Sri Raghavendra Educational Institutions Society
Sri Krishna Institute of Technology
(Approved by AICTE, Affiliated to VTU, Karnataka)

Industrial Internship Report on

” TRANSPORT MANAGEMENT SYSTEM”

Prepared by

[Rakesh R Amin]

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time. My project was (Tell about ur Project) This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface 3
2 Introduction 4
2.1 About UniConverge Technologies Pvt Ltd 4
2.2 About upskill Campus 8
2.3 Objective 9
2.4 Reference 10

3 Problem Statement 11
4 Existing and Proposed solution 12
5 Proposed Design/ Model 14
5.1 High Level Diagram (if applicable) ..... **Error! Bookmark not defined.**
5.2 Low Level Diagram (if applicable) ..... **Error! Bookmark not defined.**
5.3 Interfaces (if applicable) ..... **Error! Bookmark not defined.**
6 Performance Test 15
6.1 Test Plan/ Test Cases 15
6.2 Performance Outcome 25
7 My learnings 28
8 Future work scope 29

1 Preface

- **Healthcare Data Management**

  A cloud-based solution for managing patient data, medical records, and imaging that can help healthcare organizations to store, access, and share critical information securely and efficiently.

- **Supply Chain Management**

  A cloud-based supply chain management system can help companies to streamline their operations, increase visibility into their supply chain, and make better decisions.

- **Online Education Platform**

  An online education platform is one of the trending cloud computing projects that can provide students with access to high-quality education from anywhere in the world.

- **Smart City Solutions**

  A cloud-based solution for managing a city's infrastructure, public services, and environmental systems that can help to improve the quality of life for residents. This project can be built using platforms like AWS or Microsoft Azure and can include features like smart lighting, traffic management, and waste management.

- **HR Management System**

  A cloud-based solution for managing design HR Management System, the architecture of the system, considering factors such as scalability, security, and accessibility. After that develop the application using a programming language, such as c++.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

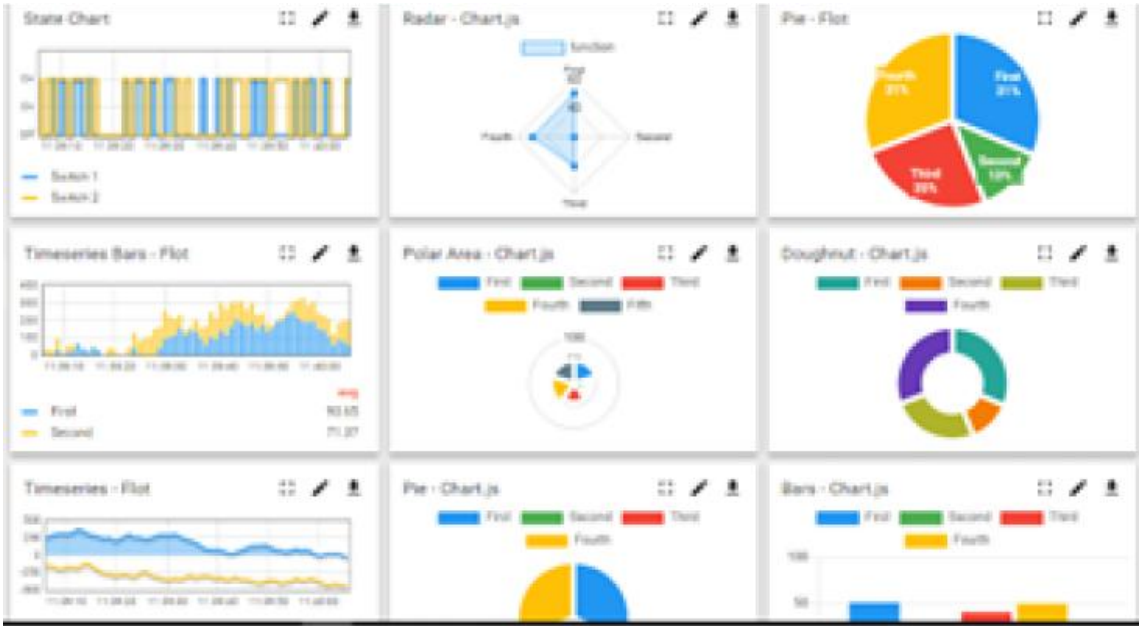
i. UCT IoT Platform (uct Insight)

