Plateful
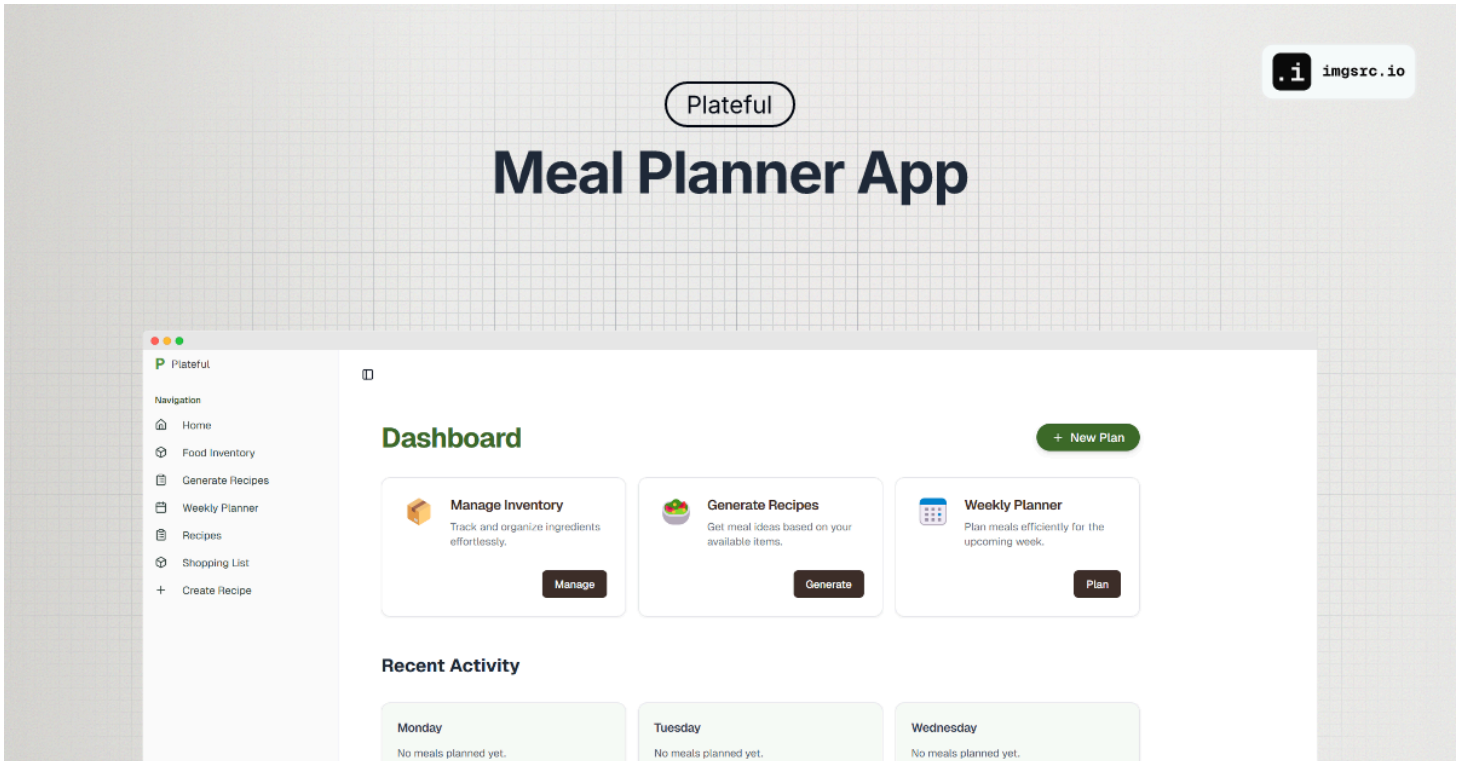
Overview

Plateful is a **meal planning web application** designed to help users efficiently **manage their food inventory, generate recipes**, and **plan weekly meals**. The app provides a **seamless and intuitive UI** for effortless meal planning, ensuring users can make the most out of their available ingredients.

Features

- 🥗 **Generate Recipes** based on available ingredients
- 📦 **Manage Inventory** to keep track of stored food items
- 📅 **Weekly Meal Planner** for structuring meals in advance
- 📊 **User-Friendly Dashboard** with recent activity tracking

Tech Stack

- **Framework:** Next.js (with TypeScript)
- **Routing:** Next.js App Router

- **State Management:** React Context API
- **Styling:** Tailwind CSS
- **UI Components:** ShadCN/UI
- **Deployment:** Vercel

Project Structure

```
plateful/
|— app/                         # Next.js App Router directory
|    ├— layout.tsx              # Root layout for shared structure
|    ├— not-found.tsx           # 404 page handling
|    ├— page.tsx                # Home page
|    |
|    ├— api/                    # API routes (server functions)
|    |   ├— recipes/route.ts    # Recipes API endpoint
|    |   ├— (other API routes)
|    |
|    ├— create-recipe/page.tsx      # Create a new recipe
|    ├— generate-recipe/page.tsx    # Generate a recipe
|    ├— inventory/page.tsx          # Inventory management
|    ├— recipes/page.tsx            # Recipes listing
|    ├— shopping-list/page.tsx      # Shopping list management
|    ├— weekly-planner/page.tsx     # Weekly meal planner
|
|— components/                  # Shared reusable UI components
|
|— context/                     # Context API state management
|    ├— AppContext.tsx          # Global state provider
|
|— hooks/                       # Custom React hooks
|    ├— useInventory.ts         # Inventory-related logic
|    ├— useRecipes.ts           # Fetching and handling recipes
|
|— lib/                         # Utility functions & API helpers
|    ├— api.ts                  # Global fetch helper
|    ├— utils.ts                # General utility functions
|
|— public/                      # Static assets
|    ├— featuredimg.png         # Featured image
|    ├— logo.png                # App logo
|
|— styles/                      # Global styling files
|    ├— global.css              # Tailwind styles
|
|— types/                       # TypeScript type definitions
|
|— ui/                          # ShadCN/UI components
```

```
|      ├── button.tsx           # Button component
|      ├── input.tsx            # Input component
|      ├── select.tsx           # Select dropdown
|      ├── dialog.tsx           # Modal/dialog component
|
|── .gitignore               # Git ignored files
|── eslint.config.js         # Linting configuration
|── next-env.d.ts            # Next.js TypeScript setup
|── package.json             # Project dependencies
|── tailwind.config.js       # Tailwind configuration
|── tsconfig.json            # TypeScript configuration
|── README.md                # Project documentation
```

Getting Started

1. Clone the Repository

```
git clone https://github.com/oguzhanozenc/plateful.git
cd plateful
```

2. Install Dependencies

```
yarn install
```

3. Start the Development Server

```
yarn dev
```

Runs the app in development mode. Open `http://localhost:3000/` in your browser.

Deployment to Vercel

1. Build for Production

```
yarn build
```

2. Deploy to Vercel

- Link your GitHub repository to Vercel
- Set **build command** to `yarn build`
- Deploy your site at `https://platefulapp.vercel.app/`

Contributing

Contributions are welcome! If you'd like to improve Plateful:

1. Fork the repository
2. Create a feature branch (`git checkout -b feature-branch`)
3. Commit your changes (`git commit -m "Add feature"`)
4. Push to the branch (`git push origin feature-branch`)
5. Create a Pull Request

License

MIT License © 2025 Oguzhan Ozenc.