# ABSTRACT

A queue system is a common solution for managing the flow of customers at a car wash facility. The goal of such a system is to optimize the waiting time for customers and the utilization of resources at the car wash. In this abstract, we will outline the various components and considerations for an effective queue system for a car wash line.

A queue management system that assigns a place in line for each incoming customer A display system to communicate the waiting time and queue status to customers

A ticketing system that assigns a unique identifier to each customer

A communication system to inform employees of the status of the queue and the next customer to be served

The capacity of the car wash and the number of wash bays available The estimated time for each wash cycle and the peak periods of demand

The convenience and accessibility of the queue management system for customers The integration of the queue system with the billing and payment systems

The provision of alternative activities or services for customers while they wait, such as vending machines or seating areas

By implementing a well-designed queue system, car wash facilities can improve the customer experience, reduce wait times, and increase operational efficiency.

# TABLE OF CONTENTS

## Title Page No.

**Abstract i**

## Chapter 1: Introduction 1-3

* 1. Introduction 1
  2. Statement of the problem 1
  3. Objectives 1
  4. Applications 2
  5. Limitations 2-3

## Chapter 2: Design 4-8

* 1. Requirements 4
  2. UML Diagrams 5-8

## Chapter 3: Coding and Result 9-15

* 1. Code 9-13
  2. Result 14-15

## Chapter 4 Conclusion and Future work 16-17

* 1. Conclusion 16
  2. Future work 16-17

# CHAPTER 1 INTRODUCTION

## Introduction:

A queue is a line of people or vehicles waiting for something. Queues are most commonly found in settings such as supermarkets, banks, post offices, and airports, where customers are waiting to be served. In a wash line, a queue is used to manage the flow of customers and ensure that everyone has an equal chance of getting their clothes washed and dried. A queue for a wash line typically consists of several elements. Firstly, there is the line itself, which can be marked with a sign or a barrier. This line serves to provide a visual cue for customers to line up in and clearly shows where the queue begins and ends. Secondly, there is an order of operations which dictates which customer is next in line. This order can be determined by the order in which customers arrive, or it can be set according to the size of the customer’s order or the type of clothing they are washing. Lastly, there is a system of payment. Customers should be given the option to pay in cash or with a credit or debit card. Having a queue for a wash line is important for several reasons. Firstly, it helps to manage the flow of customers, ensuring that everyone has an equal chance of getting their clothes washed. Secondly, it helps to speed up the process, as customers can line up in an orderly fashion and wait their turn without having to worry about being pushed out of the way. Thirdly, it helps to create a more pleasant atmosphere, as customers can take their time to decide what to wash, pay for their order, and move on without feeling rushed.

## Problem statement:

Implement queues for a car wash line (FIFO).

## Objectives:

* + 1. To be viewed as a best car wash in our area.
    2. Provide outstanding customer service and competitive pricing.
    3. Expand to two locations after fifth year of operation

## Applications:

2

* **Customer Service:** By using a queue system, customers can be served on a first- come, first-served basis. This helps to ensure that all customers are treated fairly and reduces the risk of disputes over who should be served next.
* **Efficiency:** Queues can help to streamline the process of washing cars by ensuring that each vehicle is washed in the correct order. This helps to reduce the risk of bottlenecks and improves the overall efficiency of the car wash line.
* **Staff Management:** Queues can help to manage the flow of cars through the car wash, reducing the need for staff to be constantly monitoring the line. This allows staff to focus on other tasks, such as cleaning and maintaining the equipment.
* **Wait Time Management:** Queues can help to manage customer wait times by providing an accurate estimate of the time it will take for each car to be washed. This information can be displayed on screens or communicated to customers in other ways, helping to manage customer expectations and reduce frustration.
* **Data Collection:** Queues can also be used to collect data on customer behavior and preferences, which can be used to improve the overall customer experience and to optimize the car wash line for maximum efficiency.

