### **Technical Details and Environment Setup**

This document, based on the **dev techical detail.docx**, provides a deeper dive into the architectural choices and setup instructions for SynapseRead.

**Database Architecture:**

* **Firebase Cloud Firestore (NoSQL Document Database):**
  + **Purpose:** Stores structured user data like profiles (IDs, email, subscription status), preferences (Bionic Reading settings, fonts, themes), reading progress, and dictionary lookup history.
  + **Structure:** Data organized into collections (e.g., users, documents, preferences). Uses automatic document IDs to prevent performance issues.
  + **Security:** Requires robust Firestore Security Rules to ensure users can only access their own data.
* **Firebase Cloud Storage (Object Storage):**
  + **Purpose:** Stores large, unstructured user files like uploaded PDFs, DOCX, and TXT documents.
  + **Structure:** Files organized into user-specific folders (e.g., user/<UID>/documents/mybook.pdf).
  + **Security:** Implements Cloud Storage Security Rules to restrict access to the authenticated owner and validate file properties (type, size).

**Main Components of the App:**

* **Client-Side Application (React Native):**
  + **User Interface (UI) Module:** Manages visual elements and interactions.
  + **Bionic Reading Text Renderer:** Applies the Bionic Reading algorithm (bolding, fixation, saccade, opacity) to raw text.
  + **Content Import Module:** Handles file uploads (TXT, PDF, DOCX) and potentially cloud storage integration.
  + **User Preferences Management Module:** Stores and applies personalized settings.
  + **Reading Progress & Bookmarking Module:** Tracks reading position.
  + **Dictionary Lookup Module:** Integrates with dictionary APIs.
  + **Authentication & User Session Management:** Handles login and sessions.
  + **Payment & Subscription Management Module:** Interfaces with payment gateways.
  + **Accessibility Module:** Ensures compatibility with screen readers and provides customizable settings.
* **Backend Services (Firebase):** Authentication, Cloud Firestore, Cloud Storage, and Security Rules.
* **External APIs/Services:** Dictionary API (e.g., WordsAPI), Payment Gateway API (Stripe/Paystack).

**How to Set Up the Developer Environment:**

1. **Install Node.js and npm/Yarn:** Essential for React Native development.
2. **Install React Native CLI:** npm install -g react-native-cli.
3. **Set Up Platform-Specific Development Tools:**
   * **Android:** Install Android Studio, necessary SDK components, configure environment variables, and set up an emulator/device.
   * **iOS (macOS only):** Install Xcode, command-line tools, and CocoaPods.
4. **Create a New React Native Project:** Use npx react-native init SynapseRead.
5. **Install Project Dependencies:** Install specific libraries for Firebase, document processing, etc. (e.g., @react-native-firebase/auth, @react-native-firebase/firestore). For iOS, run cd ios && pod install after adding native dependencies.
6. **Run the App:** npx react-native run-android or npx react-native run-ios.

**Key Dependencies and Technologies Used:**

* **Core:** React Native, JavaScript/TypeScript.
* **Backend:** Firebase Authentication, Cloud Firestore, Cloud Storage, Security Rules.
* **Document Processing:** react-native-fs (TXT), react-native-pdf-extractor / commercial SDKs (PDF), Spire.Doc.Base.js / docxtemplater (DOCX).
* **Bionic Reading Algorithm:** bionic-reading (npm package) and custom logic.
* **Dictionary:** WordsAPI.
* **Payments:** Stripe React Native SDK, Paystack, Google Pay.
* **Testing:** Jest, React Testing Library, Detox, Maestro.
* **Localization & Accessibility:** I18nManager API, i18next, React Native Accessibility API.