Product Requirements Document

Product Name: JoelGPT

Author: Abhigyan Das

Version: 1.0

Date: Nov 2025

---

1) Product Overview

JoelGPT is a secure, offline-first AI productivity assistant that runs **fully on personal hardware**, eliminating dependency on cloud-based LLM services like ChatGPT, Gemini, or Claude. It provides:

- Local text generation for writing & summarizing
- Personal knowledge answering through Retrieval-Augmented Generation (RAG)
- Optional live web updates (only when requested via command)
- PDF document ingestion and semantic memory
- Offline resilience against cloud outages

During the recent **Cloudflare & AWS outage**, millions of users were unable to access AI services for essential daily tasks. This dependency risk inspired the creation of JoelGPT — an AI assistant that **never goes offline**, always protects privacy, and is accessible even in low-compute environments.

JoelGPT is optimized for **light content generation**, note referencing, and knowledge recall — tasks that represent **80% of average consumer LLM usage.**

> **Mission Statement:**
> "Deliver a fast, reliable local AI assistant with real-time search and smart knowledge retrieval, optimized for minimal device resources.."

---

2) Problem Statement

Problem	Why It Matters

AI tools are centralized & cloud-dependent	Outages = complete productivity halt
Sensitive data is uploaded to external servers	Privacy and compliance risks
Existing tools require expensive GPUs or subscriptions	Financial barrier for students & individuals
LLMs don't have RAG pipelines integrated	Inefficient for personal knowledge tasks

JoelGPT solves these by enabling a **self-contained, private, resilient AI** experience.

---

3) Product Scope

In Scope

- Local inferencing on consumer hardware
- Text-based conversational assistant
- PDF ingestion into a local vector DB
- RAG-powered question answering
- Live web information retrieval via /search
- Terminal-based interaction + streaming outputs
- Small-to-medium language models (e.g., Gemma, Llama 3.1)

Out of Scope – for this release

- Advanced reasoning or coding assistance (large models)
- Speech To Text (STT) and Text To Speech (TTS) functionalities
- Multimodal support (images, videos)
- UI desktop/website frontend
- Continuous background internet crawling

---

4) Personas

Persona A — College Student (Primary)

- Needs reference assistance for academic papers, resumes, and study notes
- Often studies in locations with poor internet
- Cannot afford premium AI subscriptions

Goals

- Summarize course PDFs
- Write emails, essays, cover letters
- Store personal notes for quick recall

Pain Points

- ChatGPT or similar LLMs becomes paid or unavailable
- Privacy concerns with personal documents

Persona B — Job Seeker / Fresh Graduate

- Preparing resumes, interview answers, case studies

Goals

- Reliable writing and polishing assistance anytime
- Store career-related docs privately

Pain Points

- Reliance on hosted tools for daily workflows
- Concerns of leaking private info (salary, companies)

Persona C — Independent Creators & Writers

- Uses AI for creativity & productivity workflows

Goals

- Generate inspiration, drafts, snippets
- Maintain creative flow even offline

Pain Points

- Creative blocks amplified when internet fails

---

5) Key Features & Requirements

Below are broken into **functional**, **non-functional**, and **accessibility** requirements.

A) Functional Requirements

Feature	Description	Priority
Local LLM Chat	Generate responses offline	⭐⭐⭐⭐⭐
Document Upload (/upload)	Ingest PDFs into knowledge store	⭐⭐⭐⭐⭐
Retrieval-Augmented Generation (RAG)	Answer questions using personal docs	⭐⭐⭐⭐
Vector Memory Storage	Chunking + persistent DB (ChromaDB)	⭐⭐⭐⭐
Web Search (/search topic)	Fetch summary from internet when requested	⭐⭐⭐⭐
Streaming Response Output	Real-time token streaming	⭐⭐⭐⭐
Session Control	Exit commands & gratitude detection	⭐⭐⭐
Error Handling	Valid paths, internet failures, missing docs	⭐⭐⭐

