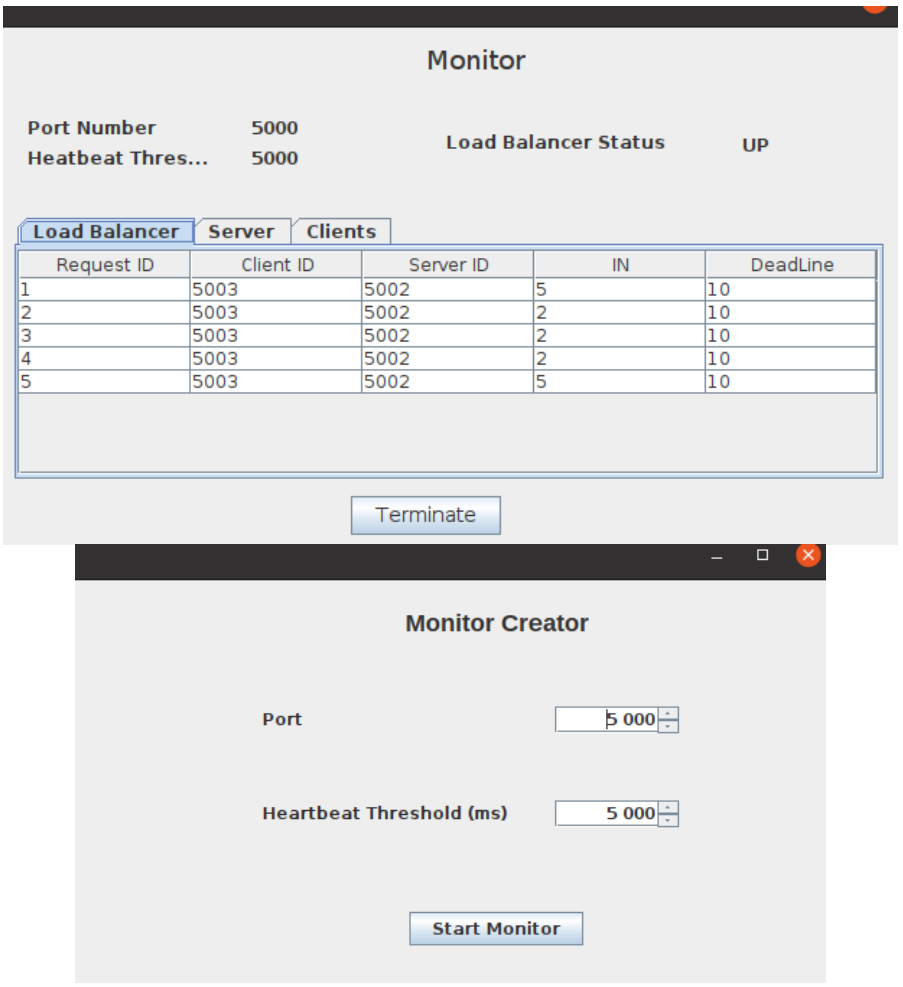
Software Architectures

Practical Assignment 3 Report

Leandro Silva, 93446
Margarida Martins, 93169

Monitor

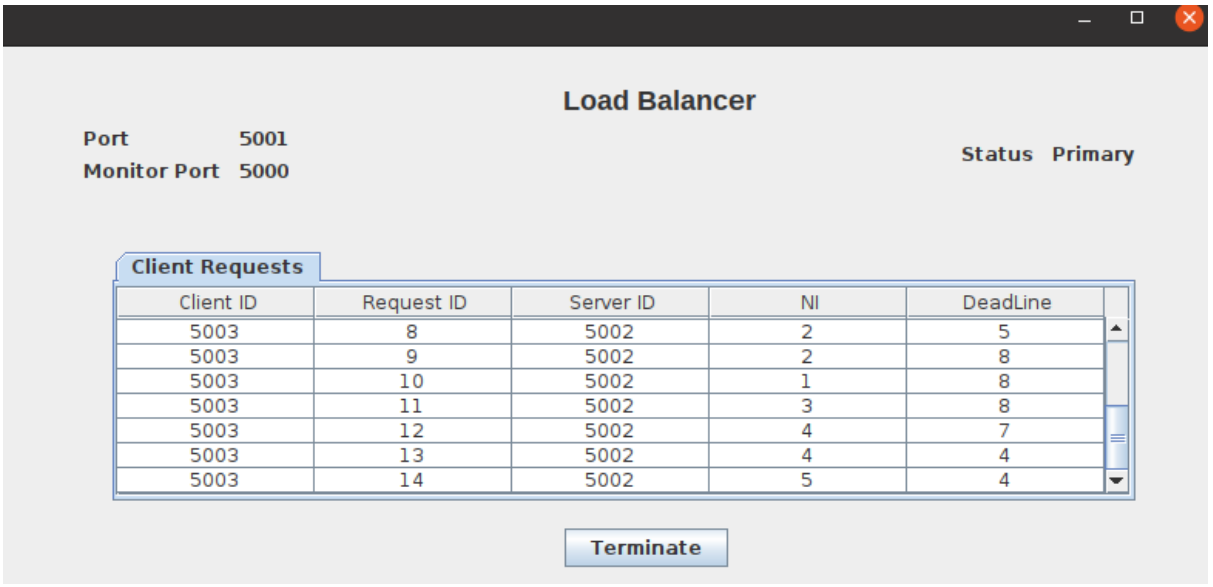
The Monitor process is responsible for knowing the status of the system.
The Monitor implements a heartbeat system in order to see if some server or load balancer went down and act accordingly. Firstly the monitor sends a heartbeat message to the respective service/load balancer and waits the amount of time defined in its initial configuration. After waiting it will consider every process as down if it did not respond to the message. When the monitor detects that the primary loadBalancer died it will send a message to the secondary so that it knows he has to take the primary place. When the monitor detects that a server is down it will send all the requests it was processing to the loadBalancer.

