A project report

On

**PARKING 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

GAURAB SUBEDI

SAFAL KOIRALA

**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

GAURAB SUBEDI

SAFAL KOIRALA

**Project Supervisor**

**BIKASH NEUPANE**

LECTURER

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

##### Acknowledgement

The project members would like to express our deepest appreciation to our lecturer, Mr.Saroj Pandey for providing proper guidance during the completion of our project.

The project members would like to thank our supervisor, Er.SujanDevkota for supervising, motivating and being co-operative throughout this project work. Without his guidance and persistent help this project would not have been possible. The project members are grateful to all the teachers who had helped us directly and indirectly throughout the project. Finally, project members are indebted to the lab in-charge for providing the facilities of lab during our project.

##### Abstract

This project manages the employs details within a small company or organizations in a systematic manner. It mainly focuses on basic operations such as adding new record, searching, modifying and deleting records. It helps to keep the records safely and saves time. It is easy to search the required details within a short period of time by entering the employee’s name.

Table of Contents

[Chapter 1 Introduction 1](#_Toc485641378)

[1.1 Background 1](#_Toc485641379)

[1.2 Significance 1](#_Toc485641380)

[1.3 Objectives 1](#_Toc485641381)

[1.4 Organization of project 1](#_Toc485641382)

[Chapter 2 Project Specification 2](#_Toc485641383)

[2.1 Functional Requirements 2](#_Toc485641384)

[2.2 Team structure 2](#_Toc485641385)

[2.3 Implementation Plan 2.3.1 Library Function 2](#_Toc485641386)

[2.4 User-defined Function 3](#_Toc485641387)

[2.5Data Structure 3](#_Toc485641388)

[2.6File Structure 3](#_Toc485641389)

[Chapter 3 Software Design and Development 4](#_Toc485641390)

[3.1 Tools and Technologies 4](#_Toc485641391)

[3.2 Algorithm 4](#_Toc485641392)

[3.3 Flowchart 9](#_Toc485641393)

[3.4 Gantt Chart 18](#_Toc485641394)

[Chapter 4 Testing 19](#_Toc485641395)

[Chapter 5 Conclusion 20](#_Toc485641396)

# Chapter 1 Introduction

## 1.1 Background

This project keeps the track of name, age, address, basic salary, contact no., nationality and gender of the employee’s in a systematic manner. In this project, record of employee can be added, listed, modified and deleted. We got the idea of this project from our seniors.

## 1.2 Significance

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

## 1.3 Objectives

* To keep the records of name, age, address, basic salary, contact no., nationality and genderof employee systematically within 2 minutes.
* To reduce searching time.
* To enhance the concept of c programming.

## 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** | **Reflection** |
| 1. | Add | To add new employee | Fig. 3.3.1 |
| 2. | List | To list employee records | Fig. 3.3.2 |
| 3. | Search | To search employee records | Fig. 3.3.3 |
| 4. | Modify | To modify the records of employee | Fig. 3.3.4 |
| 5. | Delete | To delete unnecessary records | Fig. 3.3.5 |
| 6. | About | To know about us | Fig. 3.3.6 |
| 7. | Exit | To exit the program | Fig. 3.3.7 |

## 2.2 Team structure

|  |  |  |
| --- | --- | --- |
| **Members Name** | **Symbol Number** | **Task Performed** |
| Dipesh Limbu | 313779 | Designing, Coding and Documentation |
| SamitaAwale | 313791 | Designing, Coding and Documentation |
| AshmitaBohara | 313773 | Designing, Coding and Documentation |

## 2.3 Implementation Plan 2.3.1 Library Function

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Name of Library function** | **Description** |
| 1. | stdio.h | To input output functions like printf, scanf, etc. |
| 2. | stdlib.h | To allocate and free memory |
| 3. | conio.h | Console input output which includes built in function |
| 4. | dos.h | Functions for handling interrupts, producing sound, date and time function etc. |
| 5. | string.h | For string operations |

### 

## 2.4 User-defined Function

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Name of User-defined function** | **Description** |
| 1. | Add | To add employee ID |
| 2. | Input | To add employee details |
| 3. | List | To list employee records |
| 4. | Search | To search employee records |
| 5. | Check\_id | To check employee ID |
| 6. | Modify | To modify the records of employee |
| 7. | Deleteemp | To delete unnecessary records |
| 8. | Main Page | To go menu |
| 9. | About | To know about us |

## 2.5Data Structure

|  |  |
| --- | --- |
| **Structure Name** | **Data Types** |
| Emp | char, int, float |

## 2.6File Structure

Adding, listing, searching, modifying and deleting the record of an employee is included in "empp.dat" file.

