**CONPRECT: A CONSTRUCTION PREDICTOR and MANPOWER PROVIDING SYSTEM**

*A*

*Mini Project Report*

*Submitted in partial fulfilment of the*

*Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

**V. KALYAN – 1602-19-737-135**

**SHAIK AMAAN - 1602-19-737-166**



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31**

**2020**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**2020**

**Hyderabad-500 031**

**Department of Information Technology**



**DECLARATION BY THE CANDIDATE**

We, SHAIK AMAAN and V. KALYAN bearing hall ticket numbers, 1602-19-737-166 and 1602-19-737-135 respectively, hereby declare that the project report entitled “CONPRECT: A Construction Predictor and Manpower Providing Application” is submitted in partial fulfilment of the requirement for the award of the degree of Bachelor of Engineering in Information Technology.

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

V. KALYAN

1602-19-737-135

SHAIK AMAAN

1602-19-737-166

(Faculty In-Charge) (Head, Dept. of IT)

**ACKNOWLEDGEMENT**

We as a team have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them. We would like to thank our college, **Vasavi College of Engineering,** for providing the opportunity to implement our project**, “CONPRECT: A Construction Predictor and Manpower Providing Application”.**

We are highly indebted to **Ms. Divya Lingineni, Assistant Professor, Department of Information Technology, Vasavi College of Engineering** for her guidance and constant supervision as well as for providing necessary information regarding the project and for her support in completing the project.

I would like to express my special gratitude and thanks **to Ms. Prassana Dusi, Assistant Professor, Department of Information Technology, Vasavi College of Engineering** for teaching the concepts firmly due to which we were able to complete the project.

Last but not the least, we wish to acknowledge our Parents and Friends for giving moral strength and constant encouragement.

Sincerely,

**V. KALYAN 1602-19-737-135**

**SHAIK AMAAN 1602-19-737-166**

**ABSTRACT**

We in our day-to-day life see constructions here and there but when it comes to our knowledge in the field of construction is not well enough. So, we may get easily fooled by the builders which results in the form of losses to the normal people. So, here comes our project build on this idea to provide customers with the genuine predictions of the cost, time required, and EMI’s to build houses in the way customers want. After building homes we also need to maintain it regularly. So, we also show the required person’s details to repair and maintain your homes.

**TABLE OF CONTENTS**

[1. INTRODUCTION 7](#_Toc59449566)

[1.1. ABOUT THE PROJECT 7](#_Toc59449567)

[1.2. PROJECT DOMAIN 7](#_Toc59449568)

[1.2.1. TECHNICAL DOMAIN 7](#_Toc59449569)

[1.2.2. FUNCTIONAL DOMAIN 7](#_Toc59449570)

[1.3. FEATURES 7](#_Toc59449571)

[1.3.1. EMI 8](#_Toc59449572)

[1.3.2. Total time 8](#_Toc59449573)

[1.3.3. Structure Cost 8](#_Toc59449575)

[1.3.4. Semi Furnished Interior 8](#_Toc59449576)

[1.3.5. Fully Furnished Interior 8](#_Toc59449577)

[1.3.5. Manpower 8](#_Toc59449578)

[1.3.6. Updating 8](#_Toc59449579)

[2. TECHNOLOGY 9](#_Toc59449580)

[2.1. SOFTWARE REQUIREMENTS 9](#_Toc59449581)

[2.2. HARDWARE REQUIREMENTS 9](#_Toc59449582)

[3. PROPOSED WORK 10](#_Toc59449583)

[3.1. DESIGN 10](#_Toc59449584)

[3.1.1. USER USE CASES 10](#_Toc59449585)

[3.1.1.1. EMI 11](#_Toc59449586)

[3.1.1.2. Total time 11](#_Toc59449587)

[3.1.1.3. Structure Cost 11](#_Toc59449589)

[3.1.1.4. Semi Furnished Interior 11](#_Toc59449590)

[3.1.1.5. Fully Furnished Interior 11](#_Toc59449591)

[3.1.1.6. Manpower 11](#_Toc59449592)

[3.1.1.7. Updating 11](#_Toc59449593)

[3.1.2. ADMIN USE CASES 11](#_Toc59449594)

[3.2. IMPLEMENTATION 13](#_Toc59449595)

[3.2.1. MODULE-WISE CODE 13](#_Toc59449596)

[3.2.1.1. Frame Module 13](#_Toc59449597)

[3.2.1.2. Pointer Module 14](#_Toc59449598)

[3.2.1.3. Login 14](#_Toc59449599)

[3.2.1.4. MAIN Module 18](#_Toc59449600)

[3.2.1.5. Predictor Module 20](#_Toc59449601)

[3.2.2. GITHUB/FOLDER STRUCTURE 22](#_Toc59449602)

[3.3. TESTING 22](#_Toc59449603)

[3.3.1. TEST PLAN 23](#_Toc59449604)

[3.3.2. ADMIN TEST CASES 23](#_Toc59449605)

[4. RESULTS 29](#_Toc59449606)

[4.1. Home Page 30](#_Toc59449607)

[4.2. ADMIN TEST CASE RESULTS 30](#_Toc59449608)

[4.2. USER TEST CASE RESULTS 40](#_Toc59449609)

[5. ADDITIONAL KNOWLEDGE ACQUIRED 53](#_Toc59449610)

[6. CONCLUSION AND FUTURE WORK 54](#_Toc59449611)

[7. REFERENCES 55](#_Toc59449612)

# 1. INTRODUCTION

## 1.1. ABOUT THE PROJECT

“CONPRECT – A Self-Learning and Quiz Environment” is a console-based C Project which acts as a learning platform for users of all age groups to improve their skill set. The intent behind this project is to inculcate the zeal to learn important Computer Science based concepts, such as C, Python and OOPs.

## 1.2. PROJECT DOMAIN

The domain of the project is the targeted subject area of a computer program. It is a term most commonly used in software engineering. Formally, it represents the target subject of a specific programming project, whether narrowly or broadly defined. To be concise, a **domain** in the realm of software engineering commonly refers to the subject area on which the application is intended to apply. Within a domain, there are two categories:

1. Technical Domain
2. Functional Domain

### 1.2.1. TECHNICAL DOMAIN

‘CONPRECT’ falls under the domain of a Console Application. A console application is a program designed to be used via a text-only computer interface, such as a text terminal, the command line interface of some operating systems or the text-based interface included with most GUI (Graphical User Interface) operating systems.

### 1.2.2. FUNCTIONAL DOMAIN

‘CONPRECT’ comes under the Information and Development domain because we are aiming to predict and help users to provide necessary information.

## 1.3. FEATURES

Our basic idea was to predict the expenditures, time going into, in the field of Construction and to provide the required details of manpower in different fields. We mainly focussed on the Predicting module and file handling because that was the base of our project.

Another main component we wanted to keep emphasis on was the Updating part. This is because the rates and the required technicians change day to day. So, to maintain the application we need to regularly update the information on day-to-day basics. This Updating feature helps to maintain the above constrain

In short, the features that we have offered will provide the user to get the genuine information and prediction related to construction.

### 1.3.1. EMI

This feature allows user to know the amount to be paid per month to the builder.

### 1.3.2. Total time

### Allows user to know the total time required to complete the project by inserting the required data prompted by the system.

### 1.3.3. Structure Cost

Allows user to know the total amount required to complete structure of the project.

### 1.3.4. Semi Furnished Interior

Allows user to know the total amount required to complete semi furnished type interior of the project.

### 1.3.5. Fully Furnished Interior

Allows user to know the total amount required to complete fully furnished type interior of the project.

### 1.3.5. Manpower

Allows user to know details of the plumbers, electrician and etc from the data base of the admin.

### 1.3.6. Updating

Allows Admin to maintain the application.

# 2. TECHNOLOGY

All computer software needs certain hardware components or other software resources to be present, in order for computers to be used efficiently. These prerequisites are known as System Requirements. Within this, we have two types – Software Requirements and Hardware Requirements.

## 2.1. SOFTWARE REQUIREMENTS

Software Requirements deal with defining the software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These preconditions are generally not included in the software installation package and need to be installed separately.

In order to use CONPRECT, one should have the following:

* **Operating System:** Windows 7 and above
* **C Compiler :** GNU Compiler Collection (GCC)
* **Editor:** Any text editor (preferably any IDE)

## 2.2. HARDWARE REQUIREMENTS

Hardware requirements refer to the common set requirements defined by any operating system or software application and are usually the physical computer resources. In this, we look into the processing power, memory, secondary memory, display adapter and peripherals.

In order to use CONPRECT, one should have the following:

* **Processor:** Intel Core i3 and above
* **Memory:** 4 GB RAM

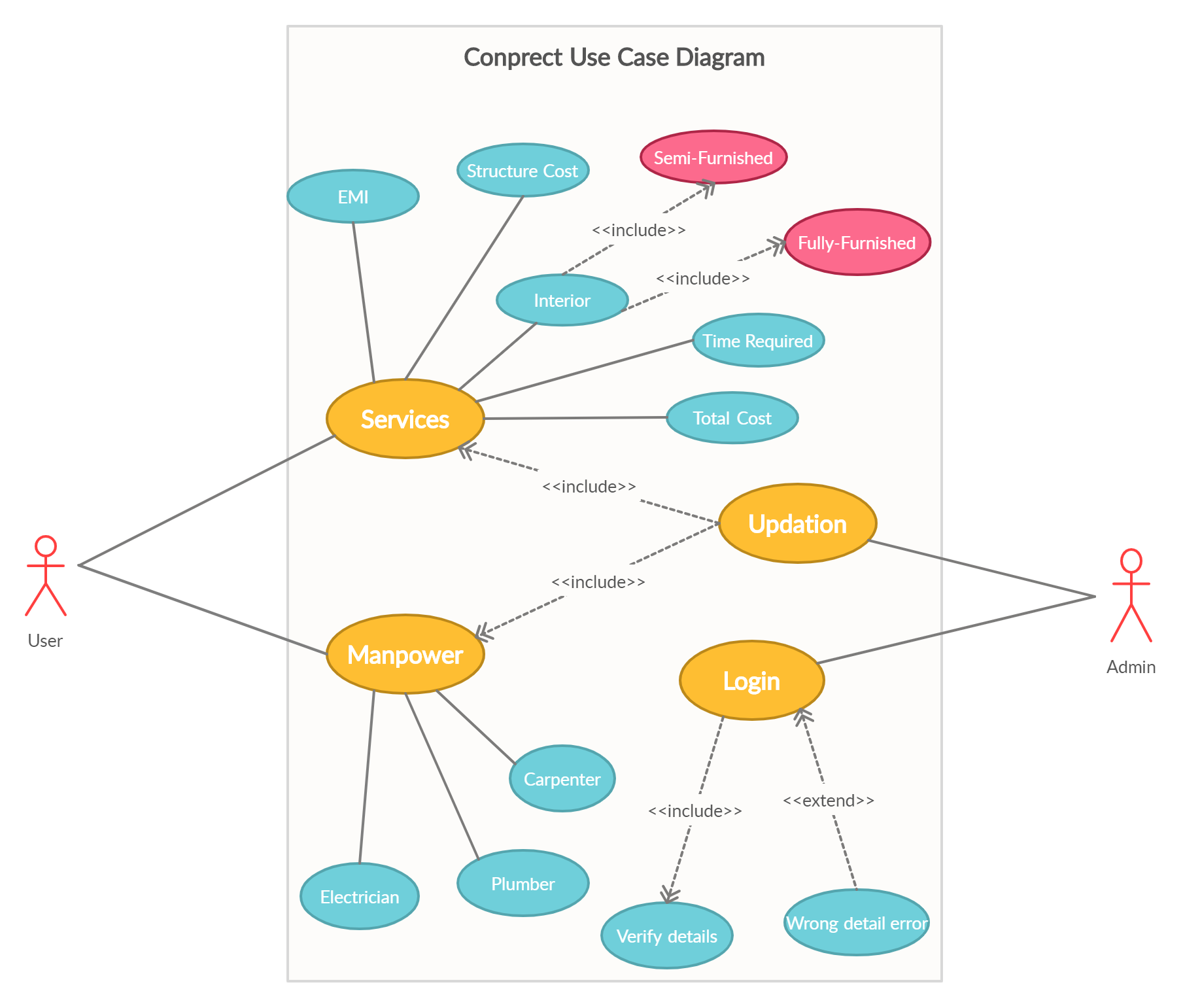
# 3. PROPOSED WORK

## 3.1. DESIGN

Our approach in designing CONPRECT was to divide our users into two groups – the users and the Admin. The user is a day-to-day consumer who uses this application to predict the expenditures and EMI. The Admin’s end goal is to maintain CONPRECT’s database and Update details day-to-day.

### 3.1.1. USER USE CASES

The user is a person who has limited access to CONPRECT’s user applications. He/she will have 5 functionalities: EMI, STRUCTURE COST, INTERIOR COST, TIME REQUIRED.



### 

### 3.1.1.1. EMI

This feature allows user to know the amount to be paid per month to the builder.

### 3.1.1.2. Total time

### Allows user to know the total time required to complete the project by inserting the required data prompted by the system.

### 3.1.1.3. Structure Cost

Allows user to know the total amount required to complete structure of the project.

### 3.1.1.4. Semi Furnished Interior

Allows user to know the total amount required to complete semi furnished type interior of the project.

### 3.1.1.5. Fully Furnished Interior

Allows user to know the total amount required to complete fully furnished type interior of the project.

### 3.1.1.6. Manpower

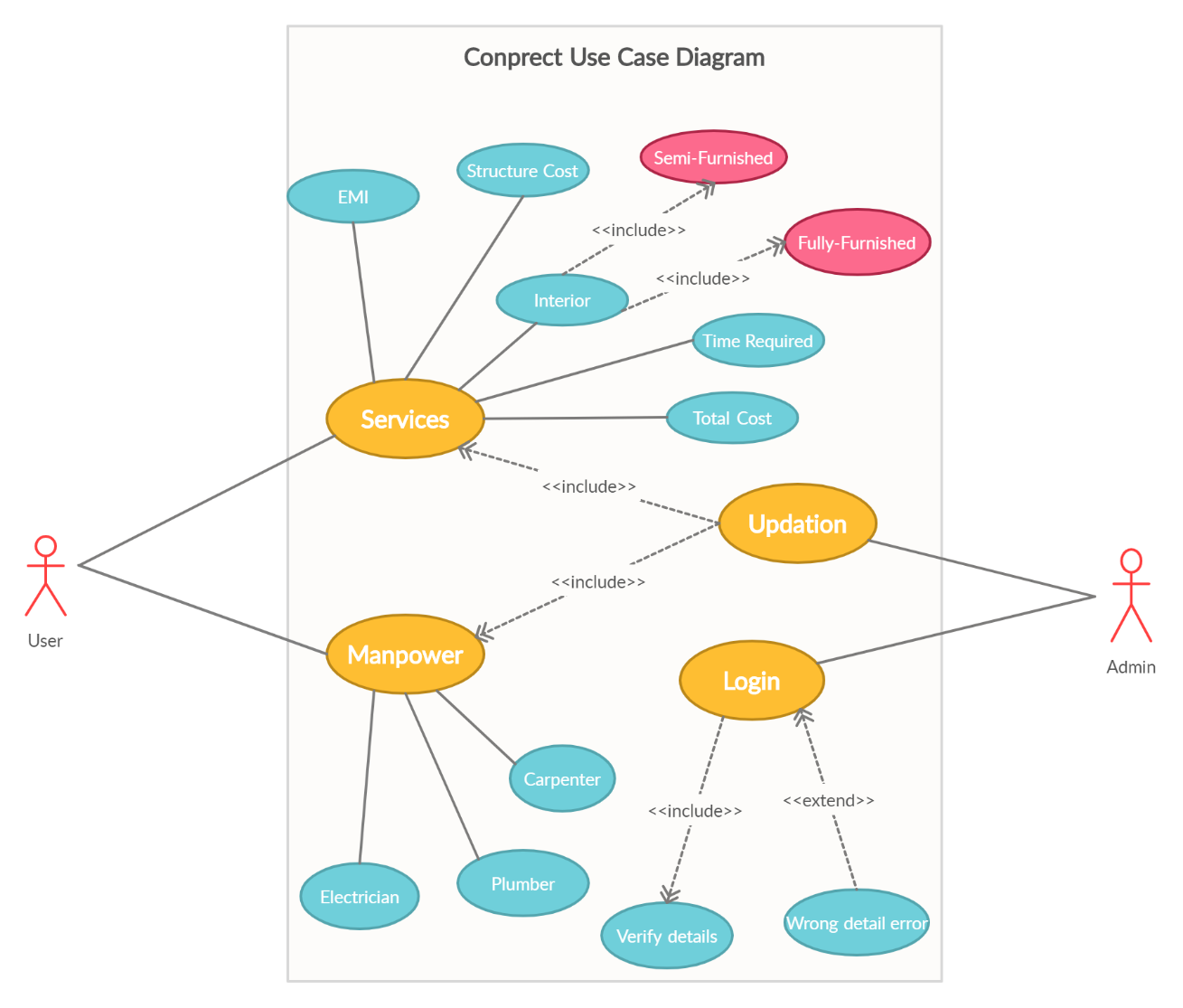
Allows user to know details of the plumbers, electrician and etc from the data base of the admin.

