Parking Payment System

Kevin Djoni | 2001586376

Contents

[Description 3](#_Toc466110821)

[Design/Plan 4](#_Toc466110822)

[Hierarchy Chart 5](#_Toc466110825)

[Explanation 5](#_Toc466110824)

[Functions 5](#_Toc466110825)

[Logbook 6](#_Toc466110826)

[Difficulties](#_Toc466110826) 7

[Solutions](#_Toc466110826) 7

[Coding](#_Toc466110821) 8

**Parking Payment System**

**Description**

Purpose : To create a program which calculate the amount of money for parking.

Features

* Display the date & time when the vehicle enters the parking lot.
* Display the date & time when the vehicle exits the parking lot.
* Display the parking duration.
* Display the parking charge.

Limitation

* Valid to only one vehicle.
* Cancelation is not supported.

**Design/Plan**

**Hierarchy Chart**

Main

In

Report

Out

Welcoming Message

Display time & date when the vehicle entered

Display time & date when the vehicle exited

Display in time & date, and out time & date.

Display Duration & Charge

**Explanation**

Functions:

**Void** main ( ) **& void** menu ( )

* Display the main menu of the program.

**Void** in ( )

* Display the time & date when the vehicle entered the parking lot.
* Starts the stopwatch to calculate duration.

**Void** out ( )

* Display the time & date when the vehicle exited the parking lot.
* Stops the stopwatch to calculate duration.
* Calculate duration in hours, minutes, and seconds.
* Set the charge needed to be paid for using the parking lot.

**Void** report ( )

* Displays the time & date when the vehicle entered the parking lot.
* Displays the time & date when the vehicle exited the parking lot.
* Displays the duration of the vehicle parked at the parking lot.
* Displays the parking fee.

**Logbook**

October 7th, 2016:

It was the day when the whole class was assigned to create a program for their final project. I began to think of a topic, and finally decided to go with the Parking Payment System. I made a hierarchy chart for this day, and submitted it through Github later that evening.

October 13th, 2016:

On this day, I started to create the functions as the structure of the whole program. I also decided to separate the functions into different files. I included the files in the main file.

October 21st, 2016:

Continuing the previous week’s work, I started to add the required keywords (some with structure) such as “**struct** tm”, “time\_t”, and “asctime” on both in and out functions in order to display the current time and date in the system, and starts the time counting, as well as stopping it. Unfortunately, the program did not run as I included all the side files in the main file containing the main function. I decided to include the previous file in the next file, example: I included “Void\_In\_Project.cpp” in the next file, which in this case is “Void\_Out\_Project.cpp” file. However, I included the “Void\_Report\_Project.cpp” in the main file in order to get all the required data without affecting the other functions.

October 27th, 2016:

2 weeks before the presentation, I decided to run the program again, but found out that the system showed an error message. However, I managed to overcome it by declaring “clock\_t start” in the function in( ), and the durations in different format in function out( ) as global variables in order to be called between functions from separate files. I also had to fix the duration in seconds due to the logic errors. I edited the switch statement so that the user will be redirected to the previous option if he/she did not enter the option in order.

November 4th, 2016:

I began to insert comments in the program in order to justify what the variables are. I also rechecked the program to prevent any errors that might show up in the future. Final works are done on this day.

**Lessons I have learned from this project:**

* I have a better understanding on <ctime> library, as well as its contents now, compared to before.
* I learnt that in order for the data to be called in different files, they should be declared as global variables.
* Comments play an important role in order for others to understand our codes.

**Problems that have been overcome:**

From this project, I learn that everything we do might contains problems which bother us in some cases. However, I managed to overcome those problems by asking with fellow colleagues, as well as the lecturer, about the problems. Syntax errors are the common problems which I encountered in this final project of mine.

**Coding**

**Int\_Main\_Project.cpp**

#include <iostream>

#include <ctime>

#include <iomanip>

#include <cstdlib>

#include <string>

#include "Void\_Report\_Project.cpp"

using namespace std;

int counter = 0;

void menu();

int main()

{

cout << "Parking Management System" << endl;

cout << "-------------------------" << endl;

menu();

return 0;

}

void menu()

{

int option;

cout << "Enter any of the following option with a number:" << "\n";

cout << "1. Input" << endl;

cout << "2. Output" << endl;

cout << "3. Report" << endl;

cout << "Your option: ";

cin >> option;

switch(option)

{

case 1 :

if(counter == 0)

{

in();

counter++;

}

else

cout<< "Invalid Input" << endl << endl;

menu();

break;

case 2 :

if(counter == 1)

{

out();

counter++;

}

else

cout<< "Invalid Input" << endl << endl;

menu();

break;

case 3 :

if(counter == 2)

{

report();

counter++;

exit(0);

}

else

cout<< "Invalid Input" << endl << endl;

menu();

default:

cout << "Invalid Input" << endl << endl;

menu();

}

}

**Void\_In\_Project.cpp**

#include <iostream>

#include <time.h>

#include <iomanip>

#include <string>

using namespace std;

string time1;

clock\_t start;

void in()

{

time\_t rawtime;

struct tm \*timeinfo;

time(&rawtime);

timeinfo = localtime(&rawtime);

cout << endl;

time1 = asctime(timeinfo);

cout << "In Time and Date: " << "\n" << time1 << endl;

start = clock(); //Starts stopwatch

}

**Void\_Out\_Project.cpp**

#include <iostream>

#include <time.h>

#include <iomanip>

#include <string>

#include "Void\_In\_Project.cpp"

using namespace std;

string time2;

int durationMin;

int durationSec;

int durationHr;

double rate = 0.5;

double charge;

void out()

{

time\_t rawtime;

struct tm \*timeinfo;

clock\_t stop;

time(&rawtime);

timeinfo = localtime(&rawtime);

cout << endl;

time2 = asctime(timeinfo);

stop = clock();

durationHr = ((stop - start) / (int)CLOCKS\_PER\_SEC) / 3600;

durationMin = (((stop - start) / (int)CLOCKS\_PER\_SEC) - durationHr \* 3600) / 60;

durationSec = (((stop - start) / (int)CLOCKS\_PER\_SEC) - durationHr \* 3600) - durationMin \* 60;

if(durationHr == 0 && durationMin < 15)

{

charge = 0;

}

else if(durationHr == 0 && durationMin >= 15) {

charge = rate;

}

else if(durationHr == 1 && durationMin > 0)

{

charge = 2 \* rate;

}

else if(durationHr == 2 && durationMin > 0)

{

charge = 3 \* rate;

}

else if(durationHr == 3 && durationMin > 0)

{

charge = 4 \* rate;

}

else if(durationHr == 4 && durationMin > 0)

{

charge = 5 \* rate;

}

else if(durationHr == 5 && durationMin > 0)

{

charge = 6 \* rate;

}

else

{

charge = 5;

}

cout << "Out Time and Date: " << endl << time2 << endl;

}

**Void\_Report\_Project.cpp**

#include <iostream>

#include <ctime>

#include <iomanip>

#include <string>

#include "Void\_Out\_Project.cpp"

using namespace std;

void report()

{

cout << endl << "Report" << endl;

cout << "------" << endl;

cout << "In Time & Date " << "\t\t: " << time1;

cout << "Out Time & Date " << "\t: " << time2;

cout << "Duration " << "\t\t: " << durationHr << " hour(s) " << durationMin;

cout << " minute(s) " << durationSec << " second(s)" << endl;

cout << "Charge " << "\t\t\t: $" << charge;

}