Frontend Engineering Take-Home Assignment: Recipe Collection & Meal Planner

Background

You're building a personal recipe management and meal planning application that helps home cooks organize their favorite recipes, plan weekly meals, and generate smart shopping lists.

The Assignment

Build a **Recipe & Meal Planning Dashboard** with the following features:

Core Requirements (Must Have)

1. **Recipe Gallery**
   - Display recipe cards in a responsive grid layout
   - Each card shows recipe image, title, cooking time, difficulty, and dietary tags
   - Implement search and filtering (by cuisine, dietary restrictions, cooking time, ingredients)
2. **Weekly Meal Planner**
   - Create a 7-day calendar grid for breakfast, lunch, and dinner slots
   - Visual indicators for nutritional balance (protein, carbs, fat in grams)
3. **Smart Shopping List Generator**
   - Auto-generate shopping list based on planned meals
   - Group ingredients by category (produce, dairy, pantry, etc.)
   - Allow manual additions/deletions to the list
   - Mark items as "already have" to exclude from shopping
   - Show estimated total cost
4. **Recipe Detail Modal**
   - Full recipe view with ingredients, instructions, and nutrition info
   - Timer integration for cooking steps
   - Rating and notes system

Technical Requirements

- **React** with TypeScript
- **State management** (Context API, Redux, or anything else that works for you)
- **Responsive design** that works on desktop, tablet, and mobile
- **CSS-in-JS or styled-components** (or Tailwind CSS)
- **Mock data -** create realistic recipe data (at least 30+ recipes with various cuisines and dietary options)
- **Local storage** persistence for meal plans and shopping lists

Bonus Features (Nice to Have)

- Recipe import from URL functionality
- Unit tests for key components
- Deploy using Vercel (free tier)

What We're Looking For

Submission Requirements:

1. **Complete React application** source code on GitHub and instructions for us to run the code locally
2. **Detailed README** that includes:
    - Setup instructions
    - **AI Usage Documentation**: Which AI tools you used, what code was generated vs. written by you, and why you made specific modifications
    - Architecture decisions and reasoning
    - Challenges faced and how you solved them
    - What you would do differently with more time
3. **Code comments** explaining complex logic or AI-generated sections you modified

Evaluation Criteria

1. **AI Proficiency** (25%)
    - Effective use of AI tools for code generation
    - Clear documentation of what was AI-generated vs. hand-written
    - Evidence of prompt engineering skills
2. **Code Understanding & Adaptation** (25%)
    - Modifications made to AI-generated code showing understanding
    - Proper integration of different AI-generated components
    - Code quality improvements beyond initial AI output
3. **Technical Implementation** (25%)
    - Clean, maintainable code structure
    - Proper React patterns and TypeScript usage
    - State management and data flow
    - Performance optimizations
4. **UX/UI Design** (25%)
    - Intuitive and visually appealing interface
    - Mobile-responsive design
    - Thoughtful user experience for meal planning workflow

Time Expectation

3 days (We understand you'll likely use AI to accelerate development, so we're looking for thoughtful implementation rather than just speed)

Submission

- GitHub repository link
- Brief video walkthrough (2-3 minutes) explaining your approach and demonstrating key features