Software Engineering Final Lab Report

Code: CSE 4722

Department of Computer Science and Engineering

Section: A

Title: MediChat: A Reliable Medical Chatbot

Submitted By:

Md. Raihan Tapader
ID: 204006

Ariful Islam Shiplu
ID: 204029

Tawhidul Islam
ID: 204018

Submitted To:

Momotaz Begum, PhD
Professor, Dept of CSE

Litan Islam,
Lecturer, Dept of CSE

Signature……………………

Dhaka University of Engineering & Technology, Gazipur.

Index

Serial No	Name of Activity	Page Num
1	Development of Problem Statements, Objectives, and Outcomes	3-5
2	Preparation of SRS, Design, and Testing Docs	6-7
3	Study and usage of Design phase UML CASE tool	8-9
4	Software Methodology and Development Environment	10-11
5	Testing and Deployment	11-19
6	Conclusion	19-20
7	References	20

Title: MediChat: A Reliable Medical Chatbot

Objectives

Our aim is to:

- To develop an AI-powered **chatbot** capable of providing users with **quick**, **reliable**, and **easy-to-understand** medical information and guidance **anytime**, anywhere.
- To reduce the **burden** on healthcare facilities by addressing common medical **queries** and promoting **preventive** healthcare awareness among users, especially in **underserved** areas.

Outcomes

- Instant, reliable medical guidance for users, helping them make better health decisions anytime, especially in underserved or rural areas.
- Reduced burden on healthcare facilities by addressing common medical queries and promoting preventive health awareness.
- User-friendly, scalable chatbot system that can be continuously updated with new medical information to stay relevant and accurate.

Background

- Bangladesh has improved basic healthcare through expanded immunization, community clinics, and reduced child and maternal mortality.
- The sector still struggles with overcrowded public hospitals, shortages of specialist doctors in rural areas, and significant urban–rural disparities.
- A growing private sector offers modern services mainly in cities, but affordability, uneven quality control, and limited technological integration remain challenges.

Work list:

WEEK 1: Development of Problem Statement.

WEEK 2: Preparation of Software Requirement Specification (SRS), Design and Testing Documents.

WEEK 3: Software Configuration & Risk Management.

WEEK 4: System Design using UML, CASE Tool Study & Implementation.

WEEK 5: Test Cases and Results.

WEEK 1

AIM: To develop problem statements for a Medical Chatbot System.

Problem Statement: The Medical Chatbot System addresses the need for accessible, preliminary medical guidance for users and efficient administrative oversight for healthcare providers. The system aims to bridge the gap between individuals seeking health information and professional medical services by offering an interactive platform for health queries, appointment management, and medical history access.

Process Flow:

- **User Interaction:** The user accesses the web-based interface and enters their symptoms into a text box.
- **Symptom Processing:** The system captures the user's input.
- **LLM Invocation:** The input, along with conversation history and a system prompt, is sent to the OpenAI GPT-3.5-turbo model via the LangChain framework.
- **Response Generation:** The LLM processes the information and generates a response based on predefined instructions, including possible conditions, OTC recommendations, lifestyle advice, and mandatory disclaimers.

- **Display Response:** The generated response is displayed to the user in the chat interface.
- **Chat Management:** The user can continue the conversation or clear the chat history to start a new session.

Components:

- **Proposed Solution:** A web based chatbot system using Python, LangChain, and Gradio for the interface, integrated with OpenAI's GPT-3.5-turbo for natural language processing.
- **Roles & Responsibilities:**

  – **General User:** Submits health queries, views suggested diagnoses, appointments, and medical history.

  – **Administrator:** Moderates responses, manages user accounts, and updates the medical database.

  – **System:** Processes queries, retrieves data, and ensures secure communication.

Inputs:

- User symptoms (text input)
- Administrator inputs for database updates and response moderation

Problems/Constraints:

- Misinterpretation of symptoms.
- Inaccurate medical advice risk.
- Chat history overload.
- External dependencies (OpenAI, Langchain, Gradio).

WEEK 2

AIM: To create comprehensive Software Requirement Specification(SRS) and Design documentation.

SRS Document containing:

1. Functional Requirements:

- Ask Health-Related Questions: Users submit symptoms to receive automated responses.
- Provide Answer: The chatbot retrieves relevant information from the knowledge base.
- View Suggested Diagnosis: Suggests possible conditions with disclaimers.
- View Appointment/Medical History: Users access their records; administrators access all records.
- Get Emergency Tips: Provides advice for critical symptoms.
- Moderate Responses: Administrators review and edit chatbot outputs.
- Manage Users/Database: Administrators add, update, or remove accounts and medical data.

2. Non-functional requirements:

- Performance: Responds to queries within 2 seconds under standard load; supports 1,000 concurrent users.
- Security: End-to-end encryption, role-based access, and HIPAA compliance.
- Usability: Intuitive interface for desktop and mobile, adhering to accessibility standards.
- Availability: 99.5% monthly uptime via cloud hosting.
- Extensibility: Modular design for future enhancements (e.g., multilingual support, wearable integration).

