Scheduling a Transportation from Hostel to in College Campus

A COURSE PROJECT

System Modelling and Simulation & Data Science

INDUSTRIAL ENGINEERING AND MANAGEMENT

UNDER THE GUIDANCE OF

Dr. Vinay V Panicker (Associate Professor)

Dr. Pradeepmon T G (Assistant Professor)

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, CALICUT

BY : Chandrakant verma,

Gaurav H. Jadhav

Certificate

This is to certify that the Project entitled **"Scheduling a Transportation from Hostel to in College Campus**" is a bonafide record of the project done by **Chandrakant verma** (Roll No.: M240797ME) , **Gaurav H Jadhav** (Roll No.:M241028ME), under my supervision, in partial fulfilment of the requirements for the award of the degree of **Master of Technology** in **Industrial Engineering and Management** from **National Institute of Technology Calicut**, and his work has not been submitted elsewhere for the award of a degree.

Dr. Vinay V. Panicker

Associate Professor
Dept. of Mechanical Engineering

Dr. Pradeepmon T.G.

Assistant Professor
Dept. of Mechanical Engineering

Place: NIT Calicut

Date:

DECLARATION

We hereby declare that except where specific reference is made to the work of others, the contents of this course project are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This course project is our own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Signature:	Signature:
Name : **Chandrakant Verma**	Name: **Gaurav H Jadhav**
Roll No : **M240797ME**	Roll No : **M241028ME**

NIT Calicut 28/11/2024

Abstract

The project titled "Scheduling a Transportation from Hostel to in College Campus" addresses the challenges of overcrowding, high waiting times, and inefficiencies in the current bus system, focusing on creating a more streamlined and student-friendly transportation experience.

Effective transportation management is essential for ensuring smooth mobility and convenience for students. Currently, the transportation system involving two buses at the hostel faces challenges such as high waiting times, overcrowding at boarding points, extended trip durations, compromised comfort and safety, and inefficiencies in scheduling. This study focuses on optimizing the scheduling of buses to address these issues and improve transportation services.

Data was collected over multiple days (Monday, Tuesday, Thursday, and Friday) at different times, with any missing information estimated through educated assumptions. The critical parameters analyzed include the buses' arrival times, departure times, waiting times, trip durations, number of trips per day, and the maximum capacity of students per bus. The current scenario revealed several inefficiencies, including long waiting times, overcrowding, and compromised safety.

Our proposed solution involves refining the bus schedule to stagger arrivals and departures, thereby reducing waiting times and overcrowding while ensuring efficient transportation. By optimizing bus schedules and aligning services with student needs, the revised system aims to enhance student satisfaction through better comfort, reduced delays, and improved overall safety. The study underscores the importance of data-driven scheduling to achieve a more efficient and student-friendly transportation system.

CONTENTS

Introduction

1.1. Introduction

In the dynamic environment of hostel transportation, where timely and efficient commutes are essential, the scheduling of buses plays a pivotal role in ensuring student comfort and satisfaction. Our project delves into the intricacies of bus operations, aiming to optimize arrival and departure schedules to reduce overcrowding, minimize waiting times, and enhance the overall efficiency of the transportation system.

1.2. Problem Statement

In NITC Campus there are two buses scheduled for transportation where both buses arrive at NITC hostel randomly, causing overcrowding and congestion as all students attend to board the buses at same time. This situation results insufficiencies, delays and discomfort.
Schedule transportation of bus with lower waiting time, less overcrowding and more efficient transportation with comfort.

Project Concept

Optimizing the scheduling of hostel buses involves analyzing and modeling various parameters to ensure efficient transportation, minimize overcrowding, and improve student satisfaction. Here's a conceptual outline for the project:1. 1.Project Overview:

The project aims to create a refined schedule for hostel buses that reduces waiting time, prevents overcrowding, and ensures efficient transportation with enhanced comfort and safety.

2. System Description:

The system includes two buses (B1 and B2) operating from the hostel, where the current random arrival pattern results in inefficiencies such as overcrowding, extended waiting times, and delays.

3. Data Collection and Inputs:

Data was collected over multiple days (Monday, Tuesday, Thursday, and Friday) and at various times during the day to capture the bus arrival and departure patterns, student counts, and trip durations.

Missing or incomplete data was estimated using logical approximations to ensure accuracy.

4. Modeling Approach:

A structured simulation model was developed to stagger bus arrivals and departures, addressing the inefficiencies of the current system.

The model considers critical parameters such as bus arrival times, waiting times, service times, and the number of students per trip to optimize scheduling.

