HEALTH AI-Performance Testing:

# 1. Introduction

Performance testing is carried out to evaluate the speed, scalability, and stability of the Medical AI Assistant under different workloads. Since the project is based on a Large Language Model (LLM) and a Gradio interface, the main focus is on response time, resource utilization, and scalability when multiple requests are processed.

# 2. Objectives of Performance Testing

- Measure response time of disease prediction and treatment plan generation.  
- Analyze the impact of hardware (CPU vs GPU) on system performance.  
- Check memory usage during inference.  
- Ensure the system handles multiple requests without crashing.  
- Verify that disclaimers are always included in responses regardless of load.

# 3. Performance Metrics

- Response Time (Latency): Time taken by the system to generate results (target: <10s on GPU, <20s on CPU).  
- Throughput: Number of requests the system can handle per second.  
- Memory Usage: RAM and VRAM consumption while loading and running the model.  
- Scalability: Ability to handle multiple simultaneous users (Gradio queueing).  
- Error Rate: Percentage of failed or incomplete responses.

# 4. Test Environment

Hardware Setup:  
- CPU: Intel i7 (8-core)  
- RAM: 16 GB  
- GPU: NVIDIA RTX 3060 (6GB VRAM)  
  
Software Setup:  
- Python 3.10  
- Transformers, Torch, Gradio  
- OS: Windows 11 / Ubuntu Linux

# 5. Test Scenarios

5.1 Disease Prediction Test  
- Input: 'fever, cough, fatigue, sore throat'  
- Expected Response Time: ≤10s (GPU), ≤20s (CPU)  
- Observed: Average 8s on GPU, 18s on CPU.  
  
5.2 Treatment Plan Generation Test  
- Input: 'Condition: Diabetes, Age: 45, Gender: Male, Medical History: hypertension'  
- Expected Response Time: ≤12s (GPU), ≤25s (CPU)  
- Observed: Average 10s on GPU, 22s on CPU.  
  
5.3 Concurrent User Test  
- 10 users input queries simultaneously.  
- Expected: No crashes, responses queued and delivered.  
- Observed: Gradio queue handled requests sequentially, slight delay (~2s extra).  
  
5.4 Memory Usage Test  
- Model load requires ~3.2 GB RAM (CPU) or ~2 GB VRAM (GPU).  
- During inference, memory consumption stable (within limits).

# 6. Test Results Summary

Test Case | Expected Result | Actual Result | Status  
------------------------- | ---------------------------------- | ---------------------- | -------  
Disease Prediction | Response <10s (GPU), <20s (CPU) | 8s / 18s | Pass  
Treatment Plan Generation | Response <12s (GPU), <25s (CPU) | 10s / 22s | Pass  
Concurrent Users (10) | No crashes, queue working | Stable queue | Pass  
Memory Usage | Within available resources | 3.2GB RAM / 2GB VRAM | Pass  
Error Handling | No crash on empty input | Blank response + disclaimer | Pass

# 7. Observations

- GPU significantly improves performance (~2x faster).  
- Gradio’s queueing system ensures multiple users can be handled but adds delay.  
- System is stable under normal workloads.  
- Larger inputs (>500 tokens) slightly increase response time.

# 8. Conclusion

The Medical AI Assistant performs efficiently under expected workloads.  
- Response times are acceptable (8–10s GPU, 18–22s CPU).  
- The system is scalable and reliable for small-to-medium use cases.  
- For larger deployments, cloud-based GPUs or model optimization (quantization) are recommended.