

Networks Lab 3

1 Students

- Moustafa Esam El-Sayed Amer - 21011364
- Ahmed Mostafa El-morsi Amer - 21010189

2 Bonus

2.1 Testing via Real Browser

- Text File

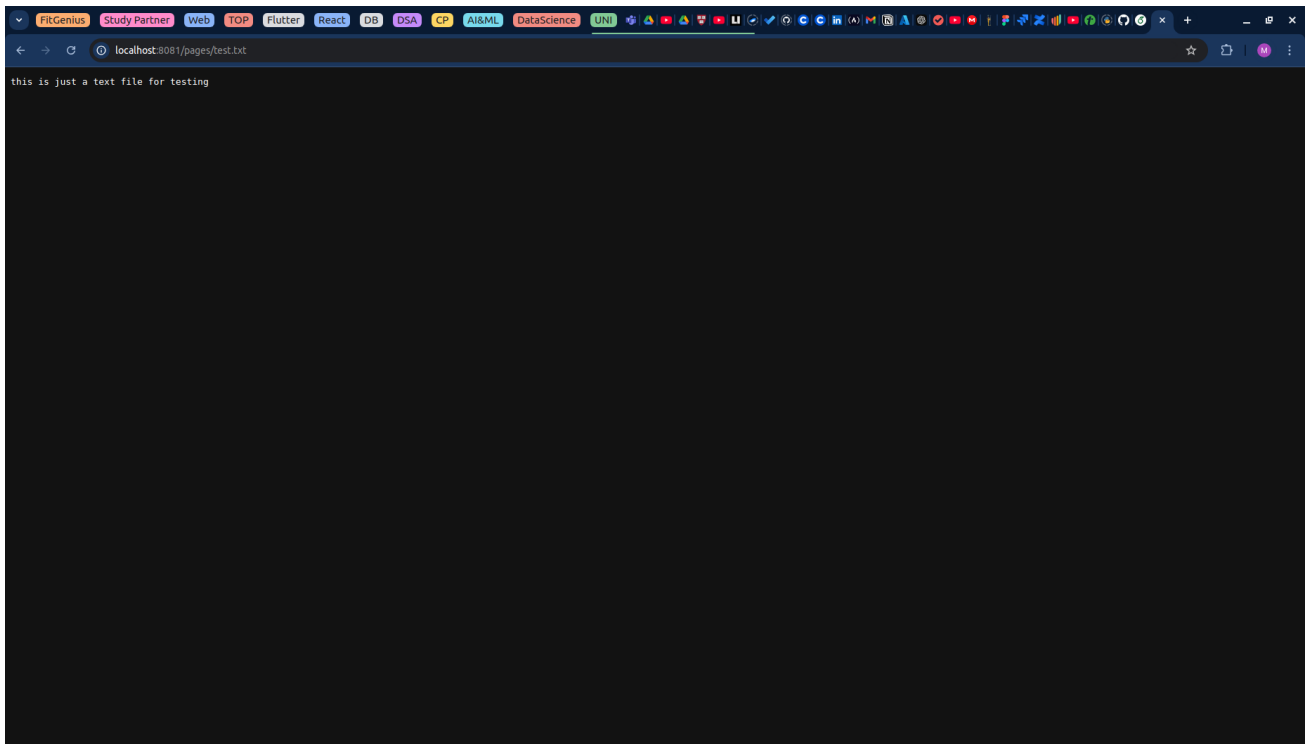


Figure 1: Testing Real Web Server with text data

- HTML Document

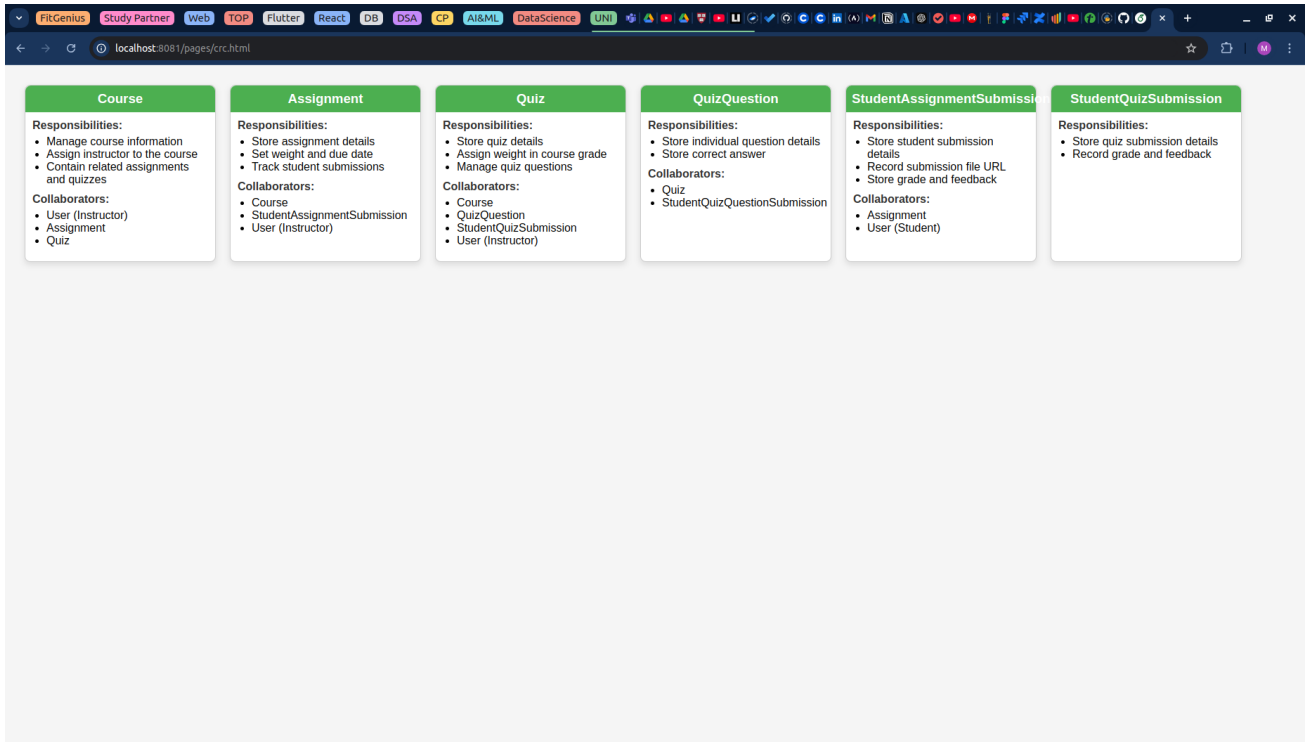


Figure 2: Testing Real Web Server With HTML

- Image

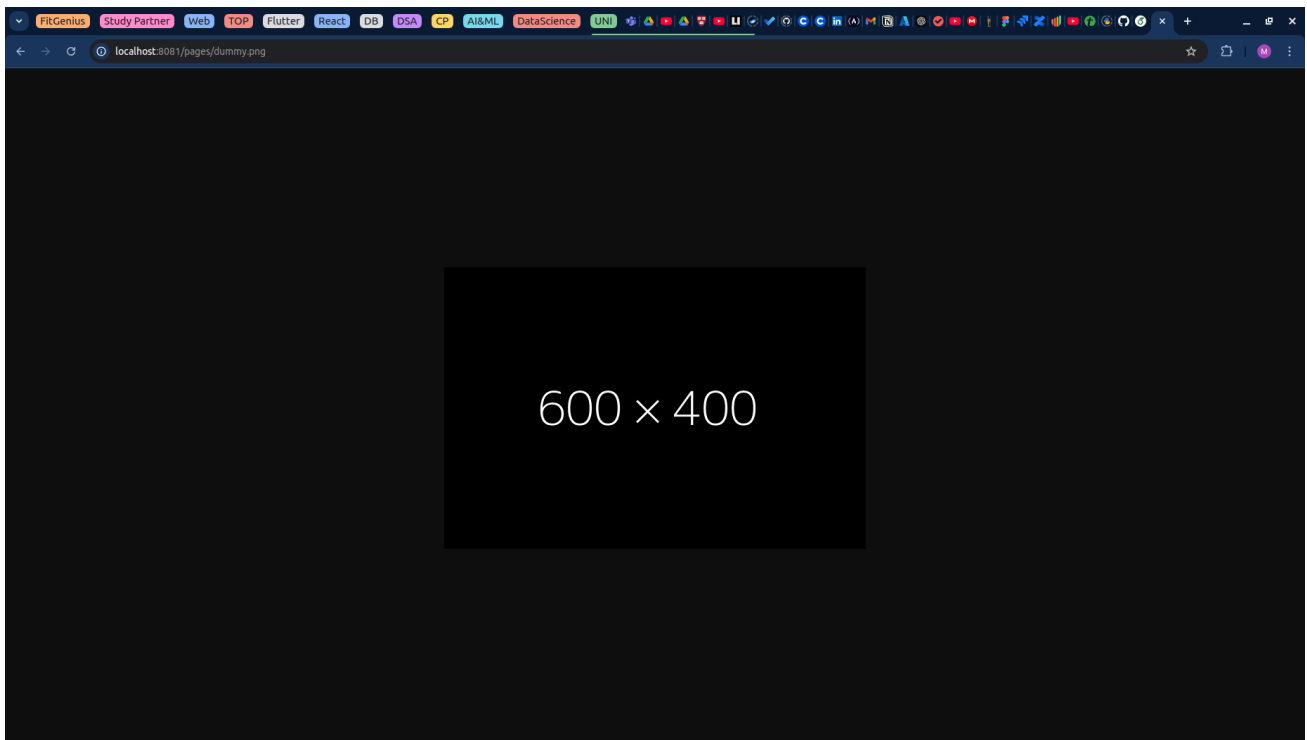


Figure 3: Testing Real Web Server With Image

2.2 Evaluation

The Dynamic Timeout Approach: The dynamic timeout setting adjusts the timeout for each connection based on server load. At low client loads, the server processes requests quickly, with low response times and high

throughput. However, as client load increases response times increase due to congestion, throughput peaks and then declines as the server reaches its capacity limit. at higher numbers of clients indicate that while the dynamic timeout can manage moderate loads effectively, it cannot prevent performance degradation under heavy loads.