## Limitations:

* + - **Unpredictable wait times:** Queues are often subject to fluctuations in demand and other unpredictable factors, which can result in long wait times for customers.
    - **Limited capacity:** Queues have a limited capacity, and when demand exceeds the capacity of the queue, customers may be turned away or forced to wait for an extended period of time.
    - **Inefficient use of resources:** Queues can be inefficient, as they may result in idle time for service providers or employees during periods of low demand, while during periods of high demand they may be unable to keep up with demand.
    - **Customer dissatisfaction:** Long wait times and limited capacity can result in customer dissatisfaction, leading to a negative experience for customers and potentially harm to the reputation of the business.
    - **Inability to prioritize customers:** Queues may not have the ability to prioritize customers based on their needs or the type of service they require, leading to a one- size-fits-all approach that may not meet the needs of all customers.
  1. **Requirments:**

# CHAPTER-2 DESIGN

* + - **Efficient queue management:** The system should be able to efficiently manage the queue of cars waiting to be washed, ensuring that each car is served in a timely manner.
    - **Customer information storage:** The system should store information about each customer, including their name, contact information, and payment details. This information can be used to provide personalized service and track customer behavior.
    - **Real-time updates:** The system should provide real.



* + - **Safety:** The queue system should be designed in a way that ensures the safety of both the customers and their vehicles.
    - Efficient Flow: The queue system should be designed to minimize wait times and ensure an efficient flow of vehicles through the car wash line.
    - **Space:** The queue system should be designed to occupy minimal space and ensure maximum utilization of the available area.
    - **Durability:** The queue system should be made of durable materials that can withstand the wear and tear of daily use.

**2.2.UML DIARAMS:**



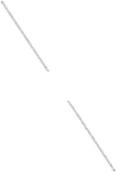
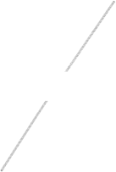
getSalary(salary):void

getPerquiste(perquiste):void

name:string age:int userName:string password:string

employee

generalization



ByMoney

ByTime

payment

inheretance inheretance

## CLASS DIAGRAM:

A class diagram is a visual representation of classes, objects, and their relationships. It's a blueprint that describes the attributes and behaviors of objects in a system. Classes are depicted as boxes with attributes and methods listed inside. Relationships such as inheritance, aggregation, and association are shown with connecting lines and symbols. Class diagrams are used in object-oriented design to define the structure of a system and are a useful tool for communication and documentation. They can help to identify potential problems early in the development process and provide a clear understanding of the system's architecture.

EntercarType(carType):void EntercarColor(carColor):void carNumber(carNumber):void carModel(carModel):void

EntercarProperty(carProperty):void

carType:string carColor:string carNumber:string carModel:string carProperty:string

Carlnformation

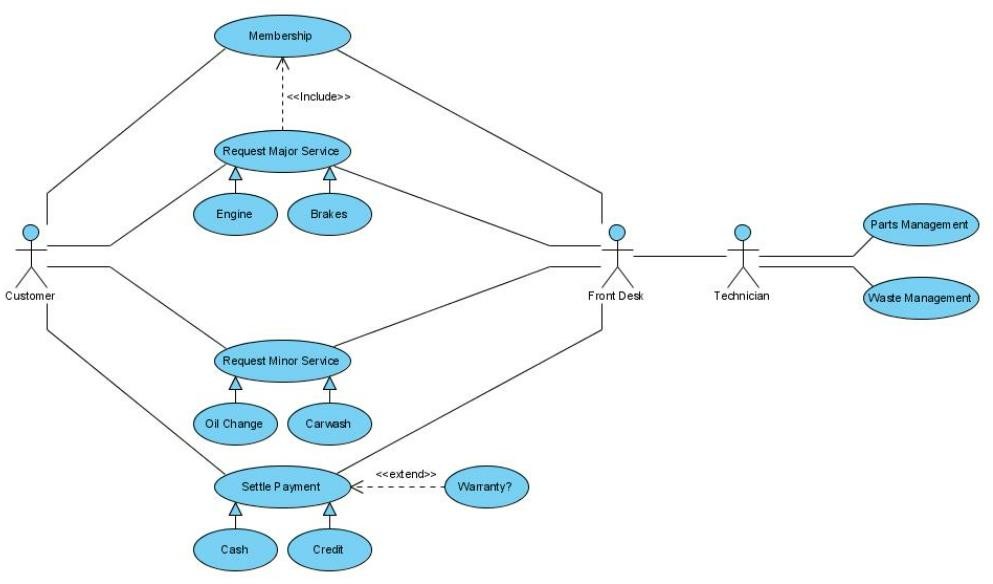
|  |
| --- |
| maneger |
| + name: string  -password: string  -username: string  +birthdate : int  -perquist:double  Salary:double |
| + testCustomer(customername):bool  +getSalary(employee name): void  +managethe money (payment method)  :void  +take money (The amount of money):void  +save in records(Theamount of  money):void |

