Simulation and Modeling Technique Coursework SMT CW

Thusharkanth\_20233168\_2410213

2025-04-11

# — Model Development —

# loading the nessary librearies   
  
library(simmer)

## Warning: package 'simmer' was built under R version 4.4.2

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:simmer':  
##   
## select

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.4.1

# Ensures consistent random values for reproducibility  
  
set.seed(2002)

# declearting the varibles/Parameters   
  
simulation\_duration <- 480 # Total simulation time in minutes (8 hours = 480 min)  
customer\_arrival\_rate <- 10 / 60 # 10 customers per hour = 1 every 6 mins ( Poisson distribution)  
average\_service\_time <- 5 # Avg service time per customer in minutes ( Exp distribution)

# — CUSTOMER TRAJECTORY —

customer\_flow <- trajectory("Customer Flow") %>%  
 seize("service\_counter", 1) %>% # Take one service counter  
 timeout(function() rexp(1, 1 / average\_service\_time)) %>% # Service time (randomly generated)  
 release("service\_counter", 1) # Release counter after service

# — FUNCTION: Run Simulation —

# Defining a Function for Simulating the bank system for a given number of counters  
  
simulate\_bank\_service <- function(number\_of\_counters) {  
   
 # Create a simulation environment  
   
 bank\_simulation <- simmer(paste("Bank ", number\_of\_counters, "Counter(s)")) %>%  
 add\_resource("service\_counter", capacity = number\_of\_counters) %>% # Add counters  
 add\_generator("Customer", customer\_flow, function() rexp(1, customer\_arrival\_rate)) %>% # Add custmr  
 run(until = simulation\_duration) # Run sim  
   
 # Collect logs  
   
 resource\_logs <- get\_mon\_resources(bank\_simulation) # Logs about counter usage  
 arrival\_logs <- get\_mon\_arrivals(bank\_simulation) # Logs about each customer  
   
 # Calculate customer waiting times  
   
 arrival\_logs$total\_waiting\_time <- arrival\_logs$end\_time - arrival\_logs$start\_time # Total time in sys  
 arrival\_logs$pure\_waiting\_time <- arrival\_logs$total\_waiting\_time - arrival\_logs$activity\_time #Only queue time  
   
 # Calculate key performance metrics  
 avg\_total\_wait <- mean(arrival\_logs$total\_waiting\_time, na.rm = TRUE) # Includes service time  
 avg\_actual\_wait <- mean(arrival\_logs$pure\_waiting\_time, na.rm = TRUE) # Just waiting time  
 avg\_queue\_length <- mean(resource\_logs$queue, na.rm = TRUE) # Avg number in queue  
   
   
 # Server utilization (how much busy the counters were)  
 total\_time\_busy <- sum(arrival\_logs$activity\_time, na.rm = TRUE)  
 total\_time\_available <- simulation\_duration \* number\_of\_counters  
 server\_utilization <- total\_time\_busy / total\_time\_available  
   
 # PRINT RESULTS -----  
   
 cat("\n--- Results:", number\_of\_counters, "Counter(s) ---\n")  
   
 cat("Avg Waiting Time (including service time):", round(avg\_total\_wait, 2), "mins\n")  
   
 cat("Avg Waiting Time (only in queue, excluding service):", round(avg\_actual\_wait, 2), "mins\n")  
  
 cat("Average Queue Length:", round(avg\_queue\_length, 2), "\n")  
  
 cat("Server Utilization:", round(server\_utilization \* 100, 2), "%\n")  
  
  
  
   
 # Return all useful outputs  
   
 return(list(  
 resource\_logs = resource\_logs,  
 arrival\_logs = arrival\_logs,  
 avg\_queue\_length = avg\_queue\_length,  
 avg\_total\_wait = avg\_total\_wait,  
 avg\_actual\_wait = avg\_actual\_wait,  
 utilization = server\_utilization  
 ))  
}