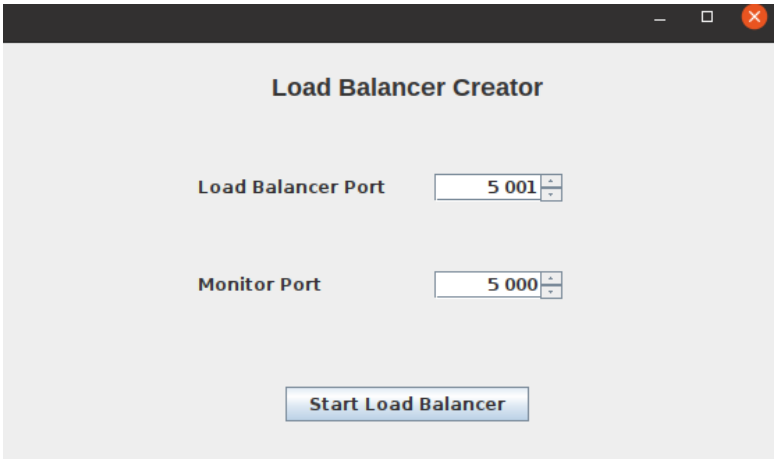


Monitor GUI's

Load Balancer

The Load Balancer is responsible to receive Client requests and forward the requests to the best server.
The calculation of the best server is done using the current system status, for that the load balancer asks the monitor in order to know the current active servers and the number of requests (processing and pending) and the total number of iterations each has. The server with the lowest number of requests is always chosen, if there is more than one server the one with the smallest number of iterations will be chosen.
The secondary load balancer will be waiting until it receives a message from the monitor in order to become the primary one.
In order to make client-load balancer connection the most transparent as possible, when the primary load balancer dies, the secondary will connect to the client in the same port as the primary one.