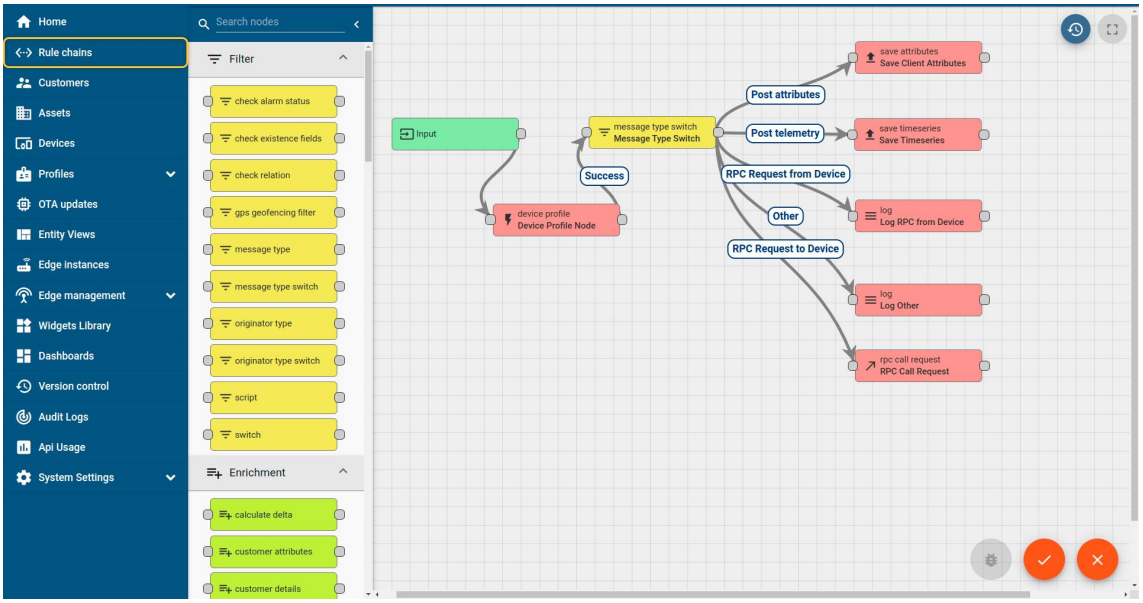
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

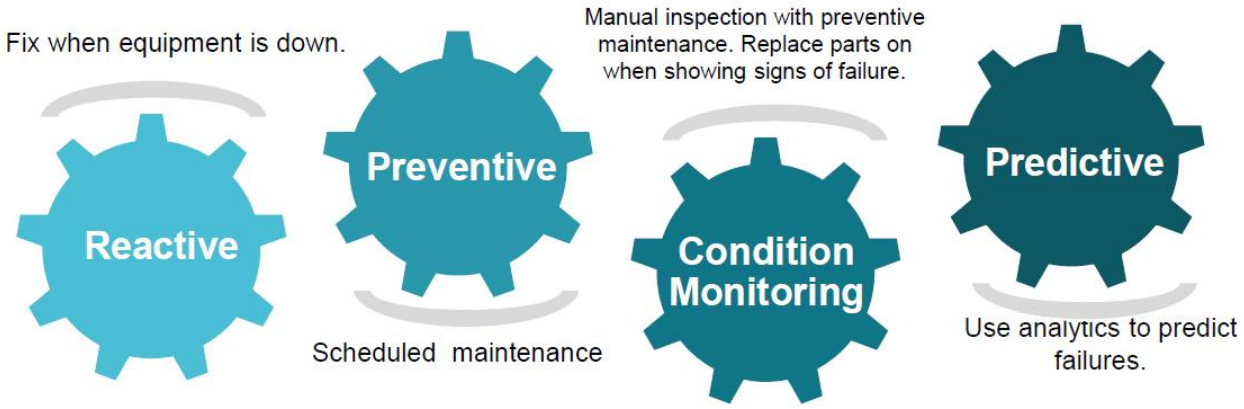
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