### 3.1.1.7. Updating

#### Allows Admin to maintain the application.

### 3.1.2. ADMIN USE CASES

The Admin is a user who has access to all information and including the rates. The Admin also has the license to edit any manpower’s details and even delete a user if needed.



#### 3.1.2.1. LOGIN

The Admin can login if he/she has the required credentials. The system first verifies whether the credentials entered match the Admin credentials. If they do, then the Admin gets complete access to CONPRECT. If they don’t match, the required message is displayed on the console.

#### 3.1.2.2. EDIT RATE DETAILS

Once logged in as the Admin, admin can choose to edit any rate details. First, we load the previous rates from the database and the Admin can choose the option whose rate we want to edit. Then, the system displays the chosen option’s details. From this, the Admin can edit. After editing, the old data in the database is replaced with the new data.

#### 3.1.2.3. EDIT MANPOWER DETAILS

Only the Admin has the authority to this, to edit the details of the manpower in the data base stored using file handling systems.

## 3.2. IMPLEMENTATION

Based on the use cases, we have implemented this project by dividing the work into 6 modules – Frame module, Pointer module, Login module, Manpower module, Prediction module, Start module.

The Frame module is for stylistically printing our product’s layout when we run our console application.

### 3.2.1. MODULE-WISE CODE

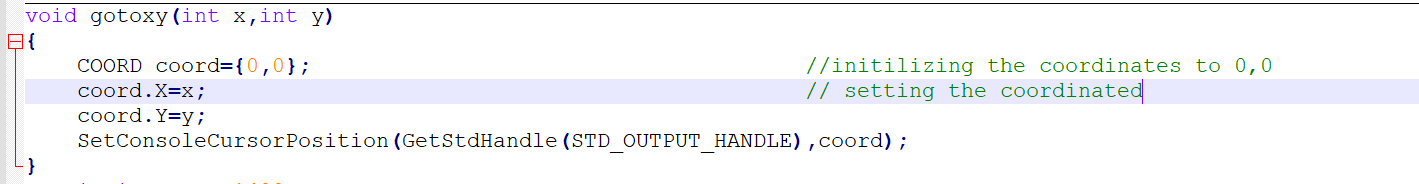
## 3.2.1.1. Frame Module

This module designs the layout.



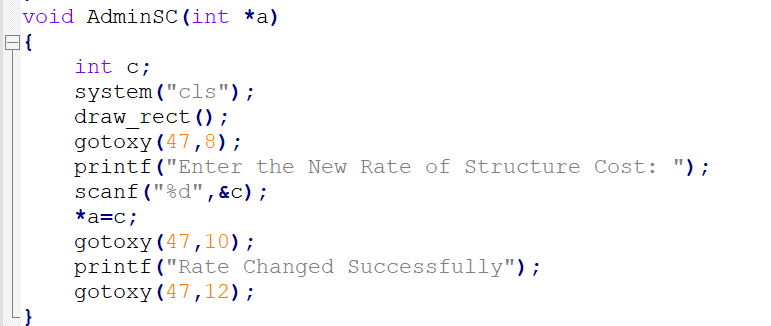
## 3.2.1.2. Pointer Module

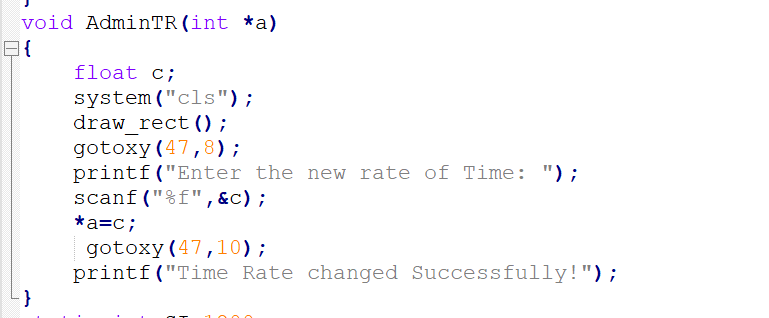
#### This module helps to change the pointer(courser) position in console.

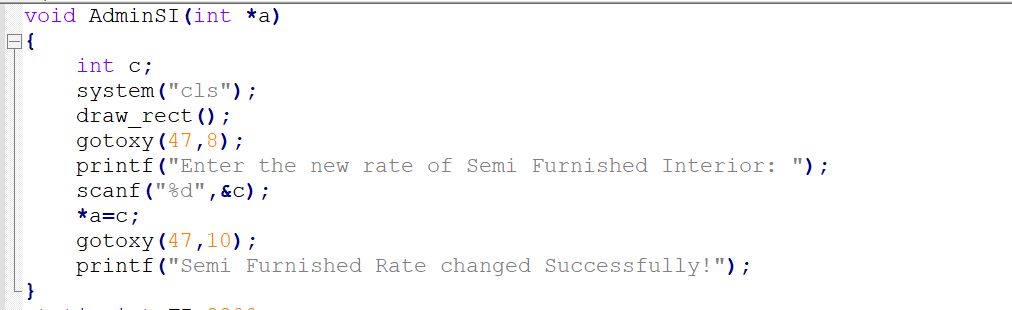


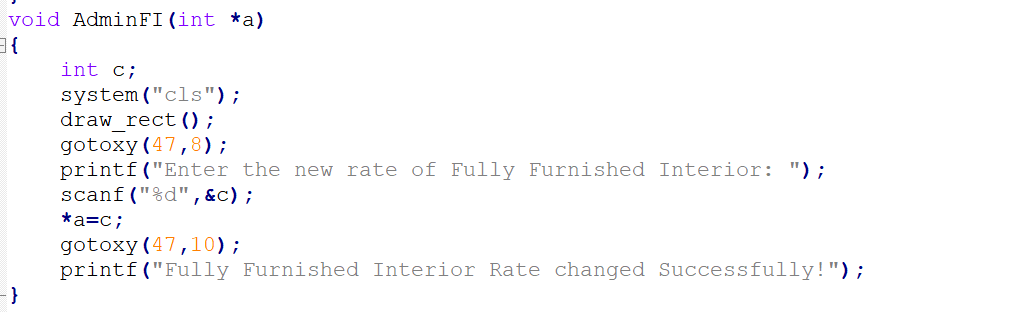
## 3.2.1.3. Login

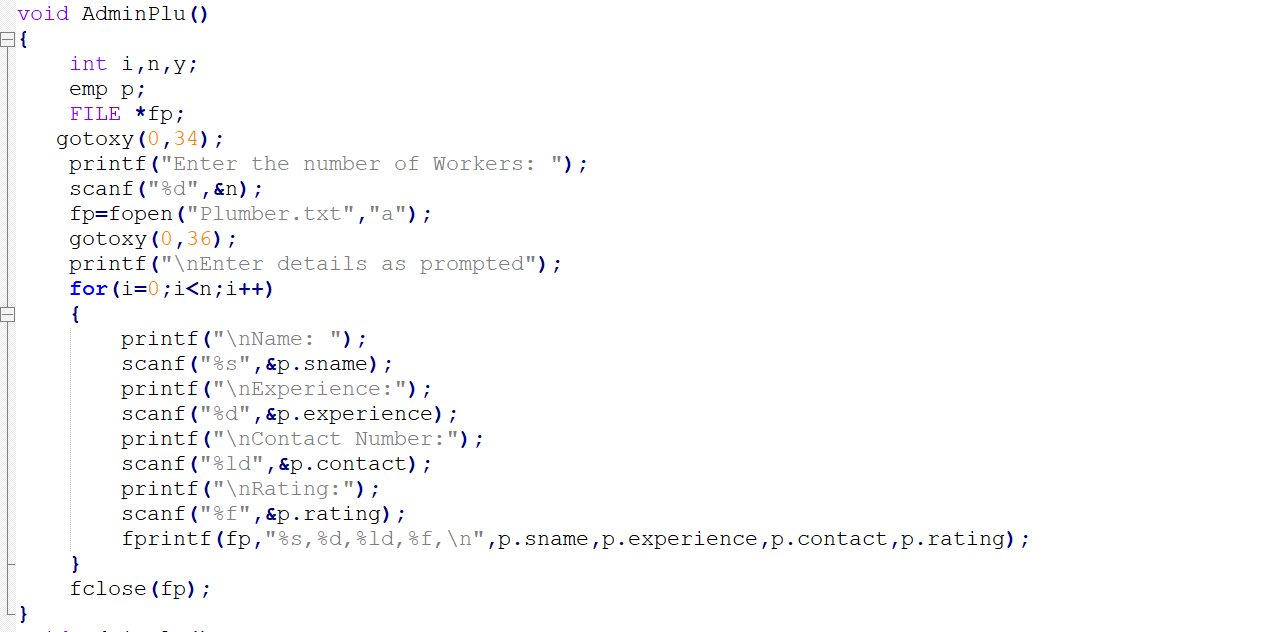
Thismoduleis like the administrator for the application

