

# DOCUMENTATIE

## TEMA 2

NUME STUDENT: RUS ANA-MARIA CARINA  
GRUPA: 30226

## CUPRINS

Obiectivul temei.....	3
Analiza problemei, modelare, scenarii, cazuri de utilizare .....	3
Proiectare.....	5
Implementare .....	7
Rezultate .....	10
Concluzii .....	10
Bibliografie.....	11

# 1. OBIECTIVUL TEMEI

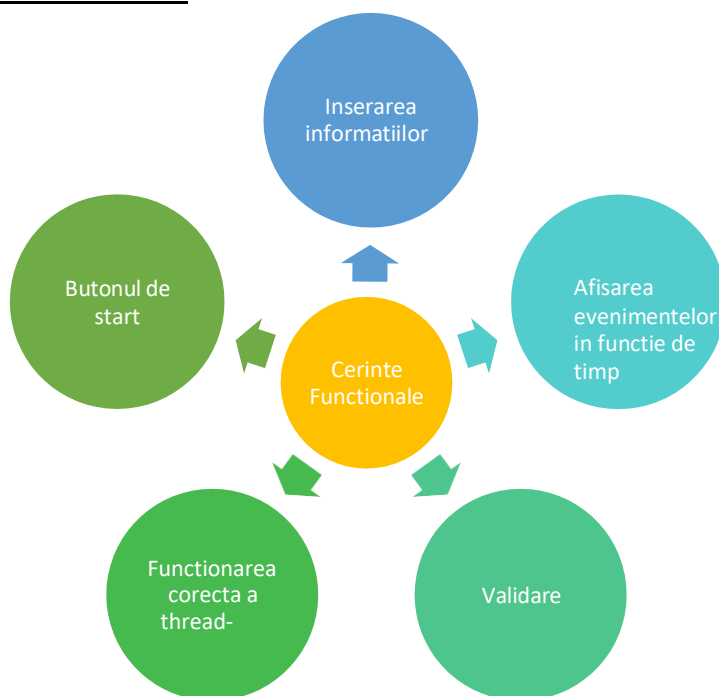
Obiectivul temei este de a crea o simularea a unei aplicatii de distribuire a unor clienti la coada in functie de timp folosind thread-uri.

## OBIECTE SECUNDARE:

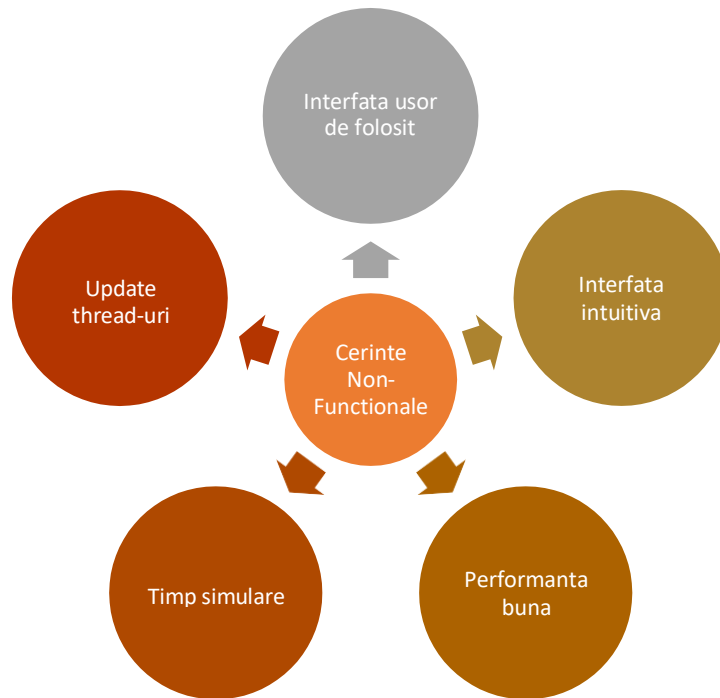
1. Intelegerea problemei
2. Crearea unei interfete potrivite pentru introducerea informatiilor
3. Crearea unei interfete ce afiseaza evenimentele
4. Crearea unei clase de management
5. Crearea unor cozi folosind thread-uri
6. Salvarea informatiilor in fisier txt

