A project report

On

**PARKING LOT MANAGEMENT SYSTEM**

Submitted in partial fulfillment of the requirement of

Project-(BIT178CO)

Of

Bachelor of Information Technology

**Submitted to**

****

Purbanchal University

Biratnagar,Nepal

**Submitted By**

SUYOG ADHIKARI (320246)

GAURAB SUBEDI (320229)

SAFAL KOIRALA (320242)

**KANTIPUR CITY COLLEGE**

Putalisadak,Kathmandu

Dec 2, 2018

A project report

On

**PARKING MANAGEMENT SYSTEM**

Submitted in partial fulfillment of the requirement of

Project-I (BITCO)

Of

Bachelor of Information Technology

**Submitted to**

****

Purbanchal University

Biratnagar,Nepal

**Submitted By**

SUYOG ADHIKARI (320246)

GAURAB SUBEDI (320229)

SAFAL KOIRALA (320242)

**Project Supervisor**

**BIKASH NEUPANE**

LECTURER

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

##### Acknowledgement

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. We would like to acknowledge with much appreciation the crucial role of the staff of Kantipur City Collage, who gave us the permission to use all required equipment and the necessary materials to complete the task.

Furthermore, special thanks to our lecturer/ project supervisor, Mr. Bikash Neupane, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report also suggesting us about the task and guiding us during the completion of this project. Finally, many thanks to lab in-charge for providing the facilities of lab during our project. We must appreciate the guidance given by other supervisor as well as the panels especially in our project presentation that has improved our presentation skills, thanks to their comment and advices.

##### Abstract

This project is designed to manage and operate a sample parking lot system. The primary objective of the program is to give the detailed information of available vehicles parking slots and park the vehicles. It stores record of the customers who have parked their vehicle and assures to save time and increase efficiency. By this program it is easy to have a clear management of transactions, customer records and available services.

.

Table of Contents

[Chapter 1 Introduction 1](#_Toc531497294)

[1.1 Background 1](#_Toc531497295)

[1.2 Significance 1](#_Toc531497296)

[1.3 Objectives 1](#_Toc531497297)

[1.4 Organization of project 1](#_Toc531497298)

[Chapter 2 Project Specification 1](#_Toc531497299)

[2.1 Functional Requirements 1](#_Toc531497300)

[2.2 Team structure 1](#_Toc531497301)

[2.3 Implementation Plan 2.3.1 Library Function 1](#_Toc531497302)

[2.4 User-defined Function 2](#_Toc531497303)

[2.5 Class 3](#_Toc531497304)

[2.6 File Structure 3](#_Toc531497305)

[Chapter 3 Software Design and Development 4](#_Toc531497306)

[3.1 Tools and Technologies 4](#_Toc531497307)

[3.2 Algorithm 4](#_Toc531497308)

[3.3 Flowchart 5](#_Toc531497309)

[3.4 Gantt Chart 6](#_Toc531497310)

[Chapter 4 Testing 7](#_Toc531497311)

[Chapter 5 Scope and Future 8](#_Toc531497312)

[Chapter 6 Conclusion 9](#_Toc531497313)

# Chapter 1 Introduction

## 1.1 Background

This document is a report for the design and development of a Vehicle Parking and management system. This system is made in such a way that it maintains a good record of vehicles check in and checkout time. Both two wheeler and four wheeler can be managed by this system and have different pricing system.

## 1.2 Significance

* It helps to reduce the complexity of parking management.
* It can easily be handled by the person who has elementary knowledge.

## 1.3 Objectives

* To enable time management and control of vehicles using recognized vehicle numbers.
* To maintain a listing of entry and exit of vehicles within the parking lot, and determine if the lot is full or not
* To determine the cost per vehicle according to their time consumption.