|  |  |  |  |
| --- | --- | --- | --- |
| customer | |  | |
| +name:string  -password: string  -username: string  +birthdate : int perquist:double | |
| +login(user name, password):void register(name, carinfo): void enterUserName(userName):void enterPassword(userPassword):void | |
|  | |
| Getperquist(perqu | ist):void | | |
| registeration | | d) | |
| customerName:string customerCartype:string customerCarColor:string  FName:string  LName:string  password:string | |
| payment:string  getUserName(customername):void  getUserPassword(customerPassowr  EnterFName(FName):void  EnterLName(LName):void  Enterpassword(password):void  Enterpayment(payment):void | |
|  | | | |
| Login | | |  |
| userName:string  userPassword:string  typeOfWash:string | | |
|  |
| EnteruserName(userName):void  EnteruserPassword(userPassword):void  EntertypeOfWash(typeOfWash):void | | |

## USE CASE DIAGRAM

6

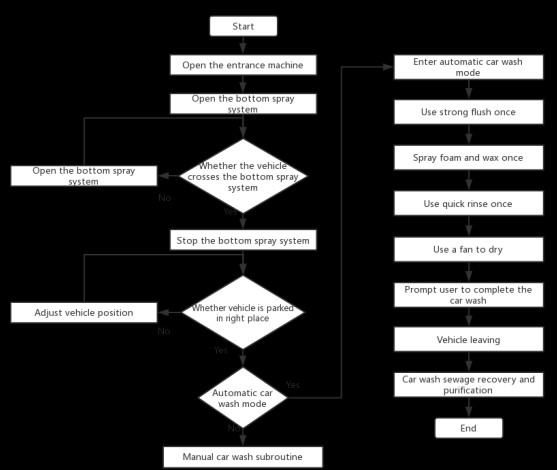
A use case diagram is a type of UML (Unified Modeling Language) diagram that depicts the interactions between a system and its actors in order to achieve a specific goal. Actors are external entities that interact with the system, such as a user or another system, while the system is represented by a rectangle. The use cases, which represent the actions performed by the actors and the system, are depicted as oval shapes. The relationships between actors and use cases are represented by lines, with arrows indicating the direction of the interaction. Use case diagrams provide a high-level view of the functionality provided by a system and are often used during the requirements gathering phase of software development.



## ACTIVITY DIAGRAM

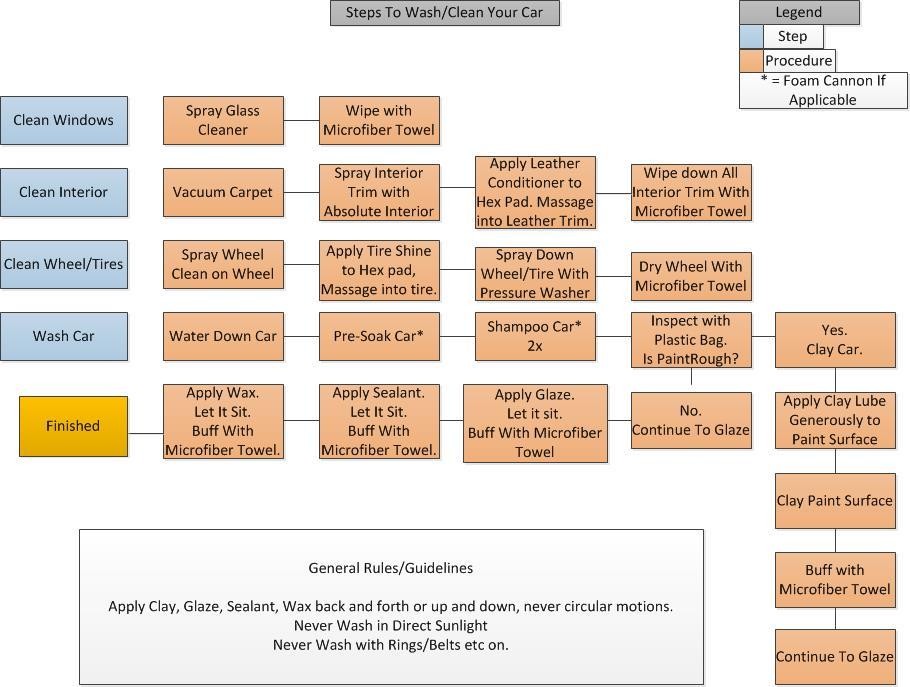
An activity diagram is a type of flowchart that shows the flow of activities and tasks in a process. It is used to represent a step-by-step process and the decisions that need to be made along the way. In an activity diagram, activities are represented by rounded rectangles, while decisions are represented by diamonds. Arrows are used to show the flow of the process, and swim lanes can be used to show the responsibilities of different participants in the process.