# 2. ANALIZA PROBLEMEI, MODELARE, SCENARIU, CAZURILE DE UTILIZARE

## Cerinte functionale



## Cerinte non-functionale



## User-Case

**User Case:** Setup-ul simularii

**Actor primar:** User-ul

**Scenariu pentru succes:**

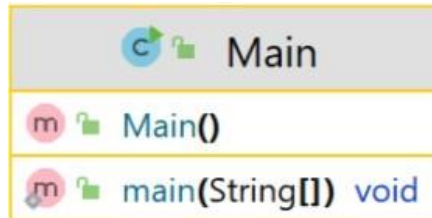
1. User-ul introduce valorile dorite pentru: numarul de clienti, numarul de cozi, intervalul simularii, limitele intervalului de sosire, limitele intervalului de servire
2. User-ul apasa butonul de validare a informatiilor
3. Aplicatia valideaza datele si afiseaza in casuta de text unu mesaj care informeaza user-ul sa inceapa simularea
4. User-ul apasa butonul de start si incepe simularea

**Secventa alternatica:** Valori invalide pentru parametrii de inceput

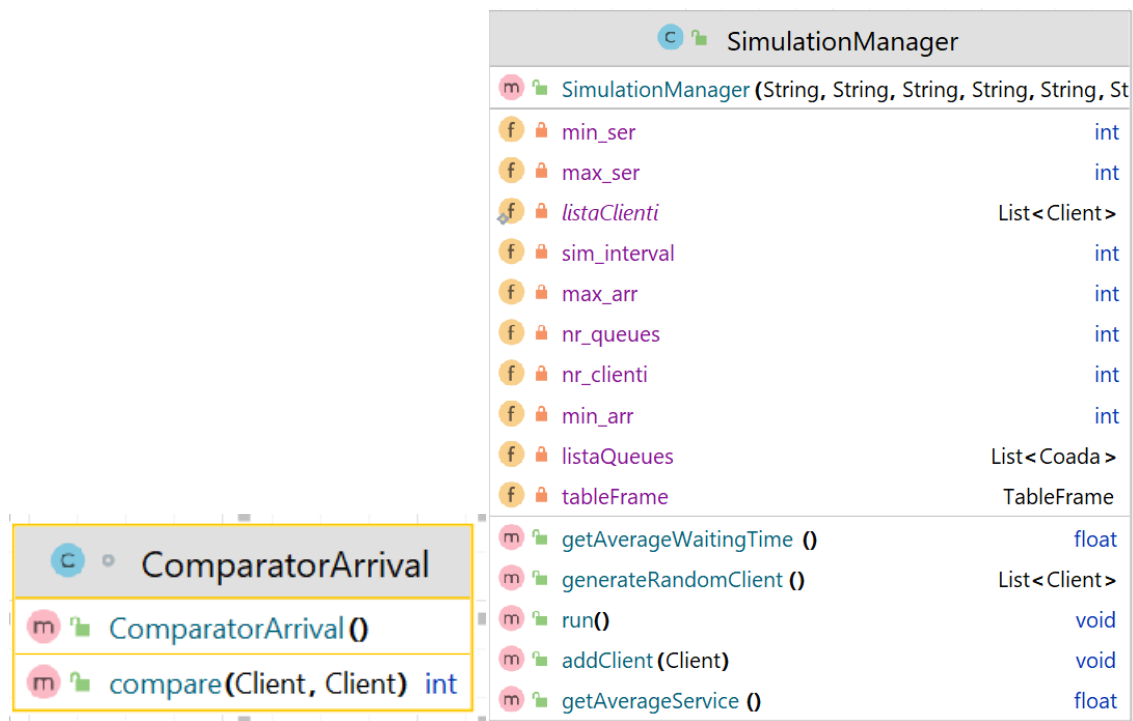
- User-ul introduce valori invalide
- Aplicatia afiseaza un mesaj care informeaza useu-ul ca valorile sunt invalide
- Scenariul se reintoarce la punctul 1.

