Project title:-

NIM , a turn-based strategy game

Introduction:-

Nim is a two-player game of strategy, played with a set of objects such as stones, matches, or coins. The game starts with a pile of objects, and the players take turns removing objects from a single pile.

This game has many variations, but the basic rules are as follows:-

- There is a pile of objects, such as stones or matches.
- Players take turns removing any number of objects from a single pile.
- The player who takes the last object wins the game.

Ideaframework:-

The base idea to develop a client-server application that will use socket programming with TCP protocol for connection orientation to play the NIM game between two clients.

Socket programming using TCP (Transmission Control Protocol) is a communication protocol that provides reliable, ordered, and error-checked delivery of data between applications. In TCP, a connection-oriented approach is followed, which means that before sending any data, a connection is established between the two parties. Once the connection is established, data can be sent in both directions.

In socket programming using TCP, two types of sockets are used: the server socket and the client socket. The server socket waits for incoming connections, while the client socket initiates the connection.

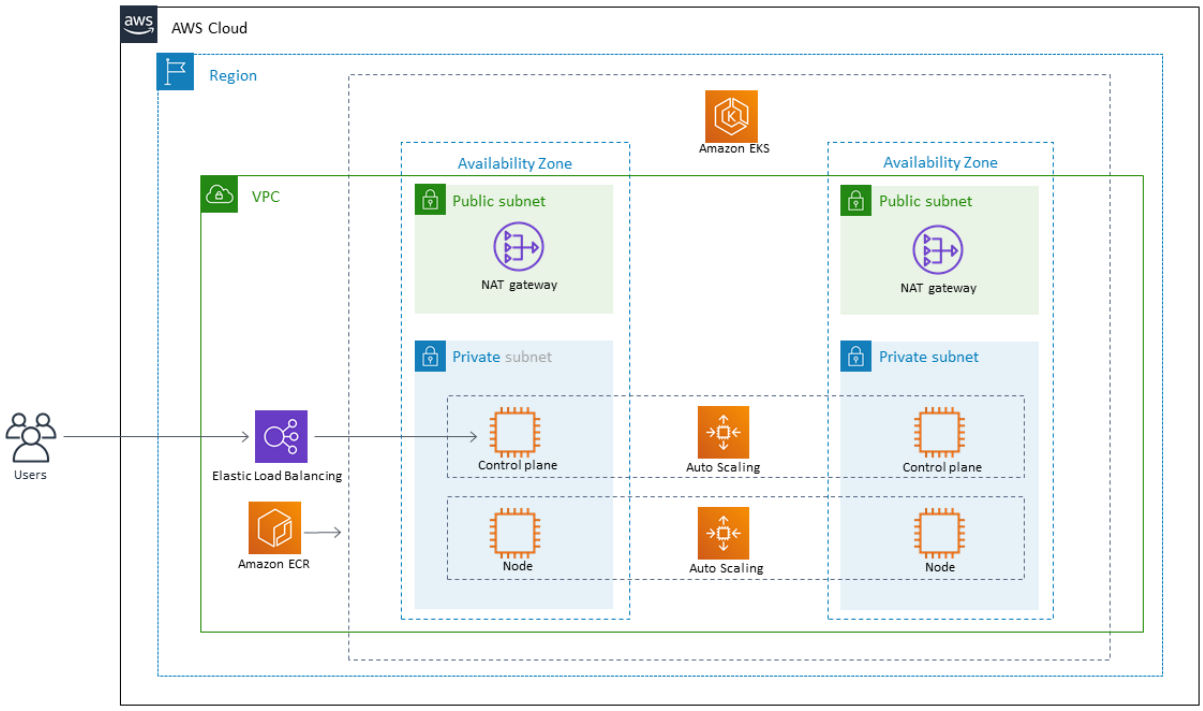
Concept note & application functions:-

The basic steps involved in socket programming using TCP are as follows:

