# ****Performance Testing****

Performance testing was an essential part of validating the Educational AI Assistant to ensure that it functions reliably across different environments and workloads. While functional testing confirmed that the application performed the correct tasks, performance testing focused on measuring the **speed, efficiency, scalability, and stability** of the system. Because the assistant is powered by a large language model, its performance depends not only on the correctness of the code but also on hardware capabilities such as CPU, GPU, and memory availability.

The primary goal of performance testing was to assess how quickly the system could process input prompts and return outputs under varying conditions. On machines equipped with GPUs, the model was expected to respond faster compared to CPU-only devices. Therefore, tests were designed to compare response times across hardware setups. Another key metric was **resource utilization**, such as memory consumption and CPU/GPU load, to ensure that the application could handle requests without crashing or exhausting system resources.

Performance testing also evaluated the system’s **scalability**. Since the Gradio interface allows multiple users to interact with the assistant via a shared link, it was important to test whether the system could handle concurrent requests. For example, if two or more users submitted queries at the same time, the system had to maintain independent sessions without interference or degradation in performance.

The testing process included various scenarios, such as short prompts, long prompts, repeated queries, and simultaneous user interactions. This ensured that the assistant was tested not only under ideal conditions but also in more demanding real-world situations. The results showed that the system performed well overall, though with some predictable limitations when operating on CPU-only environments or handling very long prompts.

### ****Performance Test Cases****

| **Test Case ID** | **Scenario** | **Input** | **Expected Outcome** | **Actual Outcome** | **Status** |
| --- | --- | --- | --- | --- | --- |
| PT01 | Response time on CPU | Prompt: “Explain gravity” | Response generated within 15–20 seconds | Generated in ~18 seconds | Pass |
| PT02 | Response time on GPU | Prompt: “Explain gravity” | Response generated within 3–5 seconds | Generated in ~4 seconds | Pass |
| PT03 | Long input handling | 500-word prompt on AI ethics | Output truncated but coherent | Output truncated after 480 tokens, coherent | Pass |
| PT04 | Concurrent user requests | Two users: “Physics” & “Chemistry” | Independent sessions, no data overlap | Both outputs generated separately | Pass |
| PT05 | Memory usage under stress | 10 repeated queries in 1 minute | No crash, stable resource consumption | No crash, memory stable | Pass |
| PT06 | Large quiz generation | Topic: “World War II” | 5 questions + answers within 10 seconds (GPU) | Delivered in ~9 seconds | Pass |

### ****Observations and Findings****

* On **GPU-enabled systems**, response times were very fast (3–5 seconds on average), making the assistant suitable for real-time classroom use.
* On **CPU-only systems**, the assistant remained functional but slower, with response times ranging from 15–30 seconds for explanations and quizzes.
* The system successfully handled **long prompts**, though with expected truncation due to maximum token limits. The outputs remained coherent and relevant.
* **Concurrent user testing** confirmed that multiple users could interact with the assistant simultaneously without interference. Each session produced independent results.
* The application showed **stable resource consumption**, with no crashes or memory leaks even under repeated queries.

### ****Conclusion****

The performance testing phase confirmed that the Educational AI Assistant is robust and adaptable across different environments. While CPU-based performance is slower, it is still usable, and GPU-based execution delivers near real-time interaction. The system handles long inputs gracefully and can support multiple users at once, making it scalable for both personal and classroom use. Overall, the assistant’s performance is strong enough to support its intended educational applications, with room for further optimization in future enhancements.