1. URL for repo:

<https://github.com/edellhou/CS6650/tree/main/Assignment3>

2:

1. Database Design:

Database chosen is Redis. It is a key-value based nonrelational database.

The data models I design can query following request.

* “For skier N, how many days have they skied this season?”
* “For skier N, what are the vertical totals for each ski day?” (calculate vertical as liftID\*10)
* “For skier N, show me the lifts they rode on each ski day”
* “How many unique skiers visited resort X on day N?”

There are 2 data models. The first model can query request 1-3, and the second data model can query request 4.

First Model: Redis Sets are used. Key: SkierID:dayID, Value: liftID

Assumption made: The season in our project is fixed, which is 2022, so I didn’t use season as part of the Key. I know dayID is also fixed, which is 1. But request 2-3 explicitly wanna know the skier info by each ski day, so I made dayId as part of the key.

Second Model: Redis HyperLoglog is used.

HyperLogLog is good at this since you get great performance at low computation cost, and a small amount of memory. HyperLogLog can be used in various situations like;

* Counting unique visitors
* Creating unique in a book or books
* Keeping best stocks of all times
* Create unique names of a products, services, category
* A situation where you are not worries about the elements of the datasets, but its counts
* Create unique names of students in a class and many more.

Key:resortID:dayID, value:skierID

Eg: say at resort 3 on day 1 which is the key, skier no.10 skied 5 times, skier no.100 skied 2 times, the hyperloglog datatype will return 2

1. Deployment topology.

6 servers are deployed.

1 for client

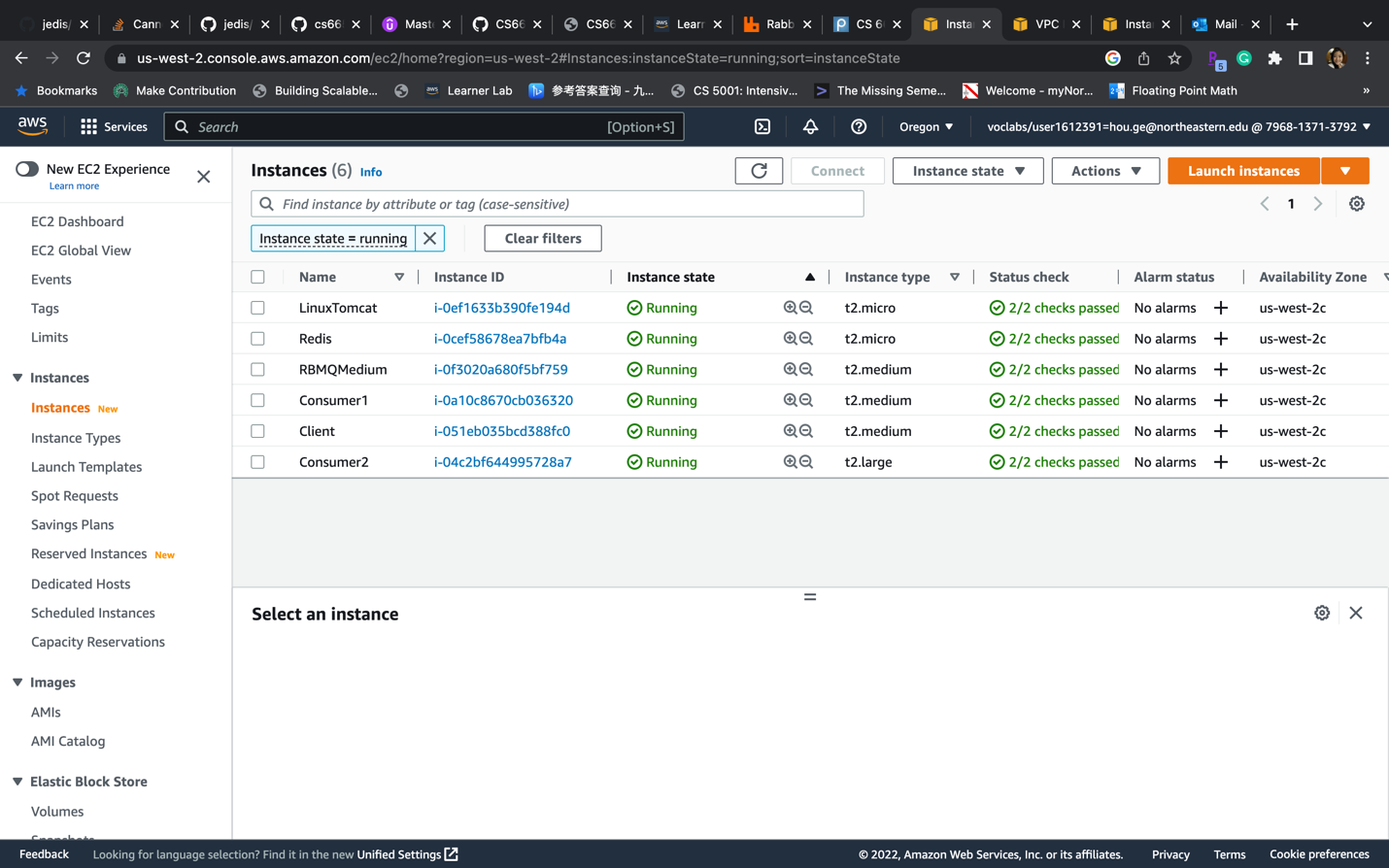
1 for Tomcat

1 for RBMQ

2 for consumer

1 for Redis Database

Picture below shows the instance type



3:Throughput and RBMQ console clipsText

Description automatically generated

Graphical user interface

Description automatically generated