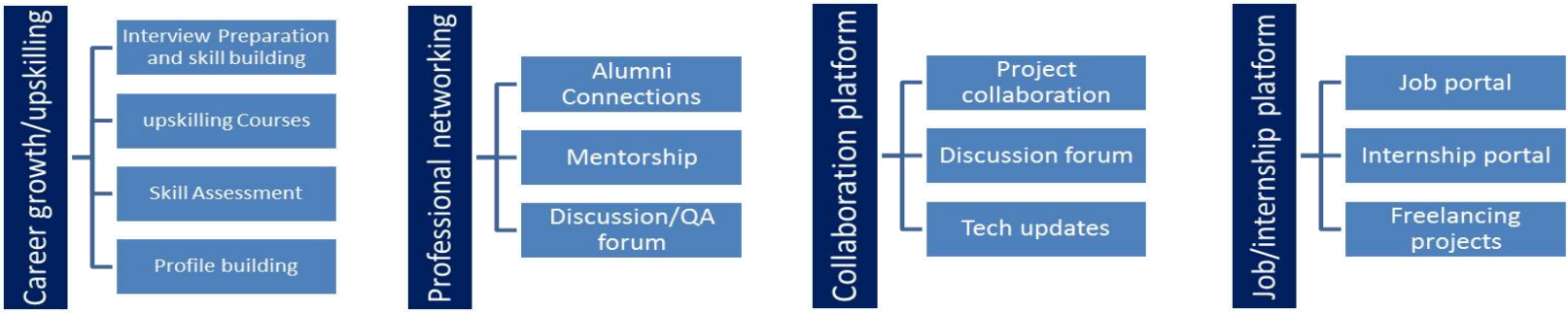
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objective for this internship program was to

- ☛ get practical experience of working in the industry.
- ☛ to solve real world problems.
- ☛ to have improved job prospects.
- ☛ to have Improved understanding of our field and its applications.
- ☛ to have Personal growth like better communication and problem solving.

2.5 Reference

[1] www.Google.com

[2] www.youtube.com

3 Problem Statement

Route Optimization:

How can we optimize transportation routes to reduce fuel consumption, delivery times, and overall operational costs while ensuring timely deliveries?

Real-Time Tracking:

How can we enhance real-time tracking and visibility of goods and vehicles in the supply chain to improve security, efficiency, and customer satisfaction?

Load Balancing:

How can we efficiently balance the load across different vehicles to minimize empty trips and maximize resource utilization?

Demand Forecasting:

How can we develop accurate demand forecasting models to optimize vehicle allocation and inventory management?

Driver Scheduling and Management:

How can we create efficient scheduling systems that consider driver preferences, regulations, and fatigue management while minimizing labor costs?

4 Existing and Proposed solution

Route Optimization:

Implement advanced route optimization algorithms that consider factors like traffic conditions, weather, and delivery time windows. Utilize GPS and real-time traffic data to dynamically adjust routes and provide drivers with updated instructions.

Real-Time Tracking:

Employ GPS tracking devices to monitor vehicle locations and cargo in real-time. Develop a user-friendly dashboard or mobile app for customers and operators to track shipments and receive updates.

Load Balancing:

Use load optimization software to distribute shipments efficiently among available vehicles. Consider implementing a dynamic pooling system to consolidate deliveries and reduce the number of vehicles on the road.

Demand Forecasting:

Use historical data and machine learning models to predict future transportation demands accurately. Integrate demand forecasting with inventory management to optimize stock levels.

Driver Scheduling and Management:

Implement scheduling software that accounts for driver availability, hours of service regulations, and preferences. Use mobile apps to provide drivers with route information, digital logs, and communication tools.

Inventory Management:

Implement just-in-time inventory practices to reduce storage costs and ensure timely restocking. Utilize RFID or barcoding to improve inventory accuracy and visibility.

Environmental Impact:

Introduce hybrid or electric vehicles to the fleet to reduce carbon emissions. Develop eco-friendly driving practices and reward systems for drivers who minimize fuel consumption.

Customer Experience:

Provide customers with a self-service portal for order placement, tracking, and delivery scheduling. Invest in customer service training for staff to handle inquiries and issues effectively.

Data Security:

Implement robust data encryption and access controls to protect sensitive information. Regularly audit and update security protocols to address emerging threats.

Intermodal Transportation:

Develop an integrated TMS that can seamlessly manage different transportation modes. Use intermodal containers and standardized processes to facilitate efficient transfers between modes.

Regulatory Compliance:

Invest in compliance management software to monitor and ensure adherence to transportation regulations. Train staff and drivers on regulatory requirements and best practices.

Maintenance and Vehicle Health:

Implement predictive maintenance systems that use sensor data to schedule maintenance proactively. Conduct regular vehicle inspections and invest in fleet management software.