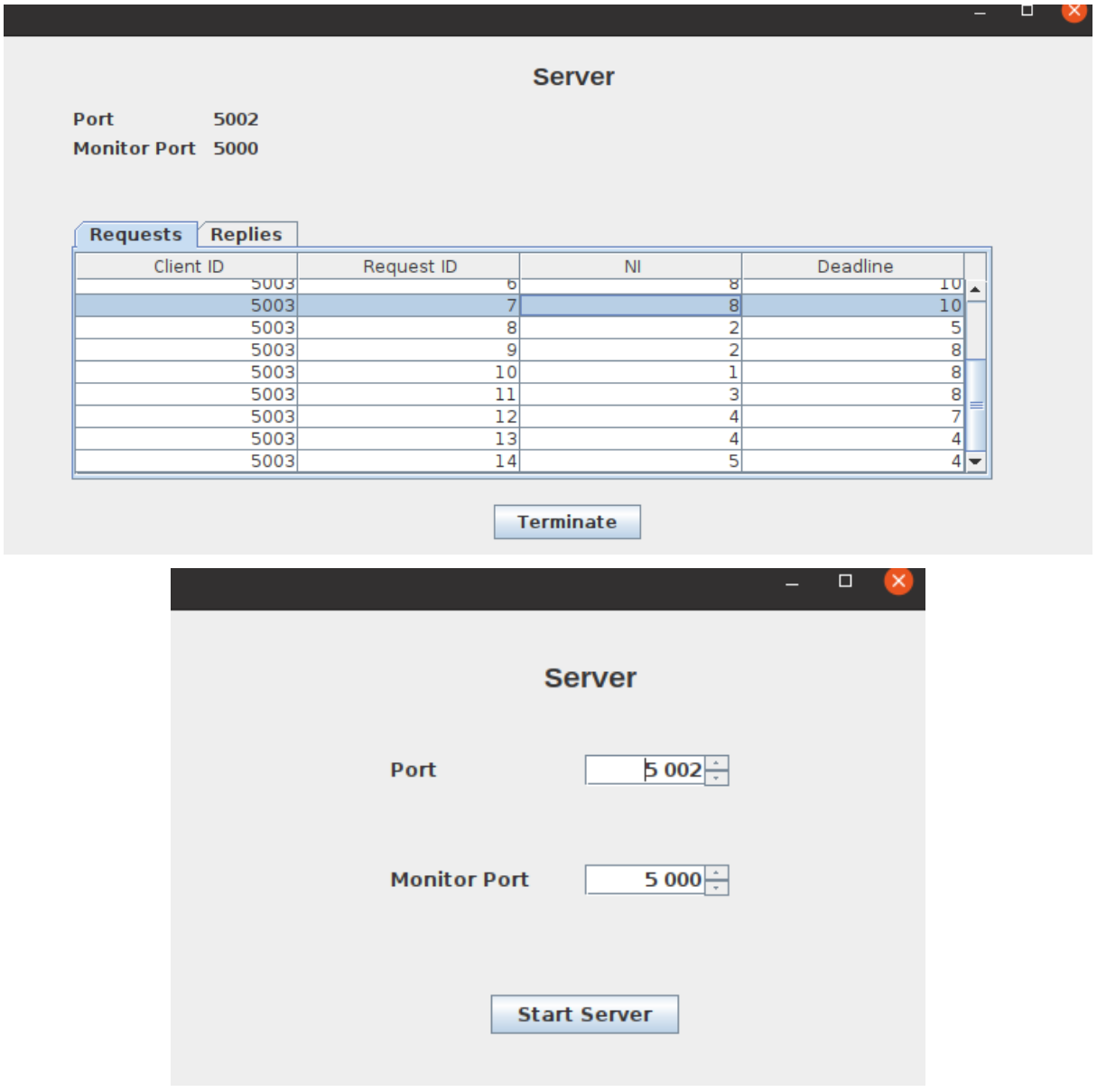




Load Balancer GUI's

Server

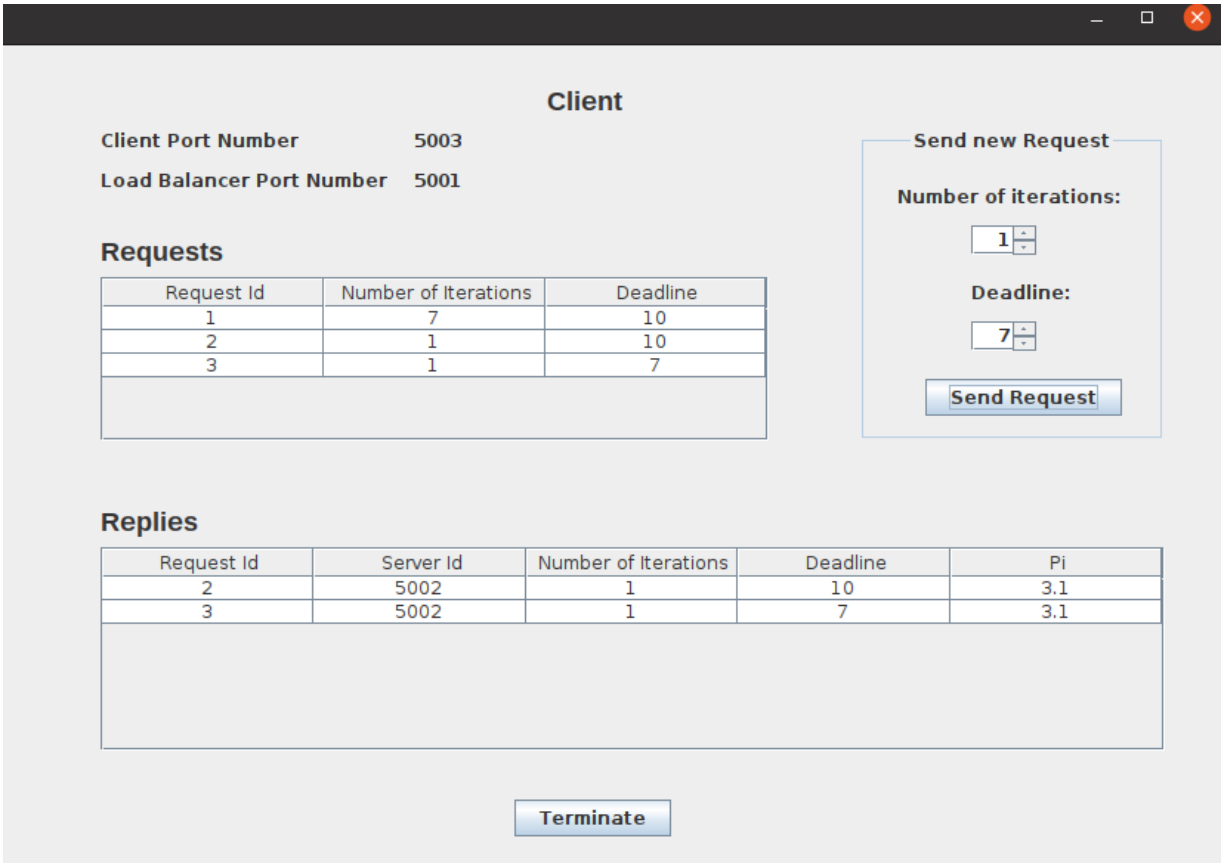
The Server is responsible to receive requests, make the calculations and reply to the respective client.
When receiving a request the server checks whether he is capable of accepting it (if the total number of requests is less than five or the total number of iterations is less than twenty).
Each server always has three worker threads running. These threads will be awakened when there are calculations to be done.