5. Simulation Parameters:

Key parameters analyzed include bus arrival times, waiting durations, departure times, trip durations, the number of trips per day, and maximum student capacity per trip.

The refined schedule focuses on reducing congestion, ensuring safety, and improving the overall transportation experience for students

Methodology

Data Collection and Analysis:

A daily data was collected from hostel bus operations to analyze transportation patterns and challenges. Using Python's Pandas library, the data was meticulously sorted and processed to extract key parameters such as arrival times, departure times, waiting times, and service times for both buses. This structured analysis provided valuable insights into the current inefficiencies and informed the development of an optimized scheduling model :

Colum	Colum	Colum	Colum	Colum	Colum	Colum	Column8
Friday	Sr.No.	sequence	B.A.time	B.D.time	wating tim	service tim	no waiting studer
	1	B1	7.3	7.45	15	11	9
	2	B2	7.32	7.46	16	12	4
	3	B1	7.56	8.09	13	14	8
	4	B2	7.52	8.06	14	15	7
	5	B1	8.23	8.38	15	16	13
	6	B2	8.13	8.3	17	17	12
	7	B1	8.52	9.04	12	18	12
	8	B2	8.34	8.49	15	19	6
	9	B1	9.22	9.33	11	14	14
	10	B2	8.53	9.09	16	20	1
	11	B1	9.47	9.59	12	17	8
	12	B2	9.29	9.45	16	19	10
	13	B1	10.12	10.26	14	20	15
	14	B2	10.5	11.01	11	20	6
	15	B1	10.57	11.11	14	18	14
	16	B2	11.4	11.5	10	18	3
	17	B1	11.45	11.59	14	16	2
	18	B2	11.57	12.04	7	22	0
	19	B1	12.11	12.25	14	17	5
	20	B2	12.19	12.33	14	18	3
	21	B1	12.34	12.46	12	20	14
	22	B2	12.48	12.54	6	17	16
	23	B1	12.53	1.05	12	18	8
	24	B2	1.05	1.18	13	19	3
	25	B1	1.19	1.32	13	24	15
	26	B2	1.28	1.37	9	20	6
	27	B1	1.42	1.5	8	21	2
	28	B2	1.52	2.03	11	17	0
	29	B1	2.12	2.26	14	20	8
	30	B2	2.46	2.58	14	21	12
	31	B1	3.4	3.52	10	20	10
	32	B2	3.45	4.02	15	20	12
	33	B1	4.1	4.27	17	20	0
	34	B2	4.31	4.41	10	24	1
	35	B1	5.05	5.16	11	28	4
	36	B2	5.44	5.62	18	21	13
	37	B1	6.23	6.35	14	18	1

Column	Colum	Colum	Colum	Colum	Column	Column
Day	Column1	Column2	Column3	Column4	Column5	Column6
Thursday	Sr.No.		B.A.time	B.D.time	wating time	service time
	1	B1	7.3	7.42	12	10
	2	B2	7.32	7.46	16	12
	3	B1	7.4	7.52	12	13
	4	B2	7.58	8.14	16	15
	5	B1	8.17	8.28	11	14
	6	B2	8.29	8.45	16	17
	7	B1	8.42	8.53	11	16
	8	B2	8.54	9.1	16	19
	9	B1	9.04	9.2	11	12
	10	B2	9.19	9.35	16	20
	11	B1	9.32	9.43	11	17
	12	B2	9.45	10.01	16	19
	13	B1	9.59	10.11	12	20
	14	B2	10.5	11.02	12	19
	15	B1	10.55	11.1	15	18
	16	B2	11.4	11.49	9	19
	17	B1	11.45	11.59	14	16
	18	B2	11.58	12.06	8	23
	19	B1	12.11	12.26	15	15
	20	B2	12.23	12.38	15	17
	21	B1	12.35	12.47	12	19
	22	B2	12.49	12.55	9	18
	23	B1	1	1.13	13	17
	24	B2	1.07	1.22	15	16
	25	B1	1.24	1.36	12	24
	26	B2	1.31	1.4	9	23
	27	B1	1.54	2.01	7	18
	28	B2	1.56	2.04	8	18
	29	B1	2.55	3.09	14	20
	30	B2	2.58	3.12	14	21
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Day	Column	Column2	Column3	Column4	Column5	Column6	Column7
WESNASDAY	Sr.No.		BUS .ARRIVAL.time	BUS.DEPARTURE.time	wating time	service time	no waiting student
	1	B1	7.3	7.47	17	11	7
	2	B2	7.44	8	16	15	6
	3	B1	7.56	8.08	12	13	6
	4	B2	8.15	8.3	15	17	9
	5	B1	8.21	8.3	9	17	11
	6	B2	8.49	9.04	15	19	14
	7	B1	8.47	8.54	7	19	10
	8	B2	9.24	9.32	8	19	8
	9	B1	9.14	9.24	10	19	12
	10	B2	9.51	10.04	13	18	3
	11	B1	9.43	9.55	12	21	6
	12	B2	10.24	10.39	15	18	12
	13	B1	10.19	10.3	11	19	13
	14	B2	10.5	11.02	12	21	2
	15	B1	10.55	11.1	15	18	6
	16	B2	11.4	11.49	9	19	16
	17	B1	11.45	11.59	14	16	6
	18	B2	11.58	12.06	8	23	5
	19	B1	12.11	12.26	15	15	1
	20	B2	12.23	12.38	15	17	0
	21	B1	12.35	12.47	12	19	3
	22	B2	12.49	12.55	9	18	5
	23	B1	1	1.13	13	17	12
	24	B2	1.07	1.22	15	16	18
	25	B1	1.24	1.36	12	24	8
	26	B2	1.31	1.4	9	23	5
	27	B1	1.54	2.01	7	18	13

