

# OllamaNet

## Feature Document

## Table of Contents

1. Introduction
2. Feature Summary
3. Detailed Feature Descriptions
  - User Features
    - User Interface for AI Interaction
    - Multi-turn Conversations
    - Personalized Recommendations
    - Real-Time Communication Layer
    - Document Upload for Contextual Responses
    - Feedback and Insights
  - Admin Features
    - Analytics Dashboard
    - Feedback Moderation
    - AI Model Monitoring
    - System Notifications and Alerts
4. Non-Functional Requirements
5. Assumptions and Constraints
6. Future Enhancements

# 1. Introduction

## Purpose

This document outlines the features and functionalities of OllamaNet, a platform designed to connect users with Ollama's AI models for advanced text processing, personalized recommendations, and real-time multi-turn conversations. OllamaNet focuses on facilitating user interaction while relying on Ollama's AI for processing.

## Target Audience

Developers, stakeholders, and team members involved in the design and implementation of OllamaNet.

## Scope

OllamaNet integrates Ollama's suite of AI models into a .NET application, providing seamless interaction through a user-friendly interface. The AI processing, including text generation, summarization, and contextual understanding, is handled entirely by Ollama's services.

## 2. Feature Summary

### User Features

1. User Interface for AI Interaction
2. Multi-turn Conversations for Coherent Dialogues
3. Personalized Recommendations Based on User Interactions
4. Real-Time Communication Layer
5. Document Upload for Context-Aware Responses
6. User & Session Management
7. Feedback and Insights

### Admin Features

7. Analytics Dashboard
8. Feedback Moderation
9. AI Model Monitoring
10. System Notifications and Alerts

## 3. Detailed Feature Descriptions

### 3.1 User Interface for AI Interaction

- **Description:** A simple and intuitive UI for users to interact with Ollama's AI models.
  - **Purpose:** Enable users to input queries and view AI-generated responses seamlessly.
  - **Users/Actors:** All users of the website.
  - **Functionality:**
    - Text input field for user queries.
    - Display AI responses in a conversational format.
  - **Dependencies:** API integration with Ollama.
  - **Design/UX Considerations:**
    - Responsive and clean interface.
    - Options for light and dark modes.
- 

### 3.2 Multi-Turn Conversations

- **Description:** Enable users to maintain context across multiple interactions with the AI.
  - **Purpose:** Facilitate coherent dialogues for an enhanced conversational experience.
  - **Users/Actors:** Registered users and guests.
  - **Functionality:**
    - Retain session context during conversations.
    - Allow users to reset or save conversation history.
  - **Dependencies:** Context management via Ollama's API and session tracking.
- 

### 3.3 Personalized Recommendations

- **Description:** Suggest relevant queries, topics, or features based on user interactions.
- **Purpose:** Increase user engagement by providing tailored suggestions.
- **Users/Actors:** Registered users.
- **Functionality:**
  - Analyze interaction patterns.
  - Suggest queries or provide tips for using Ollama effectively.
- **Dependencies:** User profile module and recommendation algorithms.

### 3.4 Real-Time Communication Layer

- **Description:** Ensure real-time communication between users and Ollama's AI.
  - **Purpose:** Provide fast and responsive interaction.
  - **Users/Actors:** All users.
  - **Functionality:**
    - Utilize WebSocket or SignalR for real-time updates.
    - Dynamically refresh the chat interface with AI responses.
  - **Dependencies:** API integration and real-time communication framework.
- 

### 3.5 Document Upload for Contextual Responses

- **Description:** Allow users to upload documents for context-aware responses.
  - **Purpose:** Enhance the relevance and accuracy of AI responses based on user-provided content.
  - **Users/Actors:** Users uploading documents for interaction.
  - **Functionality:**
    - Accept common document formats (e.g., PDF, DOCX).
    - Send documents to Ollama's API for processing.
    - Display responses derived from document content.
  - **Dependencies:** Vector database and embedding model integration with Ollama's API.
- 

### 3.6 User & Session Management

- **Description:** Manage user accounts and session data to provide a personalized experience.
- **Purpose:** Ensure secure access and seamless multi-turn conversation support.
- **Users/Actors:** Registered users and admins.
- **Functionality:**
  - User authentication (registration, login, password recovery).
  - Session tracking for ongoing interactions.
- **Dependencies:** Database for storing user and session data.

### 3.7 Feedback and Insights

- **Description:** Collect user feedback to improve the platform and the AI interaction experience.
  - **Purpose:** Enhance user satisfaction and optimize the system based on feedback.
  - **Users/Actors:** All users and administrators.
  - **Functionality:**
    - Feedback form for user input on AI responses.
    - Generate insights from feedback for platform improvement.
- 

### 3.8 Analytics Dashboard

- **Description:** A centralized dashboard for administrators to monitor platform usage and performance.
  - **Purpose:** Provide actionable insights for optimization.
  - **Functionality:**
    - Display key metrics such as:
      - Active user count.
      - Most common queries.
      - Session durations.
    - Generate detailed reports on user behavior and trends.
  - **Dependencies:** Database for storing and retrieving analytics data.
- 

### 3.9 Feedback Moderation

- **Description:** Tools to review, filter, and moderate user feedback.
- **Purpose:** Maintain constructive and appropriate feedback on the platform.
- **Functionality:**
  - Flag inappropriate feedback automatically using keywords or sentiment analysis.
  - Provide admins with options to approve, edit, or delete flagged feedback.

### 3.10 AI Model Monitoring

- **Description:** Track the performance of integrated AI models and identify issues.
  - **Purpose:** Ensure consistent and reliable AI interaction for users.
  - **Functionality:**
    - Monitor response times and error rates from Ollama's API.
    - Display usage statistics for each AI model in use.
- 

### 3.11 System Notifications and Alerts

- **Description:** Notify administrators of critical system updates or anomalies.
- **Purpose:** Enable quick response to potential issues.
- **Functionality:**
  - Send alerts for:
    - API downtime or errors.
    - High user traffic or unusual patterns.
    - System updates or maintenance schedules.



## 5. Non-Functional Requirements

- **Performance:** Ensure responses are delivered within 1 second under normal load.
  - **Scalability:** Support up to 10,000 concurrent users.
  - **Security:** Secure user data and API communication with encryption.
  - **Availability:** Maintain 99.9% uptime for real-time interactions.
- 

## 6. Assumptions and Constraints

- **Assumptions:**
    - Ollama's API will handle all AI-related tasks effectively.
    - Users will access the platform via modern browsers with a stable internet connection.
  - **Constraints:**
    - Budget limits for third-party integrations.
    - Dependency on Ollama's API for core functionality.
- 

## 7. Future Enhancements

- Support for voice-based queries and interactions.
- Multi-language support for broader accessibility.
- Offline mode with basic functionality for document uploads and AI interactions.