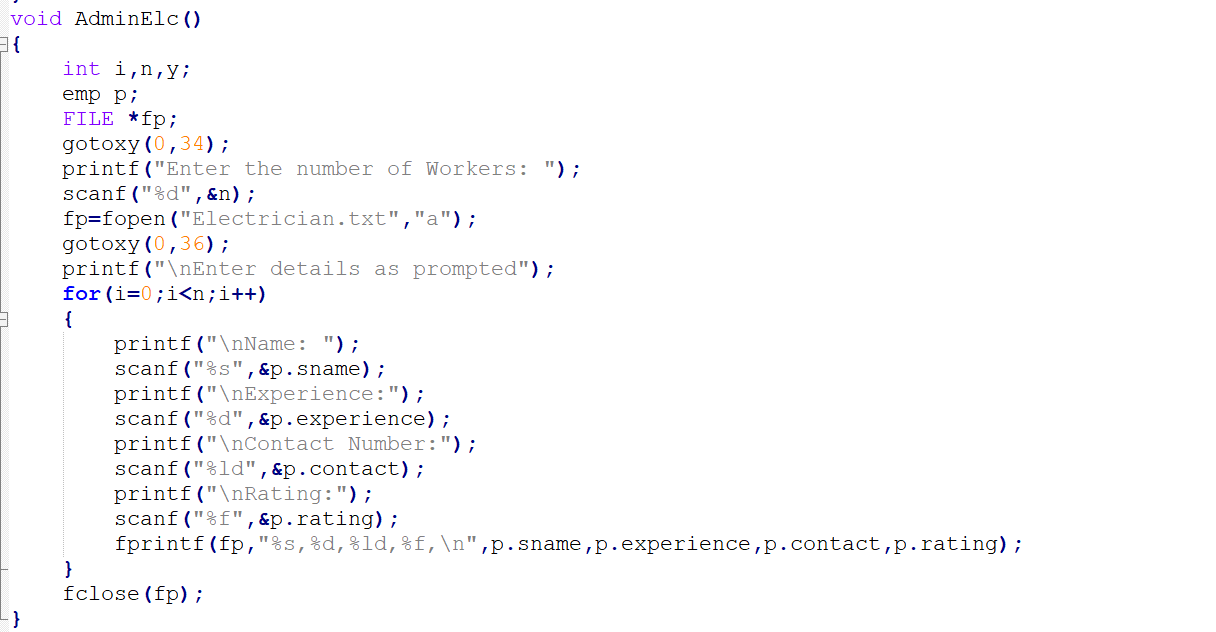


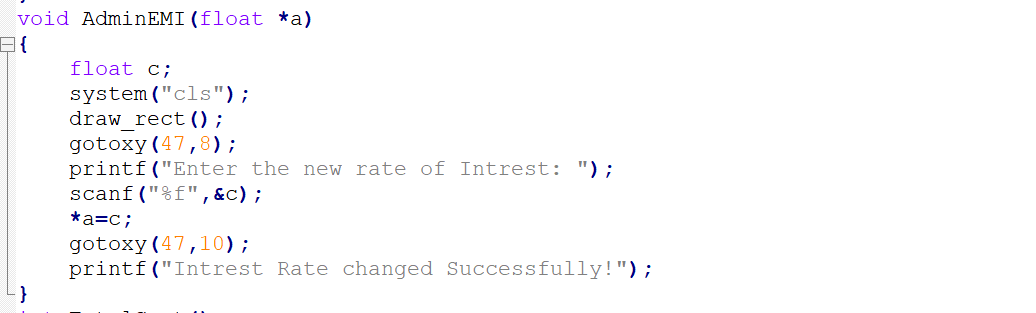


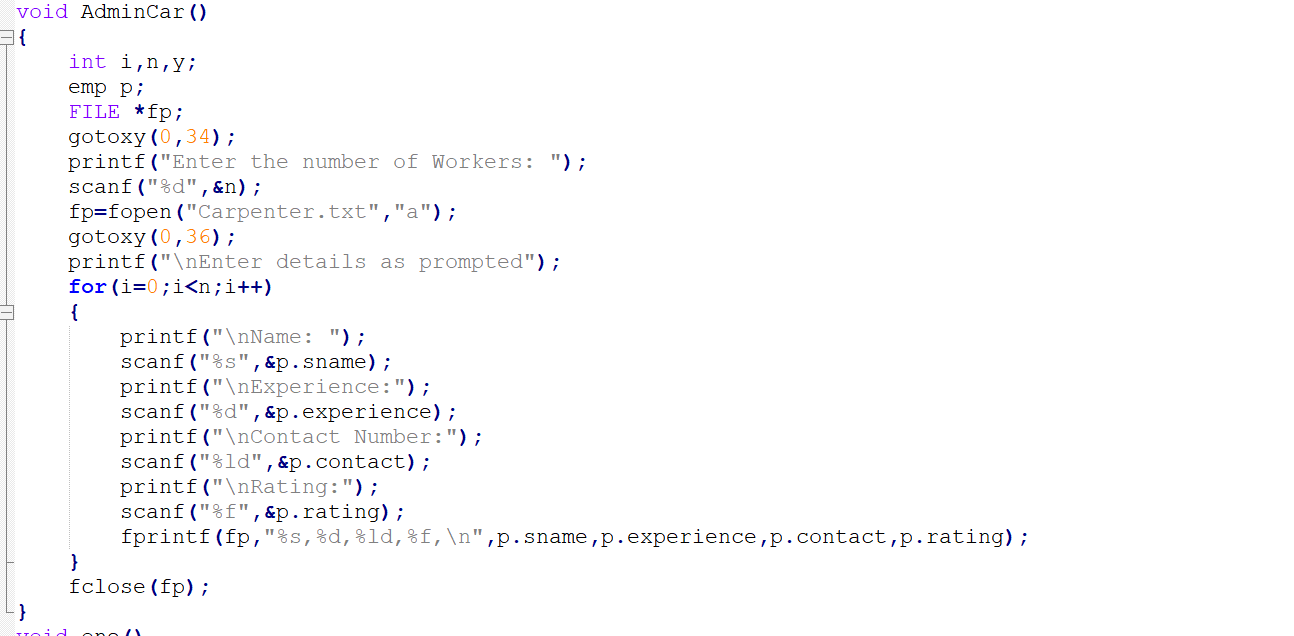


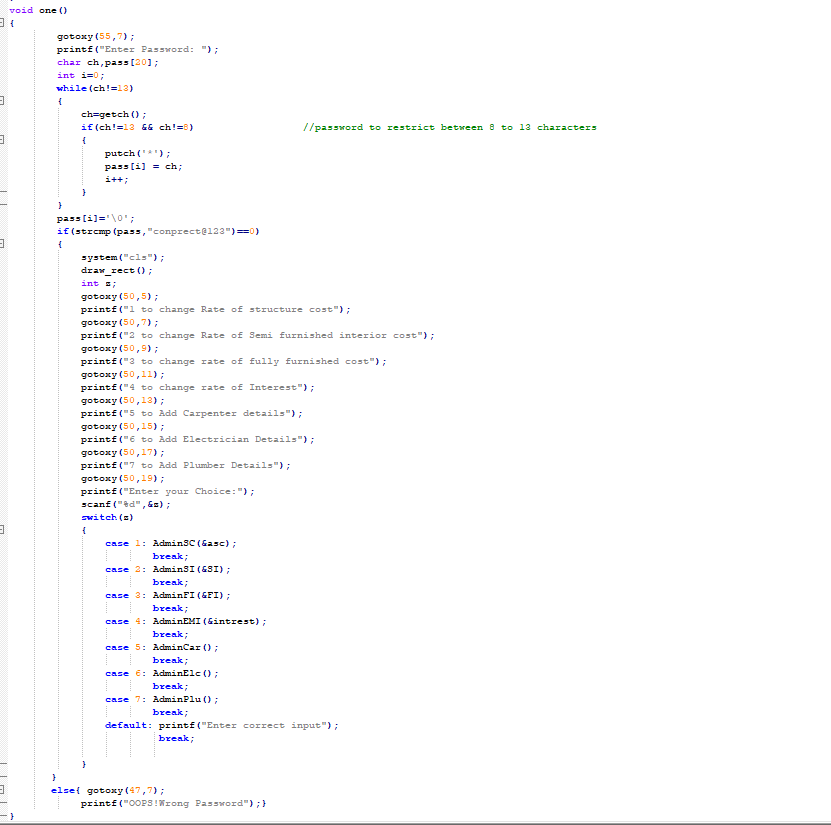








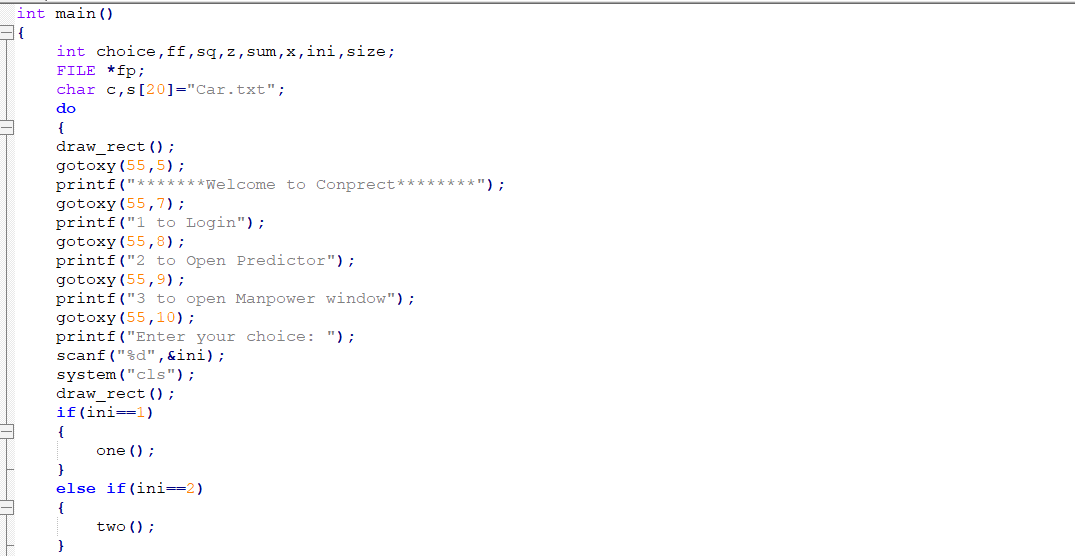


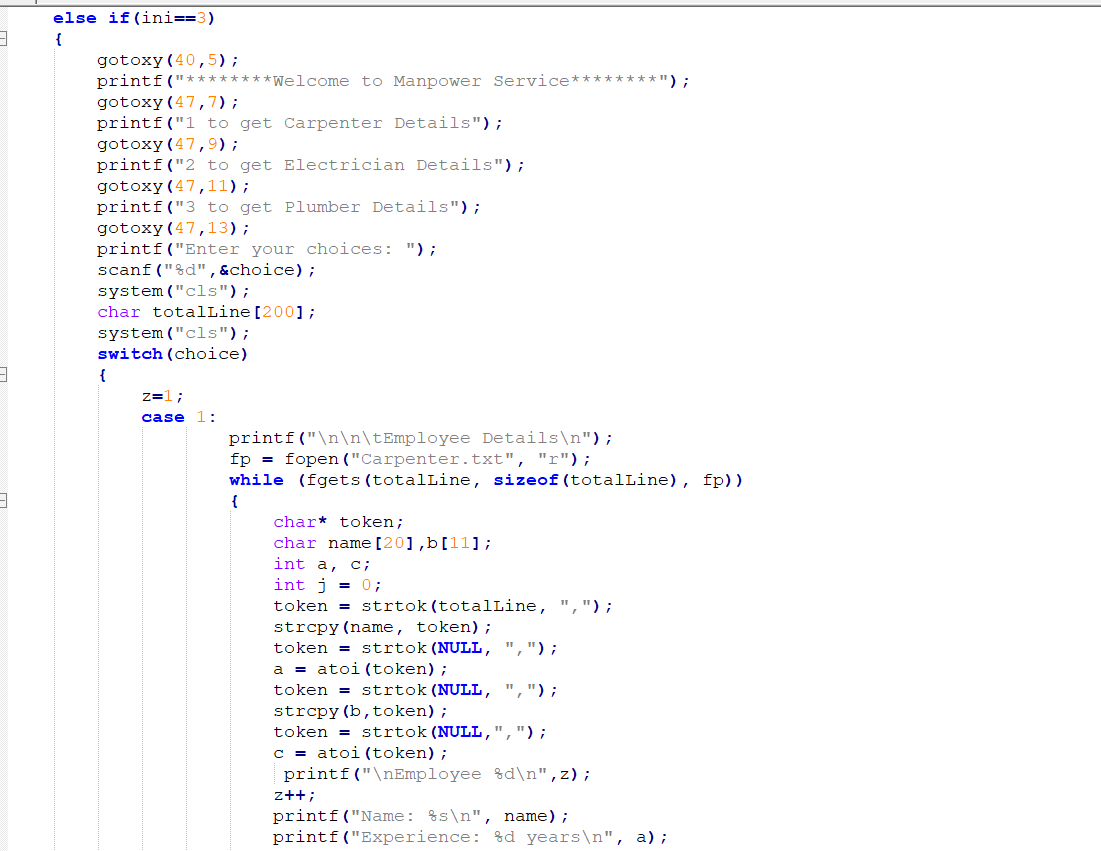


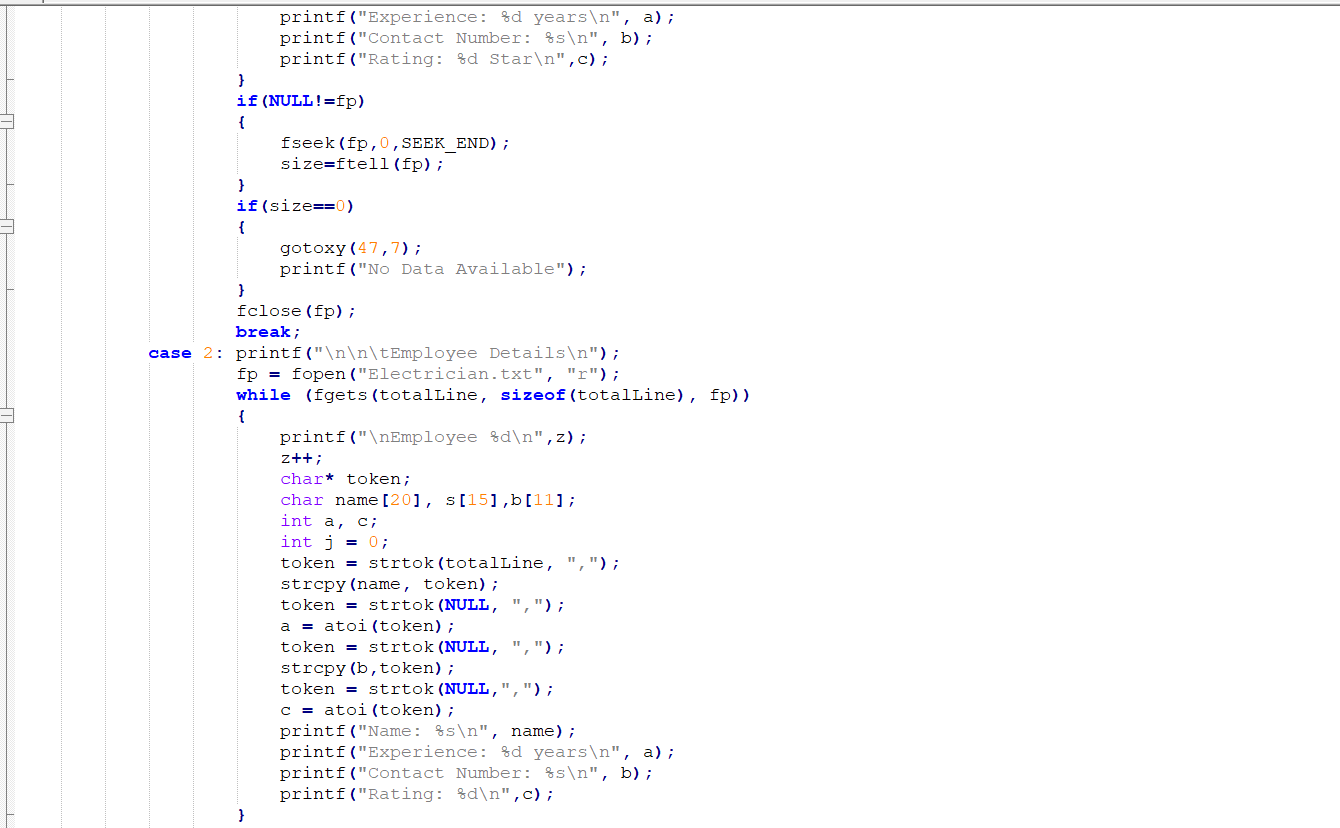
#### 

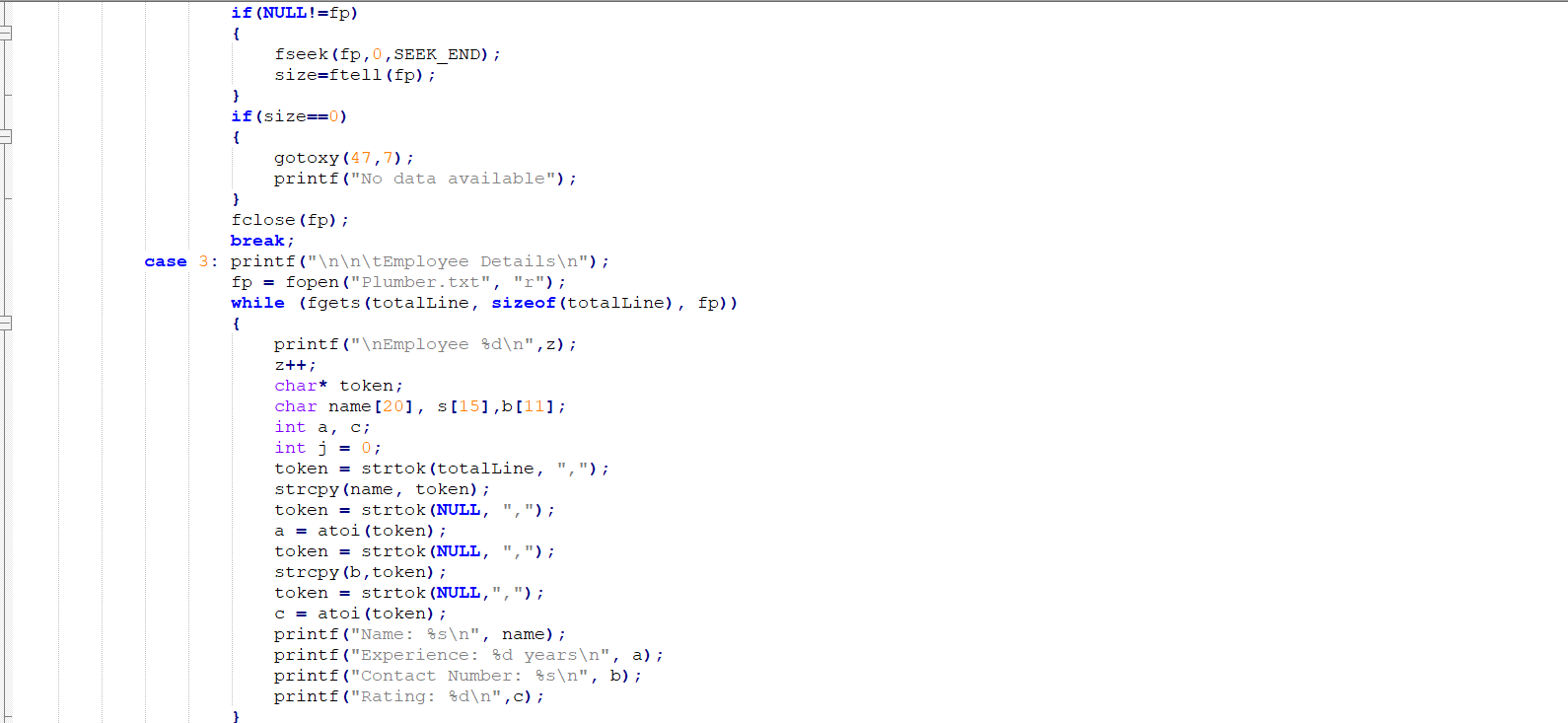
## 3.2.1.4. MAIN Module

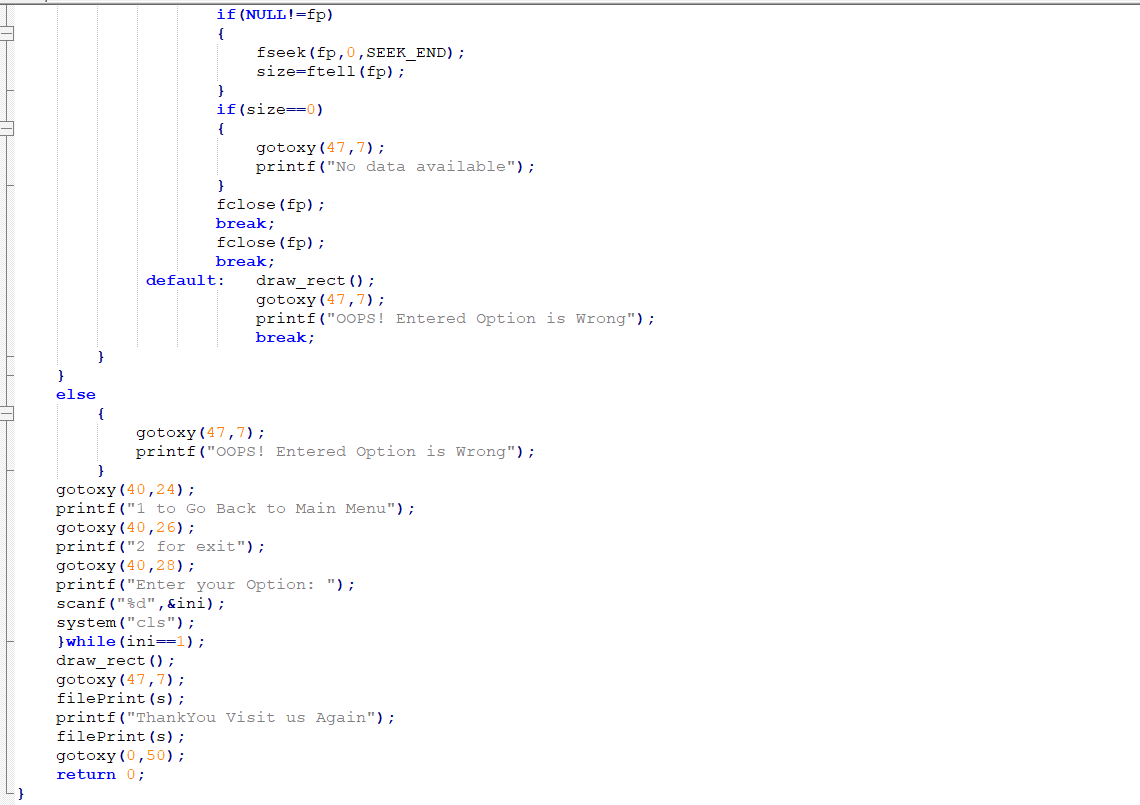
This module sets the main module





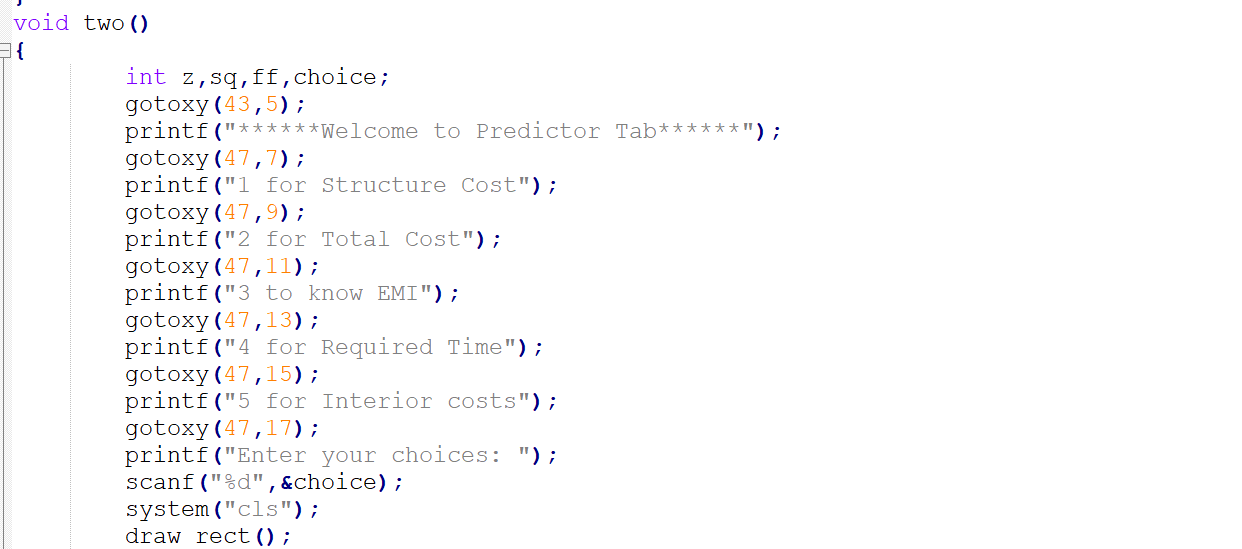






## 3.2.1.5. Predictor Module

This module helps in predicting the output of user defined functions





### 3.2.2. GITHUB/FOLDER STRUCTURE

https://github.com/Shaik-Amaan/Conprect.git

## 3.3. TESTING

Testing is a method to check whether the actual product matches the expected requirements and to ensure that the product is defect-free. This process involves execution of various parts of the product either using manual or automated tools. The purpose is to identify errors, gaps or missing requirements in contrast to the actual requirements.

### 3.3.1. TEST PLAN

We approached testing our console application by analysing each module separately. First, we coded the requirements and then manually tested each feature present in the module to cover any gaps that might occur.

### 3.3.2. ADMIN TEST CASES

The Admin has 5 major functionalities: Login, View User Table, Edit Users Details, Delete User and View Points Leader board. Below are the testcases which we have compiled together manually.

#### 3.3.3.1. LOGIN

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Test Case Template** |  |
| **Test Case ID:** TC01 | |  | **Use Case ID:** |
| **Test Case Title:** Admin-Login | | | UC 110 |
| **Test Case Description:**  Admin attempts to login into CONPRECT with password. | | |
| **Test Steps:** | **Expected Result:** | | **Actual Result:** |
| 1. System displays prompt for admin to enter password 2. Admin enters password. | System should display the updating window. | | System displays “the window on console with all the updating options”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC02 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Login | |
| **Test Case Description:** Admin attempts to login into CONPRECT with password. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays prompt for admin to enter password 2. Admin enters password. | System should display  “Invalid credentials”. | The System displays  “OOPS! Wrong Password”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC03 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Rate Change | |
| **Test Case Description:** Admin updates the rate of the structure cost. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts the admin to enter new rate of structure cost 2. Admin enters new rate. | System should display  “Rate changed successfully”. | The System displays  “Rate of Structure changed successfully”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC04 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Rate Change | |
| **Test Case Description:** Admin updates the rate of the semi furnished interior cost. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts the admin to enter new rate of structure cost 2. Admin enters new rate. | System should display  “Rate changed successfully”. | The System displays  “Rate of Semi furnished interior changed successfully”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC08 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Rate Change | |
| **Test Case Description:** Admin updates the rate of the fully furnished cost. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts the admin to enter new rate of fully furnished cost 2. Admin enters new rate. | System should display  “Rate changed successfully”. | The System displays  “Rate of fully furnished changed successfully”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC07 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Rate Change | |
| **Test Case Description:** Admin updates the rate of the time. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts the admin to enter new rate of time required 2. Admin enters new rate. | System should display  “Rate changed successfully”. | The System displays  “Rate of time required changed successfully”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC06 | | **Use Case ID:**  UC 110 |
| **Test Case Title:** Admin-Rate Change | |
| **Test Case Description:** Admin updates the rate of intrest. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts the admin to enter new rate of interest 2. Admin enters new rate. | System should display  “Rate changed successfully”. | The System displays  “Rate of interest changed successfully”. |

3.3.3. USER TEST CASES

The user has 6 major functionalities: Frame, Pointer, Login, Manpower, Prediction, Start. Below are the testcases which we have compiled together manually.

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC 15,16 | | **User Case ID:**  UC101 |
| **Test Case Title:** User-EMI | |