Data was imported into the excel file like:

trip no.	sequence	B.A.time	B.D.time	wating time	service time	no. waiting student
1	B1	7.30	7.45	15	11	6
2	B2	7.41	7.55	14	13	8
3	B1	7.56	8.06	10	13	7
4	B2	8.08	8.19	9	16	12
5	B1	8.19	8.28	9	15	16
6	7	8.28	8.35	7	16	11
7	9	8.43	8.52	9	16	8
8	7	8.53	9.03	10	17	12
9	9	9.08	9.18	10	18	2
10	7	9.22	9.30	8	17	5
11	9	9.36	9.45	9	17	10
12	7	9.49	9.58	9	18	11
13	9	10.02	10.14	12	19	6
break	break	break	break	break	break	break
14	7	10.50	11.00	10	19	0
15	9	10.55	11.07	12	19	8
break	break	break	break	break	break	break
16	7	11.40	11.47	7	19	18
17	9	11.50	12.01	11	20	6
18	7	12.06	12.18	12	23	0
19	9	12.21	12.33	12	22	3
20	7	12.41	12.56	15	25	0
21	9	12.55	1.10	15	17	5
22	7	1.21	1.30	9	18	12
23	9	1.27	1.37	10	19	8

After the data was imported to Excel python programming was used to extract waiting and service time with other important data from the data.

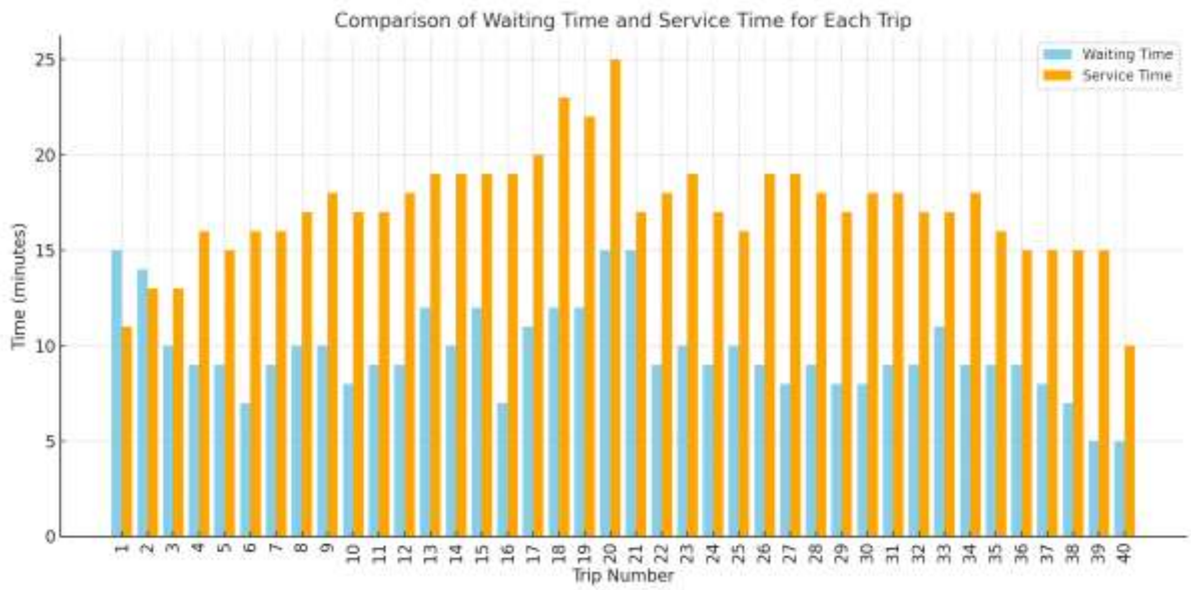
```python
import pandas as pd
import numpy as np
from scipy.stats import norm

# Load the Excel file to inspect its structure
file_path = r"C:\Users\806ch\Downloads\30081910.xlsx"
excel_data = pd.ExcelFile(file_path)

# Display sheet names to understand the structure of the file
excel_data.sheet_names
```

```
['MODAY', 'TUESDAY', 'THUERSDAY', 'FRIDAY', 'proposal']
```

