For testing assume there are 4 machines in cluster

A, B, C and D

Say each machine has 2GB RAM

and 5GB of free hard disk space.

Assume each machine has single cpu (For simplicity)

To create a cluster - cluster-1 call below api:

This will create a cluster using 4 machines.

So in total we will have 8GB of RAM, 4 CPU

and 20GB of hard disk space.

API:

POST: /cluster

{

"cluster\_name": "cluster-1",

"initial\_capacity": {

"cpu": "4",

"ram": 8,

"disk": 20

}

}

Response of above api tells names of machines which are part of cluster.

This can be IP also. AIM is that we should be able to recognize which machines are

part of the cluster.

Response:

[

"A", "B", "C", "D"

]

**Scenario 1: Use under utilized machine first.**

Create below tasks in order:

API:

Request: Create a task - taska

{

"cpu": 10,

"ram": 1,

"disk": 2,

"binary\_path": "/home/ubuntu/bin/taska.bin",

"callback": "[https://taska.callback](https://taska.callback/)"

}

Response: Name of the machine allocated for the task.

{

"machine": "A"

}

Request: Create a task - task “b”

{

"cpu": 10,

"ram": 2,

"disk": 2,

"binary\_path": "/home/ubuntu/bin/taskb.bin",

"callback": "[https://taskb.callback](https://taskb.callback/)"

}

Response:

{

"machine": "B"

}

Request: Create a task - task “c”

{

"cpu": 10,

"ram": 1,

"disk": 2,

"binary\_path": "/home/ubuntu/bin/taskc.bin",

"callback": "[https://taskc.callback](https://taskc.callback/)"

}

Response:

{

"machine": "A"

}

Test: Only two machines should be used to run above 3 tasks

Also task “c” should be running on machine running task “a”

**Scenario-2: Unutilized machines should shutdown themselves**

Wait for sometime and let taska, task “b” and task “c” complete.

Test: After task completion all the machines in cluster should shutdown.

**Scenario-3: A new machine should spin up if no running machine available to cater the needs of a new task.**

Request: Create a task - task “a”

{

"cpu": 10,

"ram": 1,

"disk": 2,

"binary\_path": "/home/ubuntu/bin/taska.bin",

"callback": "[https://taska.callback](https://taska.callback/)"

}

Response:

{

"machine": "A"

}

Test: As no machine is running after Scenario-2 , a new machine should boot up after submission of task “a”

**Scenario-4: If all the machines are used or cluster cannot run the task, then submission of new task should give some error:**

Request: Create a task - taska . Our cluster doesn't have a machine which has 5GB ram.

{

"cpu": 10,

"ram": 5,

"disk": 2,

"binary\_path": "/home/ubuntu/bin/taska.bin",

"callback": "[https://taska.callback](https://taska.callback/)"

}

Response:

{

"error": "cluster-a does not have enough resources to run the task"

}

**Scenario-5: See if callback is sent. If a team is targeting extra points.**