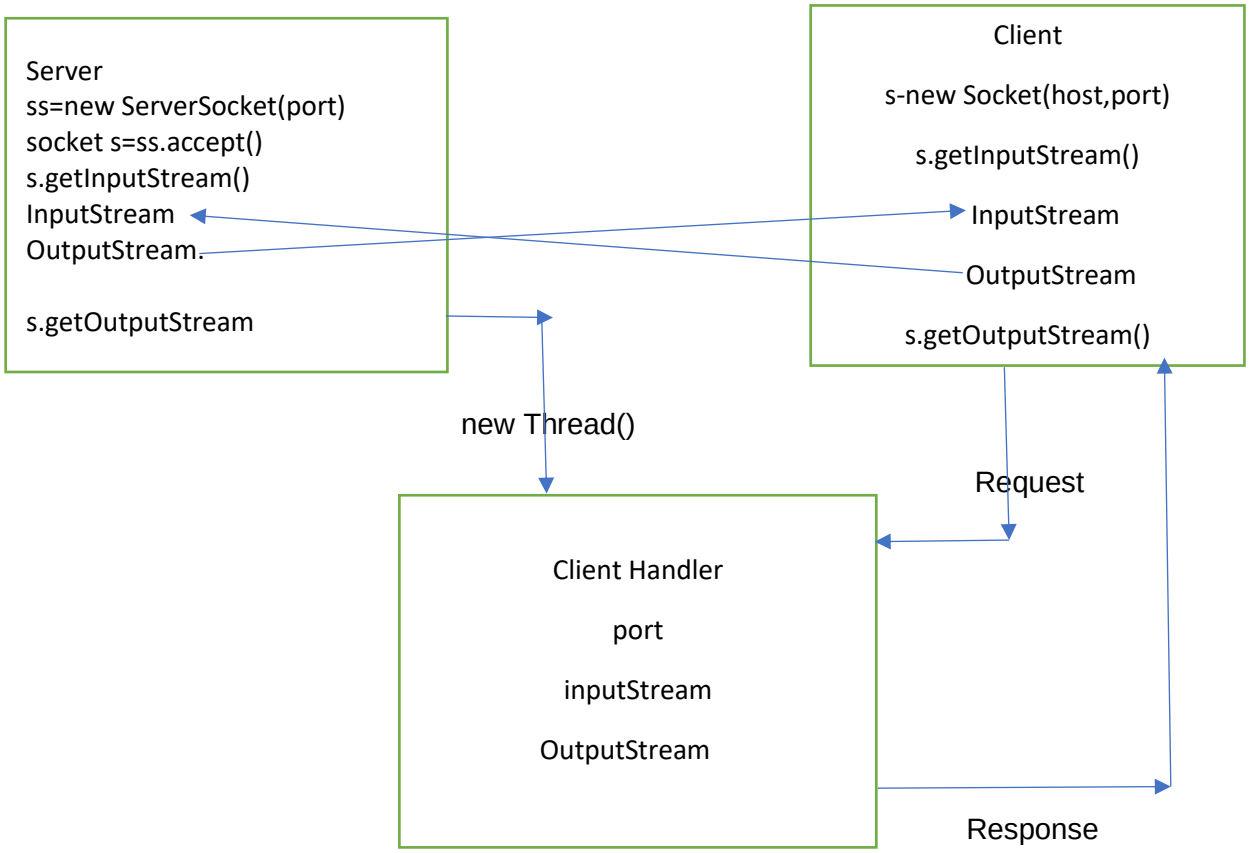
- Create a server socket: The server socket listens for incoming connections from clients.
- Create a client socket: The client socket initiates a connection to the server socket.
- Establish a connection: Once the client socket has connected to the server socket, a connection is established, and data can be sent in both directions.
- Send and receive data: After the connection is established, the client and server can send and receive data using the input and output streams of their respective sockets.
- Close the connection: When the communication is completed, the client and server sockets close the connection.

Cloud deployment:-

- 1)build a jar file
- 2)build a docker-image using Docker-file.
- 3)host Server-side on ec2 and push docker image into AWS ECR.
- 4)deploy the image file from ECR into AWS EKS.



Flowchart:-


Server
ss=new ServerSocket(port)
socket s=ss.accept()
s.getInputStream()
InputStream
OutputStream.
s.getOutputStream
Client
s-new Socket(host,port)
s.getInputStream()
InputStream
OutputStream
s.getOutputStream()
new Thread()
Request
Client Handler
port
inputStream
OutputStream
Response

Literature review:-

1. Java in its current state is still a little on the slow side, which rules it out as a possibility for high-performance, graphically intensive games. However, Java, on the other hand, provides one API and requires you only develop a single set of source code that can be executed on any Java supported platform.

2. Java's abstraction over the socket API is to use a serversocket object that automatically listens, then creates a different socket on accept. Java sockets have input streams and output streams built in, which makes programming rather pleasant.

3 .Container deployment is a method for quickly building and releasing complex applications. Docker container deployment is a popular technology that gives developers the ability to construct application environments with speed at scale.

References:-

1. https://www.gamedeveloper.com/programming/java-network-game-programming

2. https://cs.lmu.edu/~ray/notes/javanetexamples/

3. https://avinetworks.com/glossary/container-deployment/#:~:text=Container%20deployment%20is%20a%20method,environments%20with%20speed%20at%20scale.