Cost Management:

Analyze cost data to identify areas for cost reduction, such as fuel-efficient routing and maintenance optimization. Negotiate favorable contracts with suppliers and carriers.

Cross-Border Operations:

Partner with customs brokers and leverage technology to streamline cross-border documentation and clearance processes. Stay updated on international trade regulations and tariffs.

Integration with IoT and AI:

Explore the use of IoT sensors to collect data on vehicle performance, cargo conditions, and driver behavior. Implement AI-driven predictive analytics for demand forecasting and route optimization.

4.1 Code submission (Github link)

https://github.com/jahirul12/Transport_management_system.git

4.2 **Report submission (Github link) :** first make placeholder, copy the link.

https://github.com/jahirul12/Transport_management_system.git

5 Proposed Design/ Model

6 Performance Test

This is very important part and defines why this work is meant of Real industries, instead of being just academic project.

Here we need to first find the constraints.

How those constraints were taken care in your design?

What were test results around those constraints?

Constraints can be e.g. memory, MIPS (speed, operations per second), accuracy, durability, power consumption etc.

In case you could not test them, but still you should mention how identified constraints can impact your design, and what are recommendations to handle them.

6.1 Test Plan/ Test Cases

Source Code

```
#creation of database and table
import mysql.connector as ms
mycon=ms.connect(host='localhost', user='root',passwd='root@123')
cursor=mycon.cursor()
cursor.execute("create database if not exists project")
cursor.execute("use project")
cursor.execute('''create table if not exists transport(name varchar(50),bookingid int(20),contact
decimal(65), taxi varchar(50), pickup varchar(50),dropdown varchar(50),distance int(50), price int(50),
date date)''')
mycon.commit()
mycon.close()
#menu driven program starts here
g='y'
while g=='y':
  print("Welcome to TRANSPORT MANAGEMENT SYSTEM")
  print("1. Enter 1 to add passenger's details")
```

```
  print("2. Enter 2 to search passenger's details")
  print("3. Enter 3 to update passenger's details")
  print("4. Enter 4 to delete passenger's details")
  print("5. Enter 5 to display passenger's details")
  print("6. Enter 6 to exit")
  print()
  a=int(input("PLEASE ENTER YOUR CHOICE==> "))
#taking the user's choice
  if a==1:
    n='y'
    while n=='y':
      import mysql.connector as ms
      mycon=ms.connect(host="localhost",user="root",passwd="root@123",database="project")
      cursor=mycon.cursor()
      name=input("Enter the passenger's name==>")
      bid=int(input("Enter the booking id of the passenger==>"))
      contact=int(input("Enter the passenger's contact number==>"))
      pup=input("Enter the pickup location==>")
      ddn=input("Enter the dropdown location==>")
      distance=float(input("Enter the net distance travelled==>"))
      date=input("Enter the date of travel in the format yyyy-mm-dd==>")
      print("Choose the taxi types")
      print("1.Mini")
      print("2.Micro")
      print("3.Macro")
      print("4.Sedan")
      print("5.Suv")
      print("6.Luxury")
      print("7.Tempo traveller")
      print("8.Ordinary bus")
      print("9.Volvo A/C bus")
      print("10.Volvo sleeper coach A/C bus")
      b=int(input("enter the type of taxi"))
      if b==1:
        taxi="mini"
        price=distance*5
      elif b==2:
        taxi="micro"
        price=distance*6
```

```
        elif b==3:
            taxi="macro"
            price=distance*8
        elif b==4:
            taxi="sedan"
            price=distance*10
        elif b==5:
            taxi="suv"
            price=distance*14
        elif b==6:
            taxi="luxury"
            price=distance*20
        elif b==7:
            taxi="tempo_traveller"
            price=distance*20
        elif b==8:
            taxi="ordinary_bus"
            price=distance*18
        elif b==9:
            taxi="volvo_A/C_bus"
            price=distance*25
        elif b==10:
            taxi="volvo_sleeper_A/C_bus"
            price=distance*30
        cursor.execute('''insert into transport
values('{0}',{1},{2},'{3}','{4}','{5}',{6},{7},'{8}')'''.format(name,bid,contact,taxi,pup,ddn,distance,price,date))
        mycon.commit()
        mycon.close()
        print("data successfully added")
        n=input("Enter y if you want to add more details otherwise enter n (y/n)==>")
        print()
#choice no.1 done
  elif a==2:
     import mysql.connector as ms
     mycon=ms.connect(host="localhost",user="root",passwd="root@123",database="project")
     cursor=mycon.cursor()
     z="y"
     while z=='y':
        bid=int(input("Enter the booking id of the customer whose details have to be displayed==>"))
```

```
        cursor.execute('''select * from transport where bookingid={}'''.format(bid))
        data=cursor.fetchone()
        if data!=None:
            print(data)
            print()
        else:
            print("EMPTY/BOOKING ID DOES NOT EXIST")
