



**Linnéuniversitetet**

Kalmar Väst

## *Assignment 2*

### *Performance Engineering*



*Author:* Rashed Qazizada

*Supervisor:* Diego Perez

*Semester:* Spring 2020

*Course name:* Software  
Engineering Design



## Table of Contents

<b>1 Introduction</b>	<b>3</b>
<b>2 Calculation results</b>	<b>3</b>
2.1 A1	3
Average service time $S_k$ of the <i>ApplicationServer</i> and the <i>VoucherPaymentServer</i>	3
<i>ApplicationServer</i>	3
<i>VoucherPaymentServer</i>	3
2.2 A2	4
To calculate the minimum amount of resources that we need in each service center to handle an average arrival rate to the system of 10 sessions per second. When expect that the workload to our e-commerce site will increase to an average rate of 10 user sessions per second.	4
<b>3 JMT Queueing Network simulation</b>	<b>6</b>
3.1 Model	6
3.2 System Response Time	7
3.3 Utilization of each service center	8
3.4 Throughput of each service center	9
3.5 Card payment server service section	10
3.6 Card payment server Routing section	10
3.7 Database service section	11
3.8 Database Routing Section	11
3.9 User thinking Routing	12
3.10 User thinking service section	12
3.11 Voucher payment card service section	13
3.12 Voucher payment card Routing section	13
3.13	14
<b>4 UML Diagrams profiled with MARTE</b>	<b>14</b>
4.1 MARTE	14
<b>5 Bonus points</b>	<b>15</b>



## 1 Introduction

This report is going to study the performance of a software system that executes web sessions from users from the Internet.

The system is composed of five service centers: A *WebServer*, an *ApplicationServer*, a *Database*, a *CardPaymentServer*, and a *VoucherPaymentServer*.

The running system has been observed during 1000 minutes. During these 1000 minutes, 72000 user sessions have been completed and 1.2 sessions per second arrived at an average rate.

## 2 Calculation results

Time (T) = 1000 minutes = 60 000 seconds

No. of completed requests (C) = 72 000

Throughput (X) =  $C/T = 72000 / 60000 = 1.2$

### 2.1 AI

Average service time  $S_k$  of the *ApplicationServer* and the *VoucherPaymentServer*

#### ***ApplicationServer***

$U = 0.528$      $V_k = 2$

#### Utilization Law

$U_k = D_k \cdot X \Rightarrow D_k = U_k / X$

$D_k = 0.528 / 1.2$

$D_k = 0.44$

#### Service Demand Law

$D_k = S_k \cdot U_k$

$S_k = D_k / U_k$

$S_k = 0.44 / 2$

$S_k = 0.22$

Therefore, service time of application server is 0.22s

#### ***VoucherPaymentServer***

Avg no. of jobs (N) = 0.168

Avg system residence time (R) = 0.35

Demand Time ( $D_k$ ) = 0.12s



## Little's Law

$$N = X_k * R$$

$$X_k = N / R$$

$$X_k = 0.48$$

## Using Service Demand Law

$$U_k = D_k * X$$

$$U_k = 0.12 * 1.2$$

$$U_k = 0.144$$

## Using Utilization law

$$S_k = U_k / X_k$$

$$S_k = 0.144 / 0.48$$

$$S_k = 0.3 \text{ s}$$

Therefore, service time of the VoucherPaymentServer is 0.3s

## 2.2 A2

To calculate the minimum amount of resources that we need in each service center to handle an average arrival rate to the system of 10 sessions per second. When expect that the workload to our e-commerce site will increase to an average rate of 10 user sessions per second.

### **Solution:**

Using the operational laws I found the amount of resources for each service center for which the service center is not saturated.

Given:

10 sessions per second implies throughput.

$$X = 10$$

We assume that the time taken is the same.

$$T = 60000 \text{ s}$$

Using the Forced flow law I have calculated the saturation of the service centres.

According to the law the system utilization should not exceed 100 %.

I have made reasonable assumptions on the minimum number of resources.

This was also verified using JMT and in all instances, the utilization of the each service centre is less than 100%.



