

Crop NFT Product Search Application – Prototype Write-up

What the Prototype Does

This prototype is a modern web application that allows users to search for agricultural products using either crop names or NFT IDs. The system matches user input to available products and displays relevant information, including product images, prices, and purchase links. The application is designed to demonstrate a crop-to-product matching system, simulating how blockchain-based crop identification (NFTs) could be integrated with e-commerce.

Use of AI Tools in Development

AI tools were leveraged throughout the development process to:

- **Accelerate Coding:** AI-assisted code generation was used for boilerplate setup, React component scaffolding, and backend API design.
- **Debugging and Refactoring:** AI tools helped identify and resolve cross-platform issues (e.g., environment variable handling on Windows), and suggested best practices for code organization.
- **Documentation and Planning:** AI was used to generate and refine technical documentation, as well as to outline the system architecture and data flow.

Tools and Tech Stack Used

- **Frontend:**
 - React 18 (TypeScript)
 - Vite (build tool)
 - Tailwind CSS (styling)
 - shadcn/ui (component library)
 - TanStack Query (React Query) for server state
 - Wouter (routing)
 - React Hook Form + Zod (form validation)
- **Backend:**
 - Node.js with Express.js (TypeScript)
 - Drizzle ORM (for database abstraction, but in-memory for this prototype)
 - Zod (schema validation)
 - Vite (for hot reloading in development)
- **Database:**
 - In-memory storage (prototype/demo mode)
 - PostgreSQL (planned for production, but not required for demo)
- **DevOps:**
 - tsx (TypeScript execution)
 - cross-env (cross-platform environment variable support)
 - Replit (optional cloud development)
- **AI Tools:**
 - AI code assistants for code generation, refactoring, and documentation

Extending to Real-World Use

To move from prototype to production, the following steps could be taken:

- **Persistent Database:** Connect to a real PostgreSQL instance for persistent storage of crops and products.
- **User Authentication:** Add user accounts, authentication, and authorization for secure access.
- **NFT Integration:** Integrate with blockchain APIs to verify NFT ownership and authenticity.
- **Admin Dashboard:** Allow admins to add/edit crops and products via a secure interface.
- **Scalability:** Deploy on scalable cloud infrastructure with CI/CD pipelines.
- **Localization:** Add support for multiple languages and currencies.

Assumptions and Limitations

- **In-Memory Storage:** The current prototype uses in-memory storage, so all data resets on server restart.
- **No Real Blockchain:** NFT IDs are simulated; there is no actual blockchain or NFT verification.
- **No Authentication:** The system is open and does not restrict access or actions.
- **Limited Product Data:** Only a few sample crops and products are included for demonstration.
- **Single-Server Deployment:** The app is designed for single-server use and is not yet optimized for distributed or high-availability deployment.