ONDC | presents

Open Network for Digital Commerce

Build for Bharat

Supported by #startupindia

Sponsors

Google Cloud ANTLER

protean paytm

Powered by

H2S

Team Name: Devdynamos

Team Leader Name: Patnala Rajya Lakshmi

Problem Statement Category: Foundational

Problem Statement: Negotiation engine

Architecture & Design for the innovative solution.

Architecture & Design:

- **Frontend Interface:**
  1. Built using modern web technologies (HTML, CSS, JavaScript) for a responsive and intuitive user experience.
  2. Implements WebSocket or Server-Sent Events (SSE) for real-time updates and notifications.
- **Intelligent Analyzer:**
  1. Integrates machine learning models trained on historical negotiation data to identify patterns and predict outcomes.
  2. Utilizes natural language processing (NLP) techniques to understand and analyze textual terms and conditions.
  3. Leverages frameworks like TensorFlow or PyTorch for model development and deployment.

Architecture & Design for the innovative solution.

- **Backend Services:**
  1. Hosted on a cloud-based infrastructure (AWS, Azure, Google Cloud) for scalability and reliability.
  2. Utilizes microservices architecture for modularity and scalability.
  3. Implements RESTful APIs for communication between frontend and backend components.
  4. Utilizes containerization (Docker) and orchestration (Kubernetes) for easy deployment and scaling.
- **API Gateway:**
  1. Provides a unified interface for accessing backend services.
  2. Implements API management features such as authentication, rate limiting, and logging.
  3. Supports both synchronous and asynchronous communication patterns.

Architecture & Design for the innovative solution.

- **Security Measures:**
  1. Implements end-to-end encryption using SSL/TLS protocols to secure data transmission.
  2. Utilizes role-based access control (RBAC) to enforce least privilege access.
  3. Conducts regular security audits and penetration testing to identify and mitigate vulnerabilities.

Define customization & deployment options of your proposed solution.

Customization Options:

1.User Interface Customization: The frontend interface can be customized to align with the branding and user experience preferences of different e-commerce platforms and marketplaces. This includes customizing colors, logos, layouts, and interactive elements.

2.Negotiation Workflow Customization: The negotiation workflow can be tailored to accommodate specific business rules, regulations, and industry standards of different sectors or regions. This includes defining custom stages, rules for proposal acceptance, and escalation mechanisms.

3.Machine Learning Model Customization: The intelligent analyzer module can be customized by retraining machine learning models with domain-specific data to improve accuracy and relevance. This allows for adapting to unique negotiation patterns and preferences of different market segments.

4.Integration Customization: The API gateway supports customization to accommodate different integration requirements of e-commerce platforms, marketplaces, and backend systems. This includes customizing API endpoints, data formats, and authentication mechanisms.

Define customization & deployment options of your proposed solution.

Deployment Options:

1.Cloud Deployment: The solution can be deployed on cloud platforms such as AWS, Azure, or Google Cloud for scalability, reliability, and ease of management. This option offers flexibility in resource allocation and geographical distribution to meet varying performance and regulatory requirements.

2.On-Premises Deployment: For organizations with strict regulatory compliance or data sovereignty requirements, the solution can be deployed on-premises within their own infrastructure. This option provides greater control over data security and privacy while still benefiting from the solution's features.

3.Hybrid Deployment: Organizations can opt for a hybrid deployment model, combining elements of cloud and on-premises deployment to leverage the benefits of both approaches. This allows for maintaining sensitive data on-premises while utilizing cloud resources for scalability and redundancy.

4. **Managed Service Deployment:** Organizations can opt for a managed service deployment model where the solution is hosted and managed by a third-party service provider. This option reduces the burden on internal IT teams and ensures continuous support and maintenance of the solution.

Test cases & data (as applicable) against which the eval criteria can be assessed

1.Proposal Submission:

1. Test Case: User A submits a proposal with specific terms, including price, quantity, and delivery timeline.
2. Expected Outcome: The proposal is successfully submitted and displayed to User B for review.

2.Counter-Proposal Submission:

1. Test Case: User B submits a counter-proposal with revised terms in response to User A's proposal.
2. Expected Outcome: The counter-proposal is sent to User A, and both parties can view and consider the updated terms.