System Modules:

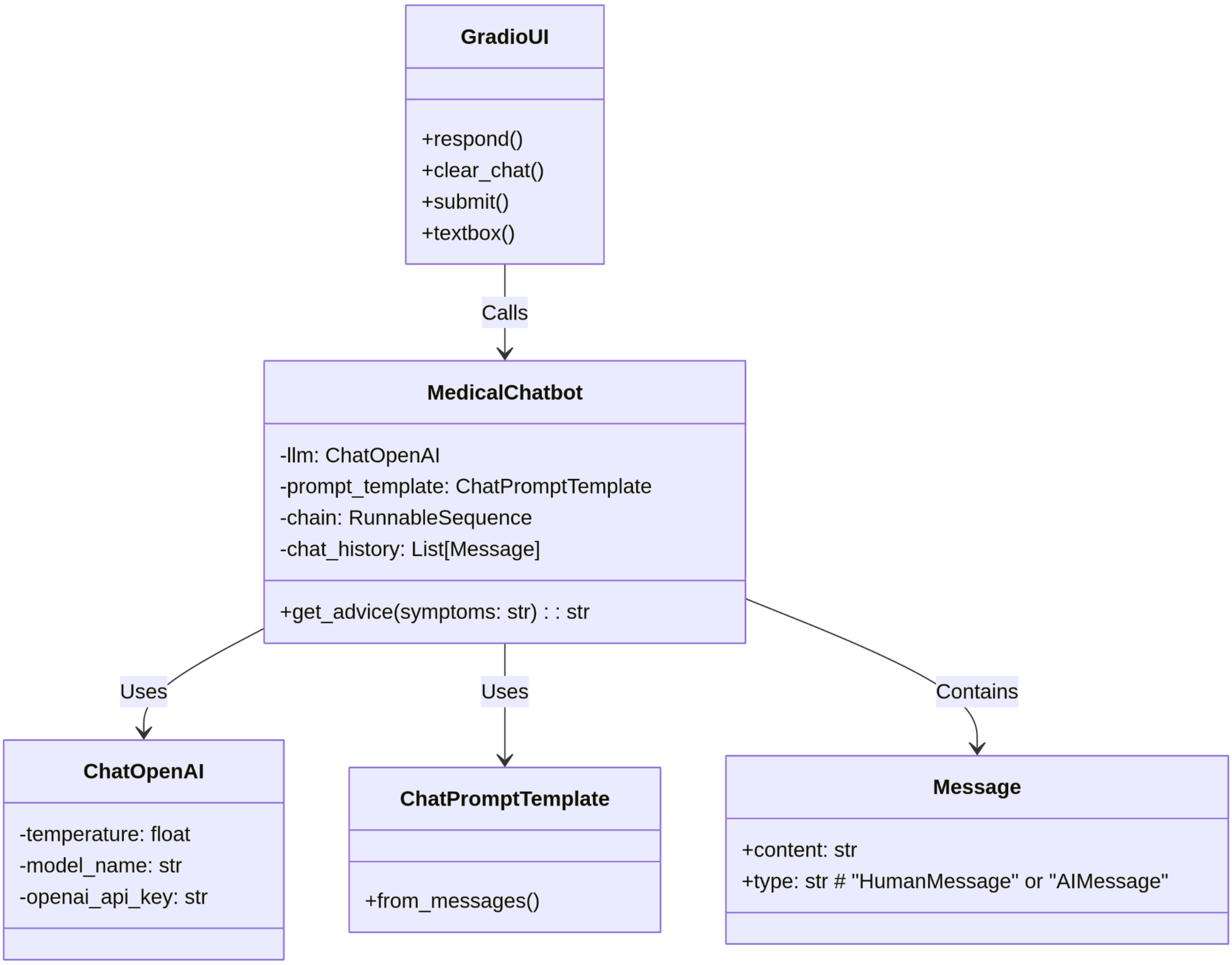
1. **Chat UI** (gradio): The front-end component built with Gradio. It provides the chat window, text input box, send button, and clear button.
2. **Medical Chatbot Core** (MedicalChatbot Class): The main logic class that initializes the LLM, manages the prompt template, and handles the conversation flow.
3. **Language Model Integration** (langchain): The framework used to connect to the OpenAI API, manage prompts, and structure the interaction with the LLM.

WEEK 3

Aim: Design using UML CASE tool and produce diagrams.

CASE Tool: UML (or alternatives) used to craft diagrams including Use Case, Class, Sequence, and Activity.

