**GrammarAssistant (Python-only: TextBlob offline + optional Gemini)**

GrammarAssistant is a Python-based grammar and spelling correction assistant. It works **offline using TextBlob** and optionally integrates **Google Gemini** for AI-powered suggestions. This package avoids the Java dependency required by language-tool-python and can run across applications like Notepad, Word, or browsers.

**Features**

* Detects sentences in real-time while typing.
* Displays **popup suggestions** above the text.
* Provides buttons to **Replace** or **Ignore** suggestions.
* Works fully **offline** with TextBlob.
* Optional **online mode** using Google Gemini AI.
* Runs globally across apps using **keyboard hooks**.

**Project Structure**

GrammarAssistant/

│

├─ launcher.py # Starts backend (FastAPI) and frontend keyboard assistant

├─ frontend/

│ └─ main.py # Keyboard listener + popup UI

├─ backend/

│ ├─ app.py # FastAPI backend with TextBlob & optional Gemini

│ └─ requirements.txt # Backend dependencies

├─ requirements.txt # Root dependencies for environment setup

└─ README.txt # Project documentation

**Technologies Used**

**Frontend:**

* keyboard : Global key event hooks
* tkinter : Popup suggestion UI
* pyautogui : Capture caret position
* pyperclip : Copy/paste text
* requests : HTTP requests to backend
* threading : Concurrent processing

**Backend:**

* fastapi : REST API backend
* pydantic : Data validation
* uvicorn : ASGI server for FastAPI
* textblob : Offline grammar/spelling correction
* google-generativeai : Optional Gemini AI integration
* asyncio : Async handling for Gemini

**Usage**

* Start typing in any application (Notepad, browser, Word, WhatsApp Web).
* When a grammar issue is detected, a popup appears with a suggestion.
* Choose **Replace** to apply correction or **Ignore** to dismiss.
* Works offline immediately with TextBlob; Gemini enhances suggestions when online.

**Optional: Build Executable (.exe)**

To create a standalone Windows executable:

pip install pyinstaller

pyinstaller --onefile launcher.py --name GrammarAssistant

* The generated .exe file can run without Python installed.

**Requirements**

* Python 3.9 or higher
* Frontend Python packages: keyboard, tkinter, pyautogui, pyperclip, requests
* Backend Python packages: fastapi, pydantic, uvicorn, textblob, google-generativeai (optional)

**License**

Open-source. Free for educational and personal use.

If you want, I can also **draft a “full ready-to-run launcher.py + frontend/main.py + backend/app.py”** template that is fully offline-ready with TextBlob and Gemini optional. This will make it plug-and-play.

Do you want me to do that next?