## 1.4 Organization of project

|  |  |
| --- | --- |
| **Chapters** | **Heading** |
| Chapter 1 | Introduction |
| Chapter 2 | Project Specification |
| Chapter 3 | Software design and development |
| Chapter 4 | Testing |
| Chapter 5 | Conclusion |

# Project Specification

## 2.1 Functional Requirements

|  |  |  |
| --- | --- | --- |
| **S. N.** | **Function Name** | **Description** |
| 1. | entry | Gets number and type info from user |
| 2. | getfreerowcol | Checks for available row and column for the given type |
| 3. | add | Adds the vehicle in free slot of given type and number |
| 4. | storeparkw | Writes the vehicle’s info in file for future references |
| 5. | storeparkr | Reads the stored info of a vehicle |
| 6. | storearrivaltime | Stores the arrival time of the customer |
| 7. | leave | Processes on checking out of the vehicle |
| 8. | calculatetime | Calculates difference between checked in and checked out time |
| 9. | getarrivaltime | Gets arrival time from the system |
| 10. | storedeparttime | Gets depart time from the user |
| 11. | displaytotal | Displays total price and total time the vehicle was stored in |

## 2.2 Team structure

|  |  |  |
| --- | --- | --- |
| **Members Name** | **Symbol Number** | **Task Performed** |
| Suyog Adhikari | 320246 | Designing, Coding and Documentation |
| Gaurab Subedi | 320229 | Designing, Coding and Documentation |
| Safal Koirala | 320242 | Designing, Coding and Documentation |

## 2.3 Implementation Plan 2.3.1 Library Function

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Name of Library function** | **Description** |
| 1. | iostream.h | To use input, output objects like cin, cout |
| 2. | fstream.h | To use file stream |
| 3. | conio.h | Console input output which includes built in function |
| 4. | time.h | Functions for handling interrupts, producing sound, date and time function etc. |
| 5. | graphics.h | For string operations |
| 6. | process.h | To use bill printing feature |
| 7. | stdio.h | To use bill printing feature |

### 

## 2.4 User-defined Function

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Name of User-defined function** | **Description** |
| 1. | option | Shows different available option to the user |
| 2. | frontpage | Displays welcome page |
| 3. | displaymenu | Displays main menu |
| 4. | entry | Checks in information of the vehicle to be parked |
| 5. | getfreerowcol | Gets available row and column for the new vehicle to be parked |
| 6. | add | Adds the new vehicle in the available slot |
| 7. | countw | Counts total vehicles that has been recently parked and stores them in a file |
| 8. | countr | Reads the vehicle counts that were parked |
| 9. | storeparkw | Stores the slot in a file in which the vehicle was parked |
| 10. | storeparkr | Reads the slot from the file where the vehicle was parked |
| 11. | storearrivaltime | Stores the check in time of the vehicle in a file |
| 12. | displaytotal | Display total time and calculates money according to the vehicle type |
| 13. | leave | Asks information of the vehicle that is about to be departed i.e. Number and vehicle type |
| 14. | getvehicle | Checks the given input is in the file or not and if found the processes to depart |
| 15. | getarrivaltime | Gets arrival time from the system and stores in a file |
| 16. | calculatetime | Calculates the difference between departed and arrival time |
| 17. | minus | Subtracts the vehicle counts and writes in a file after departure |
| 18. | storedeparttime | Stores the checked out time of the vehicle |
| 19 | displayorder | Displays the available slots and used slots in the parking lot |

## 2.5 Class

|  |  |
| --- | --- |
| **Class Name** | **Data Types** |
| checkin | char, int, float |
| checkout | Int |
| order | Int |

## 2.6 File Structure

The counts of vehicles i.e. total number of vehicles recently in the slots, and total two or four wheelers in the slot, are stored in file “count.txt”. The slot information is stored in the file “park.txt”. The recorded times and vehicles number during check in are stored in the file named “time.txt”. Whereas the recorded times during checkout are stored in file named “time2.txt”.

