**TravelSpot Project**

**Overview**

TravelSpot is a CRUD-based web app (Express + MongoDB + EJS) where users can view and manage travel destinations (beaches, mountains, falls, etc) within Panay Island.

**Features:**

1. Users can create, edit, and delete spots
2. Each spot has title, description, price (fee), location, and category.

**Tech Stack:**

* Frontend – HTML5 using EJS, Bootstrap5
* Backend – NodeJS, ExpresJS
* Database – MongoDB with Mongoose
* Middleware – Method-override (for PUT/PATCH/DELETE), Express Middleware
* Utilities – Morgan (for logging requests)

**API / Routes:**

|  |  |  |
| --- | --- | --- |
| **Method** | **Endpoint** | **Description** |
| GET | /travelspots | List all Spots |
| GET | /travelspots/:id | View single Spot |
| GET | /travelspots/create | Form to create new Spot |
| POST | /travelspots | Create new Spot |
| GET | /travelspots/:id/edit | Form to Edit Spot |
| PUT/PATCH | /travelspots/:id | Update Spot |
| DELETE | /travelspots/:id | Delete Spot |

**Mongoose Schema (travelSpot.js)**

|  |
| --- |
| const mongoose = require("mongoose");  const Schema = mongoose.Schema;  const travelSpotSchema = new Schema(    {      title: { type: String, required: true, trim: true, minlength: 2 },      price: { type: Number, min: 0, default: 0 },      description: { type: String, minlength: 10 },      category: {        type: String,        enum: ["Beach", "Mountain", "Falls", "Historical", "City", "Island", "Farm", "Park", "Resort"],      },      province: {        type: String,        required: true,      },      municipality: {        type: String,        required: true,      },      image: String,      geometry: {        type: {          type: String,          enum: ["Point"],          default: "Point",        },        coordinates: {          type: [Number],          default: 0,        }, // [lng, lat]      },    },    { timestamps: true }  );  travelSpotSchema.virtual("location").get(function () {    if (this.municipality && this.province) {      return `${this.municipality}, ${this.province}`;    } else if (this.province) {      return this.province;    } else {      return "";    }  });  module.exports = mongoose.model("TravelSpot", travelSpotSchema); |

**Known Issues / To-Do**

* Add authentication (users + login system)
* Add image uploads (currently only text data stored)
* Add flash messages for better UX (success, errors)
* Improve seed data with more unique places