<i>WebServer</i>	<i>ApplicationServer</i>	<i>Database</i>	<i>VoucherPayment</i>	<i>CardPayment</i>
$X=10$	$X_k=V_k \cdot X$	$X=10$	$U_k=10 \cdot 0.3=3$	$U_k=X_k \cdot S_k$
$S=0.09$	$X_k=2 \cdot 10$	$S=0.045$		$U_k=10 \cdot 0.4=4$
$U_k=X_k \cdot S_k$	$X_k=20$	$U_k=10 \cdot 0.045=0.45$	$c=2$	$c=6$
$U_k=0.9$	$S=0.22$	Since, iterating 5 times		
$c=2$	$U_k=0.22 \cdot 20$	$U_k=0.45 \cdot 5=2.25$		
	$U_k=4.4$	$c=5$		
	$c=5$			

Where,

$c$  = number of resources

$U_k$  = utilization of a serviced centre

### 3 JMT Queueing Network simulation

The system has been modeled using JMT Queueing Network simulation engine

#### 3.1 Model

Queueing Networks Design

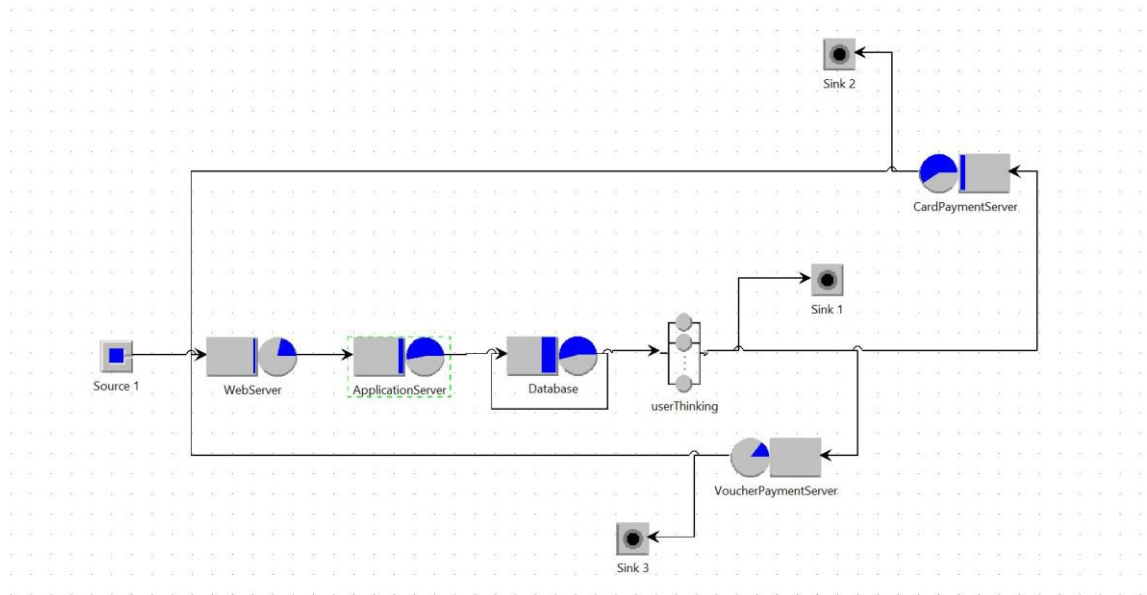
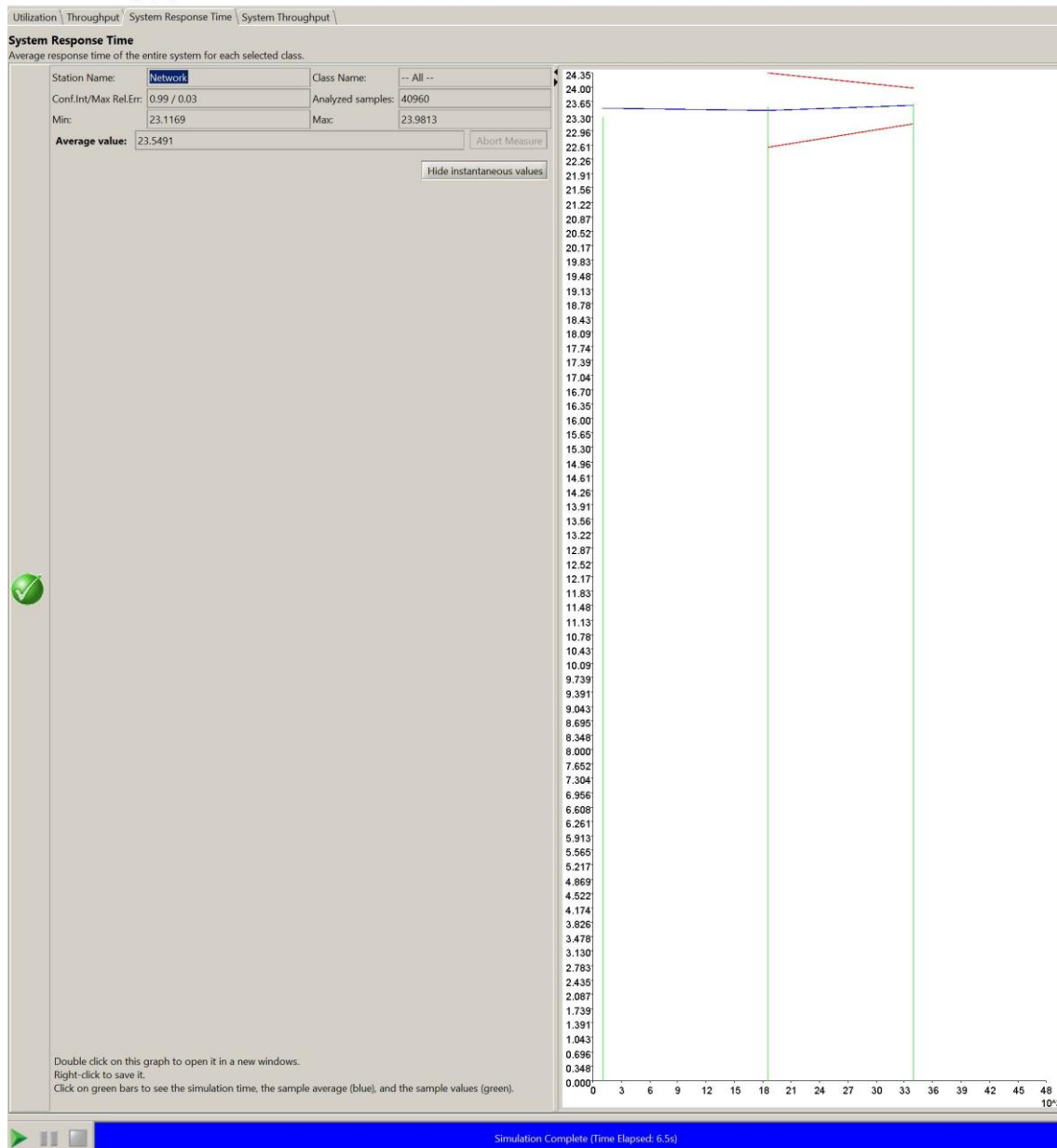