3.Acceptance of Counter-Proposal:

1. Test Case: User A accepts the counter-proposal submitted by User B.
2. Expected Outcome: The negotiation is successfully concluded, and both parties agree to the finalized terms.

4.Rejection of Counter-Proposal:

1. Test Case: User A rejects the counter-proposal submitted by User B and provides a rationale.
2. Expected Outcome: User B receives notification of the rejection and the rationale provided by

Test cases & data (as applicable) against which the eval criteria can be assessed

5. Automatic Timeout Handling:

1. Test Case: A negotiation remains inactive without any responses from either party beyond the specified time limit.
2. Expected Outcome: The system automatically terminates the negotiation due to inactivity and notifies both parties accordingly.

6. Notification Handling:

1. Test Case: User A receives a notification when User B submits a counter-proposal.
2. Expected Outcome: User A is promptly notified of the new counter-proposal, enabling timely review and response.

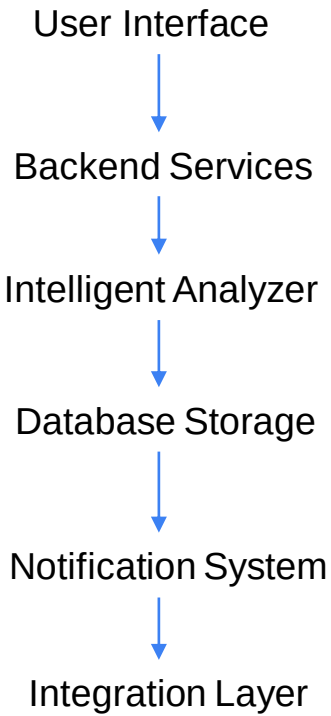
7.Error Scenario:

1. Test Case: An error occurs during the negotiation process due to a server issue or network interruption.
2. Expected Outcome: The system displays an error message to the affected users and provides guidance on how to proceed or report the issue.

Test cases & data (as applicable) against which the eval criteria can be assessed

9. **Data Integrity Check:**
    1. Test Case: Verify that all submitted proposals, counter-proposals, and negotiation status updates are accurately recorded and reflected in the system.
    2. Expected Outcome: The system maintains data integrity and consistency throughout the negotiation process, preventing data loss or discrepancies.
10. **Performance Under Load:**
    1. Test Case: Simulate a high volume of concurrent negotiations to assess system performance.
    2. Expected Outcome: The system handles the increased load efficiently, maintaining acceptable response times and stability without degradation.

Demonstrate working of the solution



Demonstrate working of the solution to the evaluation team

Components:

1.User Interface:

1. Frontend interface where users interact with the negotiation engine, submit proposals, review terms, and receive notifications.

2.Backend Services:

1. Backend infrastructure responsible for processing user requests, managing negotiation data, and handling business logic.

3.Intelligent Analyzer:

1. Module that analyzes negotiation data, identifies patterns, trends, and areas of convergence or divergence in the negotiation terms.

4.Database Storage:

1. Database system for storing negotiation data securely, ensuring data integrity and confidentiality.

5.Notification System:

1. System responsible for sending real-time updates and notifications to users regarding the status of their negotiations.

6.Integration Layer:

1. Middleware layer facilitating integration with external systems such as e-commerce platforms, marketplaces, and other backend services.

Demonstrate working of the solution to the evaluation team

Flow:

1.Users interact with the User Interface to initiate, review, and respond to negotiation proposals.

2.User requests are processed by the Backend Services, which handle the negotiation logic and data management.

3.The Intelligent Analyzer continuously analyzes negotiation data to provide insights and assistance to users during the negotiation process.

4.Negotiation data is securely stored in the Database Storage for future reference and analysis.

5.The Notification System sends real-time updates and notifications to users, keeping them informed of the status of their negotiations.

6.The Integration Layer facilitates seamless integration with external systems, allowing for

Important Links:-

- GitHub Public Repository Link:

https://github.com/Rajya-Lakshmi-2003/NegotiationAPI

ONDC | presents

Build for Bharat

Supported by #startupindia

Sponsors Google Cloud ANTLER protean paytm

Powered by H2S

THANK YOU