B) Non-functional Requirements

Category	Requirement

Performance	Response time < 4 sec for standard prompts
Security	All data stored locally; no cloud calls unless /search
Device Requirements	CPU-only operation; 4GB RAM minimum
Scalability	Vector DB should grow with multiple documents
Resilience	Operates fully offline except when searching
Maintainability	Modular architecture for component swaps

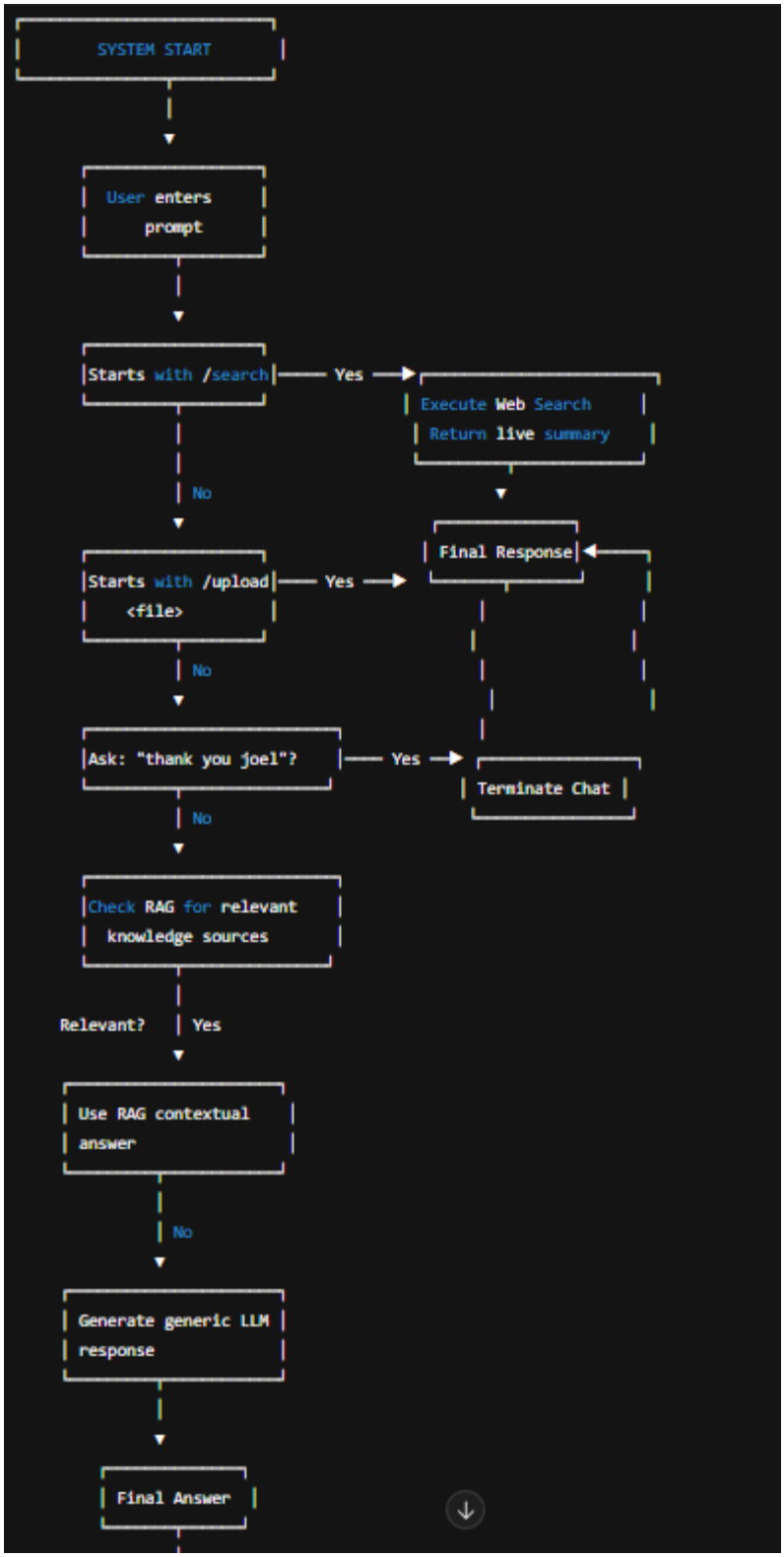
C) Accessibility / UX

Requirements

- No learning curve: CLI must be intuitive
- Provide guidance prompts (activation cues)
- Responses must be concise + context-grounded
- Support multiline input for ease of writing