# Software Design and Development

## 3.1 Tools and Technologies

Turbo-C: Compiled

## 3.2 Algorithm

Step 1: Start

Step 2: Display program menu

Step 3: Enter the option

3.1 if option == 1  
3.1.1 Enter to Add Record

3.1.2 if (ID! ==Get.ID)  
Print ID exits

Else

Print record successfully added  
Print add new records? (y/n)

3.1.3 while (ch==y/ch==n)

Go to step 3.1.1

Else

Go to step 2

3.2 if option == 2

3.2.1 Enter tolistrecord

3.2.2 if (ID == Get.ID)

Record found

Else

No response

3.2.3 while (ch == y ch==n)

Go to step 3.2.1

Else

Go to step 2

3.3 if option == 3

3.3.1 Enter ID to search records

3.3.2 if(ID == Get.ID)

Display records

Else

No response

3.3.3 while (ch == y)

Go to step 3.3.1

Else

Go to step 2

3.4 if option == 4

3.4.1 Enter ID to modify records

3.4.2 if(ID == Get.ID)

Record found

Else

No response

3.4.3 while (ch == y)

Go to step 3.4.1

Else

Go to step 2

3.5 if option == 5

3.5.1 Enter ID to delete records

3.5.2 if(ID == Get.ID)

Record found

Else

No response

3.5.3 while (ch == y)

Go to step 3.5.1

Else

Go to step 2

3.6 if option == 6

3.6.1 Enter ID to viewabout us

3.6.2 if(ID == Get.ID)

View about us

Go to step 2

3.7 if option == 7

3.7.1 Enter ID to exit records

3.7.2 if(ID == Get.ID)

Exit program

Step 4: Stop.

## 3.3 Flowchart

False

if( option ==2)

if( option ==1)

Enter your choice

Display Menu

Start

1

2

if( option ==3)

3

a

if( option ==4)

a

4

if( option ==5)

5

if( option ==7)

7

if( option ==6)

6

m

Invalid input Try again

m

if(ch == y || ch == n )

Do you like to add more record?

Record Added

Enter employee details

1

2

Dipsplay Employee Record

m

Enter Any Key to go menu

if( ID == Get ID)

Enter employee ID to search

3

Display Employee Record

True

Do you like to search again?

if ( ch == y )

m

Enter employee ID to modify

True

4

if( ID == Get ID)

Enter new employee details

Employee ID is not found

if ( ch == y )

Do you like to search again?

m

Employee ID is successfully deleted

if( ID == Get ID)

Employee ID is not found

if ( ch == y )

Do you like to search again?

m

Enter employee ID to delete

5

6

Display Employee Record

m

Enter Any Key to go menu

Stop

Exiting in 5 seconds

7

## 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. | Debugging & Testing | 1 week |  |  |  |  |  |  |  |  |
| 7. | Documentation | 3 weeks |  |  |  |  |  |  |  |  |

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

**Total Time:** 15 weeks

# Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** | **Status** |
| To Add Employee Record | Name, Age, Basic Salary, etc. | Added | True |
| To List Employee Record | Name, Age, Basic Salary, etc. | Listed | True |
| To Search Employee Record | Search by ID | Successfully search the record by using employ name | True |
| To Modify Employee Record | Name, Age, Basic Salary, etc. | Updated | True |
| To Delete Employee Record | Employee details | Deleted | True |
| To know about developer | Dipesh Limbu  SamitaAwale  AshmitaBohara | Dipesh Limbu  SamitaAwale  AshmitaBohara | True |
| Exit | Exits program | Exit | True |

# Conclusion

The project members have a great experience designing and implementing the Employee Management System by using C programming language and to work on its documentation. From this project, the project members learned many new things.

This project helped us in getting the basic understanding of basic programming concepts of C language such as loop, structure, arrays, etc.

'The project members have used almost every concepts of C language we had learned. After completing this project, the project members are in position to explain C language concepts and apply them to the modeling real world system.

##### References

* E. Balagurusamy “Programming in ANSI C”, Tata Mc Graw-Hill Publishing.
* Ram Datta Bhatta “C Programming”, VidyarthiPrakashan (P.) Ltd.