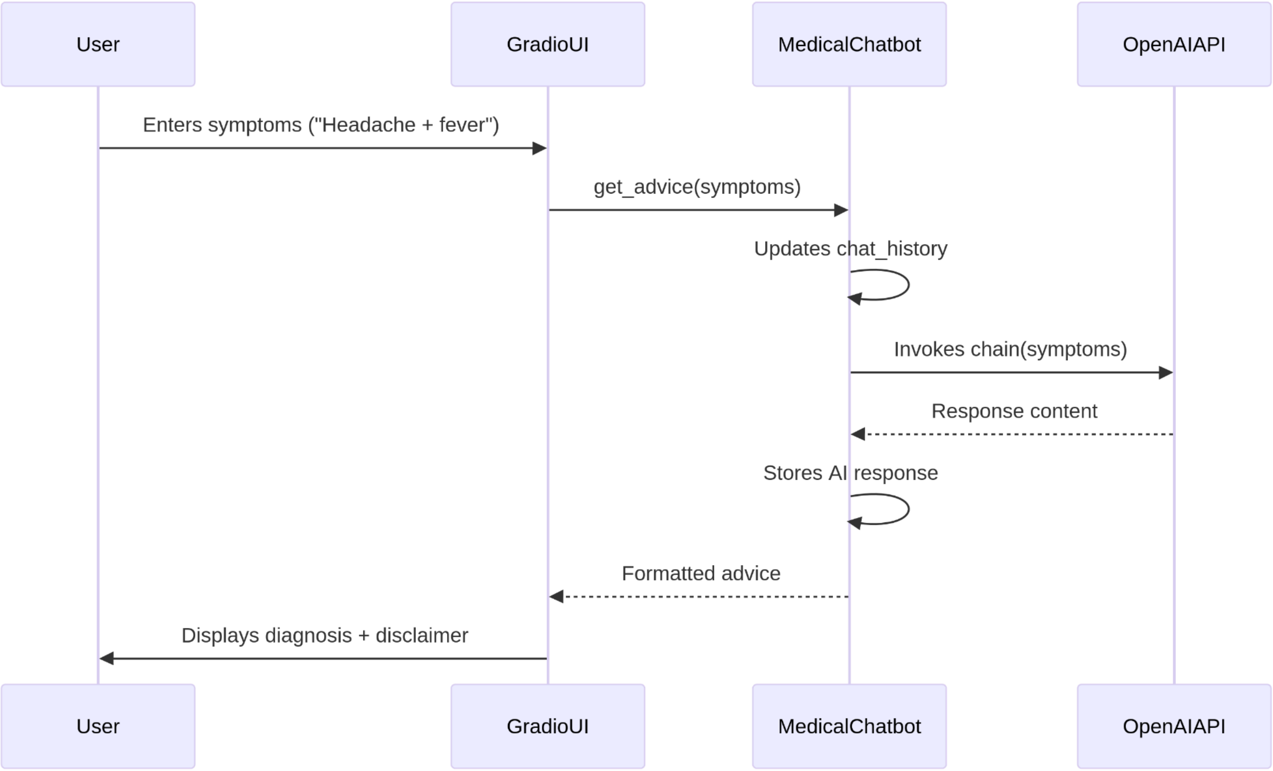
UML class Diagram:



Key Classes:

- **`MedicalChatbot`**: Core class handling:
    - LLM initialization (`ChatOpenAI`)
    - Prompt templating (`ChatPromptTemplate`)
    - Chat history management (`HumanMessage`, `AIMessage`)
- **`ChatOpenAI`**: Wrapper for OpenAI API calls
- **`ChatPromptTemplate`**: Structures the system/patient message format
- **Message Classes**: Store conversation context

Sequence Diagram (Chat Flow)



Flow Explanation:

1. User input triggers Gradio interface.
2. Chatbot processes query using LangChain prompt template.
3. OpenAI API generates medical response.
4. Output is formatted with disclaimers and shown to user.

WEEK 4

AIM: To define the methodology and establish development environment specifications.

Programming Language: Python 3.x

Libraries:

- `gradio`: For creating the web-based UI.
- `langchain-openai`: For integrating with the OpenAI API.
- `langchain`: For building the core LLM chain and prompt management.

API: OpenAI API key with access to the `gpt-3.5-turbo` model.

Operating System: Windows, macOS, or Linux.

Datasets:

- Symptom-Disease Mapping Datasets [1]
- Medical Question-Answer Datasets [2]
- Medical Conversations / Dialogue Datasets [3]
- Healthcare FAQs and Preventive Care Data [4]
- Public Health Data [5]

Software Configuration Management

- Operating System: Windows 10 or compatible
- Frontend: Gradio (Python-based)
- Backend: LangChain with OpenAI GPT-3.5-turbo, MySQL (for potential database integration)
- IDE: Visual Studio Code or PyCharm

Hardware Requirements

- Processor: Intel i3 or higher
- RAM: 4 GB or higher

- Storage: 500 GB
- Internet Connection: Required for API calls to OpenAI.

Risk Management

Risk	Severity	Mitigation
Inaccurate Responses	Medium	Implement response moderation and continuous model
Server Overload	High	Use cloud-based scalability (e.g., AWS, Azure)
Security Breaches	High	End-to-end encryption, regular audits
Broken Gradio UI	Medium	Handle UI errors; fallback standard IO
API Downtime	Medium	Fallback to local knowledge base

WEEK 5

Types of Testing Performed: Testing and Deployment

1.Unit Testing: Unit Testing is a software testing method where individual components or units of a program (like functions, methods, or classes) are tested in isolation to verify they work as intended.

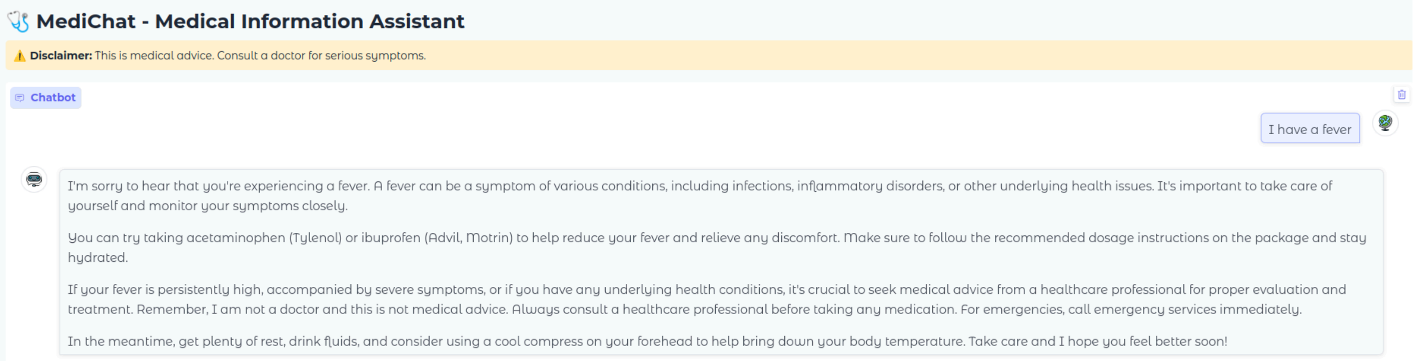
Unit testing ensures each component of the chatbot works correctly in isolation. For this project, we test:

