

## Zenith Reader

**Zenith Reader** is a high-performance, lightweight, and offline-first E-Book reader built with Electron. It is designed specifically for users who want a distraction-free reading experience for both **EPUB** and **PDF** formats, with a focus on privacy and local file management.

## The Goal

The primary goal of Zenith Reader is to provide a "Zen" reading environment. Unlike many modern readers that require cloud logins or constant internet connections, Zenith Reader aims to:

1. **Operate Fully Offline:** Once installed, the app requires no internet to render books.
2. **Ensure Privacy:** Your library is stored locally on your machine, not on a server.
3. **Provide Visual Comfort:** Through dynamic themes (Light, Dark, Sepia) and adjustable typography.
4. **Stay Updated:** A manual auto-updater ensures the app improves without being intrusive.

## Key Features

- **Dual Format Support:** Smooth rendering for both reflowable EPUBs and fixed-layout PDFs.
- **Local Library Management:** Persistent storage of your book list using browser local storage.
- **Triple-Theme System:**
  - **Light:** Crisp and clean for daylight reading.
  - **Dark:** Deep blues and grays to reduce eye strain at night.
  - **Sepia:** Warm tones for a paper-like feel.
- **Responsive Navigation:** Supports arrow key navigation and mouse-click paging.
- **Smart PDF Rendering:** Utilizes local canvas rendering for high-resolution document viewing.

## Technical Updates & Changelog

### Version 1.0.9 (Latest)

- **Production CSP Overhaul:** Implemented a strict Content Security Policy (CSP) that allows the app to function without any external CDNs.
- **Local Asset Migration:** Moved Tailwind CSS, Phosphor Icons, and PDF.js to the `src/assets` folder to prevent "connection refused" errors in offline mode.

- **PDF.js Worker Fix:** Corrected the global worker path to point to local assets, resolving the "Worker not found" crash.
- **Auto-Updater UI:** Added a custom toast notification in the frontend that communicates with the Electron Main process to show download progress and "Restart to Install" prompts.

### Version 1.0.5 - 1.0.8

- **IPC Bridge Security:** Hardened the `preload.js` script to ensure only safe commands pass between the UI and the File System.
- **Manual Update Trigger:** Changed the `autoUpdater` logic to `autoDownload = false`, giving users the choice of when to start a download.
- **GitHub Integration:** Configured `electron-builder` for automated publishing to the GitHub repository releases.

### Version 1.0.0 - 1.0.4

- **Initial Build:** Core architecture established using `epub.js` and `jszip`.
- **Library UI:** Implemented the grid view with hover effects and automatic title extraction.

## Installation & Setup

### Prerequisites

- **Node.js** (v18 or higher recommended)
- **Git**

### 1. Clone & Install

```
git clone [https://github.com/Sunny-wicked/zenith-reader.git](https://github.com/
cd zenith-reader
npm install
```

### 2. Prepare Assets (Crucial for Offline Mode)

To ensure the app works without internet, ensure your `src/assets` folder contains:

- `tailwind.js` (Standalone version)
- `phosphor.js` (Phosphor Icons library)
- `pdf.min.js` & `pdf.worker.min.js` (Legacy/UMD builds)

### 3. Run Locally

```
npm start
```

#### 4. Build for Production

```
npm run build
```

#### Project Structure

- `main.js` : Controls window lifecycle and Auto-Updater events.
- `preload.js` : The secure bridge between the web page and Node.js.
- `src/index.html` : The core UI, styling, and reader logic.
- `package.json` : Contains the build configuration and GitHub repository links for updates.

#### License