**Project Overview:**

This project aims to enhance the guest experience in the hospitality industry by integrating AI technologies, such as sentiment analysis and real-time personalization, into a comprehensive system that manages user feedback, preferences, and staff engagement. The system provides guests with personalized experiences while ensuring staff can respond swiftly to concerns or negative feedback.

**Product Owner:**

SINDU PATLOLLA

**Team**

SINDU PATLOLLA

**Stakeholders:**

Hotel

Epic 1: Setting Up Virtual Environment and Collecting CRM Dataset for Sentiment Analysis.

User Story 1.1 - Task of setting up local environment

Acceptance Criteria: Installed Python and local virtual environment

Status: Completed

Subtask 1.1.1- Install Python

Acceptance Criteria**:** Python is installed and working correctly in the local environment.

Status: Completed

User Story 1.2 - Task: Collecting Mock CRM Dataset

Acceptance Criteria**:** A CSV file containing mock CRM data for sentiment analysis is collected. Status**:** Completed

Subtask 1.2.1 - Collect CRM Data

Acceptance Criteria: Collect mock CRM data including UserID, Feedback, and Preferences from available sources.

Status: Completed

Subtask 1.2.2 - Organize Data into CSV Format

Acceptance Criteria: Format the collected data into a CSV file with columns such as UserID, Feedback, Preferences, and others necessary for analysis.

Status: Completed

Epic 2: Implementation of Sentiment Analysis Engine

User Story 2.1 - Creating Sentiment Analysis Engine Using LLM

Acceptance Criteria: The sentiment analysis engine correctly assesses and provides the sentiment (positive, negative, or neutral) of the feedback.

Status: Completed

Epic 3: Personalized Recommendation System and Dynamic Profile Management System

User Story 3.1 -Creating Personalized Recommendation System

Acceptance Criteria: The system generates personalized recommendations based on user preferences and interaction data, suggesting areas that the user has not visited or experienced yet.

Status: Completed

Subtask 3.1.1 - Implement Recommendation Logic

Acceptance Criteria: Establish and implement the logic that generates personalized recommendations based on the user's historical interaction data and preferences.

Status: Completed

User Story 3.2 -Profile Management System

Acceptance Criteria: The system correctly manages user profiles by using XAMPP to store, retrieve, and update user preferences and interactions in the database.

Status: Completed

Subtask 3.2.1 - Set Up Profile Management with XAMPP

Acceptance Criteria: Configure XAMPP with a MySQL database to store and manage user profiles, ensuring smooth retrieval and updates of user preferences.

Status: Completed

Epic 4: Staff Notification System Integrated with Slack & Email

User Story 4.1 - Integrating Staff Notification System with Slack & Email

Acceptance Criteria: The system sends notifications to staff members via Slack and email when negative feedback is received, and includes recommendations for improving user experience.

Status: Completed

Subtask 4.1.1 - Integrate Slack Notification System

Acceptance Criteria: Implement Slack integration to automatically send notifications to staff members when negative feedback is received, including user feedback and suggestions for improvement.

Status: Completed

Subtask 4.1.2 - Integrate Email Notification System

Acceptance Criteria: Set up email notifications to alert staff members about negative feedback, along with improvement recommendations for guest satisfaction.

Status: Completed

## **Sprint Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Goal** | **Key Tasks** | **Deliverables** |
| Sprint 1 | Local env setup and data collection | 1. Install python in locally, setup the virtual environment 2. Figure out data collection, explore synthetic data generation capabilities | 1. Data for feedback analysis 2. Virtual env setup |
| Sprint 2 | Develop a sentiment analysis system using LLMs | To extract **sentiment** and **suggestions for improvement** from a given text using an LLM | Feedback analyzer |
| Sprint 3 | Develop recommendation engine and dynamic profile recommendation system | Developing a  recommendation system that adapts based on user interactions, supported by a database connection to manage and store relevant data efficiently. | Recommendation engine module and profile management system |
| Sprint 4 | Add email and slack alerts | Integrate sentiment analysis and notification logic | Alerts |
|  |  |  |  |

## **Testing Plan**

**Unit Testing**:

` **Sentiment Analyzer:** Verify that the AI model correctly identifies emotional tone (positive, negative, neutral) from different feedback types.

 **Recommendation System:** Ensure the system generates personalized recommendations based on user interaction and preferences.

 **Dynamic Profile Management:** Test that the system correctly updates the user's preferences and information in the database.

 **Staff Alert Notification:** Verify that alerts are triggered appropriately for negative feedback and sent to the correct staff channels.

**Integration Testing**:

 **Sentiment Analyzer & Recommendation System:** Ensure that feedback analyzed by the sentiment analyzer correctly influences personalized recommendations.

 **Dynamic Profile Updates & Database:** Test the seamless interaction between user profile data, the database, and real-time updates.

 **Staff Alert Notification & Sentiment Analyzer:** Confirm that negative feedback triggers staff notifications in real time without delay.

**Performance Testing**:

 **Sentiment Analyzer:** Assess the response time of the sentiment analysis model with varying amounts of feedback data.

 **Recommendation System:** Test the system’s performance under heavy load (multiple guest interactions simultaneously).

 **Dynamic Profile Management:** Check how quickly the system updates user profiles in the database under high traffic.

 **Staff Alert Notification:** Ensure the alert system performs efficiently with real-time feedback processing, even with multiple generated alerts.

**User Testing**:

 **User Interaction (Guests):** Validate the process where guests provide feedback using their User ID and ensure correct system responses.

 **Sentiment Analysis:** Evaluate if the sentiment analysis accurately reflects the guest's feedback and identifies concerns.

 **Real-time Personalization:** Make personalized recommendations dynamic and change based on guest preferences and current feedback.

 **Instant Action to Negative Feedback:** Test the speed and accuracy of alert notifications sent to staff when negative feedback is detected.

 **Continuous Engagement:** Assess the relevance and timing of personalized suggestions throughout the guest’s stay, ensuring a tailored experience.

## **Key Metrics**

* **Recommendation Accuracy**: Precision, Recall.
* **Sentiment Classification Accuracy**: F1 Score
* **Profile Update Latency**: Average time to update profiles.
* **Feedback Processing Time**: Time taken to analyze feedback and trigger alerts.