# — RUN SIMULATIONS FOR 2 COUNTERS —

results\_counter\_2 <- simulate\_bank\_service(2) # Simulate with 2 counters

##   
## --- Results: 2 Counter(s) ---  
## Avg Waiting Time (including service time): 6.13 mins  
## Avg Waiting Time (only in queue, excluding service): 1.45 mins  
## Average Queue Length: 0.3   
## Server Utilization: 45.38 %

# — RUN SIMULATIONS FOR 3 COUNTERS —

results\_counter\_3 <- simulate\_bank\_service(3) # Simulate with 3 counters

##   
## --- Results: 3 Counter(s) ---  
## Avg Waiting Time (including service time): 5.47 mins  
## Avg Waiting Time (only in queue, excluding service): 0 mins  
## Average Queue Length: 0.01   
## Server Utilization: 33.4 %

# — RUN SIMULATIONS FOR 1 COUNTERS —

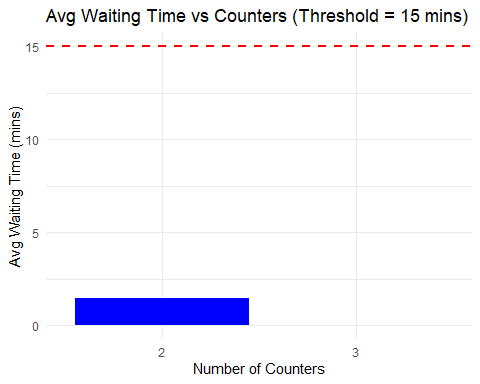
results\_counter\_1 <- simulate\_bank\_service(1) # Simulate with 1 counter

##   
## --- Results: 1 Counter(s) ---  
## Avg Waiting Time (including service time): 8.42 mins  
## Avg Waiting Time (only in queue, excluding service): 4.13 mins  
## Average Queue Length: 0.7   
## Server Utilization: 66.22 %

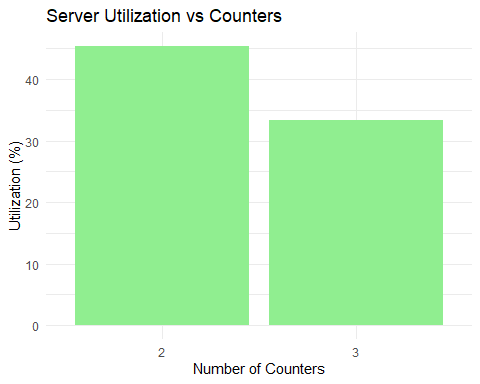
# — VISUALIZATION OF COUNTER 2 AND 3 —

# Create DataFrame for 2 and 3 counters  
  
df\_23 <- data.frame(  
 Counters = c(2, 3),  
 Avg\_Queue\_Length = c(results\_counter\_2$avg\_queue\_length, results\_counter\_3$avg\_queue\_length),  
 Avg\_Waiting\_Time = c(results\_counter\_2$avg\_actual\_wait, results\_counter\_3$avg\_actual\_wait),  
 Utilization = c(results\_counter\_2$utilization, results\_counter\_3$utilization)  
)  
  
# Bar Chart: Avg Waiting Time with 15-min Threshold  
ggplot(df\_23, aes(x = factor(Counters), y = Avg\_Waiting\_Time)) +  
 geom\_bar(stat = "identity", fill = "blue") +  
 geom\_hline(yintercept = 15, color = "red", linetype = "dashed", size = 1) +  
 labs(title = "Avg Waiting Time vs Counters (Threshold = 15 mins)",  
 x = "Number of Counters", y = "Avg Waiting Time (mins)") +  
 theme\_minimal()

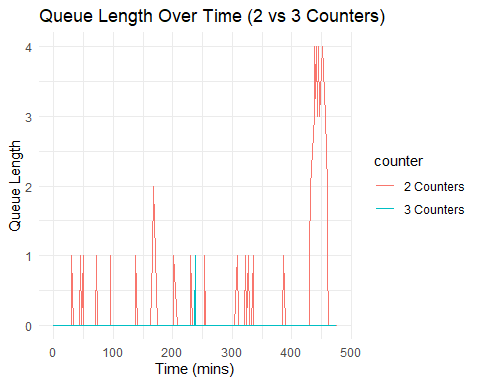
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.