- `get_advice()` **method** – Verifies symptom input (e.g., "fever and cough") generates a properly formatted response with disclaimers.
- **Prompt Template** – Checks if the system message and user inputs are correctly structured for OpenAI.
- **Error Handling** – Ensures invalid/empty inputs return graceful errors instead of crashing.

- **Chat History** – Confirms user and AI messages are stored accurately in the conversation log.
- **API Integration** – Mocks OpenAI calls to validate responses without real API usage.

Unit Testing Results:

TestID	Function	Input	Expected Output	Actual Output	Status
UT-01	Intent detection	“I have a fever”	Detect:Symptom inquiry intent	Correctly detected	Pass
UT-02	Response generation	“What is dengue?”	Return dengue info summary	Correct Summary returns	Pass
UT-03	Fallback handler	“(Empty message)”	Prompt:”Please ask a question	Correct Prompt given	Pass


MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

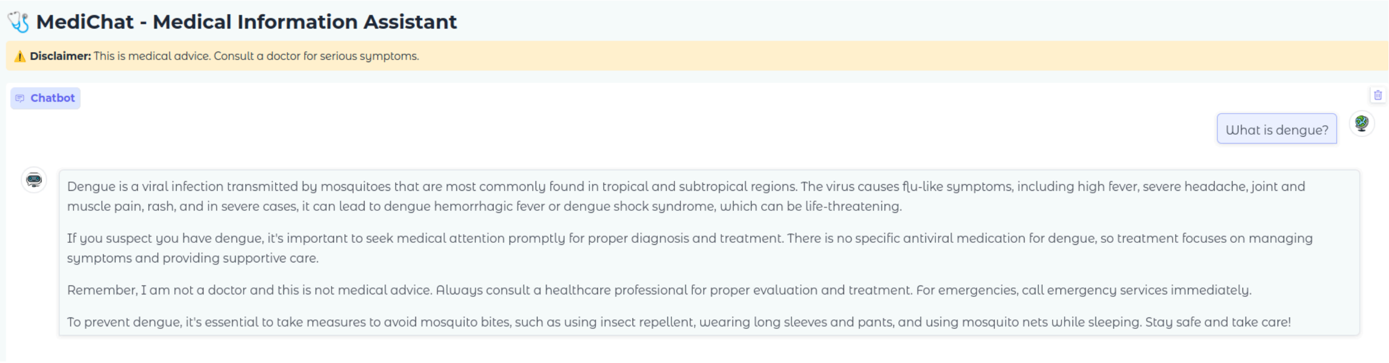
I have a fever

I'm sorry to hear that you're experiencing a fever. A fever can be a symptom of various conditions, including infections, inflammatory disorders, or other underlying health issues. It's important to take care of yourself and monitor your symptoms closely.

You can try taking acetaminophen (Tylenol) or ibuprofen (Advil, Motrin) to help reduce your fever and relieve any discomfort. Make sure to follow the recommended dosage instructions on the package and stay hydrated.

If your fever is persistently high, accompanied by severe symptoms, or if you have any underlying health conditions, it's crucial to seek medical advice from a healthcare professional for proper evaluation and treatment. Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional before taking any medication. For emergencies, call emergency services immediately.

In the meantime, get plenty of rest, drink fluids, and consider using a cool compress on your forehead to help bring down your body temperature. Take care and I hope you feel better soon!


MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

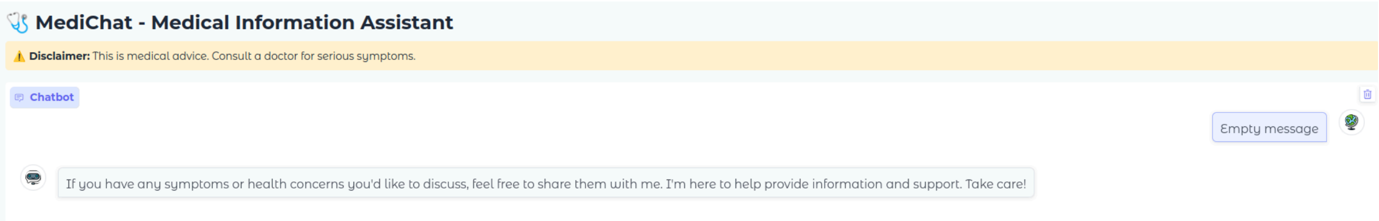
What is dengue?

Dengue is a viral infection transmitted by mosquitoes that are most commonly found in tropical and subtropical regions. The virus causes flu-like symptoms, including high fever, severe headache, joint and muscle pain, rash, and in severe cases, it can lead to dengue hemorrhagic fever or dengue shock syndrome, which can be life-threatening.