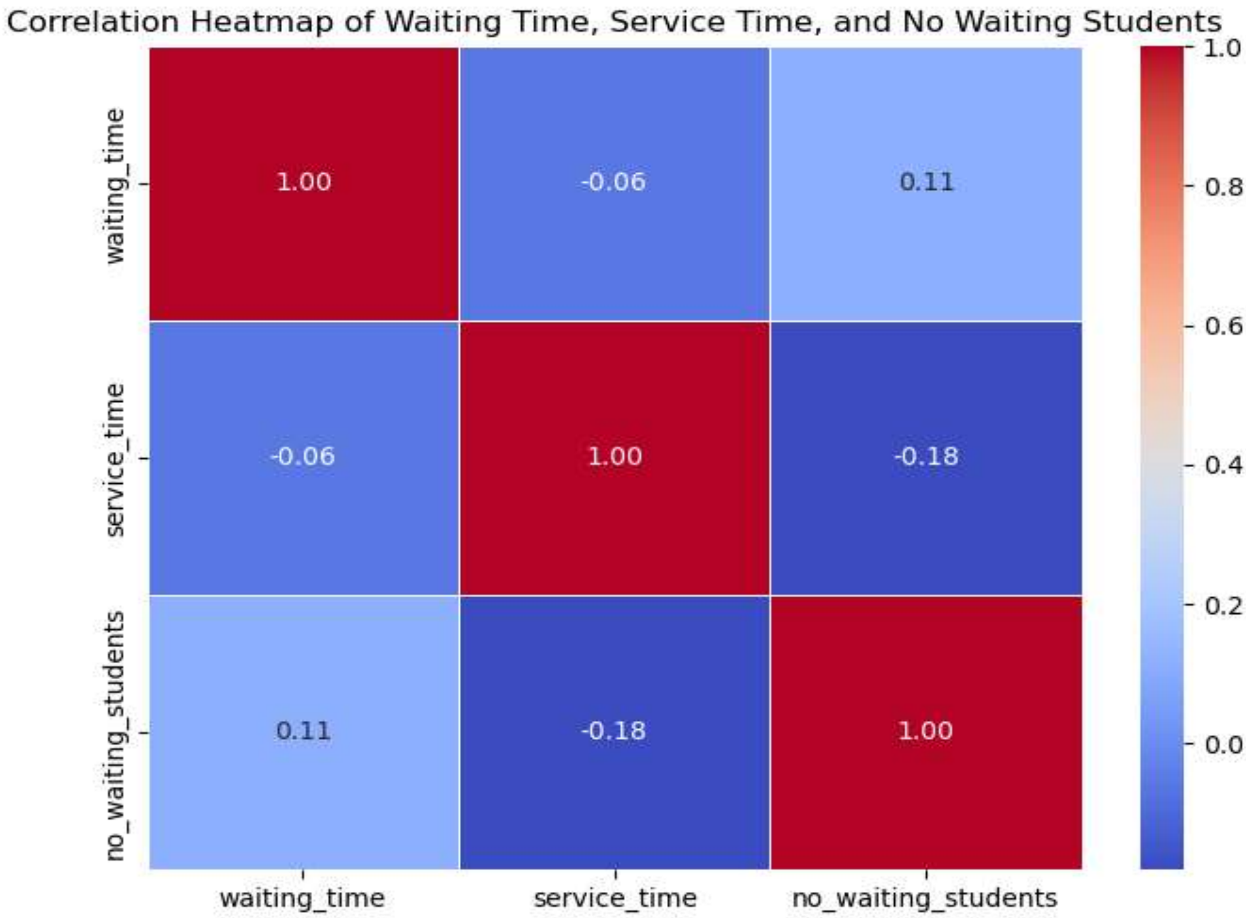
```python
df_fixed = pd.DataFrame(data_fixed)

# Checking structure and identifying issues in the fixed dataset
df_fixed.info(), df_fixed.head(), df_fixed.tail()
import numpy as np

# Replacing "break" and None with NaN for consistency
df_cleaned = df_fixed.replace(["break", None], np.nan)

# Dropping rows where most essential columns are NaN
df_cleaned = df_cleaned.dropna(subset=["trip_no.", "sequence", "B.A.time", "B.D.

# Filling numeric NaN values with 0 (or could use mean/median based on context)
numeric_columns = ["wating time", "service time", "no. waiting student"]
df_cleaned[numeric_columns] = df_cleaned[numeric_columns].fillna(0)

# Converting columns to appropriate data types
df_cleaned["trip_no."] = pd.to_numeric(df_cleaned["trip_no."], errors="coerce")
df_cleaned["wating time"] = pd.to_numeric(df_cleaned["wating time"], errors="coe
df_cleaned["service time"] = pd.to_numeric(df_cleaned["service time"], errors="c
df_cleaned["no. waiting student"] = pd.to_numeric(
    df_cleaned["no. waiting student"], errors="coerce"
)

# Checking the cleaned dataset structure and a sample
df_cleaned.info(), df_cleaned.head(), df_cleaned.tail()
```

```
  trip no. sequence B.A.time B.D.time wating time service time  \
0        1       B1      7.3     7.45          15           11
1        2       B2     7.41     7.55          14           13
2        3       B1     7.56     8.06          10           13
3        4       B2     8.08     8.19           9           16
4        5       B1     8.19     8.28           9           15

  no. waiting student
0                   6
1                   8
2                   7
3                  12
4                  16

Data Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 43 entries, 0 to 42
Data columns (total 7 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   trip no.             43 non-null     object
 1   sequence             43 non-null     object
 2   B.A.time             43 non-null     object
 3   B.D.time             43 non-null     object
 4   wating time          43 non-null     object
 5   service time         43 non-null     object
 6   no. waiting student  43 non-null     object
dtypes: object(7)
memory usage: 2.5+ KB
```