            print()
        z=input("Enter y if you want to search again otherwise enter n==>")
#choice no.2 done
  elif a==3:
      import mysql.connector as ms
      mycon=ms.connect(host="localhost",user="root",passwd="root@123",database="project")
      cursor=mycon.cursor()
      z="y"
      while z=='y':
          print("1. Update name")
          print("2. Update contact")
          print("3. Update taxi")
          print("4. Update pickup location")
          print("5. Update drop location")
          print("6. Update total distance travelled")
          print("7. Update date travelled")
          c=int(input("Please enter your choice==>"))
          print()
          if c==1:
              name=input("Enter the modified name==>")
              bid=int(input("Enter the booking id of the customer whose name has to be changed==>"))
              cursor.execute('''select * from transport where bookingid={}'''.format(bid))
              data=cursor.fetchone()
              if data!=None:
                  print("Here are the details of the booking id",bid)
                  print(data)
                  print()
                  a=input("Confirm to update:(y/n)==>")
                  if a=='y':
                      cursor.execute('''update transport set name='{0}' where bookingid={1}'''.format(name,bid))
                      mycon.commit()
                      print("Successfully updated")
```

```
                print()
            else:
                print("Updation cancelled")
        else:
            print("EMPTY/BOOKING ID DOES NOT EXIST")
            print()
    elif c==2:
        contact=int(input("Enter the modified contact number==>"))
        bid=int(input("Enter the booking id of the customer whose contact has to be changed==>"))
        cursor.execute('''select * from transport where bookingid={}'''.format(bid))
        data=cursor.fetchone()
        if data!=None:
            print("Here are the details of the booking id", bid)
            print(data)
            print()
            a=input("Confirm to update:(y/n)==>")
            if a=='y':
                cursor.execute('''update transport set contact={0} where
bookingid={1}'''.format(contact,bid))
                mycon.commit()
                print("Successfully updated")
                print()
            else:
                print("Updation cancelled")
        else:
            print("EMPTY/BOOKING ID DOES NOT EXIST")
            print()
    elif c==3:
        bid=int(input("Enter the booking id of the customer whose taxi type has to be changed==>"))
        cursor.execute('''select distance from transport where bookingid={}'''.format(bid))
        o=cursor.fetchone()
        distance=o[0]
        print("Choose the taxi types")
        print("1.Mini")
        print("2.Micro")
        print("3.Macro")
        print("4.Sedan")
        print("5.Suv")
        print("6.Luxury")
```

```
print("7.Tempo traveller")
print("8.Ordinary bus")
print("9.Volvo A/C bus")
print("10.Volvo sleeper coach A/C bus")
print()
b=int(input("Enter the type of taxi==>"))
print()
if b==1:
   taxi="mini"
   price=distance*5
elif b==2:
   taxi="micro"
   price=distance*6
elif b==3:
   taxi="macro"
   price=distance*8
elif b==4:
   taxi="sedan"
   price=distance*10
elif b==5:
   taxi="suv"
   price=distance*14
elif b==6:
   taxi="luxury"
   price=distance*20
elif b==7:
   taxi="tempo_traveller"
   price=distance*20
elif b==8:
   taxi="ordinary_bus"
   price=distance*18
elif b==9:
   taxi="volvo_A/C_bus"
   price=distance*25
elif b==10:
      taxi="volvo_sleeper_A/C_bus"
      price=distance*30
cursor.execute('''select * from transport where bookingid={}'''.format(bid))
data=cursor.fetchone()
```

```
            if data!=None:
                print("Here are the details of the booking id",bid)
                print(data)
                print()
                a=input("Confirm to update:(y/n)==>")
                if a=='y':
                    cursor.execute('''update transport set taxi='{0}',price={1} where
bookingid={2}'''.format(taxi,price,bid))
                    mycon.commit()
                    print("Successfully updated")
                    print()
                else:
                    print("Updation cancelled")
            else:
                print("EMPTY/BOOKING ID DOES NOT EXIST")
                print()
        elif c==4:
            pup=input("Enter the modified pickup location==>")
            bid=int(input("Enter the booking id of the customer whose pickup location has to be
changed==>"))
            cursor.execute('''select * from transport where bookingid={}'''.format(bid))
            data=cursor.fetchone()
            if data!=None:
                print("Here are the details of the booking id",bid)
                print(data)
                print()
                a=input("Confirm to update:(y/n)==>")
                if a=='y':
                    cursor.execute('''update transport set pickup='{0}' where bookingid={1}'''.format(pup,bid))
                    mycon.commit()
                    print("Successfully updated")
                    print()
                else:
                    print("Updation cancelled")
            else:
                print("EMPTY/BOOKING ID DOES NOT EXIST")
                print()
        elif c==5:
            ddn=input("Enter the modified dropdown location==>")
```

```
            bid=int(input("Enter the booking id of the customer whose dropdown location has to be
changed==>"))
            cursor.execute('''select * from transport where bookingid={}'''.format(bid))
            data=cursor.fetchone()
            if data!=None:
               print("Here are the details of the booking id",bid)
               print(data)
               print()
               a=input("Confirm to update:(y/n)==>")
               if a=='y':
                  cursor.execute('''update transport set dropdown='{0}' where
bookingid={1}'''.format(ddn,bid))
                  mycon.commit()
                  print("Successfully updated")
                  print()
               else:
                  print("Updation cancelled")
            else:
               print("EMPTY/BOOKING ID DOES NOT EXIST")
               print()
         elif c==6:
            distance=float(input("Enter the modified distance==>"))
            bid=int(input("Enter the booking id of the customer whose total distance has to be changed==>"))
            cursor.execute('''select * from transport where bookingid={}'''.format(bid))
            data=cursor.fetchone()
            if data!=None:
               print("Here are the details of the booking id",bid)
               print(data)
               print()
               a=input("Confirm to update:(y/n)==>")
               if a=='y':
                  cursor.execute('''select distance from transport where bookingid={}'''.format(bid))
                  o=cursor.fetchone()
                  ord=o[0]
                  cursor.execute('''select price from transport where bookingid={}'''.format(bid))
                  p=cursor.fetchone()
                  price=p[0]
                  price=(price/ord)*distance
```

```
                cursor.execute('''update transport set distance={0}, price={1} where
bookingid={2}'''.format(distance, price,bid))
                mycon.commit()
                print("Successfully updated")
                print()
            else:
                print("Updation cancelled")
        else:
            print("EMPTY/BOOKING ID DOES NOT EXIST")
            print()
    elif c==7:
        date=input("Enter the modified date of travel in the format yyyy-mm-dd==>")
        bid=int(input("Enter the booking id of the customer whose date of travel has to be changed==>"))
        cursor.execute('''select * from transport where bookingid={}'''.format(bid))
        data=cursor.fetchone()
        if data!=None:
            print("Here are the details of the booking id",bid)
            print(data)
            print()
            a=input("Confirm to update:(y/n)==>")
            if a=='y':
                cursor.execute('''update transport set date='{0}' where bookingid={1}'''.format(date,bid))
                mycon.commit()
                print("Successfully updated")
                print()
            else:
                print("Updation cancelled")
        else:
            print("EMPTY/BOOKING ID DOES NOT EXIST")
            print()
    z=input("Enter y if you want to update anything else otherwise enter n==>")
    print()
#choice no.3 done
  elif a==4:
    import mysql.connector as ms
    mycon=ms.connect(host="localhost",user="root",passwd="root@123",database="project")
    cursor=mycon.cursor()
    z="y"
    while z=='y':
```

```
        bid=int(input("Enter the booking id of the customer whose details has to be deleted==>"))
        cursor.execute('''select * from transport where bookingid={}'''.format(bid))
        data=cursor.fetchone()
        if data!=None:
           print(data)
           print()
           c=input("are you sure you want to delete the details (y/n)")
           if c=='y':
              cursor.execute('''delete from transport where bookingid={}'''.format(bid))
              mycon.commit()
              print("Successfully deleted")
              print()
        else:
           print("EMPTY/BOOKING ID DOES NOT EXIST")
           print()
        z=input("Enter y if you want to delete more details otherwise enter n==>")
        print()
#choice no.4 done
  elif a==5:
     import mysql.connector as ms
     mycon=ms.connect(host="localhost",user="root",passwd="root@123",database="project")
     cursor=mycon.cursor()
     z="y"
     while z=='y':
        cursor.execute('''select * from transport''')
        a=cursor.fetchall()
        if a!=None:
           for i in a:
              print(i)
        else:
           print("EMPTY/NO RECORD TO BE DISPLAYED")
           print()
        z=input("Enter y if you want to display again otherwise enter n==>")
        print()
#choice no.5 done
  elif a==6:
     print("Thank you for using TRANSPORT MANAGEMENT SYSTEM")
     print()
     break
```

```
else:
    print("You have entered wrong choice")
    print()
    g=input("Enter y if you want to continue otherwise enter n (y/n)==>")
    print()
```