| **Test Case Description:** Allows user to know the amount to be paid per month to builder | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts data required for system to calculate EMI per month 2. User enters the data prompted by the system. | System validate user's data then displays the predicted cost. | System displays “The EMI is: xxxx”. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case Template** | |  |  |
| **Test Case ID:** TC17 | | **User Case ID:**  UC102 |  |
| **Test Case Title:** User-Time required | |
| **Test Case Description:** Allows user to know the total time required to complete the project by inserting the required data prompted by the system | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |  |
| 1. System prompts data required for system to calculate Time required. 2. User enters the data prompted by the system | System validate user's data then displays the predicted time required. | System displays “Time required to complete is: xx days”. |  |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | | |
| **Test Case ID:** TC12 | | **User Case ID:**  UC103 |
| **Test Case Title:** User-Structure cost | |
| **Test Case Description:** Allows user to know the total amount required to complete structure of the building | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts data required for system to calculate Structure cost. 2. User enters the data prompted by the system | System validate user's data then displays the predicted cost. | The System displays “The cost of Structure is: xxx”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | | |
| **Test Case ID:** TC18 | | **User Case ID:**  UC104 |
| **Test Case Title:** User- Semi Furnished interior | |
| **Test Case Description:** Allows user to know the total amount required to complete semi furnished type interior of the project | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays prompt for user to choose an option. 2. User enters invalid choice. | System validate user's data then displays the predicted cost. | The System displays “The cost of Semi furnished Interior is: xxx”. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | | |
| **Test Case ID:** TC19 | | **User Case ID:**  UC104 |
| **Test Case Title:** User- Fully Furnished interior | |
| **Test Case Description:** Allows user to know the total amount required to complete Fully furnished type interior of the project | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays prompt for user to choose an option. 2. User enters invalid choice. | System validate user's data then displays the predicted cost. | The System displays “The cost of Fully furnished Interior is: xxx”. |
| **Test Case Template** | |  |
| **Test Case ID:** TC 24 | | **User Case ID:**  UC105 |
| **Test Case Title:** User | |
| **Test Case Description:** While choosing a option in main menu, user chooses an invalid option. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays a list of choices to choose from. 2. User chooses an option which is not present. | System should error display message. | System displays “OOPS! Entered Invalid Input” message. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | | |
| **Test Case ID:** TC13,14 | | **User Case ID:**  UC103 |
| **Test Case Title:** User-Total construction cost | |
| **Test Case Description:** Allows user to know the total amount required to complete construction of the building including interior | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System prompts data required for system to calculate Structure cost. 2. User enters the data prompted by the system | System validate user's data then displays the predicted cost. | The System displays “Total Cost is: xxx”. |

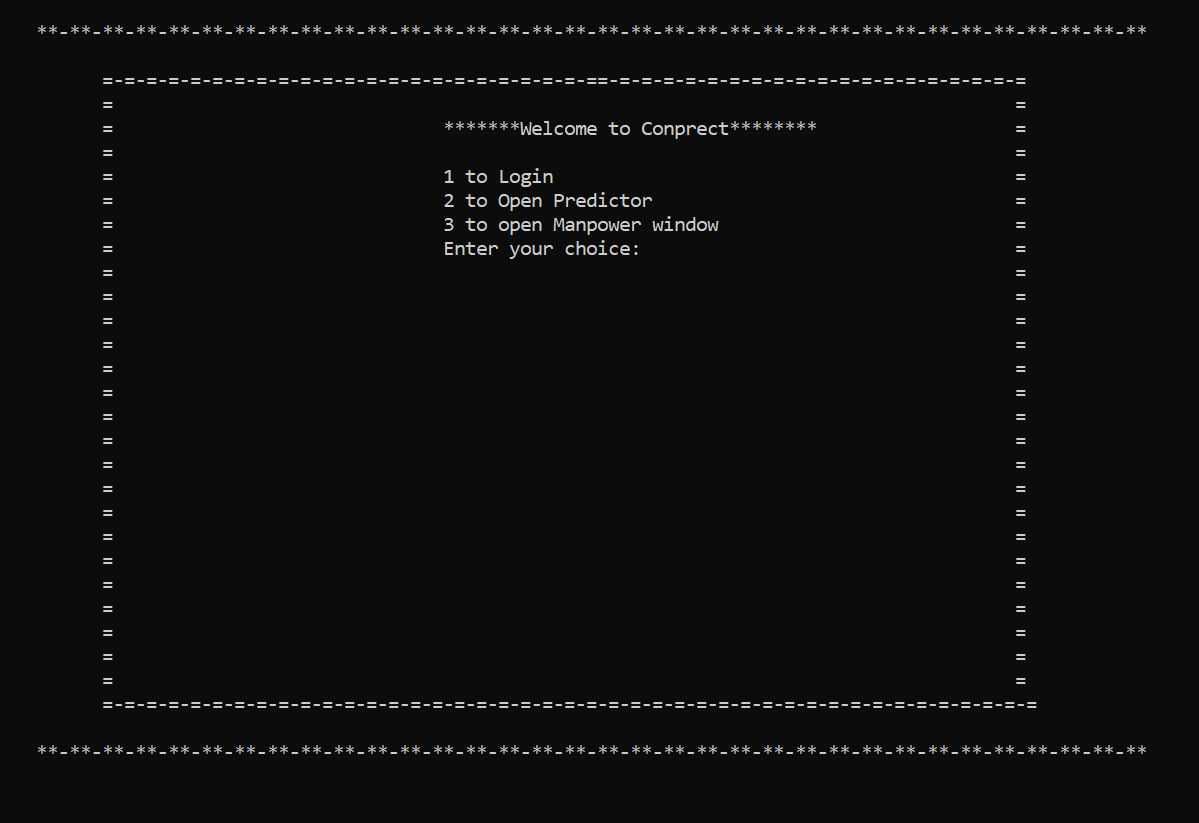
|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC20 | | **User Case ID:**  UC105 |
| **Test Case Title:** User | |
| **Test Case Description:** While choosing options in manpower window, the user chooses wrong option. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays a list of technicians to display from after choosing a manpower window. 2. User chooses an option which is not present. | System should error display message. | System displays “OOPS! Invalid Input” message. |

|  |  |  |
| --- | --- | --- |
| **Test Case Template** | |  |
| **Test Case ID:** TC11 | | **User Case ID:**  UC103 |
| **Test Case Title:** User | |
| **Test Case Description:** While choosing options in predictor tab, the user chooses wrong option. | |
| **Test Steps:** | **Expected Result:** | **Actual Result:** |
| 1. System displays a list of updating options to display from after choosing a predictor tab. 2. User chooses an option which is not present. | System should error display message. | System displays “OOPS! Invalid Input” message. |

# 4. RESULTS

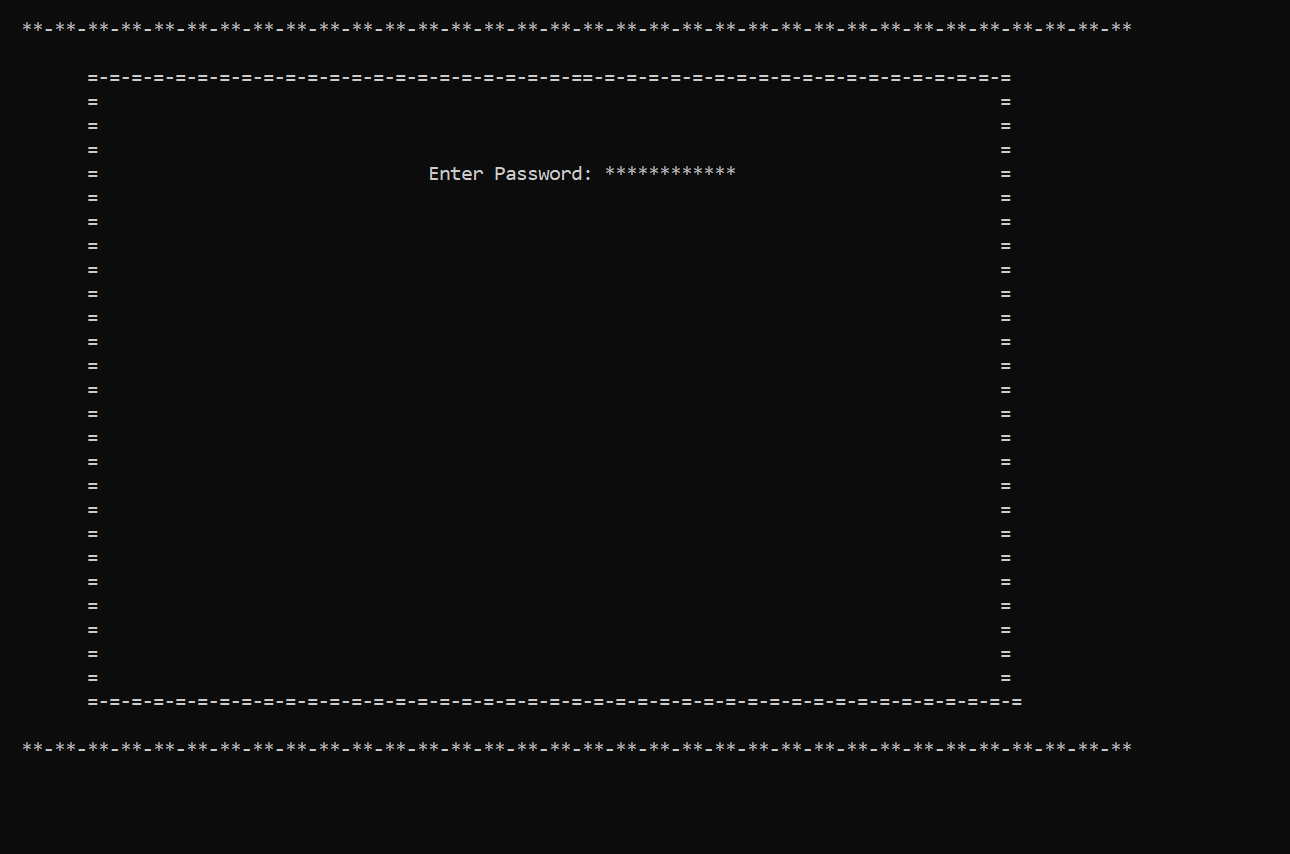
We have successfully developed a platform for users who can predict the expenditures and time required with manpower providing system. Below are the output screenshots of the testcases mentioned.

