HackLLM

A Premier LLM Hackathon under ESYA'25

Team Name: TeamX

Challenge Accepted: Medical Summarization Challenge

Project Title: ClearMed

Team Details: Rishabh Gupta (Team Leader)
Sachin Gupta
Rishabh Dubey
Ranjan Yadav

From College: ABES Engineering College, Ghaziabad

Problem statement:

- Medical documents are often long and complex.
- Doctors need clinically relevant summaries, while patients need easy-to-understand summaries.
- Existing tools rely on LLMs with APIs, but here we must build a custom lightweight summarization model.

Solution:

ClearMed (**AI System**) is an AI-powered solution that can read long medical reports or Q&A documents and generate two types of summaries:

- **Clinician Focused Mode →** Gives a summary with correct medical terms and detailed clinical information.
- **Patient Friendly Mode →** Gives a summary in simple, clear language without medical jargon.

The **AI** makes sure:

- Every line in the summary is linked to the original report, so users can verify the source.
- It checks for risky or sensitive statements (like dosages or absolute instructions) and adds a safety disclaimer.
- In Patient Mode, it automatically translates complex medical terms into simple words while keeping the meaning correct.

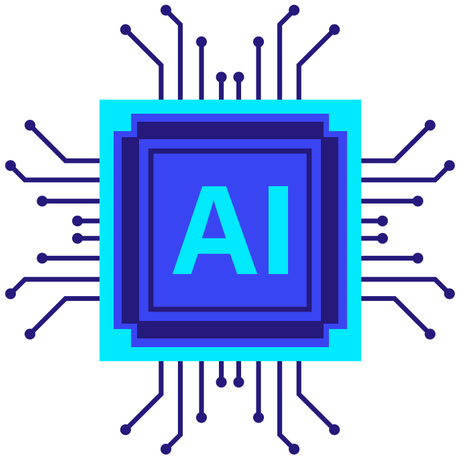
Why Our AI is Unique

Simple language converter: Complex medical terms are automatically replaced with easy-to-understand words for patients.

Two modes (Doctor & Patient): The AI has two separate outputs – one keeps full medical details for doctors, the other simplifies terms for patients.

Runs locally: The model is small and fast, so it works on a normal laptop without cloud or external APIs.

Trustworthy summaries: Every line in the summary is linked back to the original report so nothing is made up.



Safe to use: A built-in checker scans for risky advice (like dosages) and adds warnings/disclaimers

System Architecture

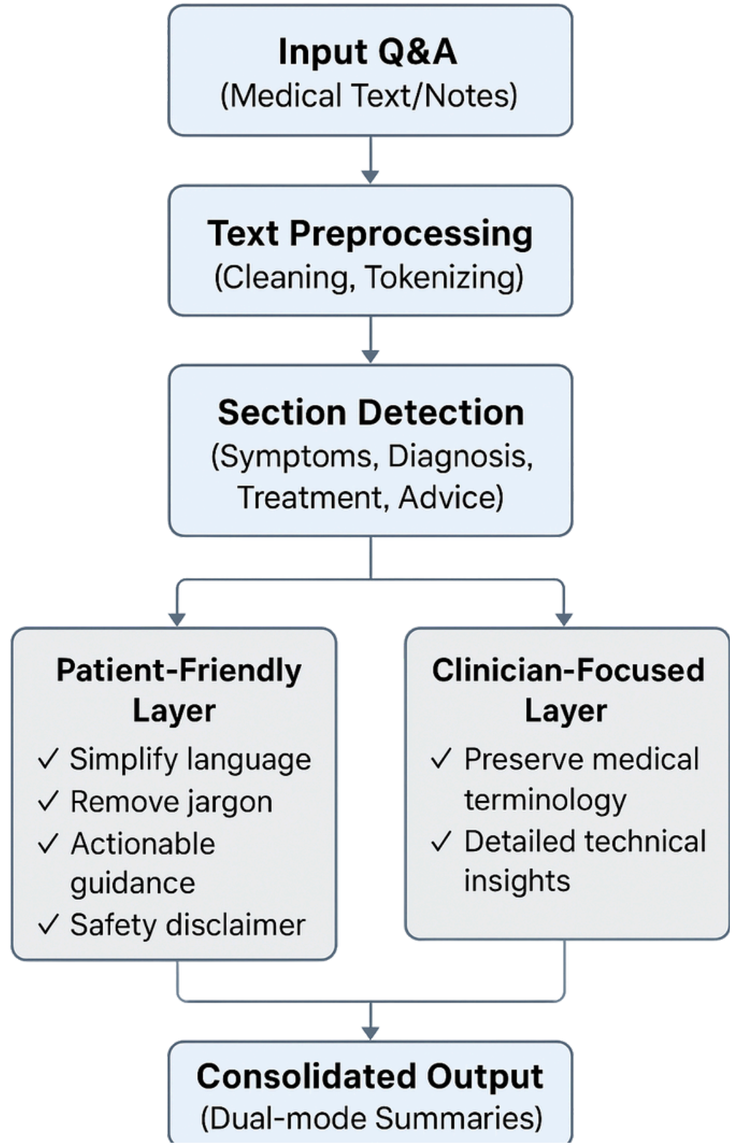
Frontend & Presentation
- Responsive medical-themed web UI (HTML/CSS/JS)
- Real-time search, query input & result display