# Software Design and Development

## 3.1 Tools and Technologies

Turbo-C++: Compiled

## 3.2 Algorithm

|  |  |
| --- | --- |
| Step 1: | START |
| Step 2: | Display Menu1  “1. Checkin   1. Display Total 2. Available Slots 3. Checkout 4. Exit” |
| Step 3: | Opt1=menu1 |
| Step 4: | If opt1=1 GOTO step 5  else if opt1=2 GOTO step 11  Else if opt1=3 GOTO step 12  Else if opt1=4 GOTO step 20  Else if opt1=5 GOTO step  else GOTO step 2 |
| Step 5: | Display Choose “1. Two Wheeler \n 2. Four Wheeler” |
| Step 6: | available\_slots1=two\_wheeler  available\_slot2=four\_wheeler |
| Step 7: | if available\_slot1>0 GOTO step 8 OR if available\_slot2>0 GOTO step 8  else display “No slots available!!! ” GOTO step 2 |
| Step 8: | Book the slot to park respective vehicle in available slot |
| Step 9: | Assign vehicle’s id with in an empty slot |
| Step 10: | Store vehicle’s data in file and give them printed output GOTO step 2 |
| Step 11: | Display total counts of vehicles that are recently parked in the parking lot GOTO step 2 |
| Step 12: | Display the parking lot that shows available parking spaces and recently parked spaces GOTO step 2 |
| Step 13: | Compare slot-ID with ID and corresponding vehicle in file |
| Step 14: | If data\_match=true GOTO step 15  Else GOTO step 16 |
| Step 15: | Free the given vehicle slot and subtract count from the file GOTO step 2 |
| Step 16: | Display “ID not found, Try Again!!!” and GOTO step 11 |
| Step 17: | STOP |

## 3.3 Flowchart

## 3.4 Gantt Chart

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S.N. | Tasks | Duration | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Concept Submission | 1 week |  |  |  |  |  |  |  |  |
| 2. | Requirement gathering | 2 weeks |  |  |  |  |  |  |  |  |
| 3. | Research and Analysis | 2 weeks |  |  |  |  |  |  |  |  |
| 4. | System Design | 2 weeks |  |  |  |  |  |  |  |  |
| 5. | Coding | 4 weeks |  |  |  |  |  |  |  |  |
| 6. | Testing Debugging | 5 week |  |  |  |  |  |  |  |  |
| 7. | Documentation | 8 weeks |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Tasks Completed** |  |

**Total Time:** 8 weeks

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** | **Status** |
| To entry new parked vehicle | Store vehicle’s info and entry time. | Stored vehicle’s info and entry time. | True |
| To display all the counts of two and four wheelers from the lot | Displays total vehicles parked in a lot. | Displayed total vehicles parked in a lot. | True |
| To find and park vehicle in available slot | Park the vehicle in available space | The vehicle is parked successfully in the free parking space | True |
| To depart the vehicle by input number | To remove the entered vehicle number from the slot and calculate price with time consumption | Vehicle successfully removed and price calculated | True |
| To view history of all entry and depart records | View all arrival and departed records of vehicles | Displayed | True |
| Exit | Exits program | Exit | True |

# Testing

# Scope and Future

This is the modern age. Many people have vehicles. Vehicle is now a basic need. Every place is under the process of urbanization. There are many corporate offices and shopping centers etc. There are many recreational places where people used to go for refreshment. So, all these places need a parking space where people can park their vehicles safely and easily. Every parking area needs a system that records the detail of vehicles to give the facility. These systems might be computerized or non-computerized. With the help of computerized system we can deliver a good service to customer who wants to park their vehicle into the any organization’s premises.

Vehicle parking management system is an automatic system which delivers data processing in very high speed in systematic manner. Parking is a growing need of the time. Development of this system is very useful in this area of field. We can sell this system to any organization. By using our system they can maintain records very easily. Our system covers the every area of parking management. In coming future there will be excessive need of Vehicle parking management system.

# Conclusion

The main objective of the project was to have an efficient system that could save time and manage the record of the parked vehicles in effective way. Also a system that could calculate pricing according to the time consumption or the duration of the vehicle that was parked.

After completion of this project all of these requirements were successfully fulfilled. Records from first parked vehicles are stored and anyone with an elementary knowledge can use the program without any difficulties.

##### References

* Kanetkar, Yashavant “Programming in ANSI C”, BPB Publishing.
* Davin N. Smith “Concepts of object oriented programming”, Green Heaven Publishing.
* Ram Datta Bhatta “C Programming”, VidyarthiPrakashan (P.) Ltd.