Data visualization:

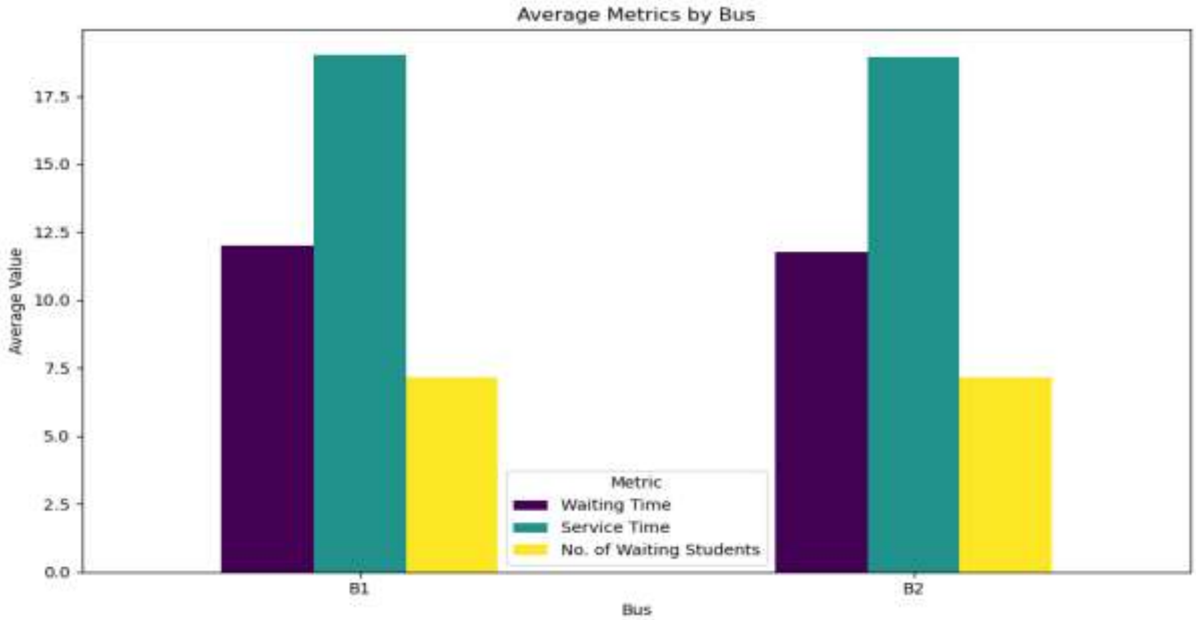


This graph demonstrates that both waiting time and service time are fixed and predictable for each trip, emphasizing regularity and efficiency in the process being measured.