API & Gateway
- FastAPI with async REST endpoints
- CORS middleware, validation, security & rate limiting

Business Logic
- Medical LLM controller + BM25 search engine
- Doctor/Patient summary generator & response quality check

AI Model & Management
- Primary: Llama 3.2 3B (LoRA fine-tuned)
- Fallback: GPT-2 base model with health monitoring

Data & Infrastructure
- Medical datasets + knowledge base (28k+ examples)
- Runs on Python 3.13, PyTorch 2.8, Transformers 4.56
- Docker-ready, production deployment with <2s response



Technical blueprint

Frontend
HTML5 + CSS3 - Modern responsive design
Vanilla JavaScript - No framework dependency
Medical-themed UI - Professional interface

Backend
FastAPI - Modern Python web framework
Uvicorn - ASGI server
Port 8000 - RESTful API endpoints

Data & Training
4 Medical Datasets - 28,562 training examples
MedQuad + Custom Data - Medical Q&A corpus
LoRA Training - Efficient domain adaptation
Google Colab - Training environment

AI/ML Core
PyTorch 2.8.0 - Deep learning framework
Transformers 4.56.0 - Pre-trained model library
PEFT (LoRA) - Parameter-efficient fine-tuning
Model: Llama 3.2 3B + Medical fine-tuning

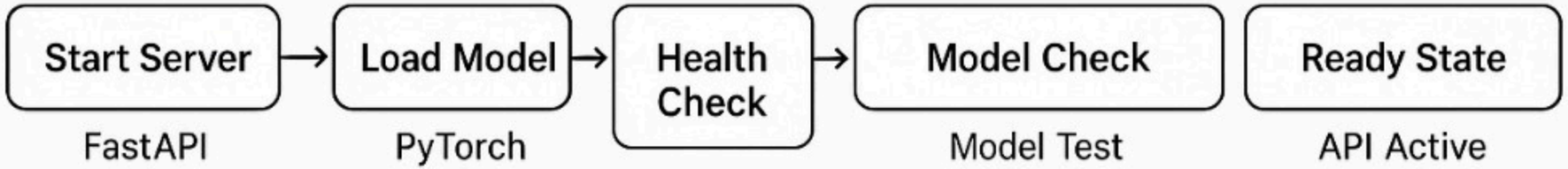
Architecture
3-Tier System: Frontend → API → AI Model
124M Parameters - Optimized model size
Cross-platform - Windows, macOS, Linux
Production-ready - Enterprise-grade system

Key Features
Hybrid AI + Search - AI responses + dataset search
Real-time Processing - Instant medical Q&A
Fallback System - Automatic model switching
Medical Specialization - Domain-specific knowledge

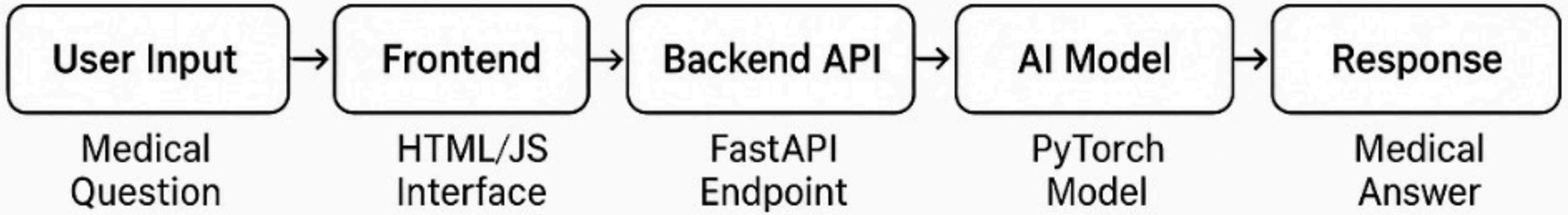
Optimization
- Model distillation & quantization for efficiency

Work Flow

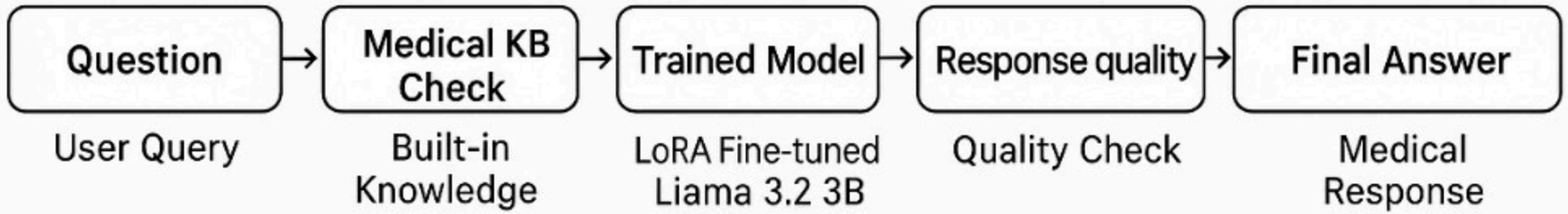
1. System Initialization



2. User Interaction Flow



3. AI Response Generation



4. Hybrid Search Process