Scale at Speed™

TECH mahindra

# Pharma Co-vigilance

LLM-Based Multi AI Agentic AE Email Triage Service

# Autonomous Case Intake Multi AI Agentic Gen AI Application - Overview

**Problem Statement:** The current case intake process in the pharmacovigilance is manual and inefficient. Intake Specialists manually read and process a large volume of emails daily, which is labor intensive and susceptible to human errors. This method does not scale well with increasing data volumes, resulting in higher costs and potential delays in safety case processing and reporting.

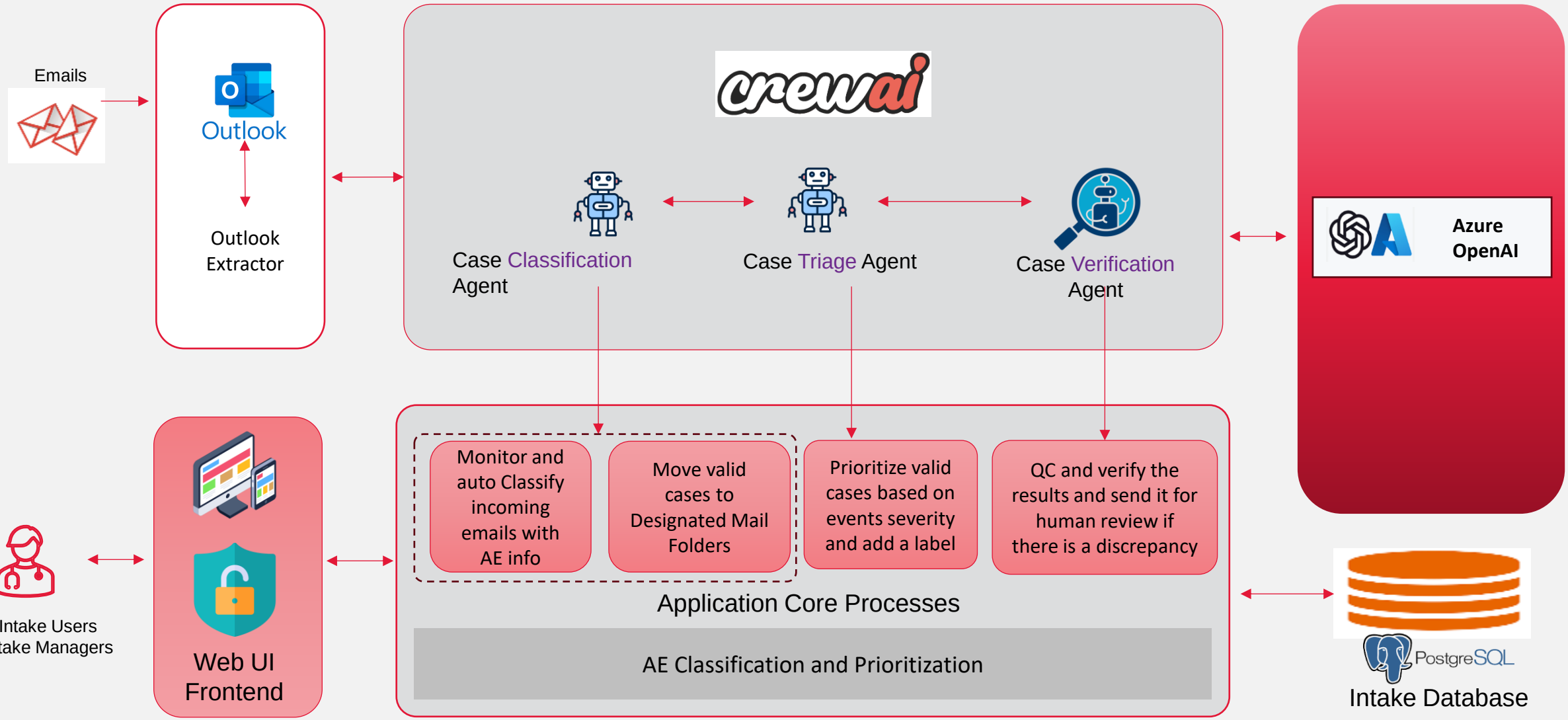
**Proposed Solution:** A Proof-of-Concept (PoC) application that enhances existing pharmacovigilance (PV) case intake processes by utilizing open-source/GEMINI generative AI models and AI frameworks. This application integrates with the company's email system to automate the monitoring, classification, prioritization, and verification of PV emails using LLM-based Multi AI agents.

