



Performance Testing: Enhancing Performance for University Attendance System

Background

The university's application required rigorous performance and scalability validation for its Attendance Registration System to ensure seamless integration with the Student Management System. This testing aimed to confirm the application could handle high volumes of simultaneous user access to APIs without compromising responsiveness. Additionally, integration and data transfer between the university's systems and the Student Records System were thoroughly tested to ensure reliability and efficiency.

Key Aspects of API Performance Testing

- ▶ **Latency:** Measured the time taken for requests to travel between the client and server.
- ▶ **Throughput:** Assessed the number of transactions processed within a specific timeframe.
- ▶ **Scalability:** Evaluated system performance under increasing user loads.
- ▶ **Response Time:** Monitored the speed of API responses.
- ▶ **Error Rate:** Tracked the percentage of failed requests.
- ▶ **Resource Utilization:** Analyzed CPU and memory usage during high-traffic scenarios failed requests.

Solution

- ▶ **Collaboration with University Teams:** Gathered detailed API information, including endpoints, data types, formats, and request frequencies for both GET and POST requests.
- ▶ **Detailed Test Plan Creation:** Documented testing scope, expected data volumes, dependencies, and test profiles.
- ▶ **Walkthrough and Feedback:** Conducted a review of the test plan with stakeholders, made necessary adjustments, and ensured alignment before test preparation.
- ▶ **Scenario Execution:** Tested combinations of API requests at varying traffic levels (1,000–3,000 requests per hour) to

Results

- ▶ The system successfully handled up to 2,000 API requests per hour with stable CPU and memory usage.
- ▶ Beyond this limit, issues such as connection timeouts and increased CPU load were identified. Recommendations included capping traffic at 42 requests per second for optimal performance.
- ▶ Another API maintained functionality with up to 1,200 requests per hour. It was advised to limit these requests to 20 per second to prevent

Conclusion

The performance tests provided valuable insights into the system's capacity and reliability, ensuring the Attendance Registration System could handle real-world demands. The iterative testing and optimization process enhanced system robustness and allowed the reuse of test scripts for ongoing improvements. Upon project completion, all test assets were delivered to the university to facilitate future performance validations.