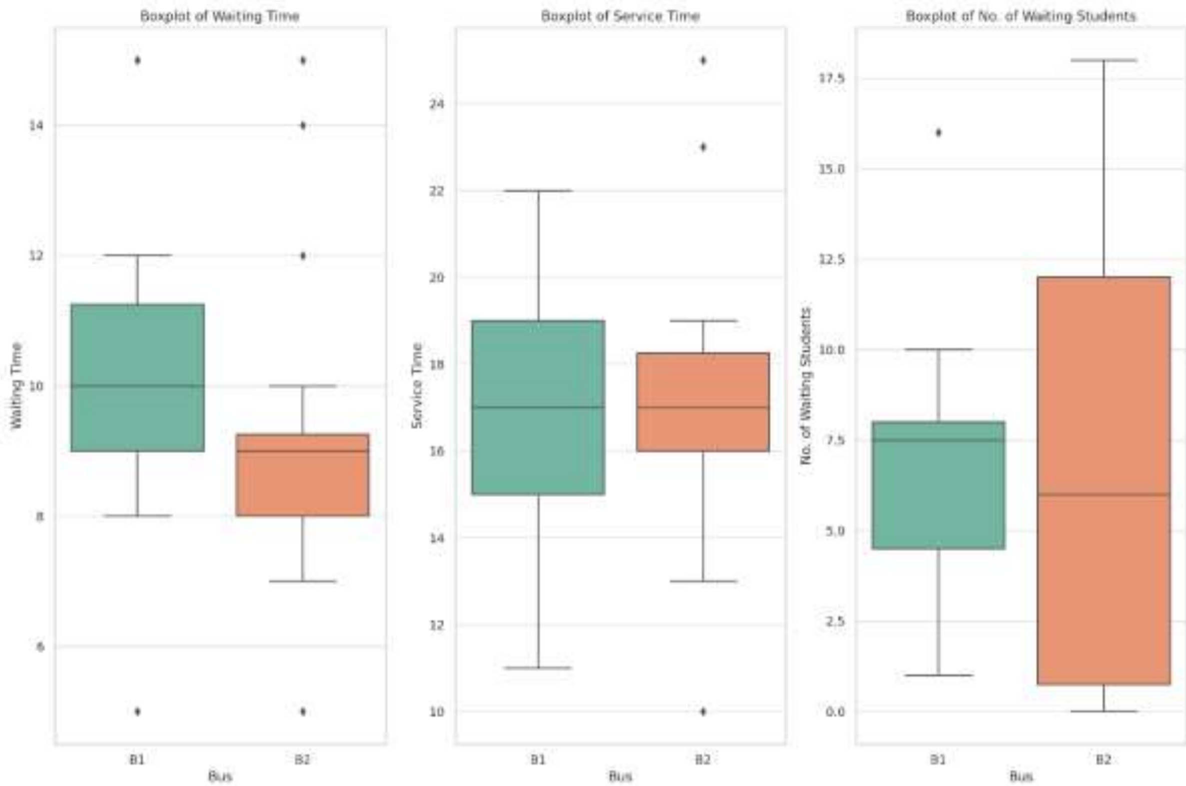


Heat map highlights patterns, peak congestion times, and inefficiencies in the transportation system. For example, it can reveal high waiting times during specific hours or overcrowding at certain stops, helping identify areas for optimization. Such insights enable data-driven decisions to refine schedules, reduce delays, and enhance the overall efficiency and comfort of transportation services.

- we can see both of bus nearly having equal waiting time, trip time, and number of students, which can be concludes no particular one bus is affecting more to increasing overall waiting time and overall trip timing.



We can see both of bus nearly having equal waiting time, trip time, and number of students, which can be concludes no particular one bus is affecting more to increasing overall waiting time and overall trip timing.



Box Plot suggest B1 has higher variability in waiting and service times but lower student counts, potentially indicating better load distribution. B2 has lower waiting times but tends to carry more students, which may lead to overcrowding. This analysis suggests that B1 provides more consistent service but might benefit from reduced waiting times, while B2 could improve by managing overcrowding.