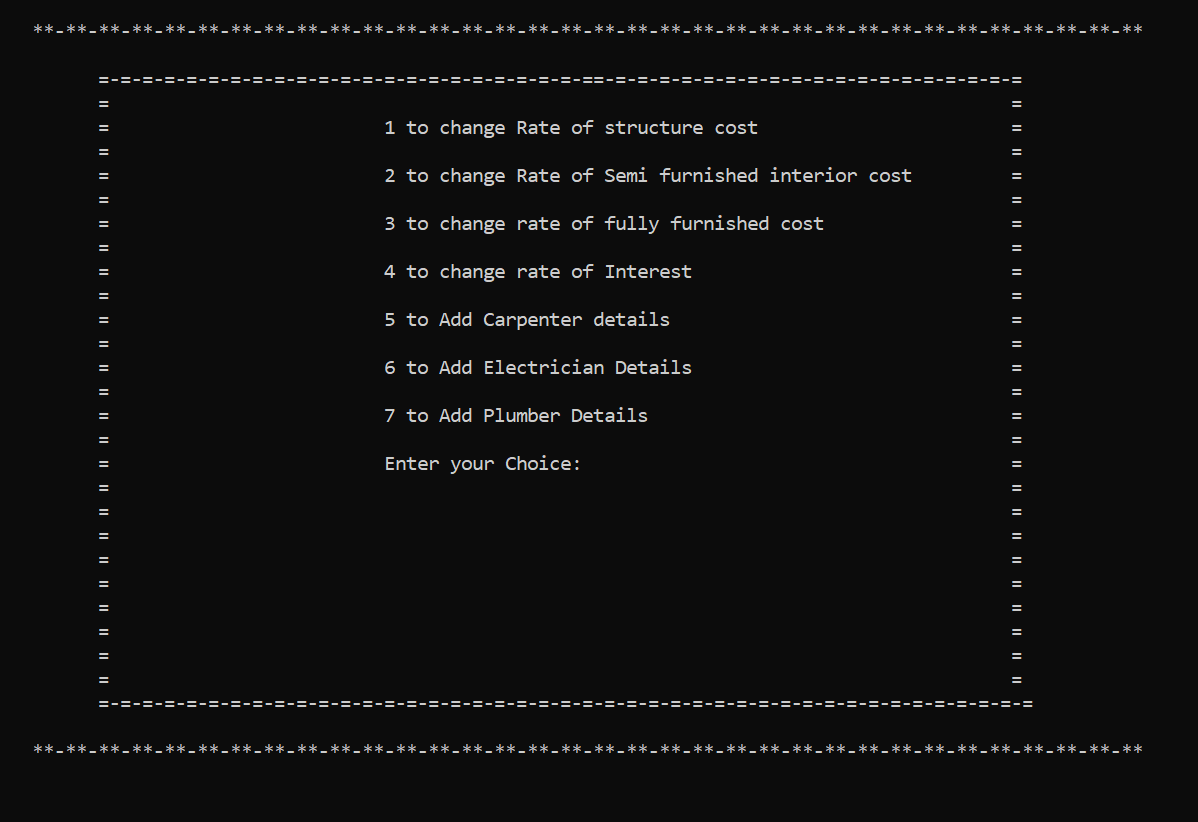
## 4.1. Home Page



## 4.2. ADMIN TEST CASE RESULTS

**Test Case 1**

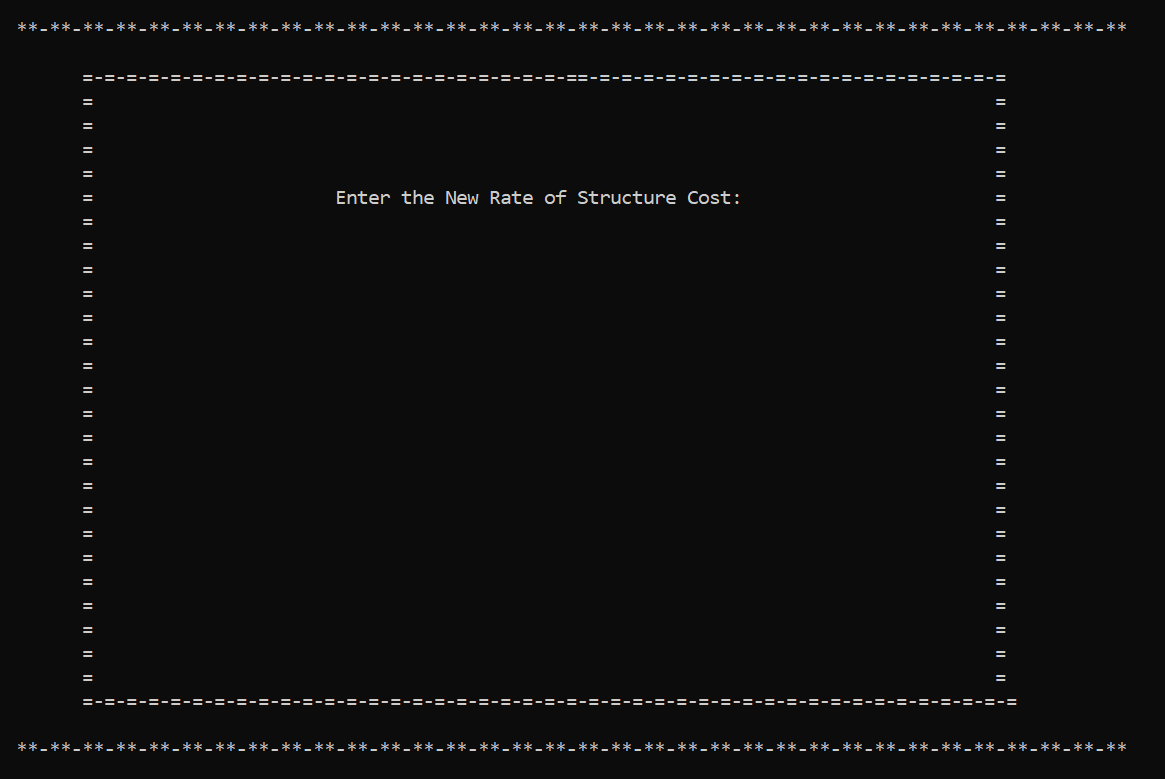


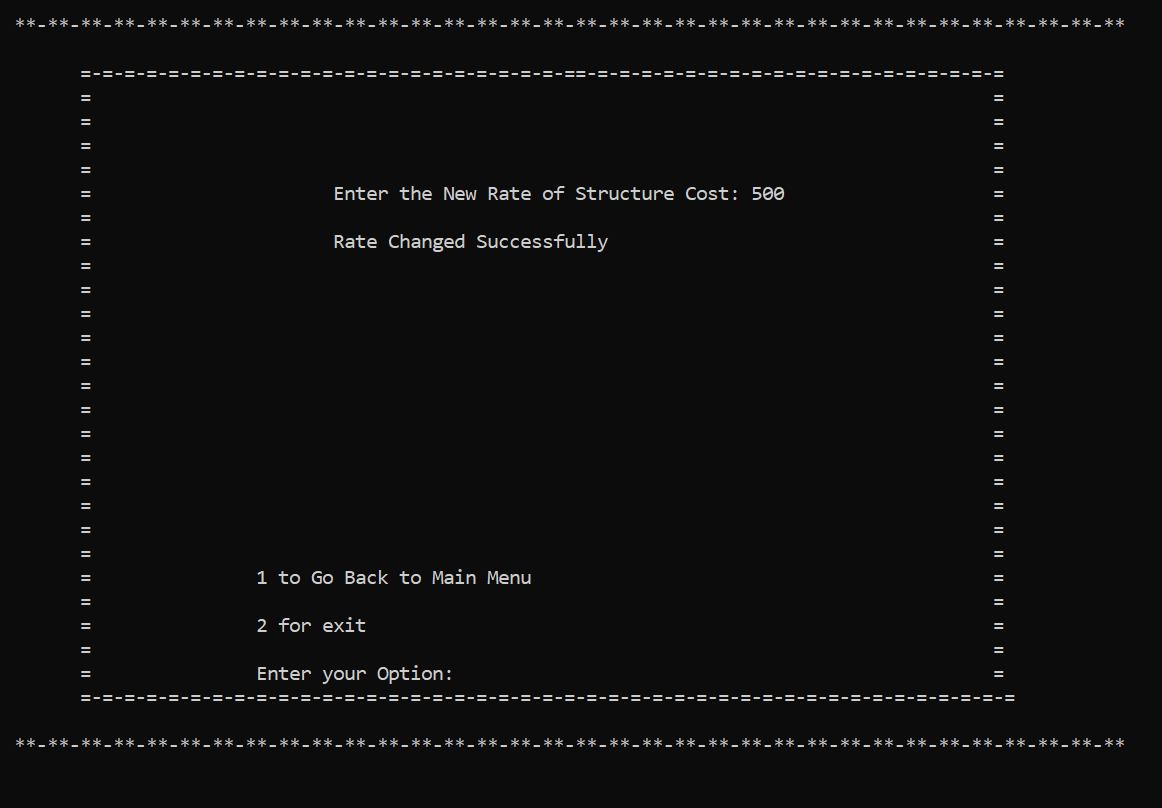


**Test Case 2**

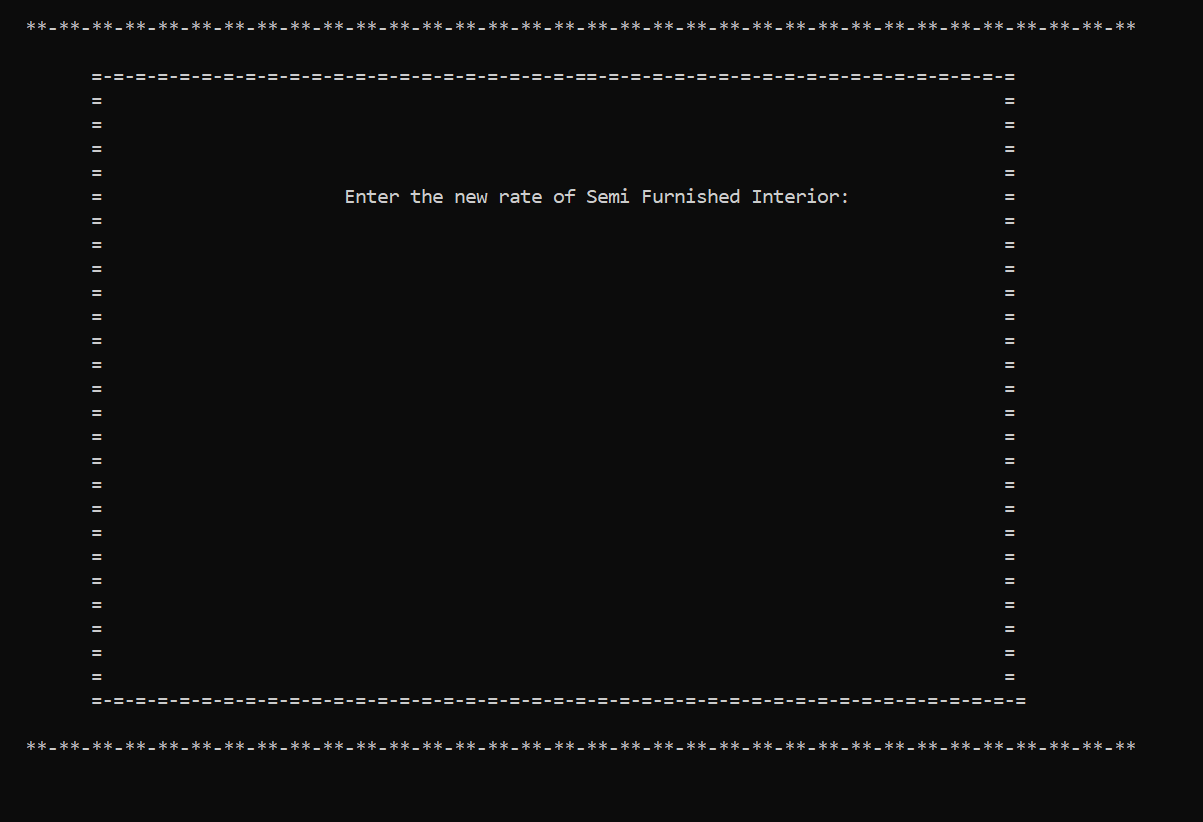


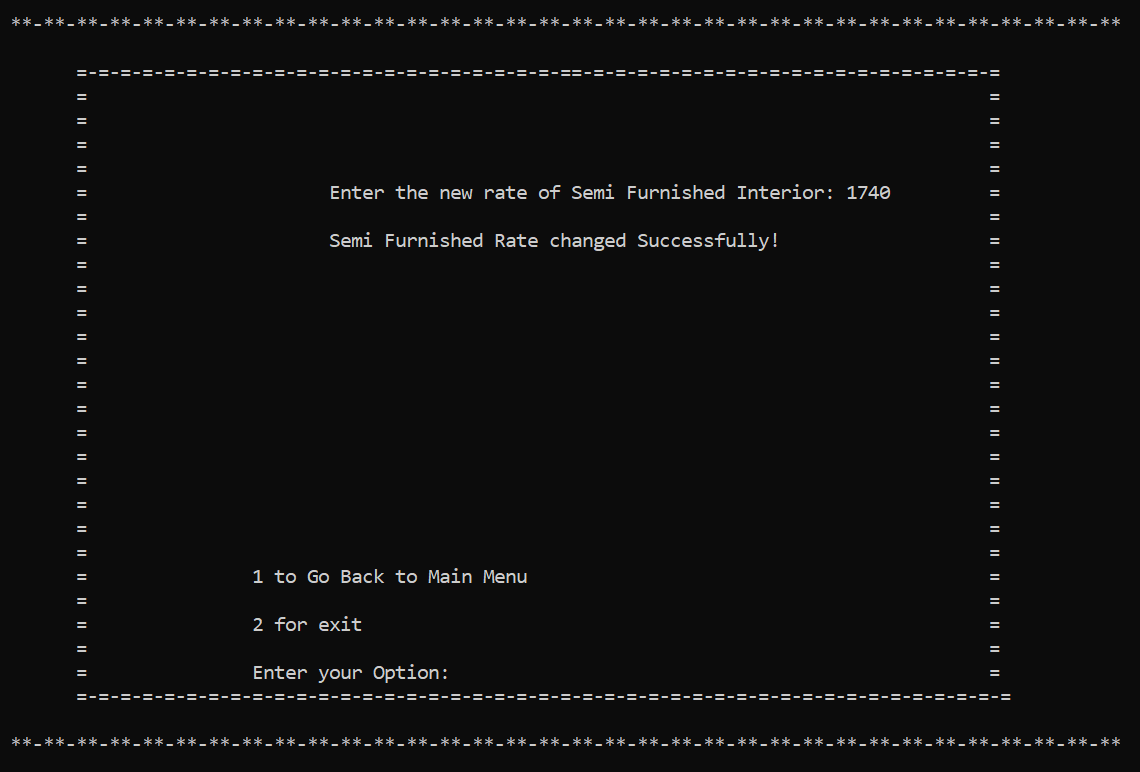
**Test Case 3**



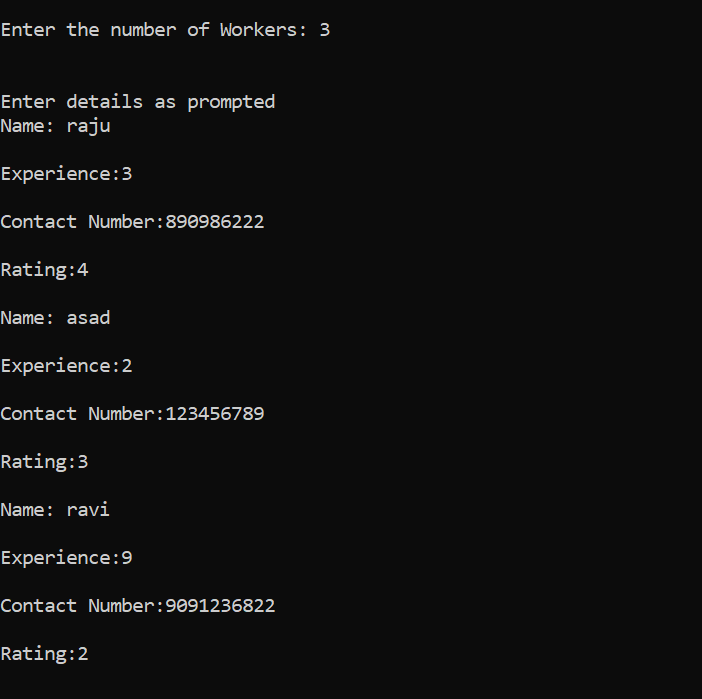


