

5-Week Roadmap: From Idea to Working Prototype

Overall Goal: To develop a functional, polished, and demonstrable Android-first MVP of the **Lumen Inventory** module. This prototype will allow a user to manage their store's inventory by scanning barcodes, tracking stock levels, and recording sales, proving the core value proposition of the business.

Week 1 (Aug 18 - Aug 24): Foundation & Design

Weekly Goal: Finalize the core user flow, design the essential UI/UX, and set up the complete development environment.

- **UI/UX Design:**
 - Map out the primary user journey: Onboarding -> Adding first product -> Viewing dashboard -> Recording a sale.
 - Create high-fidelity mockups in Figma for the key screens: Login/Sign Up, Main Dashboard, Add Product (with camera scanner view), and Product List.
 - Define a simple color palette, typography, and component style guide.
- **Technical Setup (Backend & Frontend):**
 - Initialize a React Native project for the mobile app.
 - Set up the backend project using Python (FastAPI) and initialize the PostgreSQL database schema (for users, products, inventory).
 - Establish a Git repository and define the branching strategy (e.g., main, develop, feature branches).
- **Project Management:**
 - Create a detailed project backlog in a tool like Trello or Jira, breaking down all features into small, manageable user stories.
 - Hold a kickoff meeting to ensure the team is aligned on the 5-week plan and the final MVP scope.

Week 2 (Aug 25 - Aug 31): Core Backend & UI Implementation

Weekly Goal: Build the foundational backend API endpoints and translate the UI mockups into functional frontend components.

- **Backend Development:**
 - Implement user authentication endpoints (Register, Login, Logout).
 - Create the core CRUD (Create, Read, Update, Delete) API endpoints for Products (name, barcode, etc.) and Inventory (stock quantity).
 - Set up initial database migrations.
- **Frontend Development:**
 - Build the static UI components in React Native based on the Figma designs (Login screen, Dashboard layout, buttons, input fields).
 - Implement the navigation structure for the app (e.g., moving between the login screen, dashboard, and add product screen).
- **Project Management:**

- Begin daily stand-up meetings to track progress and identify blockers.

Week 3 (Sep 1 - Sep 7): Connecting Frontend to Backend & Core Feature

Weekly Goal: Integrate the frontend with the backend API and implement the single most important feature: barcode scanning.

- **Full-Stack Integration:**
 - Connect the Login and Registration screens to the backend authentication endpoints.
 - Implement state management (e.g., React Context, Zustand) to handle user sessions and tokens.
 - Fetch and display the product list on the main dashboard from the backend API.
- **Core Feature Development (Frontend):**
 - Integrate a camera/barcode scanning library (like react-native-camera or react-native-vision-camera) into the "Add Product" screen.
 - When a barcode is scanned, make an API call to check if the product exists. If not, allow the user to add its details and initial quantity. If it does, allow them to update the quantity.
- **Backend Development:**
 - Refine API endpoints to handle the logic from the barcode scanner.

Week 4 (Sep 8 - Sep 14): Feature Polish & End-to-End Testing

Weekly Goal: Finalize all core features, add essential "quality of life" improvements, and conduct thorough testing.

- **Frontend Development:**
 - Implement the "Record Sale" functionality. This should be a simple button on the dashboard that decrements a product's stock count via an API call.
 - Add user feedback elements like loading spinners, success messages, and clear error handling (e.g., "Invalid barcode," "Network error").
 - Ensure the UI is responsive and looks clean on various Android device sizes.
- **Backend Development:**
 - Implement transaction logic to ensure inventory updates are atomic and reliable.
 - Add basic data validation to all API endpoints.
- **Testing:**
 - Conduct end-to-end testing of the full user flow.
 - Have team members (and ideally a friend who owns a shop) test the app to identify bugs and usability issues.

Week 5 (Sep 15 - Sep 19): Final Polish, Deployment & Pitch Prep

Weekly Goal: Crush all remaining bugs, deploy a stable build, and prepare a compelling presentation.

- **Bug Fixing & Final Polish:**
 - Address all critical bugs and UI inconsistencies identified during testing.
 - Add a simple app icon and splash screen.
 - Write final, clean code and ensure the repository is well-documented.
- **Deployment:**
 - Prepare a final, installable .apk file for Android.
 - (Optional Stretch Goal) Deploy the backend to a cloud service like Heroku or AWS Elastic Beanstalk for a live demo.
- **Pitch Preparation:**
 - Prepare a concise and powerful 3-5 minute presentation that walks through the problem, the solution, and a live demo of the working prototype.
 - Practice the pitch repeatedly until it is smooth and confident.
 - **Final Check:** Ensure the prototype is fully charged and functional on the demo device.