Figure 1: JMT Queueing Network simulation engine



## 3.2 System Response Time





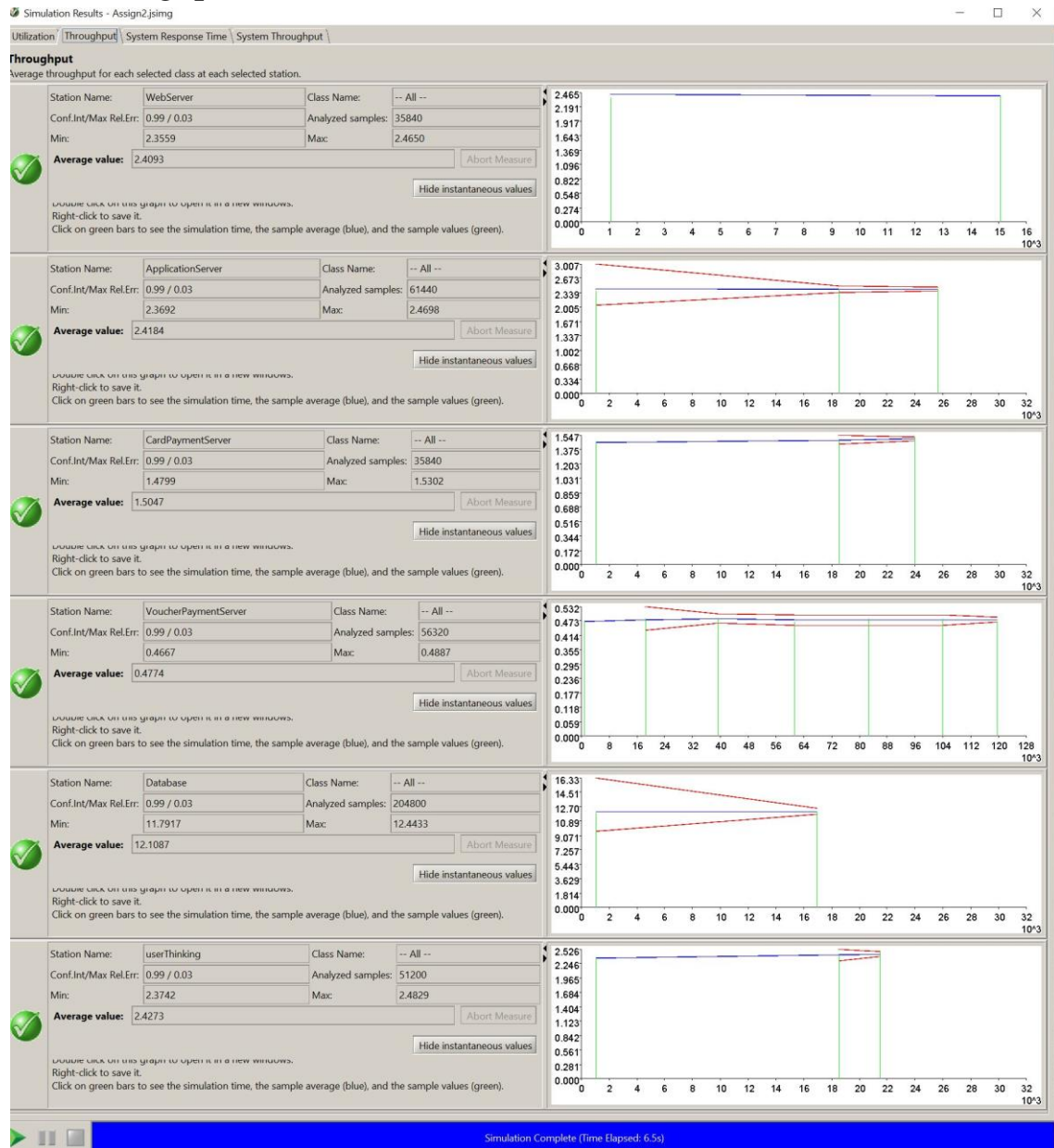
## 3.3 Utilization of each service center







## 3.4 Throughput of each service center





## 3.5 Card payment server service section

Editing CardPaymentServer Properties...

Station Name: CardPaymentServer

CardPaymentServer Parameters Definition

Queue Section | Service Section | Routing Section

Number of Servers: 1

Service Time Distributions

Class	Strategy	Service Time Distribution
Class1	Load Independent	exp(2.5)

Done

Editing Class1 Service Time Distribution...

Selected Distribution: Exponential

Exponential [exp( $\lambda$ )]:

$$f(x) = \lambda e^{-\lambda x}$$

$\lambda$ : 2.5

mean: 0.4

OK Cancel

## 3.6 Card payment server Routing section

Editing CardPaymentServer Properties...

Station Name: CardPaymentServer

CardPaymentServer Parameters Definition

Queue Section | Service Section | Routing Section

Routing Strategies

Class	Routing Strategy
Class1	Probabilities

Description

Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1.

Routing Options

Destination	Probability
WebServer	0.5
Sink 2	0.5

Done



## 3.7 Database service section

The screenshot shows a dialog box titled "Editing CardPaymentServer Properties...". It has a "Station Name" field containing "CardPaymentServer". Below this is a section titled "CardPaymentServer Parameters Definition" with three tabs: "Queue Section", "Service Section", and "Routing Section". The "Routing Section" is active. It contains a "Routing Strategies" table with two columns: "Class" and "Routing Strategy". The first row shows "Class1" and "Probabilities". To the right of this table is a "Description" box with the text: "Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1." Below the description is a "Routing Options" table with two columns: "Destination" and "Probability". It contains two rows: "WebServer" with probability 0.5, and "Sink 2" with probability 0.5. At the bottom of the dialog is a "Done" button.