**Test Case 4**

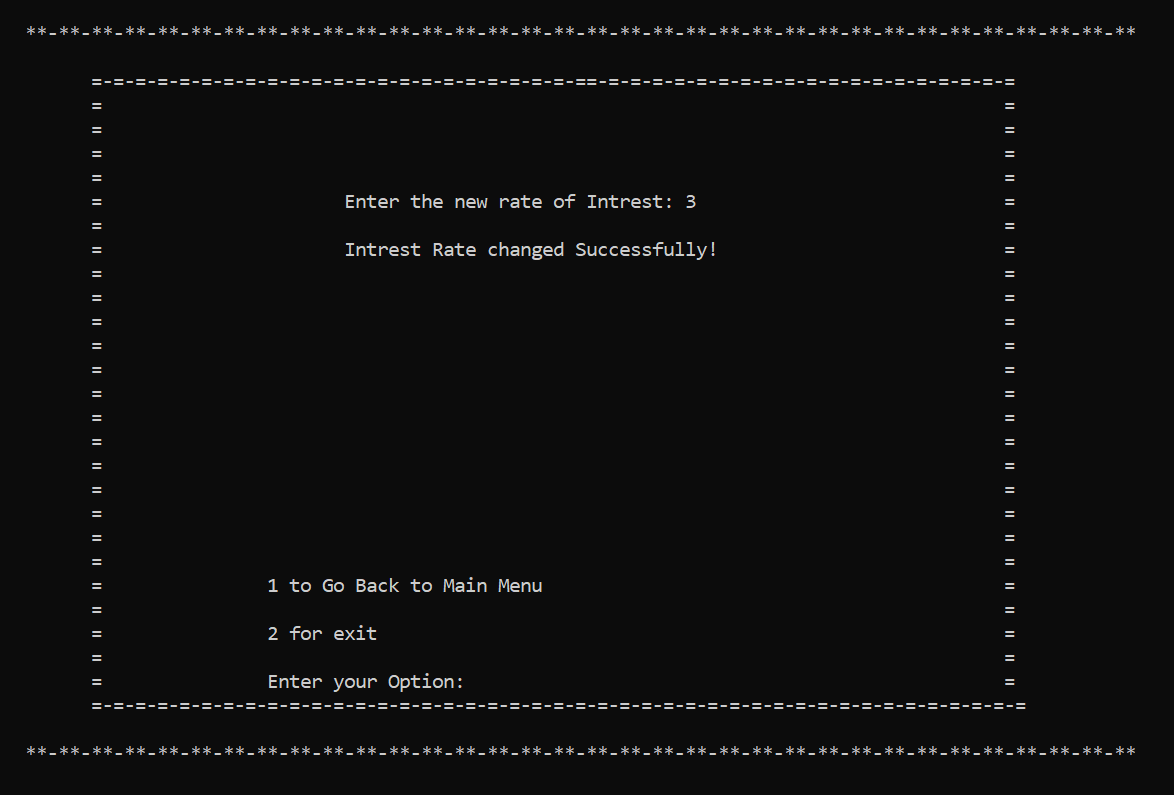




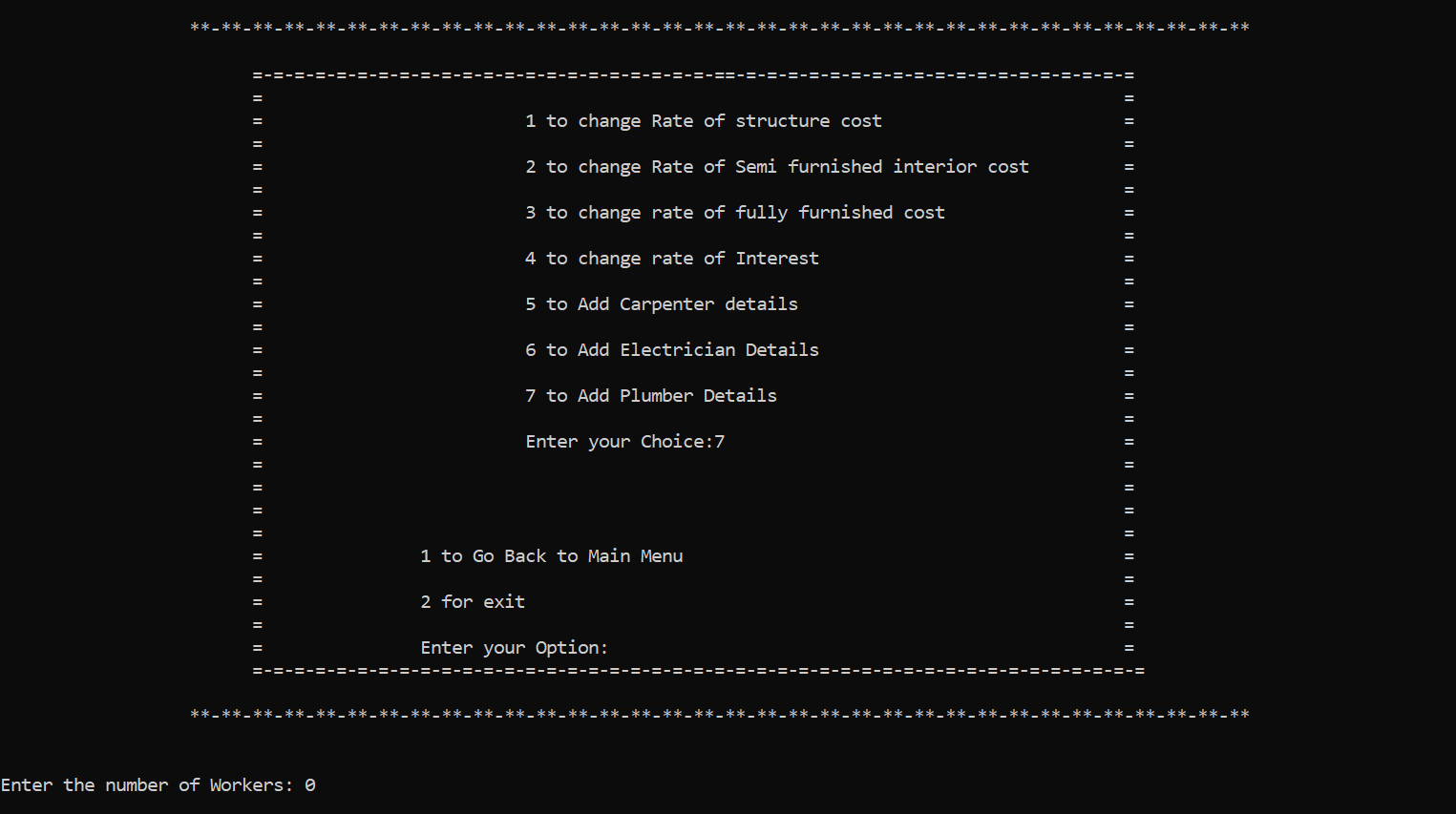
**Test Case 5**



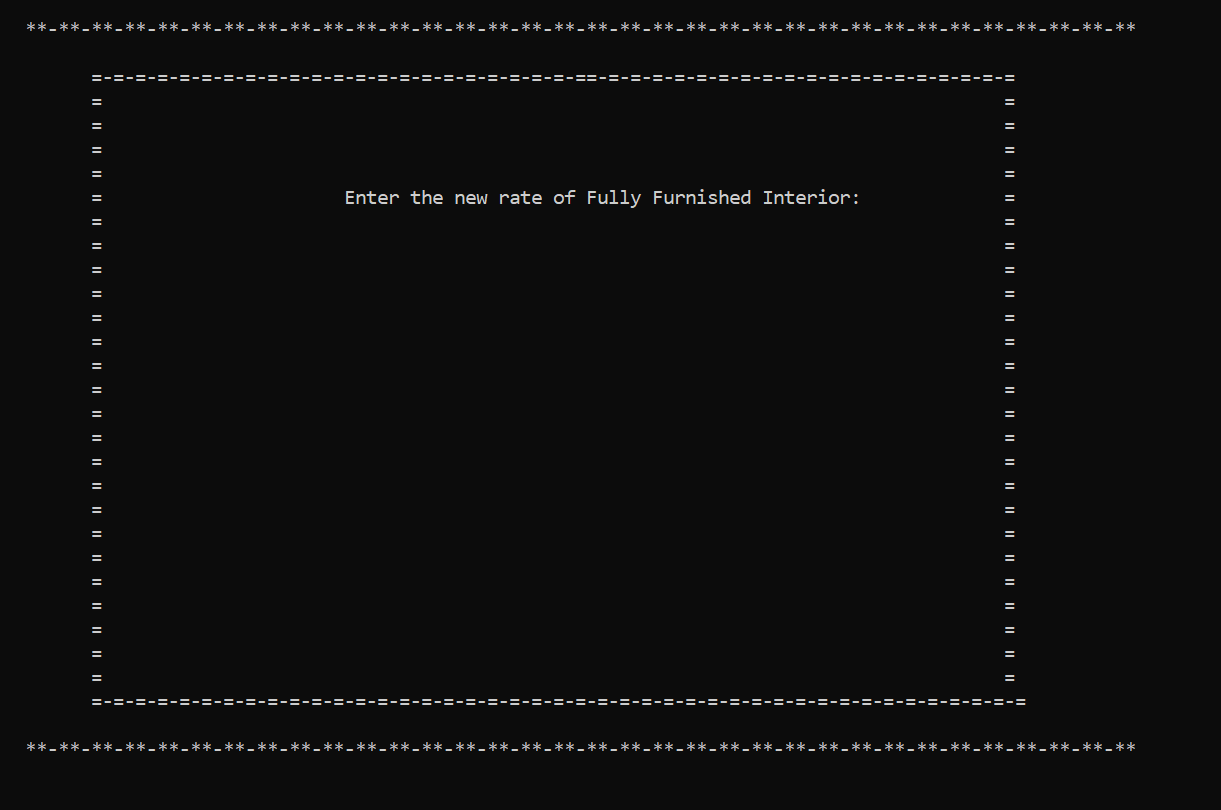
**Test Case 6**



**Test Case 7**

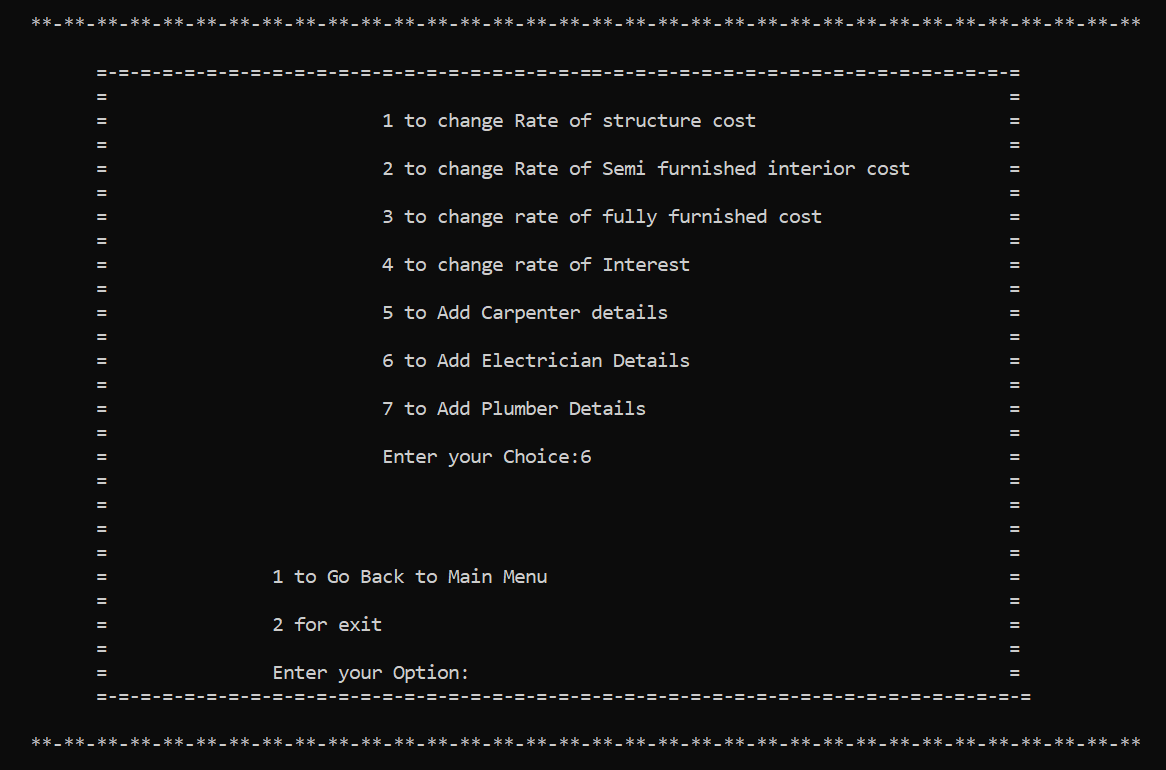


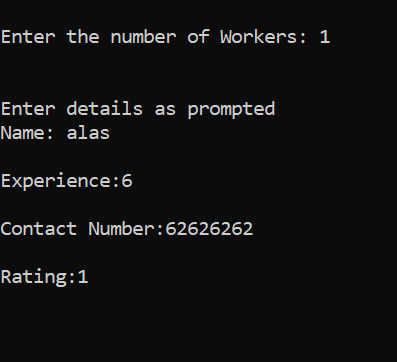
**Test Case 8**



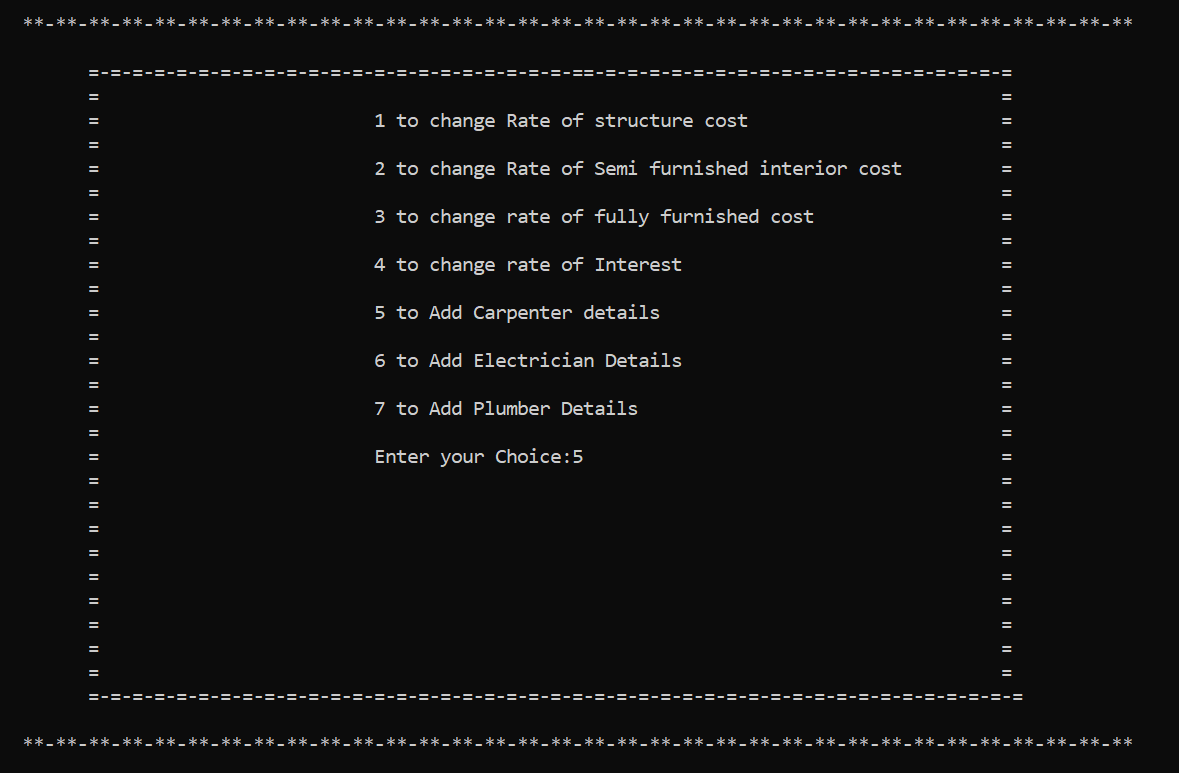