---

6) System Design & Architecture


SYSTEM START
User enters prompt
Starts with /search
Yes
Execute Web Search
Return live summary
No
Final Response
Starts with /upload <file>
Yes
No
Ask: "thank you joel"?
Yes
Terminate Chat
No
Check RAG for relevant knowledge sources
Relevant?
Yes
Use RAG contextual answer
No
Generate generic LLM response
Final Answer

7) Feature Descriptions (Detailed)

7.1) Local Chat Model

- Runs small LLM using Ollama
- Supports text generation: summaries, rewrites, creative writing

Dependencies
ollama, small LLM model installed locally

7.2) RAG Pipeline

- PDFs extracted using **PyPDF**
- Text chunked + stored in **ChromaDB**
- Query matching using embeddings similarity

Value

- Becomes smart about **your own life & notes**
- Example:
  "JoelGPT, what is my project about?"

7.3) Web Search Mode (/search)

- Only activated intentionally
- DuckDuckGo search + BeautifulSoup scraping
- Summarization prompt applied to retrieved text

Value

- Location-independent access to news + live events
- Self-healing fallback if internet fails

7.4) Session Intelligence

- Detects gratitude

- Offers to complete session gracefully

7.5) Streaming Responses

- Real-time token output → engaging experience
- Mimics major LLM interfaces

7.6) Success Metrics (KPIs)

Metric Category	Success Measurement
Reliability	100% functioning during internet outages
Usage	Avg. 20 messages per session
Adoption	Minimum 50 active unique sessions/month
Performance	< 4 secs median response time
User Value	80% of tasks accomplished without using cloud tools
Accuracy	70%+ of RAG-based queries return correct context
Privacy	0 security incidents; no data leaves device

Qualitative success indicators:

- Students report **reduced dependence on paid AI**
- Comfort using personal documents in prompts
- Users feel empowered during offline moments

---

8) Competitive Landscape

Service	Internet Required?	Free?	Personal Knowledge Storage	Privacy
ChatGPT	✔	❌ (limited)	❌	❌
Gemini	✔	✔	❌	❌
Local LLMs (common)	❌	⭐	⚠️ limited	⭐
JoelGPT	❌ (except /search)	⭐	⭐⭐⭐⭐	⭐⭐⭐⭐⭐

---

9) Release Plan

- CLI interface
- Local LLM powered chat
- RAG with ChromaDB
- Web search integration
- PDF ingestion + storage

Next Releases

Phase	Features	Timeline
v1.1	Auto model selection, better chunking, TTS	+1 month
v1.2	GUI (Electron/Streamlit)	+3 months
v2.0	Voice assistant mode + mobile build	+6 months
v3.0	Personal agent automation workflows	+12 months

---

10) Risks & Mitigations

Risk	Impact	Mitigation
CPU-only may be slow on very low-powered devices	⚠	Model quantization + caching
Web search scraping fails for dynamic sites	⚠	Multiple sources fallback
Storage grows large with PDFs	⚠	User configurable deletion + compression
Small models may hallucinate	❌	Strict context-based answering rules

---

11) Value Proposition Summary

Value	Explanation
Offline Reliability	Never stops working due to cloud outages
Data Privacy	Documents & prompts never leave device
Low Hardware Barrier	Works on standard laptops with no GPU
Personalization	RAG turns JoelGPT into *your* assistant
Cost Savings	Completely free; no subscription needed