| AI Agents: | | High-Level Functions: | |
|---|---|---|---|
| **Case Classification Agent:** | → | **Email Monitoring and Classification:** | • Integrates with the existing PV Intake email system (e.g., Microsoft Outlook/ Gmail).<br>• Analyzes Incoming emails in real-time and Automatically classifies emails with AE information or ICSRs as "Valid Cases" and those without as "Non-AE Emails."<br>• Moves valid cases into a designated "Valid Cases" folder and non-AE emails into a "non-AE" folder within the email system. |
| **Case Triage Agent** | → | **Prioritization:** | • Automatically prioritizes valid cases based on the severity of reported events, using configurable criteria and rules.<br>• Adds a priority label within the email header or subject line for easy identification. |
| **Case Verification Agent** | → | **QC and Verification:** | • Automatically verify the classification and prioritizations made by previous agent(s) and send it for human review if there is a discrepancy in the result. |
| | | **Administrative Interface and Dashboard:** | • Provides an interface for configuring classification tasks, prioritization criteria, and other business settings.<br>• Displays Daily, weekly and monthly metrics to help customers track performance and compliance.<br>• Display the Audit-Log |

# Autonomous Multi AI Agentic Gen AI PoC - Demo

- Scenario #1: AE-Fatal/Death Case - Demonstration of classifying it as a valid AE and Priority-1.
- Scenario #2: AE-Serious Case - Demonstration of classifying it as a valid AE and Priority-2.
- Scenario #3: AE-Non-Serious Case - Demonstration of classifying it as a valid AE and Priority-3.
- Scenario #4: Non-AE-Emails with Partial Info or Other Emails - Demonstration of classifying it as Non-AE.
- Scenario #5:  Human in the Loop

# Pharmacovigilance (PV) Case Intake Multi Agentic Architecture (PoC)



# LLM Agents - Flow Engineering and Key Aspects

Flow engineering – a process of designing and orchestrating the sequence of interactions and tasks that an AI agent or group of agents performs to achieve a specific goal.

| Key Aspects of Flow Engineering with LLM Agents: | |
|---|---|
| **Workflow Design:** | Map out the entire process, identifying key steps and decision points. Determine which tasks are best suited for LLM agents and which might require human intervention. |
| **Agent Configuration** | Define the roles and capabilities of each LLM agent in the workflow. Set up prompts, context, and constraints for each agent to guide their behavior. |
| **Agent Specialization** | Designing specialized agents for specific tasks or roles within the overall workflow |
| **Orchestration** | Implement a system to manage the flow of tasks between agents. This could be a centralized orchestrator or a decentralized system where agents communicate directly |
| **Task Decomposition** | Breaking down complex tasks into smaller, manageable subtasks that can be handled by individual agents or components. |
| **Interaction Patterns** | Defining how agents communicate and share information with each other. This can include sequential, parallel, or hierarchical interaction patterns |
| **Decision Points** | Identifying key decision points in the workflow where agents need to make choices or evaluations. Implementing logic for routing tasks based on these decisions. |
| **Data Flow Management** | Design how information is passed between agents, ensuring that each agent has the necessary context to perform its task. Implement data validation and transformation steps as needed |
| **Error Handling and Recovery** | Designing mechanisms for detecting and handling errors or unexpected situations. Implementing fallback strategies and human intervention points when needed |
| **Feedback Loops** | Incorporating feedback mechanisms for continuous improvement of the system. This can include learning from past interactions or human feedback |
| **Integration with External Systems** | Set up connections to databases, APIs, and other external tools that agents may need to access |
| **Monitoring and Logging** | Implement systems to track the progress of tasks through the workflow. Log agent interactions and decisions for auditing and improvement purposes |
| **Iteration and Optimization** | Analyze the performance of the workflow and individual agents. Refine the flow based on observed bottlenecks or inefficiencies. |

TECH mahindra