User Requirements Document: Travello App

Introduction

Travello is a web-based social media platform for sharing travel experiences. The app allows users to create posts about their travel experiences, interact with other users, and search for travel experiences based on various criteria. The app also includes a chat feature to enable users to communicate with each other.

User Profiles

The app is designed for two types of users:

1. Regular Users: These users can create accounts, post travel experiences, view other users' travel experiences, and interact with other users by commenting on and liking posts. They can also use the chat feature to communicate with other users.
2. Admin Users: These users have additional privileges, including the ability to delete inappropriate posts and comments, and manage user accounts.

Functional Requirements

Account Creation and Authentication

- The app should allow users to create accounts with the following information: Name, email address, and password.
- Users should be able to log in and out of their accounts.
- Passwords should be securely encrypted and stored in the database.
- Users should be able to recover their password through a password recovery feature.

Posting and Viewing Travel Experiences

- Users should be able to create posts about their travel experiences, including title, description, location, images, date of travel, and tags.
- Users should be able to edit and delete their own posts.

- Users should be able to view other users' travel experiences in a feed or gallery view.
- The feed or gallery view should include a thumbnail image and a brief description of each post.
- Users should be able to view the full details of a post, including images and comments, by clicking on the post in the feed or gallery view.

Interactions with Posts

- Users should be able to interact with posts by liking, disliking, and commenting on posts.
- Users should be able to view a list of their own posts and the posts they have liked.
- Users should be able to report inappropriate posts or comments to the admin.

Chat Functionality

- Users should be able to send and receive chat messages with other users.
- Users should be able to create and participate in group chats with other users.
- Users should be able to view a list of their chat conversations and search for specific conversations.

Search

- Users should be able to search for travel experiences by location, date, keywords, and tags.
- Search results should be sorted by relevance, with the most relevant results appearing at the top of the list.
- Users should be able to filter search results by location, date, and tags.

Admin Features

- Admin users should have the ability to delete inappropriate posts and comments.
- Admin users should be able to view a list of reported posts and comments.

- Admin users should be able to manage user accounts, including resetting passwords and deleting user accounts.

Non-functional Requirements

Performance

- The app should load quickly and respond to user interactions without delay.
- The app should be able to handle a large number of users and posts without slowing down.

Usability

- The app should be easy to use and navigate, with a clean and intuitive user interface.
- The app should be responsive and optimized for different screen sizes and devices.

Security

- User data should be securely stored in the database and encrypted during transmission.
- Passwords should be securely hashed and stored in the database.
- The app should be protected against common security threats, such as SQL injection and cross-site scripting.

Accessibility

- The app should be accessible to users with disabilities, including those who use screen readers and other assistive technologies.

Reliability

- The app should be reliable and available for users to access at all times.

- The app should be able to recover from errors and failures without data loss.

Scalability

- The app should be designed to scale easily as the user base grows.
- The app should be able to handle a large number of users and posts without sacrificing performance.

Technology Stack

Travello will be built using the MERN (MongoDB, Express.js, React.js, Node.js) stack. The following technologies and tools will be used:

- MongoDB for the database
- Express.js for the server-side web application framework
- React.js for the client-side user interface framework
- Node.js for the server-side JavaScript runtime environment
- Socket.io for real-time communication in the chat feature
- Bootstrap for front-end styling and layout
- Git for version control
- Heroku for deployment

Conclusion

The Travello app is a social media platform for sharing travel experiences. The app allows users to create posts about their travel experiences, interact with other users, and search for travel experiences based on various criteria. The app also includes a chat feature to enable users to communicate with each other. The app will be built using the MERN stack and will be designed to be reliable, scalable, secure, and easy to use.