

# App Development Protocol

---

## 1. Core Technology Stack

---

We prioritize speed, simplicity, and low overhead. We do not use bloated frameworks where native solutions work.

- **Frontend Framework:** React (via Vite)  
*Why: Fast build times, minimal configuration, high performance.*
- **Styling Engine:** Tailwind CSS  
*Standard: Utility-first. No external CSS files. Dark Mode support by default.*
- **Backend & Auth:** Firebase (Firestore + Authentication)  
*Rationale: Serverless architecture for real-time data and secure auth without infrastructure maintenance.*
- **Icons:** Lucide React  
*Style: Lightweight, vector-based, consistent.*

## 2. Development Workflow (The "Interactive Handshake")

---

To ensure code integrity and prevent errors during AI-assisted development, we strictly adhere to this process.

### Phase 1: The Sync (Start of Session)

- **Requirement:** Before requesting changes, the developer must provide the current state of the application's core logic.
- **Action:** Upload `src/lib/utils.jsx` (or equivalent logic file) and `src/App.jsx`.
- **Goal:** Ground the session in the actual codebase, preventing "hallucinated" code based on outdated context.

### Phase 2: Execution (The Micro-Chunk Protocol)

- **Trigger:** Any file generation exceeding 50 lines of code.
- **Protocol:** We use a **Turn-Based Delivery System** to prevent buffer truncation.
  1. Turn 1: AI generates [Part 1 of X] and stops.
  2. User Action: User reviews, pastes the code, and types "Next".
  3. Turn 2: AI generates [Part 2 of X].
  4. *Repeat until complete.*

### Phase 3: Completion (The Build Check)

- **Requirement:** The session is not closed until the build passes.
- **Action:** User runs `npm run build`.
- **Rule:** If the build fails, the immediate priority is fixing the break. No new features are added until the build is green.

### 3. Coding Standards (Non-Negotiable)

---

#### A. Formatting

- **Indentation: Hard Tabs (`\t`).**

*Rule: Do not use spaces. Simplicity and consistency are paramount.*

#### B. Defensive UI Engineering

- **Overflow Protection:** All Cards/Modals with `rounded-xl` must have `overflow-hidden`.
- **Text Safety:**
  - Variable text columns must use `whitespace-nowrap` AND `truncate`.
  - Never rely on auto-sizing for user-generated content.
- **Mobile Integrity:** If a feature breaks on mobile, it is considered broken. Horizontal scrolling on the page body is prohibited.

#### C. The "Anti-Regression" Policy

- **Principle:** Speed is nothing without control.
- **Rule:** Never break an existing feature to implement a new one.
- **Logic:** Features must be atomic. UI, Logic, and Data must ship together in a working state.

### 4. Environment Setup

---

- **IDE:** VS Code (Primary editor).
  - **Terminal:** CMD or PowerShell (For build/deploy commands).
  - **Deployment:** Build locally ( `npm run build` ), then deploy via FTP or hosting service.
-