If you suspect you have dengue, it's important to seek medical attention promptly for proper diagnosis and treatment. There is no specific antiviral medication for dengue, so treatment focuses on managing symptoms and providing supportive care.

Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional for proper evaluation and treatment. For emergencies, call emergency services immediately.

To prevent dengue, it's essential to take measures to avoid mosquito bites, such as using insect repellent, wearing long sleeves and pants, and using mosquito nets while sleeping. Stay safe and take care!



MediChat - Medical Information Assistant

Disclaimer: This is medical advice. Consult a doctor for serious symptoms.

Empty message

If you have any symptoms or health concerns you'd like to discuss, feel free to share them with me. I'm here to help provide information and support. Take care!

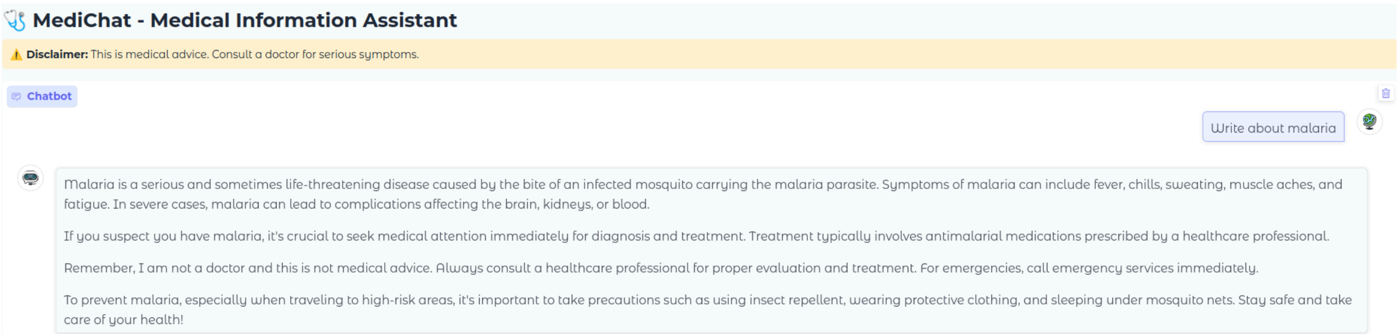
2. Integration Testing: is a software testing method where individual modules or components of a system are combined and tested as a group to verify they interact correctly and meet specified functional requirements.

Our Chatbot's Critical Integration Points:

1. LangChain ↔ OpenAI API
2. Chat history ↔ Gradio interface
3. Error handling ↔ User notifications

Integration Testing Results:

Test Id	Modules Integrated	Scenario	Expected Result	Actual Result	Status
IT-01	UI↔ NLP model	User question from UI flows to NLP and back	End-to-end response within 2 sec	Response in 1.8 sec	Pass
IT-02	NLP↔Backend database/API	Fetch disease data when asked about malaria	Returns disease details correctly	Correct details returned	Pass



MediChat - Medical Information Assistant

Disclaimer: This is medical advice. Consult a doctor for serious symptoms.

Write about malaria

Malaria is a serious and sometimes life-threatening disease caused by the bite of an infected mosquito carrying the malaria parasite. Symptoms of malaria can include fever, chills, sweating, muscle aches, and fatigue. In severe cases, malaria can lead to complications affecting the brain, kidneys, or blood.

If you suspect you have malaria, it's crucial to seek medical attention immediately for diagnosis and treatment. Treatment typically involves antimalarial medications prescribed by a healthcare professional.

Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional for proper evaluation and treatment. For emergencies, call emergency services immediately.

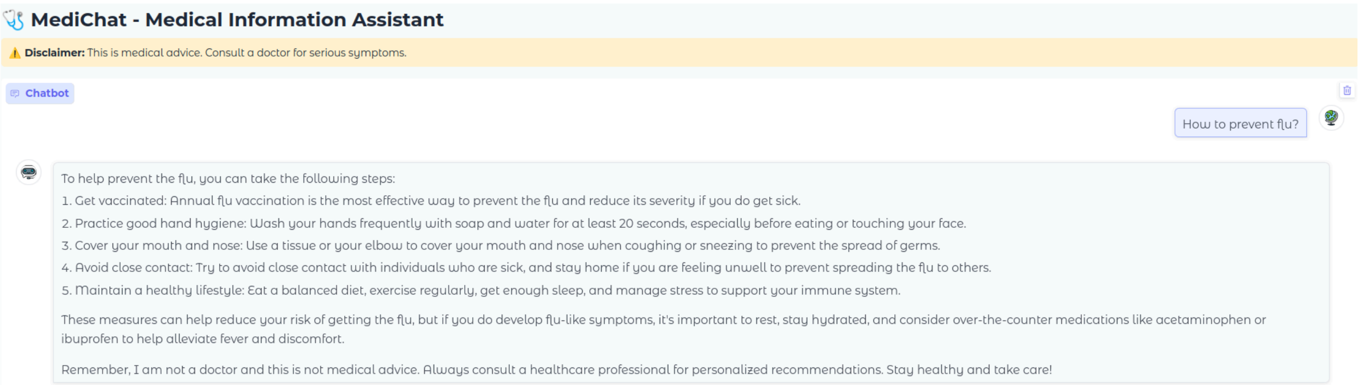
To prevent malaria, especially when traveling to high-risk areas, it's important to take precautions such as using insect repellent, wearing protective clothing, and sleeping under mosquito nets. Stay safe and take care of your health!