6.2 Performance Outcome

```
Enter y if you want to update anything else otherwise enter n==>y

1. Update name
2. Update contact
3. Update taxi
4. Update pickup location
5. Update drop location
6. Update total distance travelled
7. Update date travelled
Please enter your choice==>6

Enter the modified distance==>20
Enter the booking id of the customer whose total distance has to be changed==>2
Here are the details of the booking id 2
('Khushi SU', 2, Decimal('9900069868'), 'sedan', 'Malleswaram', 'Hebbal', 10, 100, datetime.date(2020, 11, 18))

Confirm to update:(y/n)==>y
Successfully updated

Enter y if you want to update anything else otherwise enter n==>y

1. Update name
2. Update contact
3. Update taxi
4. Update pickup location
5. Update drop location
6. Update total distance travelled
7. Update date travelled
Please enter your choice==>7

Enter the modified date of travel in the format yyyy-mm-dd==>2020-11-19
Enter the booking id of the customer whose date of travel has to be changed==>1
Here are the details of the booking id 1
('Jayanth', 1, Decimal('9686725695'), 'ordinary_bus', 'Hebbal', 'Malleswaram', 20, 360, datetime.date(2020, 11, 18))

Confirm to update:(y/n)==>y
Successfully updated
```

```
Enter y if you want to update anything else otherwise enter n==>y

1. Update name
2. Update contact
3. Update taxi
4. Update pickup location
5. Update drop location
6. Update total distance travelled
7. Update date travelled
Please enter your choice==>4

Enter the modified pickup location==>Malleswaram
Enter the booking id of the customer whose pickup location has to be changed==>2
Here are the details of the booking id 2
('Khushi SU', 2, Decimal('9900069868'), 'sedan', 'Yeshwanthpur', 'Malleswaram', 10, 100, datetime.date(2020, 11, 18))

Confirm to update:(y/n)==>y
Successfully updated

Enter y if you want to update anything else otherwise enter n==>y

1. Update name
2. Update contact
3. Update taxi
4. Update pickup location
5. Update drop location
6. Update total distance travelled
7. Update date travelled
Please enter your choice==>5

Enter the modified dropdown location==>Hebbal
Enter the booking id of the customer whose dropdown location has to be changed==>2
Here are the details of the booking id 2
('Khushi SU', 2, Decimal('9900069868'), 'sedan', 'Malleswaram', 'Malleswaram', 10, 100, datetime.date(2020, 11, 18))

Confirm to update:(y/n)==>y
Successfully updated
```

```
Enter y if you want to continue otherwise enter n (y/n)==>y

Welcome to  TRANSPORT MANAGEMENT SYSTEM
1. Enter 1 to add passenger's details
2. Enter 2 to search passenger's details
3. Enter 3 to update passenger's details
4. Enter 4 to delete passenger's details
5. Enter 5 to display passenger's details
6. Enter 6 to exit

PLEASE ENTER YOUR CHOICE==> 4
Enter the booking id of the customer whose details has to be deleted==>1
('Jayanth', 1, Decimal('9686725695'), 'ordinary_bus', 'Hebbal', 'Malleswaram', 20, 360, datetime.date(2020, 11, 19)

are you sure you want to delete the details (y/n)y
Successfully deleted
```

```
Enter y if you want to delete more details otherwise enter n==>n

Enter y if you want to continue otherwise enter n (y/n)==>y

Welcome to  TRANSPORT MANAGEMENT SYSTEM
1. Enter 1 to add passenger's details
2. Enter 2 to search passenger's details
3. Enter 3 to update passenger's details
4. Enter 4 to delete passenger's details
5. Enter 5 to display passenger's details
6. Enter 6 to exit

PLEASE ENTER YOUR CHOICE==> 5
('Khushi SU', 2, Decimal('9900069868'), 'sedan', 'Malleswaram', 'Hebbal', 20, 200, datetime.date(2020, 11, 18))
Enter y if you want to display again otherwise enter n==>n

Enter y if you want to continue otherwise enter n (y/n)==>y

Welcome to  TRANSPORT MANAGEMENT SYSTEM
1. Enter 1 to add passenger's details
2. Enter 2 to search passenger's details
3. Enter 3 to update passenger's details
4. Enter 4 to delete passenger's details
5. Enter 5 to display passenger's details
6. Enter 6 to exit

PLEASE ENTER YOUR CHOICE==> 6
Thank you for using TRANSPORT MANAGEMENT SYSTEM
```

```
mysql> use project;
Database changed
mysql> select * from transport;
+-----------+-----------+------------+-------------+------------+------------+----------+-------+------------+
| name      | bookingid | contact    | taxi        | pickup     | dropdown   | distance | price | date       |
+-----------+-----------+------------+-------------+------------+------------+----------+-------+------------+
| Jayanth   |         1 | 9686725695 | ordinary_bus | Hebbal     | Malleswaram |       20 |   360 | 2020-11-19 |
| Khushi SU |         2 | 9900069868 | sedan       | Malleswaram | Hebbal     |       20 |   200 | 2020-11-18 |
+-----------+-----------+------------+-------------+------------+------------+----------+-------+------------+
2 rows in set (0.00 sec)

mysql> select * from transport;
+-----------+-----------+------------+-------+------------+----------+----------+-------+------------+
| name      | bookingid | contact    | taxi  | pickup     | dropdown | distance | price | date       |
+-----------+-----------+------------+-------+------------+----------+----------+-------+------------+
| Khushi SU |         2 | 9900069868 | sedan | Malleswaram | Hebbal   |       20 |   200 | 2020-11-18 |
+-----------+-----------+------------+-------+------------+----------+----------+-------+------------+
1 row in set (0.00 sec)
```