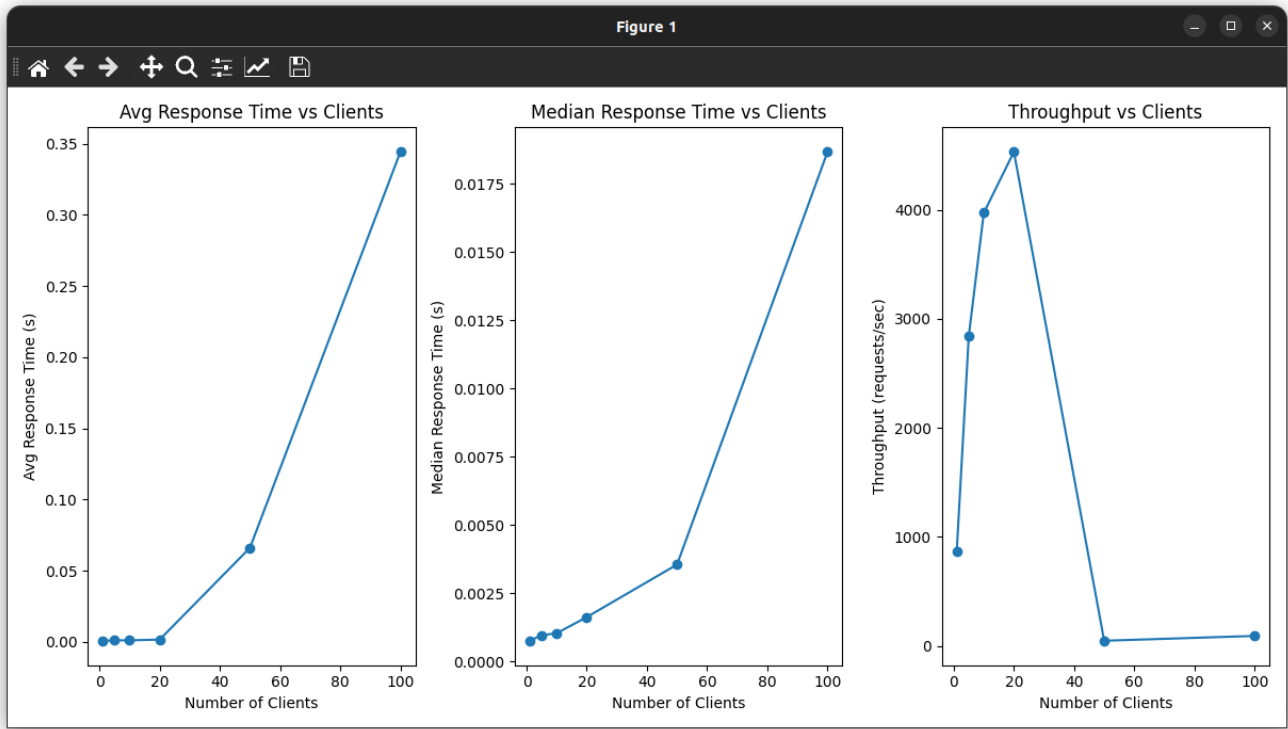


Figure 4: Evaluation for dynamic timeout

This chart shows three plots—average response time, median response time, and throughput—against the number of clients for a server using a dynamic timeout setting.

- **Average response Time vs Clients:** This plot shows the average response time as the number of clients increases. At lower numbers of clients, the average response time perceives lower values, suggesting the server can handle the load without delays. As the client count increases for a certain bound (40), response times increase significantly, and by 100 clients, the response time rises largely to around 0.35 seconds. This means that the server experiences more delay under a heavy load, this is due to lack of resources and increased queuing time for requests.
- **Median Response Time vs Clients:** The median response time also rises as the number of clients increases, but the values remain much smaller than the average response time. This suggests that some requests take significantly longer than the median time, causing the average to be higher.
- **Throughput vs Clients:** Throughput is the rate of requests handled per second by the server. The throughput initially rises with an increase in clients, peaking at around 40 clients, reaching over 4000 requests per second. After this point, throughput drops sharply, which indicates that the server's capacity is exceeded at high client counts, leading to poor performance indicated by low throughput. As the number of clients increases above 100, throughput reaches much lower values, indicating the server is likely overloaded and unable to process requests efficiently.