### 3. PROIECTARE

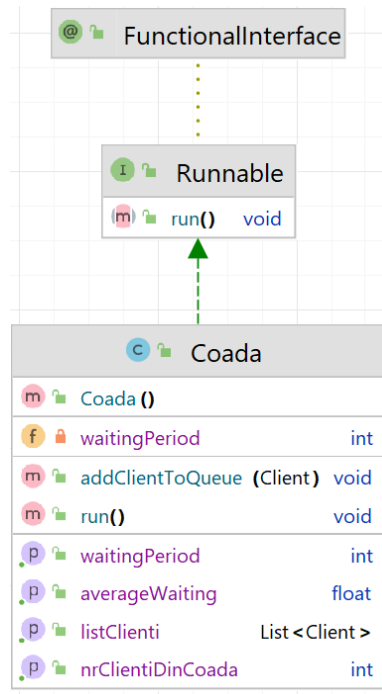
#### Diagrama Main



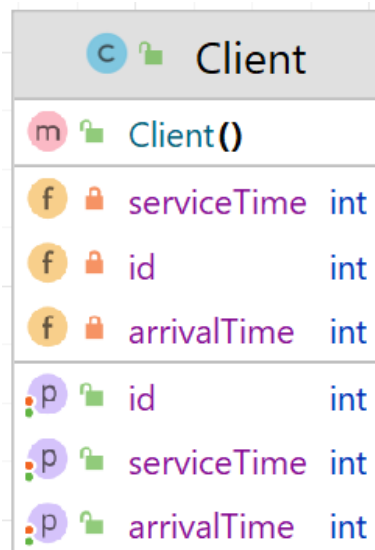
#### Diagrama SimulationManager



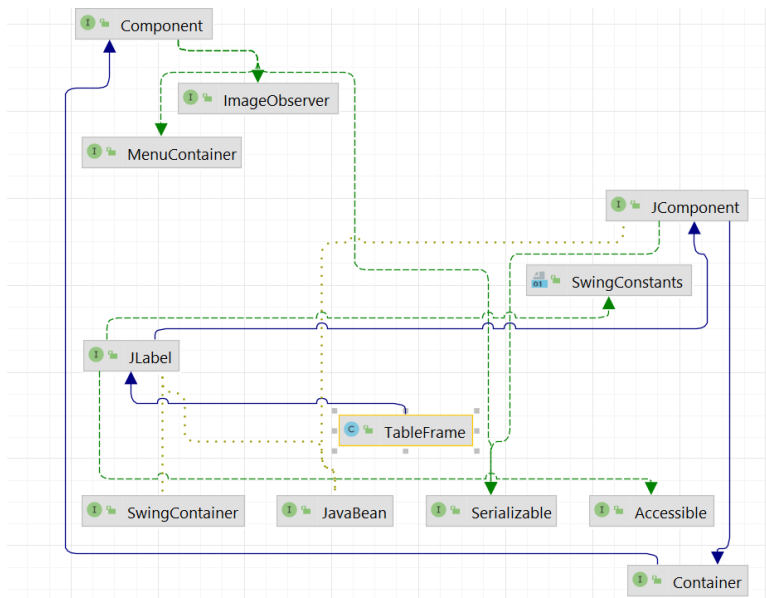
## Diagrama Coadă



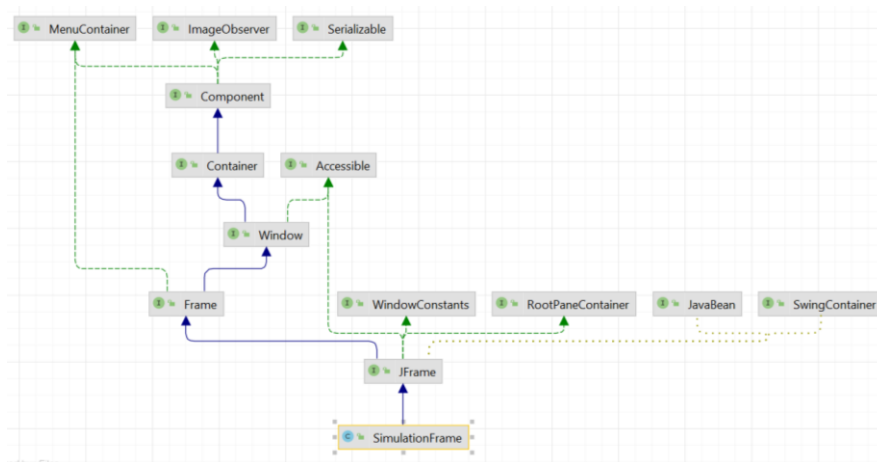
## Diagrama Client



## Diagrama TableFrame



## Diagrama SimulationFrame



Am folosit doua pachete:

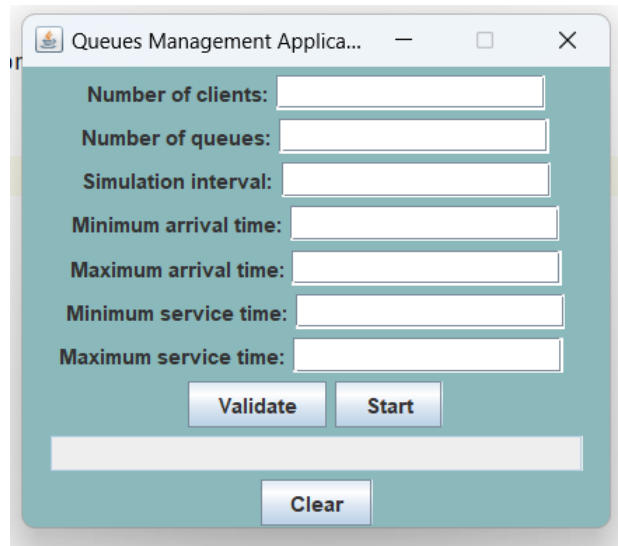
1. GUI
2. Org-example

### 3. IMPLEMENTARE

#### 1. Clasa Main

In clasa Main am initializat interfata SimulationFrame.