Activity diagrams are commonly used in modeling business processes and software development. They provide a visual representation of a process, making it easier to understand and communicate.



## FLOW CHART

A flow chart is a graphical representation of a process, system, or algorithm. It uses symbols and arrows to depict the sequence of steps in a task or procedure, making it easier to understand and follow. Flow charts can be used in a variety of fields, including software development, project management, business process modeling, and problem-solving. They help break down complex processes into manageable parts, enabling effective communication and collaboration among team members. Flow charts also allow for continuous improvement by identifying areas where the process can be streamlined or optimized.



# CHAPTER -3 CODING AND RESULTS

9

**3.1 Code:**

#include<stdio.h> #include<stdlib.h> #define maxsize 10 void in();

void out(); void display();

int rear = -1, front = -1; int queue[maxsize]; void main ()

{

int choice; while(choice != 4)

{

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Car wash Line\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\n======================================================

===========\n");

printf("\n1.Enter the car vehicle number...\n2.Enter the Vehicle Number that need to go Out...\n3.Display the Vehicles in the queue\n4.Exit\n");

printf("\nEnter your choice :");

scanf("%d",&choice); switch(choice)

{

case 1: in(); break;

case 2: out(); break;

case 3: display(); break;

case 4: exit(0); break;

default: printf("\nEnter valid choice ?\n");

}

}

}

void in()

{

int item;

printf("\nEnter the Vehicle Number. \n");

scanf("\n%d",&item); if(rear == maxsize-1)

{

printf("\nVehicles are Full in the Queue\n");

return;

}

if(front == -1 && rear == -1)

{

front=0; rear=0;

}

else

{

rear = rear+1;

}

queue[rear] = item;

printf("\nVehicle number %d Came into the Queue \n ",item);

}

void out()

{

int item;

if (front == -1 || front > rear)

{

printf("\nNo Vehicle number %d in the Queue to go out \n",queue[front]);

return;

}

else

{

item = queue[front]; if(front == rear)

{

front = -1; rear = -1 ;

}

else

{

printf("\nVehicle no %d Crossed the TollGate. \n",queue[front]);

front = front + 1;

}

}

}

void display ()

{

int i;

if (rear == -1)

{ printf("\n No Vehicle in the Queue \n");

}

else

{ printf("\n printing Vehicle Numbers \n");

for(i =front;i<= rear;i++)

{

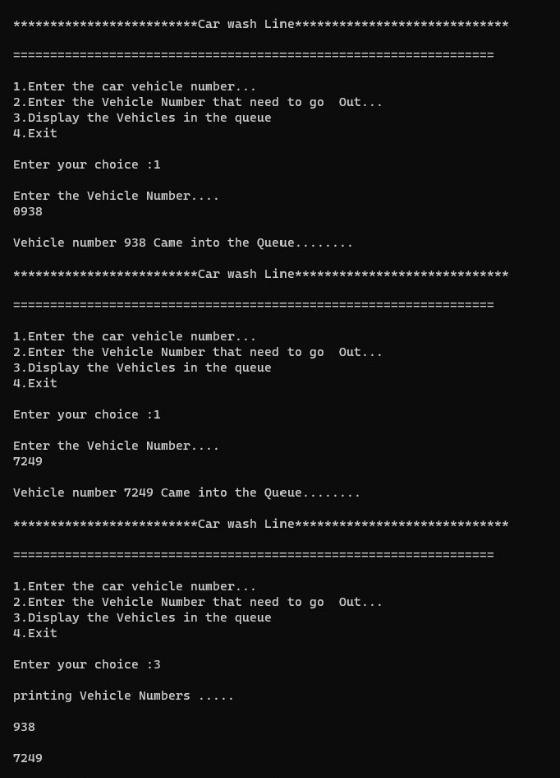
printf("\n %d\n",queue[i]);

