# MediChat Documentation

## Table of Contents

1. Introduction

2. Project Overview

3. Objectives

4. Scope

5. System Model

6. System Architecture

7. Use Case Diagrams

8. Entity-Relationship (ER) Diagram

9. Database Schema

10. Functional Requirements

11. Non-Functional Requirements

12. Features & Modules

13. Technology Stack

14. User Interface Design

15. Security Measures

16. API Documentation

17. Deployment Strategy

18. Testing Plan

19. Future Enhancements

20. Conclusion

## 1. Introduction

MediChat is a web-based application aimed at providing users with an online platform for health-related queries, emergency assistance, and medical support. The platform integrates various functionalities such as real-time communication, health-related information, and an emergency contact system.  
   
 With the increasing demand for telemedicine and online healthcare services, MediChat serves as a comprehensive solution for connecting users with medical professionals while ensuring user privacy and security.

## 2. Project Overview

MediChat is designed to bridge the gap between medical professionals and patients by offering an easy-to-use web application. The project incorporates interactive features to ensure accessibility and efficiency in healthcare communication.  
   
 The system ensures a seamless experience for both patients and healthcare providers by integrating real-time messaging, AI-powered assistance, and emergency response functionalities.

## 3. Objectives

The main objectives of MediChat are:  
 - Provide an intuitive and responsive platform for health-related discussions.  
 - Offer emergency assistance with quick response times.  
 - Ensure user privacy and data security by using encryption methods.  
 - Implement AI-driven chat support for common health queries.  
 - Improve accessibility by designing a user-friendly interface.

## 4. Scope

MediChat is designed to serve a broad audience, including patients, doctors, and healthcare institutions.   
 - Users can access general health information and consult professionals.  
 - Emergency contact feature for quick medical assistance.  
 - Secure communication channels for users to interact with doctors confidentially.  
 - Future enhancements include AI-driven diagnosis suggestions and integration with healthcare providers.

## 5. System Model

MediChat follows the Client-Server Model, where users interact with the application through a web interface, and all data processing is handled on the server. The system consists of:  
 - Client Side: Web-based user interface.  
 - Server Side: Processes user requests and facilitates messaging.  
 - Database: Secure storage for user data and messages.

## 6. System Architecture

MediChat is built using a three-tier architecture:  
 1. Presentation Layer: Frontend with HTML, CSS, JavaScript, Bootstrap.  
 2. Business Logic Layer: Backend using Node.js and Express.  
 3. Data Layer: MongoDB for data storage and management.  
   
 The architecture ensures scalability, security, and efficiency by segregating different functionalities into modular components.