# **Assignment 3**

# **Cloud Resource Manager with Load Balancing**

**Background:**

Cloud computing has become an integral part of modern computing infrastructure, offering scalable and on-demand access to computational resources. Efficiently managing these resources is crucial for ensuring optimal performance and cost-effectiveness. This assignment focuses on designing a cloud resource manager that incorporates load balancing strategies to distribute jobs across virtual machines (VMs) in a cloud environment.

By completing this assignment, students will:

* Gain practical experience in designing and implementing a cloud resource manager, similar to those used in real-world cloud services.
* Understand the challenges and considerations involved in load balancing within a cloud environment.
* Develop skills in dynamic resource allocation, fault tolerance, and scalability.

**Input, Output and Sample workflow:**

**Initial parameters:**

Initial configuration of virtual machines, including their computing power and max jobs they can take at a time. Max jobs will impact both size of the heap and the number of jobs after which a new VM has to be spawned.

**Inputs:**

Stream of incoming jobs with varying computational requirements and priority levels.

**Outputs:**

Allocation of jobs to virtual machines, considering load balancing and job priority.

Dynamic adjustments to resource allocation based on changing workload.

**Sample Workflow:**

* The system starts with an initial configuration of VMs and awaits incoming job requests.
* As jobs arrive, the resource manager allocates them to VMs based on their computational requirements and priority levels.
* The system continuously monitors the load on each VM and dynamically adjusts resource allocation to maintain load balancing. New VMs can be spawned as needed. Research load balancing algorithms used by real cloud service providers such as AWS and implement simple versions of them.
* In the event of a VM failure, the resource manager redistributes the affected jobs to available VMs.
* The simulation tracks the completion times of jobs and the current load on each VM.
* **REQUIRED FEATURE:** In the end, show a summary of the jobs completed, time taken, the number of VM’s used and the time taken by each VM.

The core of your assignment should be a heap data structure to manage priorities. You may also use other data structures you have studied in this course. Choosing an appropriate data structure for the appropriate task carries marks – you will be asked to showcase which data structures you have used and to justify your choices in the demo.

The assignment is very open-ended and you can choose the implementation details best suited, and show-case the workflow as you see fit. The point is to implement the job allocation and dynamic load balancing with heaps and other data structures you have studied and be able to justify the choices. A key requirement is that the output of the simulation should thoroughly showcase the internal dynamics. Better outputs will gain more (and even bonus) marks.

**Marking Rubric:**

Allocation Functionality (30%):

Implementation of the cloud resource manager with proper job allocation using heaps. Correct handling of job priorities.

Efficiency and Scalability (20%):

Load balancing algorithms chosen appropriately and implemented correctly.

Fault Tolerance (15%):

Effective handling of VM failures (system recovery and continuity after a VM failure).

Output and workflow (20%):

Clarity of output and a clean, clear workflow for demonstration purposes that showcases all features.

Presentation and Documentation (15%):

Clarity and organization of code.

Detailed documentation explaining and justifying ***all*** the design choices and algorithms used.

**Demonstration:**

Students will present their assignment as a live demo, showcasing the following:

Running the simulation with sample inputs.

* Demonstrating the dynamic allocation of jobs to VMs and load balancing adjustments.
* Simulating a VM failure and showcasing the fault tolerance mechanism.
* Presenting performance analysis results, discussing efficiency, and scalability considerations.
* Answering questions regarding their design choices and the functionality of the cloud resource manager.

**Instructions**:

* This assignment should be done in pairs.
* Please submit your code files as well a report outlining the design choices and a user manual for your code describing the workflow. Show sample input and output screenshots. Zip all files into a single folder and name it <Assignment3\_groupMember1RollNo\_groupMember2RollNo.zip>.
* Late submissions will not be accepted.