

# System Design and Evaluation Document

This document provides a holistic view of the infrastructure supporting your application and clarifies the usage of different components within the project.

**Purpose:** The AlmaO1A FastAPI application is engineered to leverage OpenAI's GPT models for the analysis of CVs (Curriculum Vitae), transforming uploaded PDFs into insightful, actionable data. The application's design focuses on scalability, security, and ease of use, accommodating a broad range of deployment environments and user scenarios.

## Design Choices:

- **FastAPI Framework:** Opted for its high performance and easy asynchronous support, ideal for IO-bound and high-concurrency operations.
- **OpenAI GPT Models:** Utilized for their leading-edge natural language processing, enabling sophisticated text analysis using GPT 4 and GPT-3.5 turbo..
- **PDFPlumber for PDF Processing:** Chosen for its reliable text extraction capabilities, critical for accurate data analysis.
- **Jinja2 Templating:** Employed for dynamic HTML response rendering, facilitating easy updates and management of the web interface.
- **Environment Variables:** Implemented for secure management of sensitive data such as API keys, enhancing security by avoiding hard-coded credentials.

## Architecture:

- The application is structured around a microservices architecture, delineating the web interface, API layer, and text processing services. This separation enhances maintainability and supports easier scalability.

## Scalability:

- **Horizontal Scaling:** Achieved by deploying additional instances behind a load balancer to manage increased traffic and computational demand.
- **Asynchronous Processing:** FastAPI's asynchronous handling allows efficient management of numerous concurrent requests, crucial for performance under load.

## GitHub Integration:

- **GitHub Actions:** Automates workflows for continuous integration and deployment, ensuring that the application is always in a deployable state following any changes to the codebase.
- **GitHub Pages:** Used to host the project's static front-end, providing users with information about the application and access to the web interface.

## Template Folder Usage:

- Contains `base.html` for the core HTML structure, `upload_form.html` for the CV upload interface, and `results.html` for displaying analysis results. These templates ensure a consistent look and feel across the application and facilitate easy updates to the UI.

### Evaluation of Output:

- **Accuracy Measurement:** Text extraction and analysis accuracy are validated against a benchmark set of annotated CVs.
- **User Feedback:** Further, can be solicited to refine and enhance the model's accuracy and the overall user experience, also compare the accuracy with the currently approved O1-A applicant CVs to understand the analysis response the LLM provided.
- **Logging and Monitoring:** Integral for ongoing performance assessment, helping to identify and rectify issues swiftly.

**Conclusion:** The AlmaO1A FastAPI application stands out as a robust solution for CV analysis, designed with a focus on scalability, security, and user engagement. Continuous refinement based on user feedback and system monitoring ensures the application remains effective and relevant.

**Live Application:** Access the live application and learn more about its features at [AlmaO1A GitHub Pages](#).