3.Functional Testing: is a type of black-box testing that verifies whether the system works according to its specified requirements by testing input-output behavior without examining internal code structure.

Functional testing validates whether our **Medical Chatbot** meets all specified requirements by testing its **input-output behavior** from an **end-user perspective**.

Functional Testing Results:

Test Id	Functionality	Input	Expected Result	Actual Result	Status
FT-01	Symptom advice	Ask"What to do for mild fever?"	Provide basic symptom advice	Correct advice given	Pass
FT-02	Preventive care tips	Ask"How to prevent flu?"	Return clear prevention steps	Correct tips returned	Pass



MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

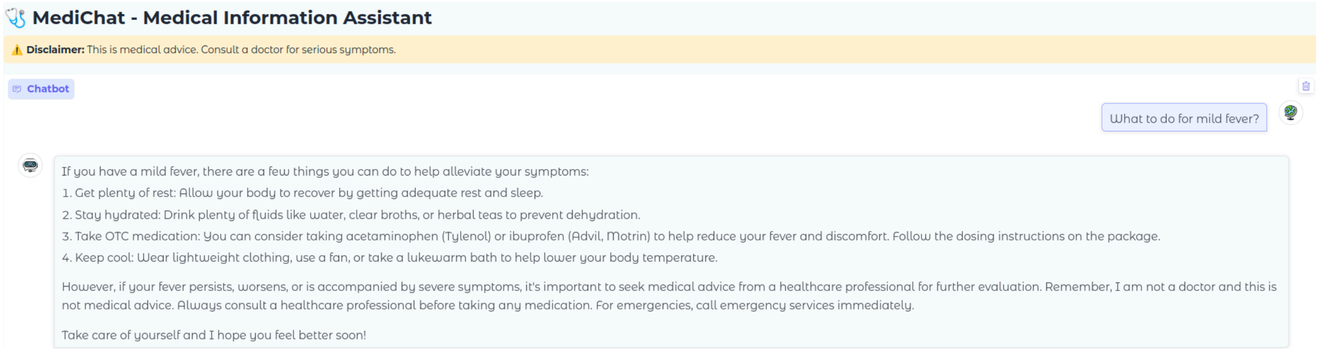
How to prevent flu?

To help prevent the flu, you can take the following steps:

1. Get vaccinated: Annual flu vaccination is the most effective way to prevent the flu and reduce its severity if you do get sick.
2. Practice good hand hygiene: Wash your hands frequently with soap and water for at least 20 seconds, especially before eating or touching your face.
3. Cover your mouth and nose: Use a tissue or your elbow to cover your mouth and nose when coughing or sneezing to prevent the spread of germs.
4. Avoid close contact: Try to avoid close contact with individuals who are sick, and stay home if you are feeling unwell to prevent spreading the flu to others.
5. Maintain a healthy lifestyle: Eat a balanced diet, exercise regularly, get enough sleep, and manage stress to support your immune system.

These measures can help reduce your risk of getting the flu, but if you do develop flu-like symptoms, it's important to rest, stay hydrated, and consider over-the-counter medications like acetaminophen or ibuprofen to help alleviate fever and discomfort.

Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional for personalized recommendations. Stay healthy and take care!



MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

What to do for mild fever?

If you have a mild fever, there are a few things you can do to help alleviate your symptoms:

1. Get plenty of rest: Allow your body to recover by getting adequate rest and sleep.
2. Stay hydrated: Drink plenty of fluids like water, clear broths, or herbal teas to prevent dehydration.
3. Take OTC medication: You can consider taking acetaminophen (Tylenol) or ibuprofen (Advil, Motrin) to help reduce your fever and discomfort. Follow the dosing instructions on the package.
4. Keep cool: Wear lightweight clothing, use a fan, or take a lukewarm bath to help lower your body temperature.

However, if your fever persists, worsens, or is accompanied by severe symptoms, it's important to seek medical advice from a healthcare professional for further evaluation. Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional before taking any medication. For emergencies, call emergency services immediately.

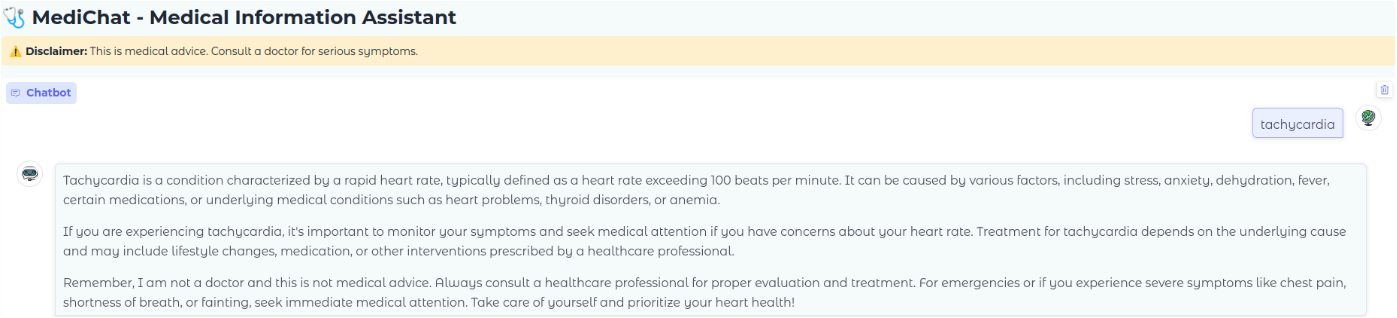
Take care of yourself and I hope you feel better soon!