Class	Routing Strategy
Class1	Probabilities

**Description**  
Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1.

Destination	Probability
WebServer	0.5
Sink 2	0.5

## 3.8 Database Routing Section

The screenshot shows a dialog box titled "Editing Database Properties...". It has a "Station Name" field containing "Database". Below this is a section titled "Database Parameters Definition" with three tabs: "Queue Section", "Service Section", and "Routing Section". The "Routing Section" is active. It contains a "Routing Strategies" table with two columns: "Class" and "Routing Strategy". The first row shows "Class1" and "Probabilities". To the right of this table is a "Description" box with the text: "Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1." Below the description is a "Routing Options" table with two columns: "Destination" and "Probability". It contains two rows: "Database" with probability 0.8, and "userThinking" with probability 0.2. At the bottom of the dialog is a "Done" button.

Class	Routing Strategy
Class1	Probabilities

**Description**  
Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1.

Destination	Probability
Database	0.8
userThinking	0.2



## 3.9 User thinking Routing

The screenshot shows the 'Editing userThinking Properties...' dialog box. The 'Station Name' is 'userThinking'. The 'Routing Section' is active. The 'Routing Strategies' table shows 'Class1' with 'Probabilities' as the routing strategy. The 'Description' text states: 'Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1.' The 'Routing Options' table shows the following data:

Destination	Probability
CardPaymentServ...	0.62
VoucherPayment...	0.2
Sink 1	0.18

The 'Done' button is at the bottom right.

## 3.10 User thinking service section

The screenshot shows the 'Editing userThinking Properties...' dialog box with the 'Service Section' active. The 'Number of Servers' is set to 10. The 'Service Time Distributions' table shows 'Class1' with 'Load Independent' strategy and 'exp(0.1)' service time distribution. The 'Editing Class1 Service Time Distribution...' dialog box is open, showing the 'Exponential [exp(λ)]:' distribution with the formula  $f(x) = \lambda e^{-\lambda x}$ . The 'λ' is set to 0.1 and the 'mean' is set to 10. The 'OK' and 'Cancel' buttons are at the bottom right.



### 3.11 Voucher payment card service section

Editing VoucherPaymentServer Properties...

Station Name: VoucherPaymentServer

**VoucherPaymentServer Parameters Definition**

Queue Section | Service Section | Routing Section

**Number of Servers**

Number: 1

**Service Time Distributions**

Class	Strategy	Service Time
Class1	Load Independent	exp(3.333)

Done

Editing Class1 Service Time Distribution...

Selected Distribution: Exponential

**Exponential [exp( $\lambda$ )]:**

$$f(x) = \lambda e^{-\lambda x}$$

$\lambda$ : 3.33333333333333

mean: 0.3

OK Cancel

### 3.12 Voucher payment card Routing section

Editing VoucherPaymentServer Properties...

Station Name: VoucherPaymentServer

**VoucherPaymentServer Parameters Definition**

Queue Section | Service Section | Routing Section

**Routing Strategies**

Class	Routing Strategy
Class1	Probabilities

**Description**

Jobs are routed to stations connected to the current one according to the specified probabilities. If the sum of the probabilities is different from 1, all the values will be scaled to sum 1.