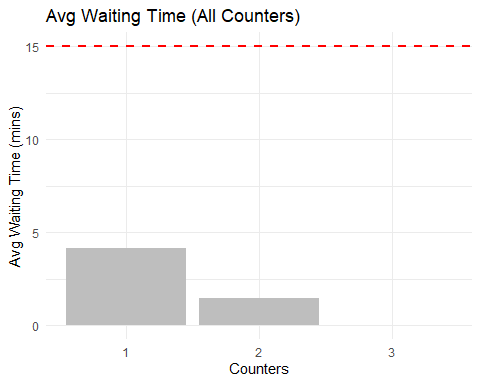
# Bar Chart: Server Utilization  
ggplot(df\_23, aes(x = factor(Counters), y = Utilization \* 100)) +  
 geom\_bar(stat = "identity", fill = "lightgreen") +  
 labs(title = "Server Utilization vs Counters", x = "Number of Counters", y = "Utilization (%)") +  
 theme\_minimal()



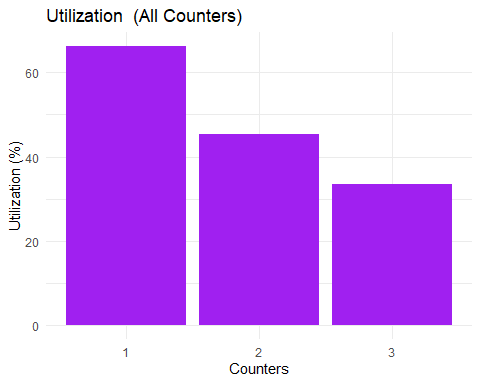
# Line Chart: Avg Queue Length Over Time  
r2 <- results\_counter\_2$resource\_logs  
r3 <- results\_counter\_3$resource\_logs  
r2$counter <- "2 Counters"  
r3$counter <- "3 Counters"  
queue\_df <- rbind(r2, r3)  
  
ggplot(queue\_df, aes(x = time, y = queue, color = counter)) +  
 geom\_line(size = 0.5) +  
 labs(title = "Queue Length Over Time (2 vs 3 Counters)",  
 x = "Time (mins)", y = "Queue Length") +  
 theme\_minimal()

# — VISUALIZATION OF COUNTER 1, 2 AND 3 —

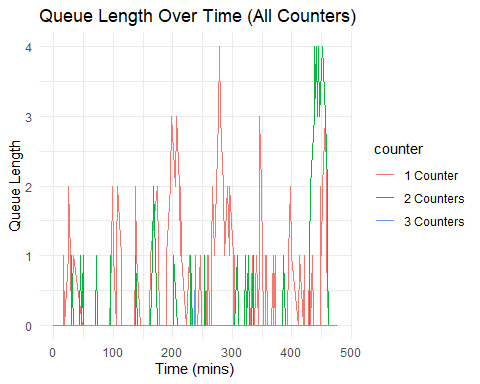
# Combined Data for all 3  
  
df\_all <- data.frame(  
 Counters = c(1, 2, 3),  
 Avg\_Queue\_Length = c(results\_counter\_1$avg\_queue\_length,  
 results\_counter\_2$avg\_queue\_length,  
 results\_counter\_3$avg\_queue\_length),  
 Avg\_Waiting\_Time = c(results\_counter\_1$avg\_actual\_wait,  
 results\_counter\_2$avg\_actual\_wait,  
 results\_counter\_3$avg\_actual\_wait),  
 Utilization = c(results\_counter\_1$utilization,  
 results\_counter\_2$utilization,  
 results\_counter\_3$utilization)  
)  
  
# Waiting Time with Threshold  
ggplot(df\_all, aes(x = factor(Counters), y = Avg\_Waiting\_Time)) +  
 geom\_bar(stat = "identity", fill = "gray") +  
 geom\_hline(yintercept = 15, color = "red", linetype = "dashed", size = 1) +  
 labs(title = "Avg Waiting Time (All Counters)", x = "Counters", y = "Avg Waiting Time (mins)") +  
 theme\_minimal()