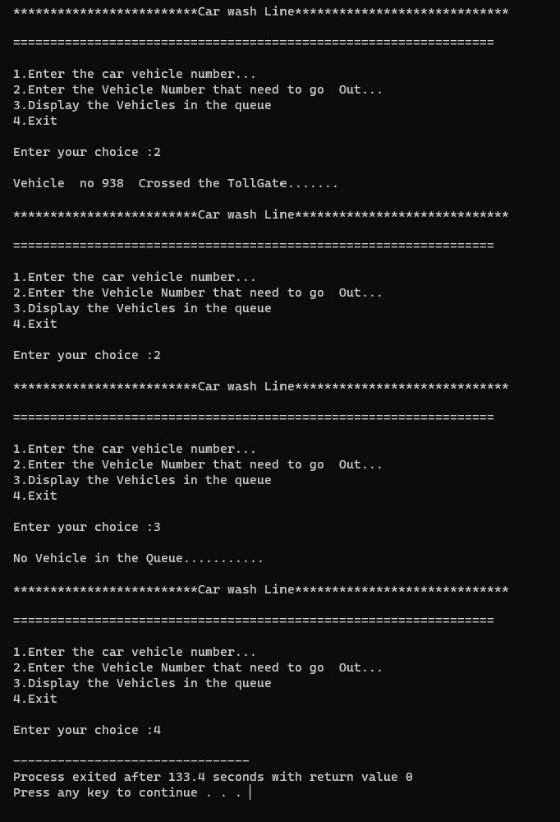
}

}

}

## RESULT:





**CHAPTER -4 CONCLUSION AND FUTURE WORK**

## Conclusion:

A car wash line using queues is an efficient way to manage the flow of vehicles waiting for a wash. By using a queue, cars are processed in the order in which they arrive, reducing wait times and promoting fairness. This system can be implemented using physical markers, such as cones, to designate where cars should line up, or it can be managed digitally through an app or website.

In a car wash line using queues, customers can choose the type of wash they would like, such as a basic wash or a more comprehensive detailing. Once the customer has made their selection, they can then enter the queue and wait for their turn. This can be done through a physical line, or through a virtual line using an app or website.

Advantages of using a queue in a car wash line include increased efficiency, as cars are processed in the order in which they arrive, reducing wait times and promoting fairness. Additionally, the use of a queue can also increase customer satisfaction, as customers are able to see how long they can expect to wait and can make an informed decision on whether they would like to proceed with the wash.

In conclusion, a car wash line using queues is an effective solution for managing the flow of vehicles waiting for a wash. By implementing a queue, cars are processed in a fair and efficient manner, reducing wait times and increasing customer satisfaction. Whether implemented through physical markers or through digital means, a queue system can provide a smooth and seamless experience for customers, making their visit to the car wash more enjoyable.

## FUTURE WORK:

**Predictive queue management:** Predictive algorithms could be used to estimate the wait time for each customer based on real-time data such as the number of cars in the queue, the speed of the wash, and the number of available wash bays. This information could then be

used to optimize the flow of cars through the wash line, reducing wait times and increasing customer satisfaction.

17

**Mobile app integration:** Customers could use a mobile app to join the queue remotely, allowing them to avoid waiting in line. The app could also provide real-time updates on the wait time, allowing customers to make more informed decisions about when to arrive at the car wash.

**Automated payment:** Automated payment systems could be integrated into the car wash process, allowing customers to pay for their wash without having to leave their car. This would streamline the payment process and reduce the time spent waiting in line.

**Smart bays:** Car wash bays could become "smart," using sensors and other technologies to monitor their usage and optimize the flow of cars through the wash line. For example, if a wash bay is free, the next car in the queue could be automatically directed to that bay, reducing wait times.

**Remote monitoring and control:** Car wash operators could use remote monitoring and control systems to monitor the wash line in real-time, making adjustments as needed to optimize the flow of cars and reduce wait times.