**Routing Options**

Destination	Probability
WebServer	0.95
Sink 3	0.05

Done

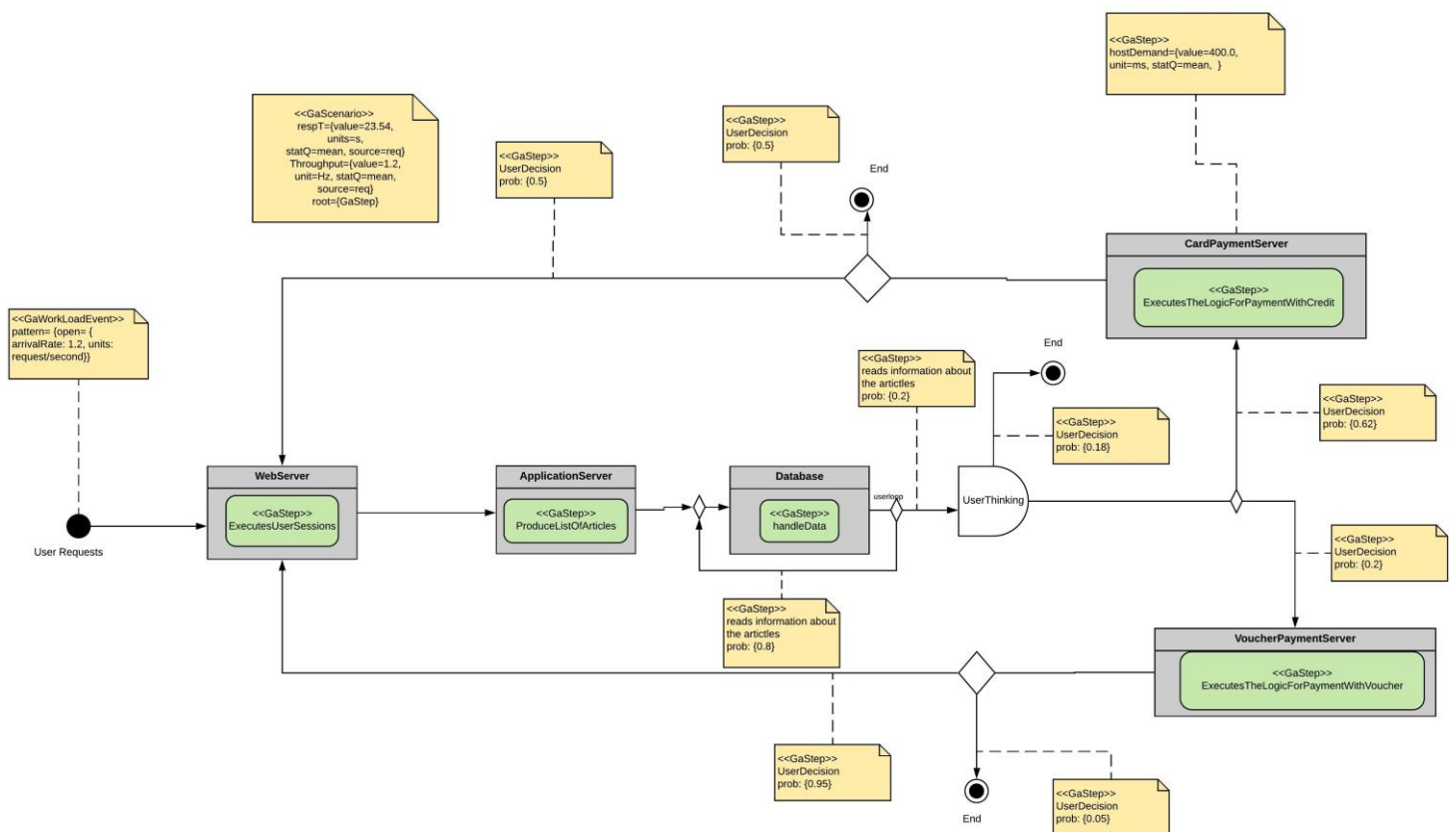


## 3.13

## 4 UML Diagrams profiled with MARTE

Only represents the workload of the system and the time that the CardPaymentServer needs to execute

### 4.1 MARTE





## 5 Bonus points

Find the number of visits ( $V_k$ ) of the Application server using the given data

### Solution:

According to the question, a request that arrives to the Application server is executed 5 times in the Database. This implies that

$$S_{\text{ApplicationServer}} = 5 * (S_{\text{Database}})$$

The text does not imply the next formula.  
There is not any reason for which the service time of the ApplicationServer does not need to be 5 times the service time of the DB.

According to the question, the service time ( $S_k$ ) of the Database is 0.045s.

Therefore,  $S_{\text{ApplicationServer}} = 5 * 0.045 = 0.22$

### Utilization Law

$$U_k = D_k * X \Rightarrow D_k = U_k / X$$

$$D_k = 0.528 / 1.2 = 0.44$$

### Using Service Demand Law,

$$V_k = D_k / S_k = 0.44 / 0.22$$

$$V_k = 2$$

Hence, visits of application server is 2.

End