# Utilization  
ggplot(df\_all, aes(x = factor(Counters), y = Utilization \* 100)) +  
 geom\_bar(stat = "identity", fill = "purple") +  
 labs(title = "Utilization (All Counters)", x = "Counters", y = "Utilization (%)") +  
 theme\_minimal()



# Queue Length Over Time – All  
r1 <- results\_counter\_1$resource\_logs  
r1$counter <- "1 Counter"  
queue\_all <- rbind(r1, r2, r3)  
  
ggplot(queue\_all, aes(x = time, y = queue, color = counter)) +  
 geom\_line() +  
 labs(title = "Queue Length Over Time (All Counters)",  
 x = "Time (mins)", y = "Queue Length") +  
 theme\_minimal()



# --- CONCLUSION ---  
cat("\n\*\*\*\*\*\*\*\*\*\* CONCLUSION \*\*\*\*\*\*\*\*\*\*\n")

##   
## \*\*\*\*\*\*\*\*\*\* CONCLUSION \*\*\*\*\*\*\*\*\*\*

if (results\_counter\_2$avg\_actual\_wait <= 15) {  
 cat(" 2 counters are ENOUGH.\n Avg wait time is acceptable.\n")  
} else {  
 cat(" 2 counters are NOT enough. Consider increasing.\n")  
}

## 2 counters are ENOUGH.  
## Avg wait time is acceptable.

# Analysis and Interpretation

The simulation was run for the bank operation with 1,2,3 counters respectively with the given details in the question.

Key performance matrix are summarized below

|  |  |  |  |
| --- | --- | --- | --- |
| **Metrix** | **1Counter** | **2 counter** | **3 counter** |
| **Avg Wait Time (Queue)** | |  | | --- | | 4.13 mins | | |  | | --- | | 1.45 mins | | |  | | --- | | 0 mins | |
| **Avg Queue Length** | |  | | --- | | 0.7 | | |  | | --- | | 0.3 | | |  | | --- | | 0.01 | |
| **Server Utilization** | |  | | --- | |  |  |  | | --- | | 66.22% | | 45.38% | 33.4 % |

Under the current 2-counter system, the average-wait-time is well below the 15-min thresholld, and queue length remains miniimal. Utilization is moderate, suggesting the sysaem handlles currrent demand efficiiently.

an additiional 3rd counter reduces waiting time and queue length even more but causes utilizatiion to falls to 33.4%, indicating potentiial underutiilization of resouirces. Whiile perforrmance is relatiively imprioved, the additiional counter may not be justified under present circumsttances.

2 counters, then, represent the best compromiise between operatiing effiiciency and qualiity of customer serviice. A thiird counter woulld be needed  only if future trafffic shows a signifiicant increasse.

Also compared with 1 cuntor in oder to check its performance , which clear state that 1 cuntoer is not enough and have a high queue, which suggest for having another counter

# Visualization and Reporting

Visual outputs were generatted to support the analyssis and comparre the systtem under diffferent counter.

A summmary of the key visualls is presentd below:

|  |  |
| --- | --- |
| **Visualization** | **Insights** |
| Bar Chart: Avg Waiting Time | Both 2 and 3 counters stay below the 15-min threshold. 3 counters remove queue completely, but 2 is still sufficient. |
| Bar Chart: Server Utilization | Utilization drops as counters increase. 2 counters show optimal balance between load and idle time. |
| Line Chart: Queue Length Over Time | Queue length fluctuates slightly for 2 counters, remains flat for 3. Shows that queues are minimal in both. |

Visualizatiions confiirm that 2 countes are suffiicient under curent trafic. The bar chars highligt acceptble waiiting tiimes and server utiilization for 2 counters, and line graps confirm that quaues are small across the simulation.  
  
Introducing a 3rd counter improves the metics but leads to underutlization. Since the marginal gains are not signifiicantly contribuing to customer experiance, the 2-counter setup remains optiimal.  
  
Overall, the visual data is aliigned with performance datia, affirmming the suggestiion to meintain the current setup unlless increased future damand requires expansion.