Flow Diagram:

1. Problem Formulation:
   Developing a simulation model to optimize the bus scheduling process, aiming to reduce waiting times, overcrowding, and improve overall transportation efficiency for students.
2. Setting of Objectives and Overall Project Plan:
   The objective of this project is to determine the optimal bus arrival and departure schedule that minimizes waiting times, reduces overcrowding, and enhances the comfort and efficiency of the transportation system.
3. Model Conceptualization:
   The bus scheduling process is divided into two major types of time: a. Waiting Time: The time students spend waiting at the boarding point for the bus to arrive. It is influenced by factors such as bus arrival time and the number of students waiting at the stop. b. Service Time: The time taken by the bus to pick up students, travel to the destination, and drop them off. This is affected by the traffic conditions, the distance between stops, and the bus capacity.

The waiting time is dependent upon factors such as bus arrival intervals, student distribution at different times of the day, and the total number of students needing transportation.

The service time is influenced by the mode of transportation, the number of trips per day, traffic conditions, and the route distance between key stops. The goal is to optimize these times for better operational efficiency and student satisfaction.

4. Data Collection:
   We collected real time data by observing system movement , as bus arrival departure and no of students waiting for the bus.

5. Model Translation:
   The waiting and service times at each bus stop in the transportation process were analyzed based on the number of students and time intervals.
   A relationship was observed where higher student density resulted in longer waiting and service times, while lower density led to quicker boarding and departures. However, the waiting time also depended on how synchronized the bus arrivals were with student schedules.
   Data was further analyzed to identify the intervals of time when overcrowding or delays were most common, and how these intervals impacted the efficiency of the transportation system.
   A database was created using Python's Pandas library and Excel to analyze and store the data. This database helped identify patterns such as the optimal number of buses required, the peak boarding times, and the time distribution across various stops.

trip no.	sequence	Bus.Arrival.time	Bus.Departure.time	wating time	service time
1	B1	7.25	7.30	5	15
2	B2	7.35	7.40	5	15
3	B1	7.45	7.50	5	15
4	B2	7.55	8.00	5	15
5	B1	8.05	8.10	5	15
6	B2	8.15	8.20	5	15
7	B1	8.25	8.30	5	15
8	B2	8.35	8.40	5	15
9	B1	8.45	8.50	5	15
10	B2	8.55	8.60	5	15
11	B1	9.05	9.10	5	15
12	B2	9.15	9.20	5	15
13	B1	9.25	9.30	5	15
14	B2	9.35	9.40	5	15
15	B1	9.45	9.50	5	15
16	B2	9.55	10.00	5	15
17	B1	10.05	10.10	5	15
18	B2	10.15	10.20	5	15
19	B1	10.50	10.55	5	15
20	B2	11.00	11.05	5	15
21	B1	11.45	11.50	5	15
22	B2	11.55	12.00	5	15
23	B1	12.05	12.10	5	15
24	B2	12.15	12.20	5	15
25	B1	12.25	12.30	5	15
26	B2	12.35	12.40	5	15
27	B1	12.45	12.50	5	15
28	B2	12.55	1.00	5	15
29	B1	1.05	1.10	5	15
30	B2	1.15	1.20	5	15

The above is the dataset created which tells the uniform time should

Be followed .

6. Verification of the Model:

Verification focuses on ensuring the model is built correctly and aligns with the conceptual framework. It involves comparing the computer representation of the model with the real-world data.

the verification of the bus scheduling model was carried out by comparing the simulated waiting and service times for the buses (B1 and B2) with the actual data collected from the transportation system. The alignment between the model's output and the real-world observations confirmed that the model accurately reflects the current transportation dynamics, validating its use for optimization and decision-making.

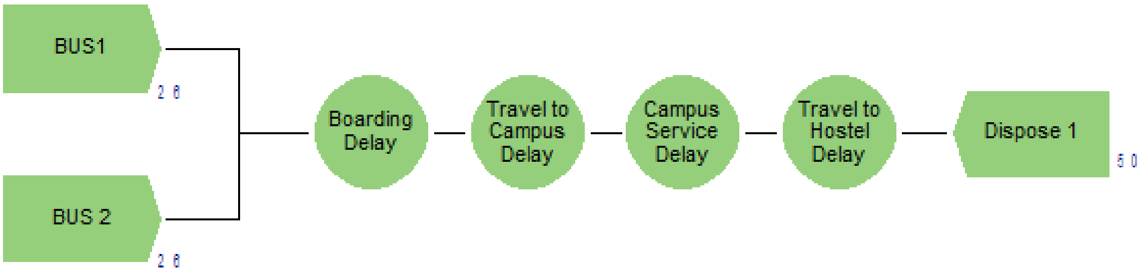


Fig. shows verification of model in Arena.