When there are two pending requests and there is a free worker thread the request that will be picked will be the one with the smallest deadline.

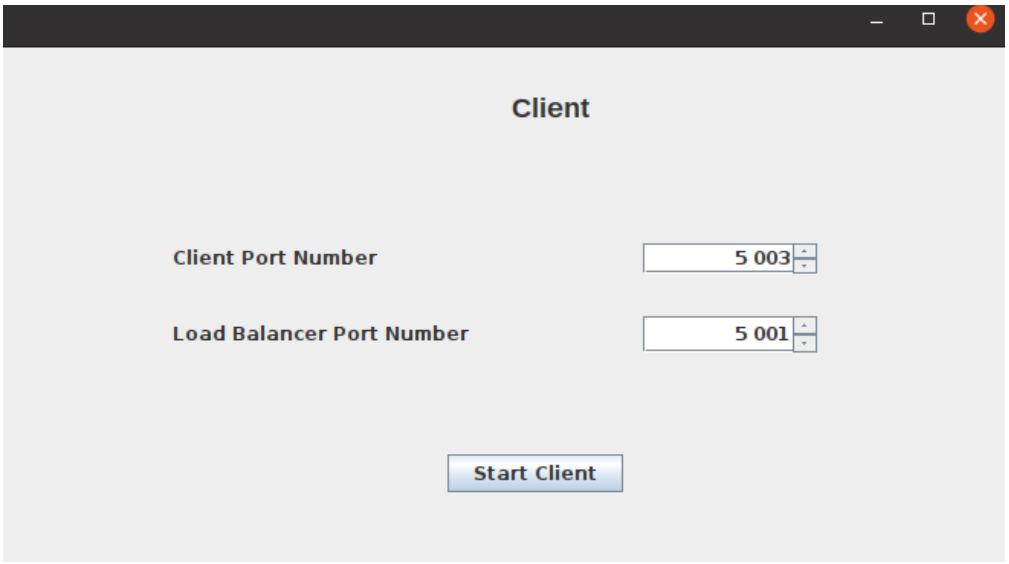


Server GUIs

Client

The client is responsible to make requests to the load balancer and receive replies from a server.
The client can send as many requests as he wishes which can then be rejected or accepted by a server.
The requests can have different number of iterations and deadlines.





Client GUIs

Features not Implemented

In this assignment we were able to implement all the use cases and its respective features.

Contribution

Leandro Silva - 50%
Margarida Martins - 50%