EncureIT

Technical Assessment: Generative AI Developer.

Company: EncureIT Systems Pvt Ltd.

Role: Generative AI Developer

Task: Project Bank of Maharashtra: Loan Product Assistant

Time Limit: 48 hours from receipt

1. Project Overview & Goal

Welcome to the technical assessment for the Generative AI Developer role at EncureIT Systems Pvt Ltd.

The goal of this project is to build a proof-of-concept "Loan Product Assistant." This tool will be powered by a Retrieval-Augmented Generation (RAG) pipeline designed to answer questions specifically about the Bank of Maharashtra's loan offerings.

This is an open-ended task. We value your problem-solving approach, architectural choices, best practices and the clarity with which you can explain your decisions.

2. The Challenge: Your Core Tasks

Your mission is to build a system that can accurately answer questions about the Bank of Maharashtra's loan products, such as:

- "What are the interest rates for a Bank of Maharashtra home loan?"
- "What is the maximum tenure for a personal loan if my salary account is with the bank?"
- "Tell me about the Maha Super Flexi Housing Loan Scheme.”
- "Are there any processing fee concessions for women or defence personnel on home loans?"

To achieve this, you will need to:

1. **Scrape & Collect Data:** Gather public information focused exclusively on Bank of Maharashtra's loan products.

2. **Consolidate & Process Data:** Clean and structure the collected loan information into a unified knowledge base.
3. **Build a Lightweight RAG Pipeline:** Implement a simple RAG system that uses your consolidated data to answer user queries.

3. Detailed Requirements

Part A: Data Scraping & Collection

- Write scripts to scrape publicly available information about the Bank of Maharashtra's loan products. Your experience scraping websites for structured data will be valuable here.
- **Scope:** The data collection must be tightly focused on loans. Sources should ONLY include the official Bank of Maharashtra website, particularly the sections for Personal Loans, Home Loans, etc.

Part B: Data Consolidation & Processing

- Consolidate all the loan-specific information into a single, clean text document or a structured format.
- This critical step involves cleaning the data by removing irrelevant content (HTML tags, navigation, ads) to create a high-quality, focused knowledge base for your AI.

Part C: Lightweight RAG Pipeline

- Using your consolidated loan knowledge base, build a RAG pipeline.
- **"Lightweight" means** you do not need to use a complex, dedicated vector database. An in-memory solution like FAISS or a simple file-based vector store is sufficient.
- The pipeline must perform these core functions:
    a. Accept a user's question.
    b. Convert the question into a vector embedding.

c. Search the knowledge base to find the most relevant text chunks related to the question.

d. Provide the question and the retrieved text to a Large Language Model (LLM).

e. Return the LLM's generated answer.

- **LLM Choice:** You may use any accessible LLM via an API (e.g., from OpenAI, Anthropic, Google, etc. or use local LLM model). Please use a free tier or your own API key.

4. Submission Guidelines

Please submit your project as a link to a Git repository (e.g., GitHub). The repository **must** include:

1. **All Source Code:** Scraping scripts, data processing code, and RAG pipeline implementation.
2. **Processed Data File(s):** The final, clean knowledge base containing the loan information.
3. **A README.md File:** This is the most important part of your submission for our evaluation. It should be clear and comprehensive, containing:
    - **Project Setup:** Clear instructions on how to set up the project and run your code.
    - **Architectural Decisions:** Explain the reasoning behind your choices.
        - **Libraries:** Which libraries did you choose for scraping, data processing, and the RAG pipeline? Why?
        - **Data Strategy:** How did you approach chunking your text data for the vector search? Why that strategy?
        - **Model Selection:** Which embedding model and LLM did you use? What was your rationale?
        - **AI Tools Used:** Which tools did you leverage in this product and why?

- **Challenges Faced:** Briefly describe any issues you encountered (e.g., dynamic web pages, messy data formats) and how you solved them.
- **Potential Improvements:** If you had more time, what would you do next to enhance this solution?

4. **Video Walkthrough**: Please provide a link to a screen recording, no longer than five minutes, that demonstrates your project. In the video, walk through your implementation and explain the "what" and "why" behind your technical decisions. You may use any screen recording tool you are comfortable with (e.g., Loom, Snagit, etc) and host the video on a cloud storage platform of your choice.

5. Evaluation Criteria

This assessment will be evaluated based on the following criteria, reflecting the needs of a non-technical interviewer who must understand your work's value and thought process.

- **Functionality (30%):** Does the RAG pipeline work as intended and accurately answer questions about loan products?
- **Code Quality (20%):** Is the code clean, well-structured, and reusable?
- **Data Handling (20%):** How effectively did you scrape, clean, and structure the loan data?
- **Documentation & Reasoning (20%):** How clearly did you document your project and articulate your technical decisions in the `README.md`? This is a key measure of your communication skills.
- **Ability to use AI Tools (10%)**: Number of tools, how they were used and their output will be a key measure of your problem solving and ability to adapt to emerging trends in this field.

In case there are some questions or clarifications required regarding the task, evaluation criteria or anything else, please reach out to the HR with your questions.

Good luck! We look forward to seeing your work.