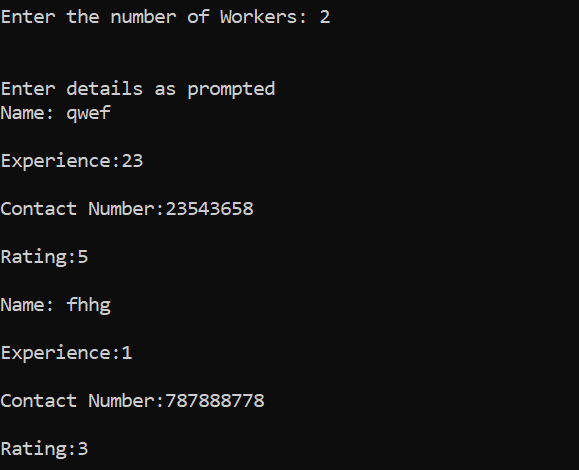
**Test Case 9**





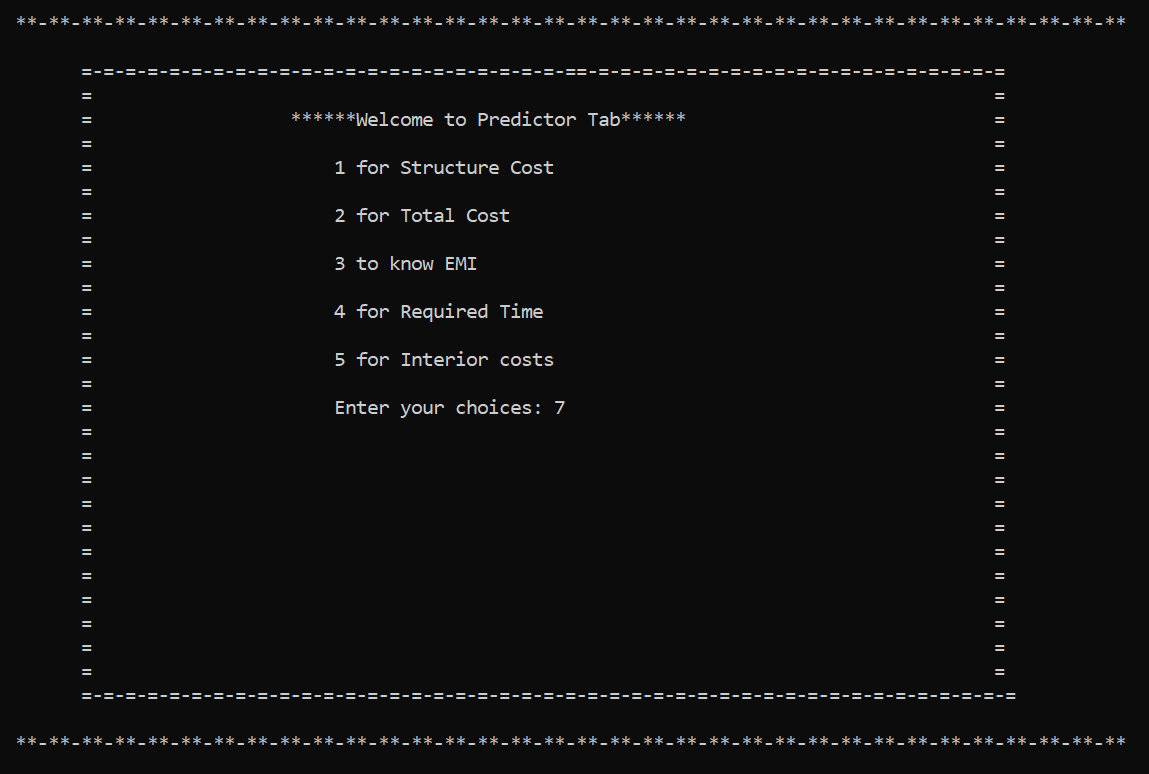
**Test Case 10**

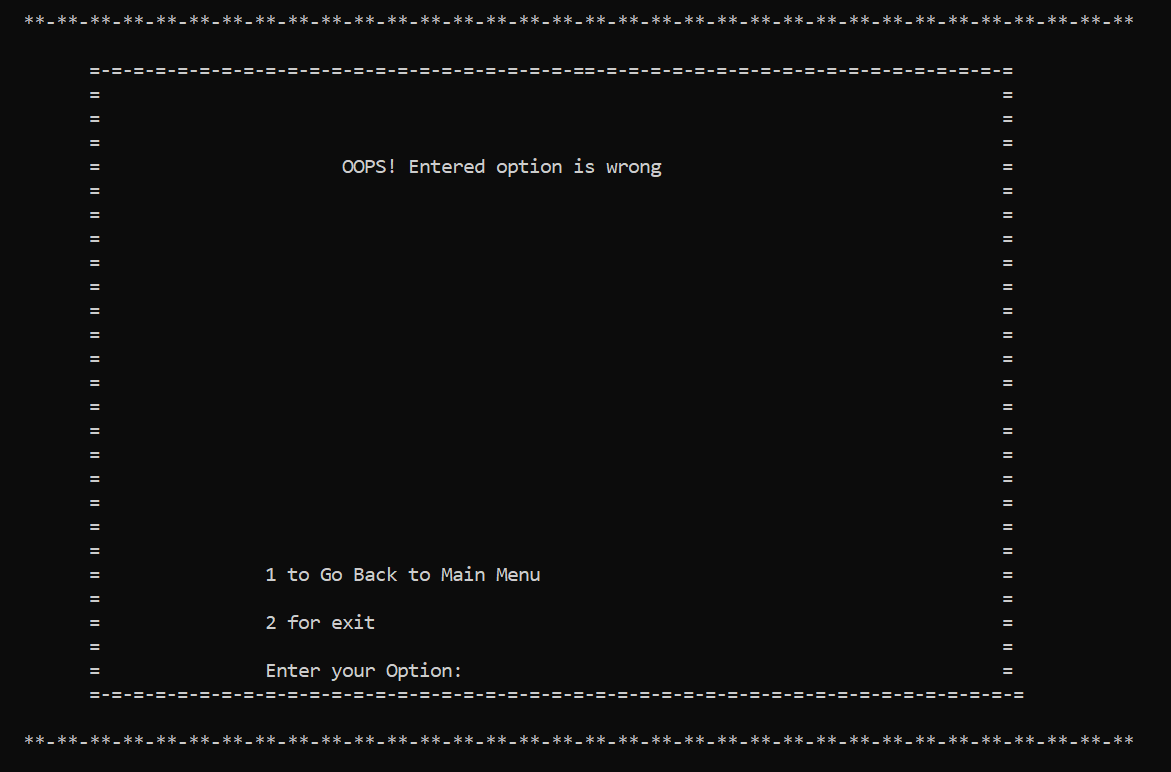




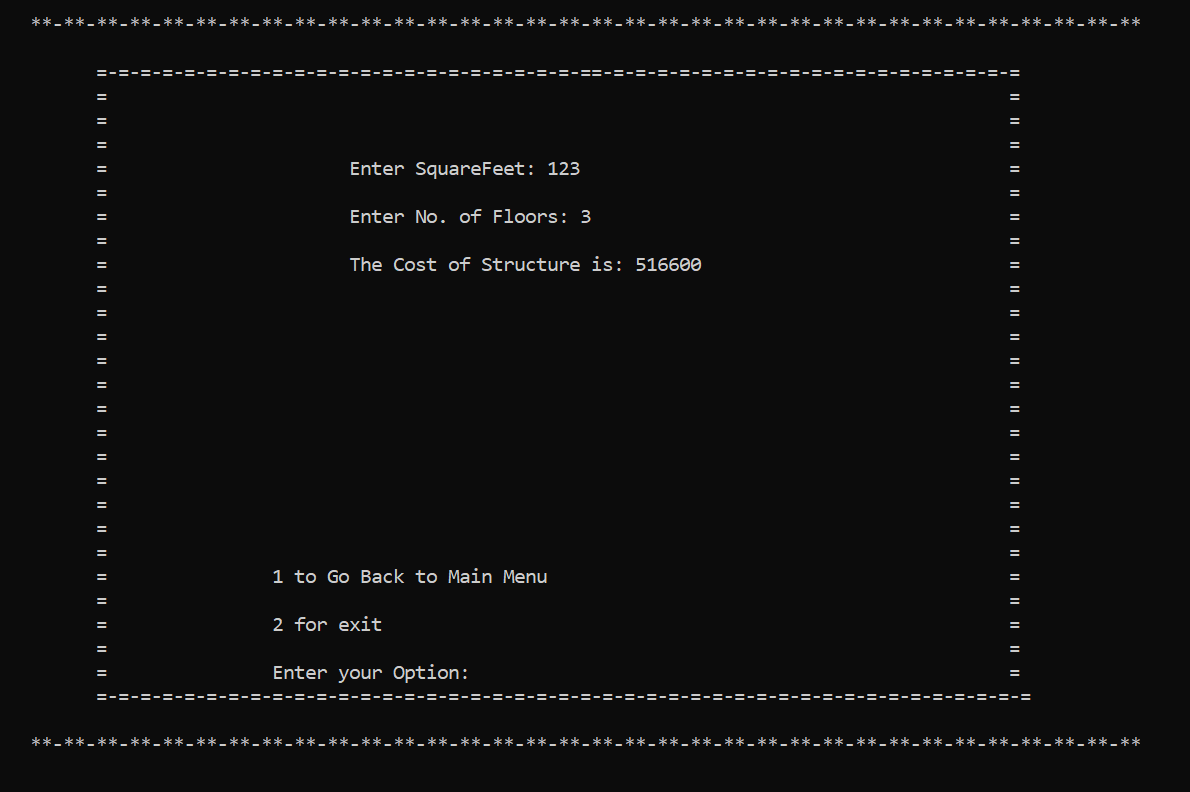
## 4.2. USER TEST CASE RESULTS

**Test Case 11**

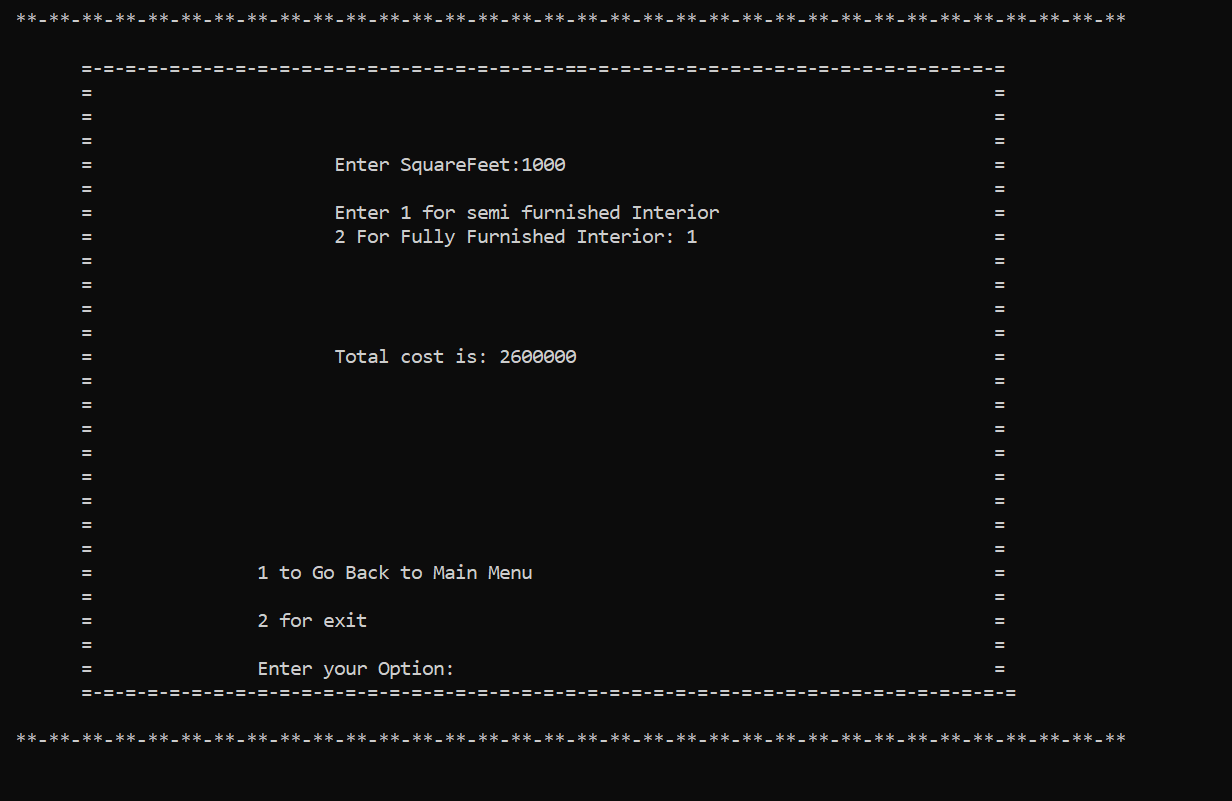




