML, CSS, Bootstrap, JavaScript.*
- **Backend:** *Java Spring MVC.*
- **Database:** *MySQL.*

Libraries/Frameworks Used:

- *Spring Boot*
- *Spring MVC*
- *Thymeleaf*
- *Bootstrap*
- *[Other relevant libraries]*

Development Environment:

- **IDE:** *Intellij Ultimate*
- **Database Management:** *MySQL Workbench*

Database Schema

Entity Relationship Diagram (ERD):

Plan
- placeId : Integer
- place : String
- address : String
- hotel : String
- dateTravel : Date
- dateAdded : Date
- climate : String
- placeUrl : String
- hotelUrl : String
- userId : Integer
- emailId : String
- idList : List<Integer>

Review
- reviewId : Integer
- review : String
- rating : float
- place : String
- dateAdded : Date
- userId : Integer

has   has

User
- userId : Integer
- userName : String
- password : String
- email : String
- sessionId : String