|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | M C SOMASEKHAR RAJU | | |
| **Email** | [Somasekhar.raju9@gmail.com](mailto:Somasekhar.raju9@gmail.com) | | |
| **Current Role** | Senior Full Stack Developer | **Number of Years/Months in the role** | 48+ |
| **Current Responsibilities** | End to End Development, Testing and Deployment  1) Develop the Front-End Components  2) Develop the Backend Components  3) Develop the Devops Components with Pipelines (CI / CD)  4) Unit Testing, Integration Testing  5) Ensuring Quality Assurance | | |

|  |
| --- |
| **Please describe technical competences you specialize in.**  **Note: use N/A if no experience.** |
| **Front-end Technologies: Angular, React, HTML, CSS, JavaScript and Typescript**  **Back-end Technologies: Java, Spring boot, JPA, Microservices**  **Databases: Postgres, Oracle, MongoDB**  **Cloud Platforms: AWS, Azure** |

|  |
| --- |
| **What is your exposure to building web applications that leverage AI/ML Models? How is the ML model integrated with the application? Please describe using a project you have worked on.** |
| **Exposure to Building Web Applications Leveraging AI/ML Models :-**  I have substantial experience in developing web applications that integrate AI and machine learning models. A notable pilot project I recently worked on was in the medical domain, where we developed a chatbot to assist patients by suggesting probable medicines based on their conditions. Below, I outline the step-by-step process of this project to illustrate my expertise and the integration of ML models within the application.  **Technologies Used:** React.js, Spring boot, RAG Techniques, Vector Databases(pinecone)   1. Implemented RAG to enhance the chatbot’s ability to provide accurate information by combining retrieval mechanisms with generative capabilities. 2. Utilized internal medical data, including medical literature, patient records (anonymized), and drug databases to train and fine-tune the model.   **Flow**   1. Patient enters symptoms into the chat interface. 2. The backend sends this input to the vector database to retrieve relevant medical documents and data. 3. The retrieved information is fed into the generative model, which synthesizes a response recommending probable medicines tailored to the patient’s condition. 4. The generated response is sent back to the frontend and displayed to the user.   **APIS**   1. Developed RESTful APIs to facilitate communication between the frontend, backend, and the ML models. 2. Ensured secure and efficient data transfer, adhering to medical data privacy standards.   \*\* Implemented feedback loops to continuously improve the model’s accuracy and the chatbot’s responsiveness based on user interactions and new medical data |
|  |

|  |
| --- |
| **Design and implement a web application that would allow users to:**  **Functional Requirements**   * **Upload and persist pricing feeds from retail stores using CSV files which contain Store ID, SKU, Product Name, Price, Date** * **Search for pricing records using various criteria and be able to edit/save changes to any record**   **Non-Functional Requirements**   * **Standard set of non-functional requirements you would expect a retail stores chain with 3000 stores across multiple countries**   **Please feel free to choose the technology stack and frameworks you are comfortable with and implement a single page web application.**  **Expected Deliverables:**   * **Context Diagram** * **Solution Architecture** * **Design Decisions** * **Non-functional requirements considered and how the design addresses them** * **Assumptions** * **Source for the implementation**   **Upload the artifacts and source to your Github repository and include a reference to it as part of the response.** |
|  |