6. Calibration and Validation:

The validation of the bus scheduling model was a critical aspect of the study, as the logic behind the simulation model incorporates simplified assumptions. Factors such as the number of students at the boarding point, the efficiency of the boarding process, and the frequency of bus arrivals play a significant role in determining waiting and service times. However, since some of these variables were assumed to remain constant during the simulation, the model's behavior aligned partially with the real-world data. This partial validation indicates that while the model captures the core dynamics of the system.

```python
# Function to clean and convert the 'waiting time' column to numeric
def clean_waiting_time(data, column_name):
    # Convert to numeric, coercing errors to NaN, then drop NaNs
    return pd.to_numeric(data[column_name], errors='coerce').dropna()

# Clean the waiting time columns in both datasets
waiting_time_monday1_clean = clean_waiting_time(monday1_data, 'wating time')
waiting_time_proposal_clean = clean_waiting_time(proposal_data, 'wating time')

# Recalculate statistics with cleaned data
mean_monday1 = waiting_time_monday1_clean.mean()
mean_proposal = waiting_time_proposal_clean.mean()
std_monday1 = waiting_time_monday1_clean.std(ddof=1)
std_proposal = waiting_time_proposal_clean.std(ddof=1)

# Sample sizes
n_monday1 = len(waiting_time_monday1_clean)
n_proposal = len(waiting_time_proposal_clean)

# Standard error of the difference in means
se_diff = np.sqrt((std_monday1**2 / n_monday1) + (std_proposal**2 / n_proposal))

# Z-test statistic
z_stat = (mean_proposal - mean_monday1) / se_diff

# Determine the p-value for the test
p_value = norm.cdf(z_stat)  # Left-tailed test

# Decision and conclusion
if z_stat < 0.05:
    conclusion = "Proposed schedule has less waiting time."
else:
    conclusion = "Proposed schedule fails to reduce the waiting time."

mean_monday1, mean_proposal, z_stat, p_value, conclusion
```

```
:  (12.837837837837839,
    5.0,
    -18.054611384576017,
    3.628543775457095e-73,
    'Proposed schedule has less waiting time.')
```

Figure3: Validating The Model In Python

ARENA Simulation Results
IE_LAB_PC1 - License: 2958005287

Output Summary for 25 Replications

Project: Unnamed Project — Run execution date :11/28/2024
Analyst: IE_LAB_PC1 — Model revision date:11/28/2024

OUTPUTS

Identifier	Average	Half-width	Minimum	Maximum	# Replications
Entity 1.NumberIn	26.000	.00000	26.000	26.000	25
Entity 1.NumberOut	25.000	.00000	25.000	25.000	25
Entity 2.NumberIn	26.000	.00000	26.000	26.000	25
Entity 2.NumberOut	25.000	.00000	25.000	25.000	25
System.NumberOut	50.000	.00000	50.000	50.000	25

Simulation run time: 0.03 minutes.
Simulation run complete.

Figure4: Validating The Model In Arena

The validation process was simulating the bus scheduling model on both Arena simulation software and Python. The results from both platforms were compared, and it was observed that the outcomes were consistent. This alignment confirms the accuracy and reliability of the model in replicating real-world scenarios, ensuring that the simulation logic and implementation were correctly applied across both tools.

Result & Analysis

Result and Findings:

Bus B1 showed higher waiting times on average compared to Bus B2, which led to increased congestion for students boarding Bus B1.

Bus B2 had a greater variance in the number of waiting students, indicating inconsistent boarding patterns throughout the day.

Longer service times were observed during peak hours, which caused delays in subsequent trips for both buses.

Recommendation:

During peak hours arrival of both the buses should minimize, no of stations to pick up the students for campus should be increase.stations should plant between two different departments so service time by stopping at each department will decrease.

Conclusion

Conclusion:

In this mini project, we successfully combined data science and simulation techniques up to some extent to model the packing time of the parcels. By identifying the bottlenecks, simulating alternative scenarios, and providing actionable recommendations, the project aims to contribute to the enhancement of India Post's delivery efficiency, ultimately improving customer satisfaction and operational effectiveness. Ongoing monitoring and adaptation are essential to sustaining the benefits of the proposed optimizations in a dynamic delivery environment.

Future Scope:

Simulating Transportation time is a complex process and will involve a machine learning algorithm and mapping the different Offices of India Post. This can be done as a major project.

References

Banks, J., Carson, J.S., Nelson, B. L., and Nicol, D. M., 2014, Discrete-Event System Simulation, Pearson Education.

Altilock, T.and Melamed, B.,2014, Simulation modelling and analysis with Arena,Elesvier