DOCUMENTATION

ASSIGNMENT 2

STUDENT NAME: GRAD LAURENȚIU-CĂLIN

GROUP: 30425

Contents

[1. Assignment Objective 3](#_Toc163198336)

[2. Problem Analysis, Modeling, Scenarios, Use Cases 3](#_Toc163198337)

[3. Design 3](#_Toc163198338)

[4. Implementation 3](#_Toc163198339)

[5. Results 3](#_Toc163198340)

[6. Bibliography 3](#_Toc163198341)

# Assignment Objective

Design and implement a queues management application which assigns clients to queues such that the waiting time is minimized. Queues are commonly used to model real world domains. The main objective of a queue is to provide a place for a "client" to wait before receiving a "service". The management of queue-based systems is interested in minimizing the time amount their "clients" are waiting in queues before they are served. One way to minimize the waiting time is to add more servers, i.e., more queues in the system (each queue is considered as having an associated processor) but this approach increases the costs of the service supplier. The queues management application should simulate (by defining a simulation time ) a series of N clients arriving for service, entering Q queues, waiting, being served, and finally leaving the queues. All clients are generated when the simulation is started and are characterized by three parameters: ID (a number between 1 and N), 𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙 (simulation time when they are ready to enter the queue) and 𝑡\_𝑠𝑒𝑟𝑣𝑖𝑐𝑒 (time interval or duration needed to serve the client, i.e. waiting time when the client is in front of the queue). The application tracks the total time spent by every client in the queues and computes the average waiting time. Each client is added to the queue with the minimum waiting time when its 𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙 time is greater than or equal to the simulation time (𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙 ≥ 𝑡\_𝑠𝑖𝑚𝑢𝑙𝑎𝑡𝑖𝑜𝑛). The following data should be considered as input data for the application that should be inserted by the user in the application’s user interface:

* Number of clients (N)
* Number of queues (Q)
* Simulation interval (𝑡\_𝑠𝑖𝑚𝑢𝑙𝑎𝑡𝑖𝑜𝑛\_𝑀𝐴𝑋)
* Minimum and maximum arrival time (𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙\_𝑀𝐼𝑁 ≤ 𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙 ≤ 𝑡\_𝑎𝑟𝑟𝑖𝑣𝑎𝑙\_𝑀𝐴𝑋)
* Minimum and maximum service time (𝑡\_𝑠𝑒𝑟𝑣𝑖𝑐𝑒\_𝑀𝐼𝑁 ≤ 𝑡\_𝑠𝑒𝑟𝑣𝑖𝑐𝑒 ≤ 𝑡\_s𝑒𝑟𝑣𝑖𝑐𝑒\_𝑀𝐴𝑋)

# Problem Analysis, Modeling, Scenarios, Use Cases

# Design

# Implementation

# Results

# Bibliography

1. [Unified Modeling Language - Wikipedia](https://en.wikipedia.org/wiki/Unified_Modeling_Language)
2. [Use Case Diagrams | Unified Modeling Language (UML) - GeeksforGeeks](https://www.geeksforgeeks.org/use-case-diagram/)
3. Programming Techniques lectures, laboratories, and support materials
4. Paul Deitel, Harvey Deitel, Java How to program (10th edition), Publisher: Pearson Education, Inc., Upper Saddle River, NJ, United States, ISBN: 978-0-13-380780-6, Published: 2015