7 My learnings

It's great that you're interested in learning about Transport Management Systems (TMS). To get started and gain a solid understanding of TMS, you can follow these steps:

Understand the Basics:

Begin by learning the fundamental concepts of transportation management, including the goals, challenges, and key components involved.

Hands-On Experience:

If possible, gain hands-on experience by working with a TMS software or system. Many companies offer internships or entry-level positions in logistics or transportation departments.

Networking:

Connect with professionals in the logistics and transportation industry through networking events, LinkedIn, and industry-specific forums. Engaging with experts can provide valuable insights and opportunities for learning.

Case Studies and Projects:

Study real-world case studies of companies that have successfully implemented TMS solutions. Analyze their challenges, strategies, and outcomes to gain practical insights.

8 Future work scope

The future scope of Transport Management Systems (TMS) is promising, driven by advancements in technology, evolving customer expectations, sustainability concerns, and the need for more efficient transportation operations. Here are some key areas of future growth and development in the TMS industry:

Integration with Emerging Technologies:

Artificial Intelligence (AI) and Machine Learning: TMS will increasingly leverage AI and ML for predictive analytics, demand forecasting, route optimization, and anomaly detection to enhance decision-making and efficiency.

Internet of Things (IoT):

IoT sensors on vehicles and cargo will provide real-time data on vehicle health, cargo conditions, and location, enabling proactive maintenance and improved tracking.

Blockchain:

Blockchain can enhance supply chain transparency, security, and traceability, making it valuable for tracking goods and verifying the authenticity of transactions.

Sustainability and Green Logistics:

TMS will focus on reducing the environmental impact of transportation by optimizing routes for fuel efficiency, promoting the use of electric or hybrid vehicles, and monitoring emissions and carbon footprints.

Autonomous Vehicles:

As autonomous vehicle technology matures, TMS will integrate self-driving trucks and drones for last-mile deliveries, reducing labor costs and improving delivery efficiency.

Customer-Centric Solutions:

TMS will prioritize enhancing the customer experience by providing real-time tracking, delivery time windows, and convenient self-service options, aligning with the growing demand for transparency and convenience.