Weather Forecast Application Requirement Document

Overview

The Weather Forecast Application is a Next.js web application that provides users with current weather conditions and forecasts. Utilizing the OpenWeatherMap API for weather data and the Unsplash API for dynamic background images based on the current weather conditions, the application aims to deliver a visually engaging and informative user experience. The application will be responsive, ensuring a seamless experience across various devices.

Objectives

- To provide accurate and up-to-date weather information.
- To enhance user experience with dynamic background images reflecting current weather conditions.
- To demonstrate the candidate's proficiency in Next.js, including React Server Components, Client Components, and data fetching strategies.

Technical Requirements

Core Technologies

- **Next.js:** For server-side rendering, static site generation, and building the overall application structure. Please use the latest App Router.
- **TailwindCSS:** For styling the application.
- **Shadcn UI:** Use this as UI components so you don’t have to “design.”
- **Unsplash API:** For fetching dynamic background images based on weather conditions. You can get it FREE here.
- **OpenWeatherMap API:** For fetching current weather data and forecasts. You can get the FREE tier here.
- **Other:** Whatever you think will be useful. Just remember, don’t add too much.

Features and Functionalities

Weather Data Display

- Display current weather conditions including temperature, humidity, wind speed, and a brief description (e.g., sunny, cloudy).
- Provide a 5-day weather forecast showing daily high and low temperatures and weather conditions.
- Implement error handling for API requests to manage and display errors gracefully.

Dynamic Background

- Utilize the Unsplash API to fetch and display background images that correspond to the current weather conditions (e.g., sunny, rainy, cloudy).
- Ensure that the background image changes dynamically with the weather conditions.

Responsive Design

- The application must be fully responsive, providing an optimal viewing experience across a wide range of devices (from desktop monitors to mobile phones).

Performance Optimization

- Implement image optimization techniques for Unsplash images to ensure fast loading times without sacrificing quality.
- Use Next.js features like Server Components for data fetching and Client Components for interactive elements to optimize performance.

Development Considerations

API Integration

- Securely store API keys for OpenWeatherMap and Unsplash using environment variables.
- Implement **efficient data fetching strategies** to minimize API calls and reduce load times.

Code Quality and Documentation

- Write clean, maintainable, and scalable code following best practices.
- Include comprehensive documentation covering setup instructions, application structure, and major functionalities.

Testing

- Implement unit tests for critical components and functionalities to ensure reliability.
- Use Next.js built-in ESLint configuration for code quality assurance.

Submission Guidelines

- The project should be submitted as a GitHub repository with a clear README.md file explaining how to set up and run the application.
- Include instructions for setting up API keys for both OpenWeatherMap and Unsplash.
- The application should be deployable on platforms like Vercel or Netlify with minimal configuration. However, you aren’t required to do this. But it’ll be nice, though.

Evaluation Criteria

- **Functionality:** The application works as described without errors.
- **Code Quality:** The code is clean, well-organized, and follows best practices.
- **UI/UX Design:** The application is user-friendly, visually appealing, and responsive. No need to overthink design too much, though. That’s why you’ll be thankful to use Shadcn UI!
- **Performance:** The application loads quickly and efficiently to handle data fetching and image loading.
- **Documentation:** The project includes comprehensive documentation that makes setting up and understanding the application's structure and features easy.

(Extra) Wireframe

Weather Forecast Application Wireframe (Mobile View First)

```
## Header
- Application Title
```

```
## Weather Data Display Section
- Current Weather:
  - Icon (Weather condition)
  - Temperature
  - Weather description (e.g., Sunny, Cloudy)
  - Humidity, Wind speed (icons with text)

## 5-Day Forecast
- Scrollable horizontal list:
  - Day of the week
  - Icon (Weather condition)
  - High/Low temperatures

## Dynamic Background
- Full-screen background image from Unsplash API reflecting current weather

## Additional Information (Accordion for details)
- Accordion Component:
  - Hourly forecast
  - Detailed weather description
  - Weather-related advice

## Footer
- Credits and API acknowledgments
```

UI Sketch from `V0.dev`

The wireframe description above is already generated as a prompt and sketched by v0.dev already. You can access **`the UI Sketch here`** to get an idea of what the UI might look like. But be mindful about using the code. You can copy and paste. But the end product should be in your hands. Otherwise, we will know if you crafted your code or not ;) We're not against using AI, though!

This document outlines the requirements for the Weather Forecast Application Takehome Project. Candidates are expected to demonstrate their skills in Next.js, API integration, responsive design, and performance optimization through the completion of this project.