

# **Vels University**

**MTECH – CSE**

**3<sup>rd</sup> Semester (2025- 2026)**

**24880107**

**Thanigaivel G**

**Project Guide :**

**Dr. Thirumal MTECH, PHD**

**Dr. Kumar MTECH, PHD**

# **A Low-Code LLM-Based Conversational AI Assistant for ERP Systems**

## **Abstract:**

Enterprise Resource Planning (ERP) systems are vital for managing business operations but often come with complex user interfaces and steep learning curves. This project introduces a low-code, Large Language Model (LLM)-powered conversational AI assistant that integrates seamlessly with ERP systems. The assistant enables users to interact with ERP modules using natural language, thereby improving usability, accessibility, and productivity. Leveraging low-code development platforms and state-of-the-art LLMs like GPT-4, the solution offers a scalable and adaptable approach to ERP interaction, reducing dependency on technical expertise.

## **1. Introduction:**

- Enterprise Resource Planning (ERP) systems are comprehensive software platforms used by organizations to manage business functions such as finance, supply chain, human resources, and customer relations. Despite their capabilities, ERP systems often pose usability challenges due to their complex interfaces. Users typically require extensive training to navigate and utilize ERP systems efficiently.
- Conversational AI, especially when powered by advanced LLMs, provides a promising solution to these usability issues. By allowing users to interact with ERP systems through natural language, the barrier to entry is significantly lowered. Additionally, using low-code platforms to develop such assistants accelerates deployment and simplifies maintenance.

## **2. Problem Statement:**

Current ERP interfaces demand significant user training and technical knowledge, leading to reduced productivity and increased operational costs. A solution is needed to:

- Simplify ERP interactions using natural language.
- Minimize the need for extensive coding or customization.
- Improve ERP accessibility for non-technical users.

## **3. Objectives:**

To design and develop a conversational AI assistant using LLMs. - To utilize low-code development tools (**Oracle Apex**) for easy integration and customization. - To enable secure and efficient interaction with ERP systems.

#### 4. Literature Survey:

Several studies highlight the potential of conversational agents in enterprise settings. Research in NLP has led to the development of powerful LLMs capable of understanding and generating human-like text. Tools like Oracle APEX and Microsoft Power Platform provide low-code solutions for building enterprise applications. However, limited work has been done on combining these technologies to create ERP-specific assistants.

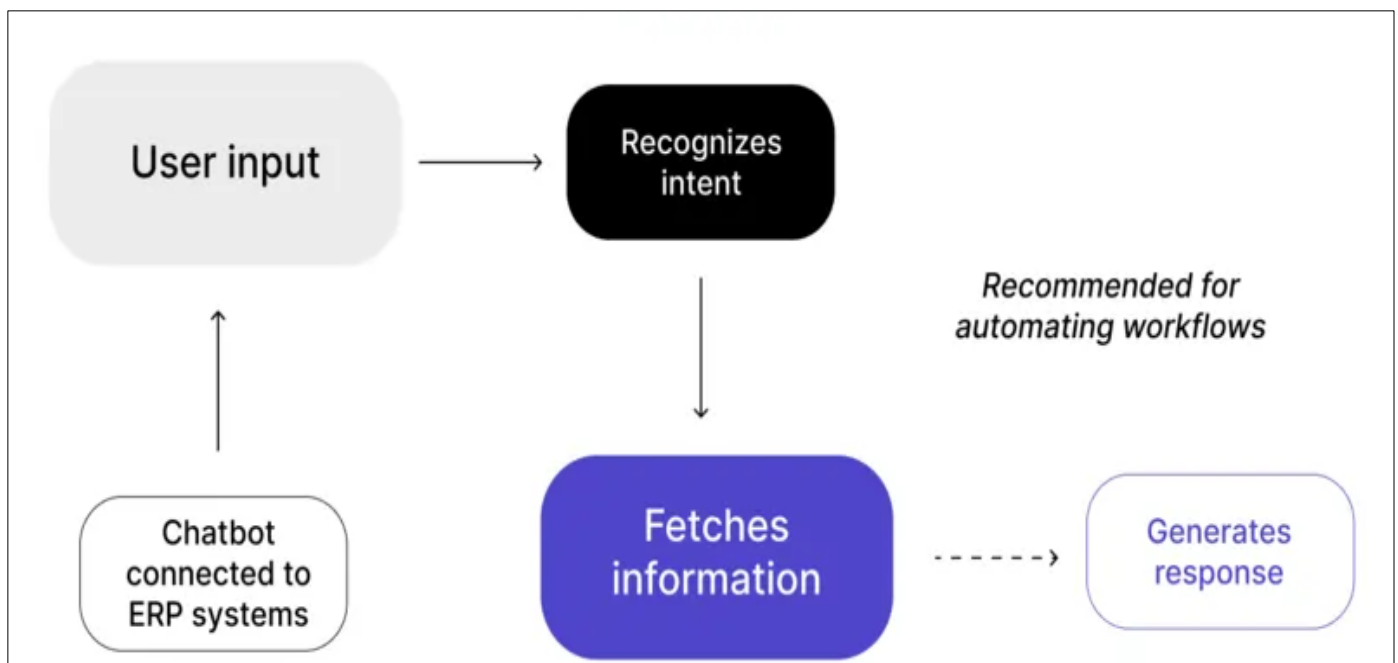
#### 5. System Architecture:

##### Components: -

- **User Interface:** Web or mobile-based chat interface built using low-code tools.
- **Middleware Layer:** Handles prompt formatting, user session management, and logging.
- **ERP Connector:** Integrates with ERP backend using APIs or direct SQL access.

##### Data Flow:

- User inputs query in natural language.
- Middleware formats input and sends it to the LLM.
- LLM processes input and returns structured commands.
- Middleware translates command into ERP-compatible actions.
- Results are fetched from ERP and displayed to the user.



## **6. Methodology:**

Using Oracle APEX to build the frontend chat interface. - Integrating GPT-4 using OpenAI or Azure API. - Using REST or SOAP APIs to interact with ERP backend. - Implementing role-based access control to ensure data security. - Provide fallback mechanisms for unsupported queries.

## **7. Features:**

- Low-Code Development using Oracle APEX
- Natural Language Processing with LLMs (Open-source or proprietary APIs)
- ERP Integration (HR, Finance, Inventory modules)
- Chat Interface for ease of access
- Role-Based Access & Logging for security

## **8. Tools and Technologies:**

- Oracle APEX (Low Code Platform)
- GPT-4 API
- Python Flask (Middleware)
- ERP Platform (Oracle EBS / SAP / Dynamics)

## **10. Expected Outcomes:**

A working prototype of a conversational AI assistant for ERP systems. Reduces training time and dependency on IT teams. Provides quick access to critical ERP data. Scalable to other departments and use cases. Enhances digital transformation in enterprises.

## **11. Conclusion:**

The integration of LLMs with ERP systems via low-code platforms presents a powerful solution to longstanding usability challenges. This project not only simplifies user interaction but also paves the way for smarter, more adaptive enterprise tools. The proposed assistant can significantly enhance user experience, operational efficiency, and accessibility within ERP ecosystems.

**12. Future Work:** - Integrating voice input/output capabilities. Extending support for multi-language interaction. Enable learning from user interactions to improve assistant accuracy. Adding support for additional ERP platforms and vertical-specific features.

**References:** 1. Oracle APEX Documentation 2. OpenAI API Documentation 3. SAP ERP Integration Guides 4. Microsoft Power Platform Whitepapers 5. Research Papers on Conversational AI and LLMs 6. Gartner Reports on Low-Code Development Platforms