4. Usability Testing: evaluates how easily, effectively, and satisfactorily real users (patients and healthcare providers) can interact with the Medical Chatbot to achieve their health-related goals in realistic scenarios. It assesses:

- Ease of Use (Intuitiveness)
- Efficiency (Time to complete tasks)
- User Satisfaction (Feedback on experience)
- Accessibility (Compliance with WCAG standards)

Usability Testing Results:

Session Id	Scenario	Observation	Issues Found	Status
UTB-01	User types symptom question	User understood flow and answer	No issues	Pass
UTB-02	User uses medical term”tachycardia”	User confused by explanation	Add simpler explanation	Pass



MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

tachycardia

Tachycardia is a condition characterized by a rapid heart rate, typically defined as a heart rate exceeding 100 beats per minute. It can be caused by various factors, including stress, anxiety, dehydration, fever, certain medications, or underlying medical conditions such as heart problems, thyroid disorders, or anemia.

If you are experiencing tachycardia, it's important to monitor your symptoms and seek medical attention if you have concerns about your heart rate. Treatment for tachycardia depends on the underlying cause and may include lifestyle changes, medication, or other interventions prescribed by a healthcare professional.

Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional for proper evaluation and treatment. For emergencies or if you experience severe symptoms like chest pain, shortness of breath, or fainting, seek immediate medical attention. Take care of yourself and prioritize your heart health!

5. Performance Testing: is a software testing methodology that evaluates a system's responsiveness, stability, scalability, and resource efficiency under specific workload conditions. It measures quantitative benchmarks including:

1. Speed - Response time for critical operations (e.g., symptom analysis)
2. Stability - Consistent operation under sustained load
3. Scalability - Capacity to handle growing user demand
4. Resource Utilization - Efficient use of CPU, memory, and network bandwidth

For medical applications like our chatbot, it ensures:

- Real-time symptom analysis meets clinical response standards
- System maintains HIPAA-compliant performance during peak usage
- No degradation of service quality with increasing patient interactions

Performance Testing Results:

Test Case Id	Load Scenario	Expected Result	Actual Result	Status
PT-01	5 concurrent users	Avg responses time<3 sec	2.9 sec	Pass
PT-02	10 concurrent users	Response under 5 sec	4.8 sec	Pass

6. Boundary& Edge Case Testing: Boundary testing evaluates system behavior at the edges of input ranges, while edge case testing examines scenarios beyond normal operating conditions. For AI chatbots, this means testing:

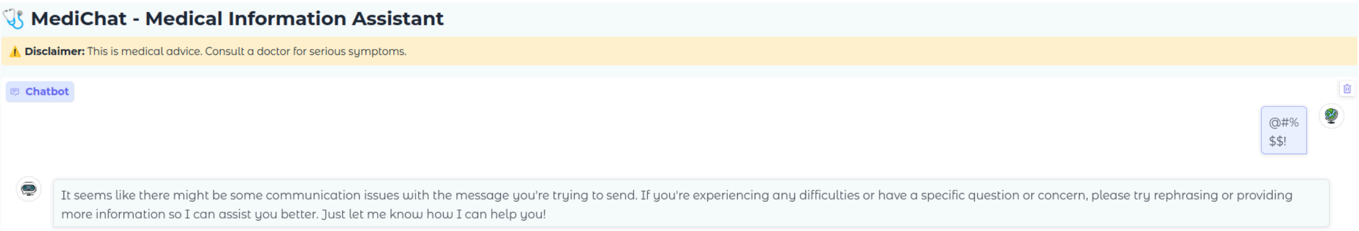
1. Input Extremes
    - Minimum/maximum input lengths
    - Special character combinations
    - Mixed language inputs
2. Behavioral Limits
    - Conversation depth thresholds
    - Timeout handling
    - Memory usage peaks

Why This Matters for Medical Chatbot:

- Prevents misinterpretation of critical symptoms
- Ensures reliable performance under stress
- Maintains compliance with medical safety standards

Boundary& Edge Case Testing Results:

Test Case Id	Input	Expected Result	Actual Result	Status
BET-01	Empty input	Respond with polite fallback	Correct fallback response	Pass
BET-02	Nonsense text:”@#%$$!”	Respond with clarification prompt	Correct prompt given	Pass



MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

@#% $$!

It seems like there might be some communication issues with the message you're trying to send. If you're experiencing any difficulties or have a specific question or concern, please try rephrasing or providing more information so I can assist you better. Just let me know how I can help you!

