

# Java Backend LLM Assessment - PDF Reader API Using LLM

## Overview

This project is a Java-based REST API designed to process and parse PDF files (typically CASA statements) by extracting key details—such as name, email, opening balance, and closing balance—using a language model (LLM) API. Additionally, it provides a feature for generating a bank account password based on user inputs.

## Detailed Solution

- **Language and Tools Used:**
  - **Java & Spring Boot:** Provides the foundation for a robust REST API.
  - **Apache PDFBox:** Extracts text content from PDF files.
  - **Jackson:** Handles JSON processing for converting responses.
  - **LLM API Integration:** Communicates with an LLM (e.g., OpenAI's GPT-3.5-turbo) to parse extracted text and return structured data.
- **Key Components:**
  - **PdfApplication:** The main class that boots up the application.
  - **PdfController:** Exposes two REST endpoints:
    - **api/parse:** Accepts a PDF file and returns extracted details.
    - **api/createPassword:** Generates a bank account password based on first name, date of birth, and account type.
  - **PdfParserService:** Contains the business logic to extract text from PDFs and to interact with the LLM API.
  - **PdfDataResponse:** A model class that encapsulates the parsed data.
  - **LlmClient:** A utility class for making HTTP requests to the LLM API and processing its responses.

## **Execution**

- **Run via Maven:** (mvn spring-boot:run) The application will start on port **8080**.
- **Testing the Endpoints:**
  - **Parsing a PDF:**
  - (curl -X POST -F "file=@/path/to/your.pdf" <http://localhost:8080/api/parse>)
  - **Generating a Password:**
  - (curl -X POST "http://localhost:8080/api/createPassword?firstname=John&dateOfBirth=1990-01-01&accountType=Savings")

## **Unsolved Challenges**

- **Running Errors:**

Despite the code being logically sound, the application currently experiences some runtime errors when calling the LLM API. These errors are under investigation, and additional error handling or retries might be necessary in future iterations.
- **LLM API Response Handling:**

The integration with the LLM API may require further improvements to gracefully handle different error conditions.

## **References**

- **Oracle Java Documentation:**  
<https://docs.oracle.com/javase/8/docs/api/javax/net/ssl/HttpsURLConnection.html>
- **Spring Boot Documentation:**  
<https://docs.spring.io/spring-boot/docs/current/reference/html/>
- **Apache PDFBox Documentation:**  
<https://pdfbox.apache.org/>
- **Jackson Documentation:**  
<https://github.com/FasterXML/jackson>

## **Acknowledgments**

- **Assistance:**  
This project was developed with valuable guidance and support from both ChatGPT and Claude.
- **Note:**  
The current version of the project contains some running errors that are being worked on.