Project Draft: Hiking Place Project (MERN Stack)

Project Title: TrailTracker: Explore and Share Hiking Experiences

Executive Summary:

"TrailTracker" is a web application designed to help hiking enthusiasts discover, explore, and share hiking trails. Built using the MERN stack (MongoDB, Express.js, React.js, Node.js), this application provides users with detailed information on various hiking trails, including maps, difficulty levels, reviews, and user-generated content. The app aims to build a community of hikers who can contribute by adding new trails, sharing their experiences, and providing feedback on existing trails.

Objectives:

1. **Trail Discovery**: Provide a comprehensive database of hiking trails with detailed descriptions, maps, and user ratings.
2. **User Engagement**: Encourage users to share their hiking experiences, upload photos, and leave reviews.
3. **Community Building**: Foster a community of outdoor enthusiasts who contribute to the growth of the hiking trail database.
4. **Trail Safety and Updates**: Offer real-time updates on trail conditions and safety tips.

Target Audience:

- **Hiking Enthusiasts**: Individuals looking for new trails to explore.
- **Nature Lovers**: People who enjoy spending time outdoors.
- **Travelers**: Tourists interested in exploring hiking trails during their trips.
- **Local Communities**: Residents who want to discover nearby trails and share their experiences.

Key Features:

1. **Trail Database**:
    - Detailed descriptions of trails (name, location, length, difficulty, elevation gain).
    - (stretch goals): Interactive maps showing trail routes.(stretch goals)
    - (stretch goals): Photos and videos of trails.
    - (stretch goals): User ratings and reviews.
    - Trail conditions and safety updates.
2. **User Accounts**:
    - User registration and authentication (using JWT for security).
    - (stretch goals): Profile management (avatar, bio, favorite trails, past hikes).
    - (stretch goals): Ability to follow other users and see their activity.

3. **Trail Contribution**:
    - Users can add new trails with detailed descriptions and maps.
    - (stretch goals): Photo and video uploads for trails.
    - (stretch goals): Community-driven updates on trail conditions.
4. **Interactive Map**:
    - (stretch goals): Integration with Google Maps or Mapbox for real-time location tracking.
    - (stretch goals): GPS integration to start, track, and end hikes.
    - (stretch goals): Mark points of interest (e.g., scenic views, water sources).
5. **Search and Filters**:
    - (stretch goals): Search for trails by name, location, or keywords.
    - (stretch goals): Filter trails by difficulty, length, user ratings, and more.
6. **Social Features**:
    - (stretch goals): Users can leave comments on trails.
    - (stretch goals): Upvote/downvote comments and reviews.
    - (stretch goals): Share trail experiences on social media platforms.

Technology Stack:

- **Frontend**: React.js with Redux for state management, Boostrap for styling, React Router for navigation.
- **Backend**: Node.js with Express.js to handle API requests.
- **Database**: MongoDB for storing trail data, user information, and reviews.
- (stretch goals): **Hosting**: AWS EC2 for backend and MongoDB Atlas for the database.
- **Authentication**: JWT (JSON Web Token) for secure user authentication.

MongoDB Schema Design:

1. **User Collection**:
    - `username`: String, unique, required
    - `email`: String, unique, required
    - `password`: String, required (hashed)
2. **Trail Collection**:
    - `name`: String, required
    - `location`: String, required
    - `length`: Number, required (in miles or kilometers)
    - `difficulty`: String, required (e.g., Easy, Moderate, Hard)
    - `elevationGain`: Number, optional (in feet or meters)
    - `description`: String, required
    - `map`: String, URL to a trail map
    - `photos`: Array of Strings, URLs to trail photos
    - `isDogFreindly`: Boolean