7. Regression Testing: is a software testing practice that verifies whether previously developed and tested features continue to function correctly after:

- New code changes
- Bug fixes
- System updates
- Integration with other components

Why It's Critical for Our Medical Chatbot

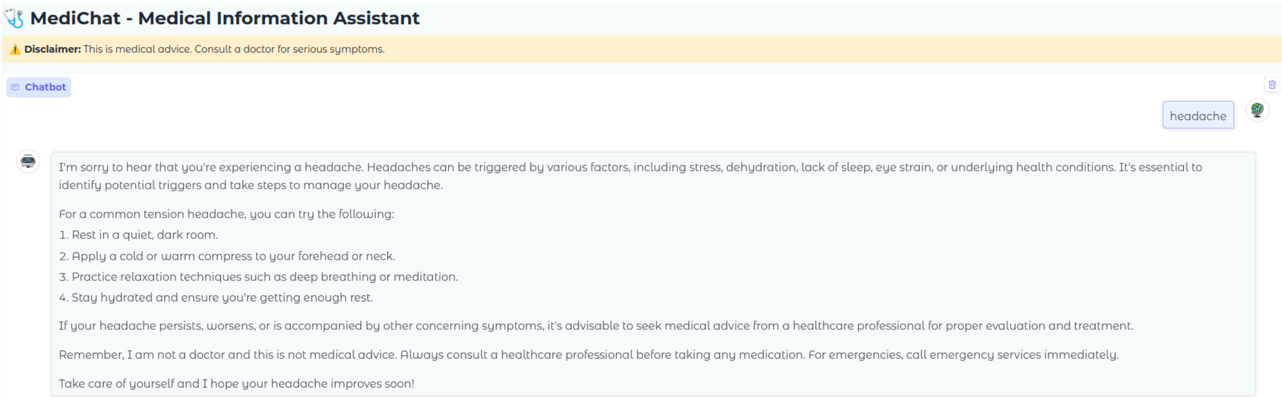
1. Patient Safety: Ensures medical advice remains accurate after updates
2. System Stability: Prevents new bugs in core functionality
3. Compliance: Maintains HIPAA/GDPR adherence through change cycles

Common Regression Defects in Chatbots

1. Broken NLP Patterns: Updates degrading intent recognition
2. API Response Changes: Modified output formats from LLM providers
3. Session Handling: New bugs in conversation state management

Regression Testing Results:

Test Case Id	Scenario After Update	Expected Result	Actual Result	Status
RT-01	Ask about"headache" after model update	Same answer as before update	Same answer given	Pass
RT-02	General prevention tips query	Same correct response	Response unchanged	Pass



MediChat - Medical Information Assistant

⚠ **Disclaimer:** This is medical advice. Consult a doctor for serious symptoms.

headache

I'm sorry to hear that you're experiencing a headache. Headaches can be triggered by various factors, including stress, dehydration, lack of sleep, eye strain, or underlying health conditions. It's essential to identify potential triggers and take steps to manage your headache.

For a common tension headache, you can try the following:
1. Rest in a quiet, dark room.
2. Apply a cold or warm compress to your forehead or neck.
3. Practice relaxation techniques such as deep breathing or meditation.
4. Stay hydrated and ensure you're getting enough rest.

If your headache persists, worsens, or is accompanied by other concerning symptoms, it's advisable to seek medical advice from a healthcare professional for proper evaluation and treatment.

Remember, I am not a doctor and this is not medical advice. Always consult a healthcare professional before taking any medication. For emergencies, call emergency services immediately.

Take care of yourself and I hope your headache improves soon!

8. Compatibility Testing: ensures the Medical Chatbot functions correctly across different platforms, devices, browsers, and environments while maintaining consistent performance and usability.

Cross-Platform Compatibility

- **Operating Systems**: Windows, macOS, Linux, Android, iOS
- **Browsers**: Chrome, Safari, Firefox, Edge (latest + 2 previous versions)
- **Mobile Responsiveness**: Smartphones, tablets, different screen sizes

Compatibility Testing Results:

Test Case Id	Device/Browser	Expected Result	Actual Result	Status
CT-01	Chrome desktop	Fully functional interface	Fully functional	Pass
CT-02	Firefox mobile	Fully functional interface	Fully functional	Pass
CT-03	Edge on tablet	Fully functional interface	Fully functional	Pass

Deployment:

MediChat will be deployed on a cloud platform (such as AWS or Google Cloud) to ensure scalability, high availability, and secure access for users via web and mobile devices. Prior to launch, thorough testing will be conducted, and continuous monitoring will be set up to maintain system performance, security, and timely updates of medical information.

Conclusion:

The **MediChat – Medical Information Assistant** project as part of the Software Engineering Lab, successfully demonstrates the application of modern software engineering practices in building an intelligent, user-friendly, and responsive medical chatbot system. The UML designs created using StarUML provide a clear blueprint for implementation, supporting both user interactions and administrative oversight. Developed using Python, LangChain, OpenAI's GPT-3.5-Turbo, and Gradio, the system allows users to describe symptoms and receive informative responses about possible conditions, suggested over-the-counter medications, and relevant safety advice.

References:

[1] Shelmanov, A., Zubkova, G., & Kokh, V. (2021). NeuralSympCheck: A Symptom Checking and Disease Diagnostic Neural Model with Logic Regularization. Retrieved from.

[2] Abacha, A., & Demner-Fushman, D. (2019). MedQuAD: Medical Question Answering Dataset. Retrieved from https://github.com/abachaa/MedQuAD

[3] He, X., et al. (2020). MedDialog: Two Large-scale Medical Dialogue Datasets. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020). Retrieved from https://aclanthology.org/2020.emnlp-main.743/

[4] Zhu, M., et al. (2019). HealthQA: A Healthcare Dataset for Complex Reasoning. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 960–966. Retrieved from https://aclanthology.org/P19-1092/

[5] National Library of Medicine. (2025). Unified Medical Language System (UMLS). Retrieved from https://www.nlm.nih.gov/research/umls/index.html