

**Department of Engineering Technology**

SET-222

Software Operations & Maintenance

Experiment # 11

**Experiment Title**

**High Availability & Scalability (Load balancing, horizontal vs. vertical scaling, auto-scaling)**

**Assessment of CLO(s): 03**

**Performed on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student Name:** |  | | |
| **Roll No.** |  | **Group** |  |
| **Semester** |  | **Session** |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S. No.** | **Perf. Level**  **Criteria** | **Excellent**  **(2.5)** | **Good**  **(2)** | **Satisfactory**  **(1.5)** | **Needs Improvement**  **(0 ~ 1)** | **Marks Obtained** |
| **1** | Project Execution & Implementation | Fully functional, optimized, and well-structured. | Minor errors, mostly functional. | Some errors, requires guidance. | Major errors, non-functional, or not Performed. |  |
| **2** | Results & Debugging  Or Troubleshooting | Accurate results with effective debugging  Or Troubleshooting. | Mostly correct, some debugging Or Troubleshooting needed. | Partial results, minimal debugging  Or Troubleshooting. | Incorrect results, no debugging Or Troubleshooting, or not attempted. |  |
| **3** | Problem-Solving & Adaptability  (VIVA) | Creative approach, efficiently solves challenges. | Adapts well, minor struggles. | Some adaptability, needs guidance. | Lacks innovation or no innovation, unable to solve problems. |  |
| **4** | Report Quality & Documentation | Clear, structured, with detailed visuals. | Mostly clear, minor gaps. | Some clarity issues, missing details. | Poorly structured, lacks clarity, or not submitted. |  |
| **Total Marks Obtained Out of 10** | | | | | |  |

**Experiment evaluated by**

|  |  |  |  |
| --- | --- | --- | --- |
| **Instructor’s Name** | **Ms. Shagufta Aftab** | | |
| **Date** |  | **Signature** |  |

## Copyright © Department of Engineering & Technology – UIT University Karachi

**Objective:**

To demonstrate the use of load balancing, horizontal and vertical scaling, and auto-scaling for improving the availability and scalability of IT services.

**Tools and Platforms Required:**

* Virtual machines (Local/Cloud)
* Web server software (Apache, Nginx, IIS)
* Load balancer (HAProxy, Nginx, AWS ELB)
* Cloud environment (AWS, Azure, or Google Cloud – optional)

**Theory Concepts:**

**Load Balancing**

* Distributes traffic across multiple servers to avoid overloading a single server.
* Improves performance, availability, and fault tolerance.

**Horizontal vs. Vertical Scaling**

* **Horizontal Scaling:** Adding more servers to handle load.
* **Vertical Scaling:** Increasing the resources (CPU/RAM) of a single server.

**Auto-Scaling**

* Automatically adjusts resources based on traffic/load conditions.
* Helps in optimizing cost and maintaining performance.

**Lab Tasks:**

**Task 1: Setup Two Web Servers**

* Deploy and configure two instances running a web server (e.g., Apache on VM1 and VM2).
* Modify homepage to show the server name (for testing load distribution).

**Task 2: Configure Load Balancer**

* Install and configure HAProxy (or Nginx) on a third instance.
* Forward requests to VM1 and VM2 in a round-robin fashion.
* Test by refreshing browser to confirm alternating server responses.

**Task 3: Simulate Horizontal and Vertical Scaling**

* Horizontal: Add a third server and configure the load balancer to include it.
* Vertical: Increase CPU/RAM of one web server and observe performance impact.

**Task 4: Auto-Scaling (Optional - Cloud)**

* Use AWS/Azure to configure an Auto Scaling Group.
* Define policies to launch instances based on CPU usage.

**Expected Outcomes:**

* Understand and implement a basic load balancer setup.
* Identify the difference between scaling methods.
* Observe benefits of auto-scaling in cloud environments.

**Assessment Questions:**

1. What is the role of a load balancer in high availability?

Ans: A load balancer shares incoming traffic across multiple servers so no one server gets overloaded. For example, if 100 users visit a website, it spreads them over 3 servers. This keeps the system fast and always available.

1. Compare horizontal and vertical scaling with examples.

Ans: Horizontal scaling means adding more machines (like adding more delivery bikes). Vertical scaling means upgrading one machine (like making a bike faster). Horizontal is better for big websites needing more power

1. What are the advantages of using auto-scaling in cloud platforms?

Ans: Auto-scaling adds or removes servers automatically based on traffic. It saves money and keeps performance smooth, like getting more staff during busy hours and fewer when it's quiet.

1. How does scaling improve system performance?

Ans: Scaling handles more users without slowing down. It adds resources when needed so the app runs fast even during high traffic.

1. List tools used for implementing high availability solutions.

Ans: Common tools: AWS ELB, Kubernetes, HAProxy, Nginx, and Azure Load Balancer. These help keep apps running even if one part fails.