#### 2. Clasa SimulationFrame



In clasa CalcView am creat interfata folosind butoane, text fields, panel-lui pentru ca user-ul sa poata introduce informatiile initiale. Se porneste un thread principal.

Am folosit ActionListener pentru a implementa operatiile facute de fiecare buton. Aici se verifica validitatea informatiilor introduse si se porneste simularea.

#### 3. Clasa TableFrame

Aceasta clasa se ocupa cu afisarea evenimentelor: fiecare secunda a simularii impreuna cu lista clientilor care asteapta sa fie introdusi la coada, cozile cu clientii lor si timpul de servire a clientilor.

Am folosit FileWriter pentru a salva evenimentele simularii intr-un fisier txt numit log.txt.



#### 4. Clasa SimulationManager

Clasa SimulationManager pastreaza informatiile introduse la in interfata SimulationFrame. Metoda run incrementeaza timpul simularii si apeleaza metoda addTextArea din TableFrame pentru a adauga o noua textArea care sa contina informatiile intr-o secunda anume.

Am folosit metoda generateRandomClient pentru a genera random secunda in care un client ajunge si timpul de servire a clientului, dupa care am ordonat clientii in functie de timpul la care au ajuns.

#### 5. Clasa Coadă

Clasa Coadă retine lista de clienti din ea si perioada de asteptare la coada intr-o anumita secunda.

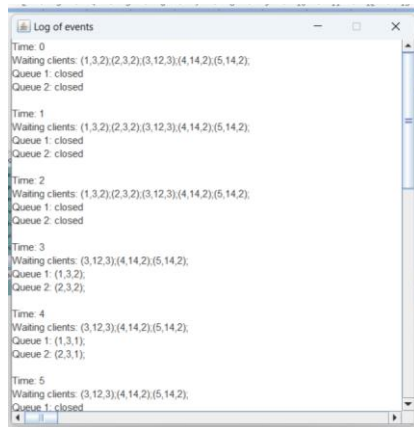
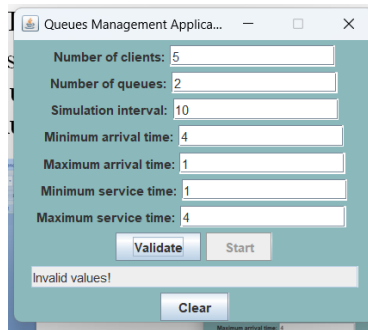
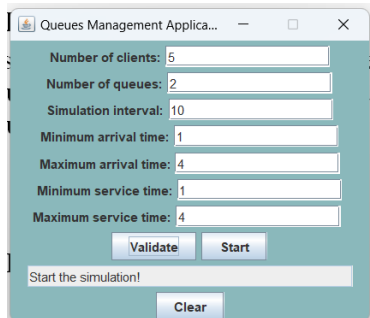
Metoda run scade timpul de servire a clientului care este servit in aceasta secunda si decrementeaza perioada de asteptare la coada actuala. In cazul in care clientul a fost servit, se sterge din lista de clienti.

#### 6. Clasa Client

Clasa Client reprezinta clientii cu id-ul unic, service time si arrival time.

## 7. REZULTATE

Se poate observa ca este afisat un mesaj care informeaza user-ul daca informatiile introduse sunt valide sau nu.



Informatiile sunt afisate in fiecare secunda.

## 8. CONCLUZII

Am invatat sa creez interfata mai sofisticata si sa afisez in timp real informatiile intr-o alta interfata. Am invatat sa folosesc thread-urile si sincronizarea.

## 9. BIBLIOGRAFIE

1. <http://docs.oracle.com/javase/tutorial/essential/concurrency/index.html>
2. [https://www.w3schools.com/java/java\\_threads.asp](https://www.w3schools.com/java/java_threads.asp)
3. <https://stackoverflow.com/questions/6641897/scrollbars-not-appearing-in-jscrollpane>
4. <http://www.javacodegeeks.com/2013/01/java-thread-pool-example-using-executors-and-threadpoolexecutor.html>