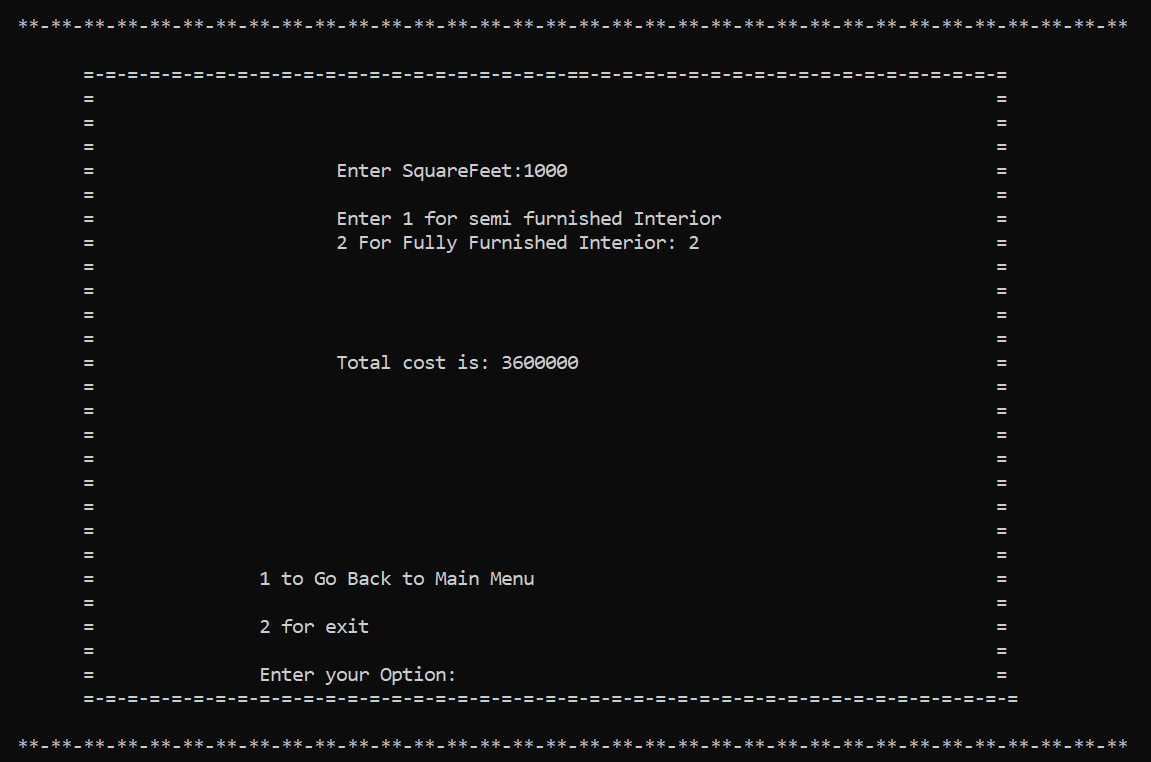
**Test Case 12**



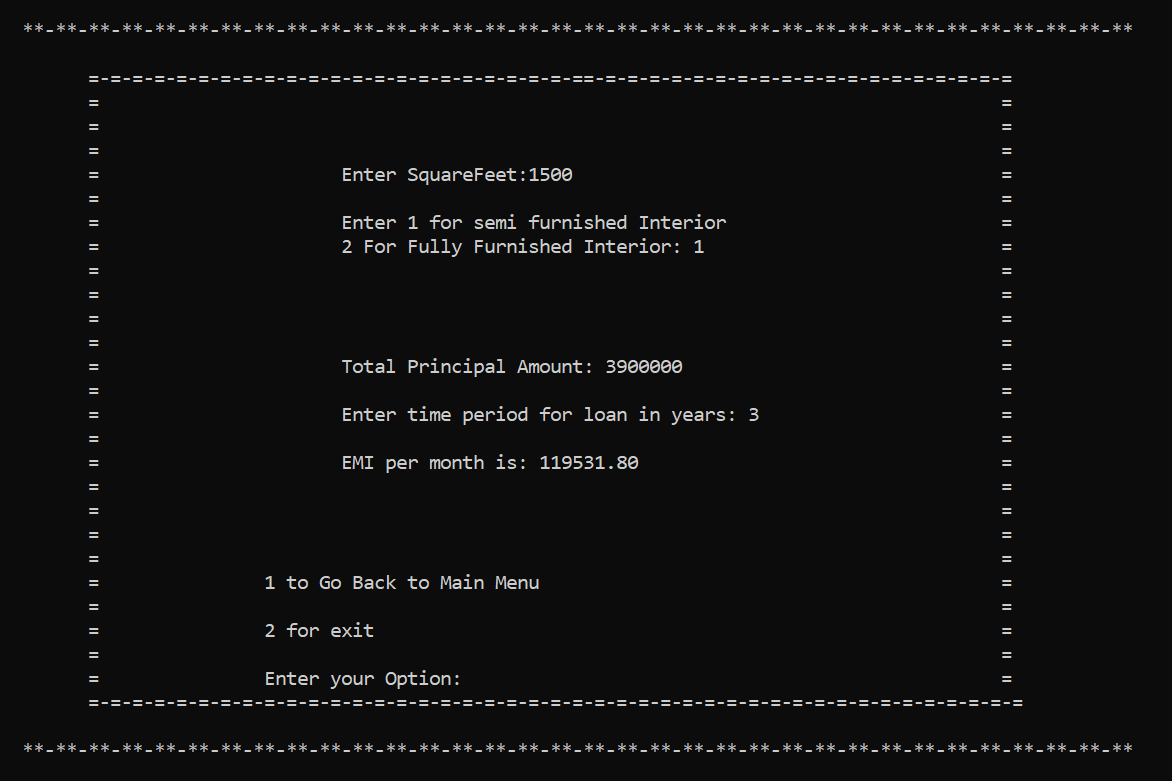
**Test Case 13**



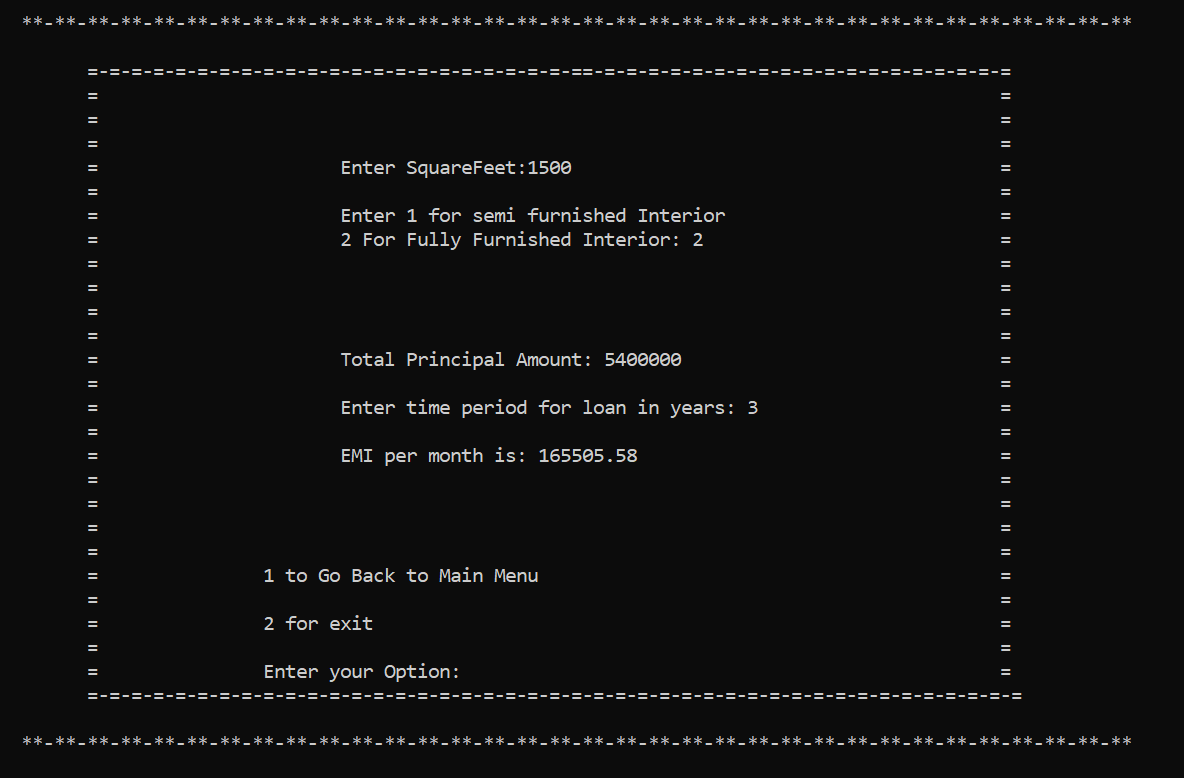
**Test Case 14**



**Test Case 15**



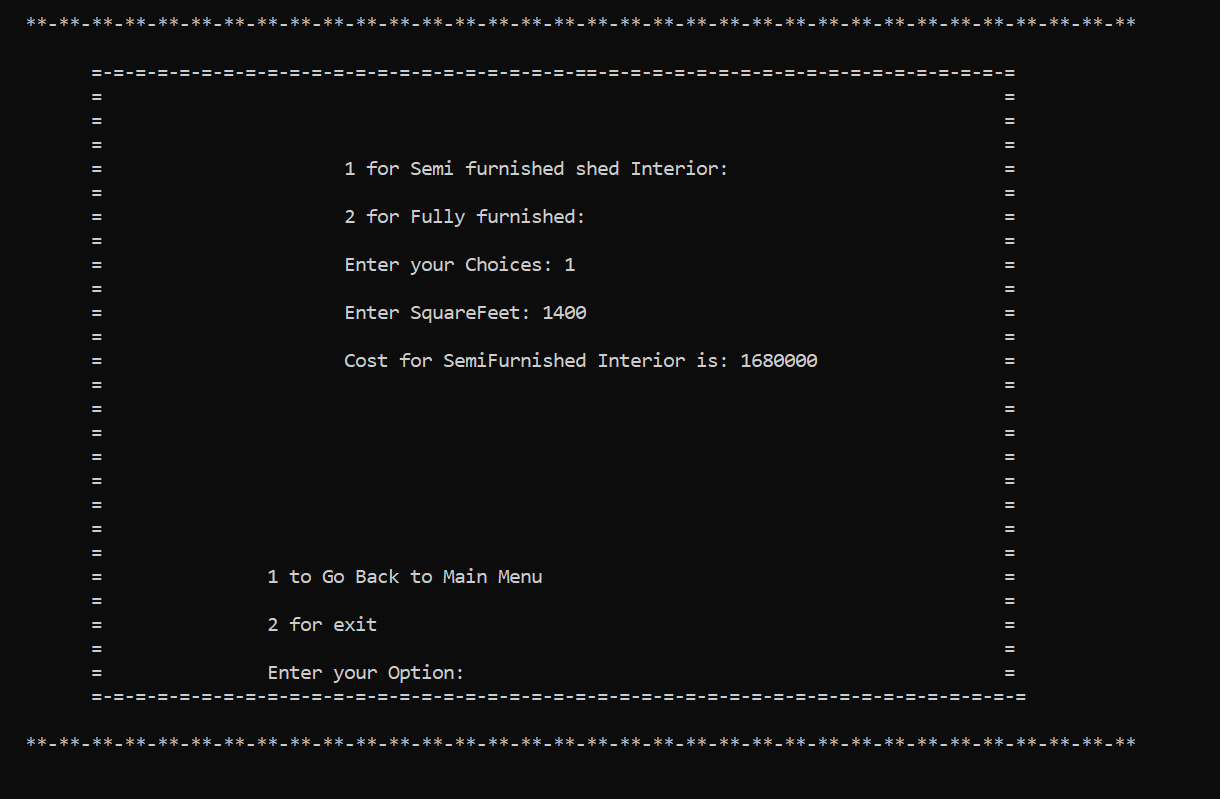
**Test Case 16**



**Test Case 17**

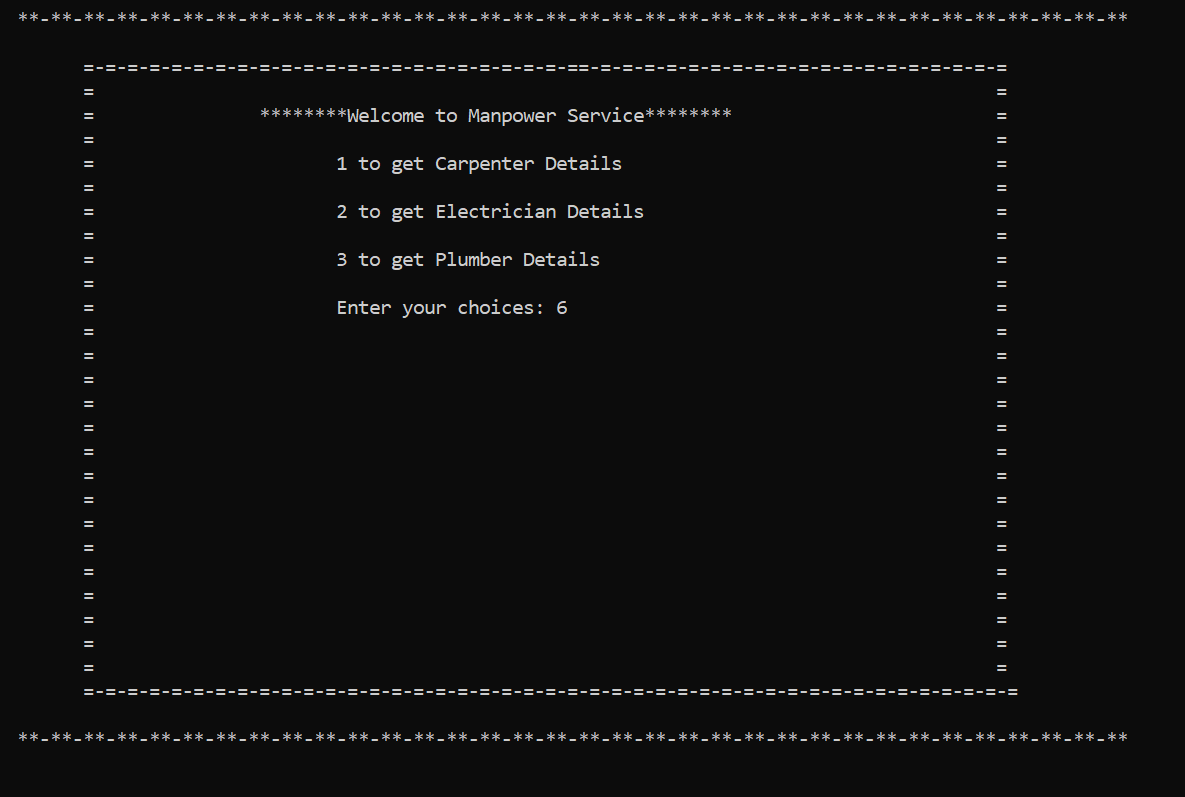


**Test Case 18**



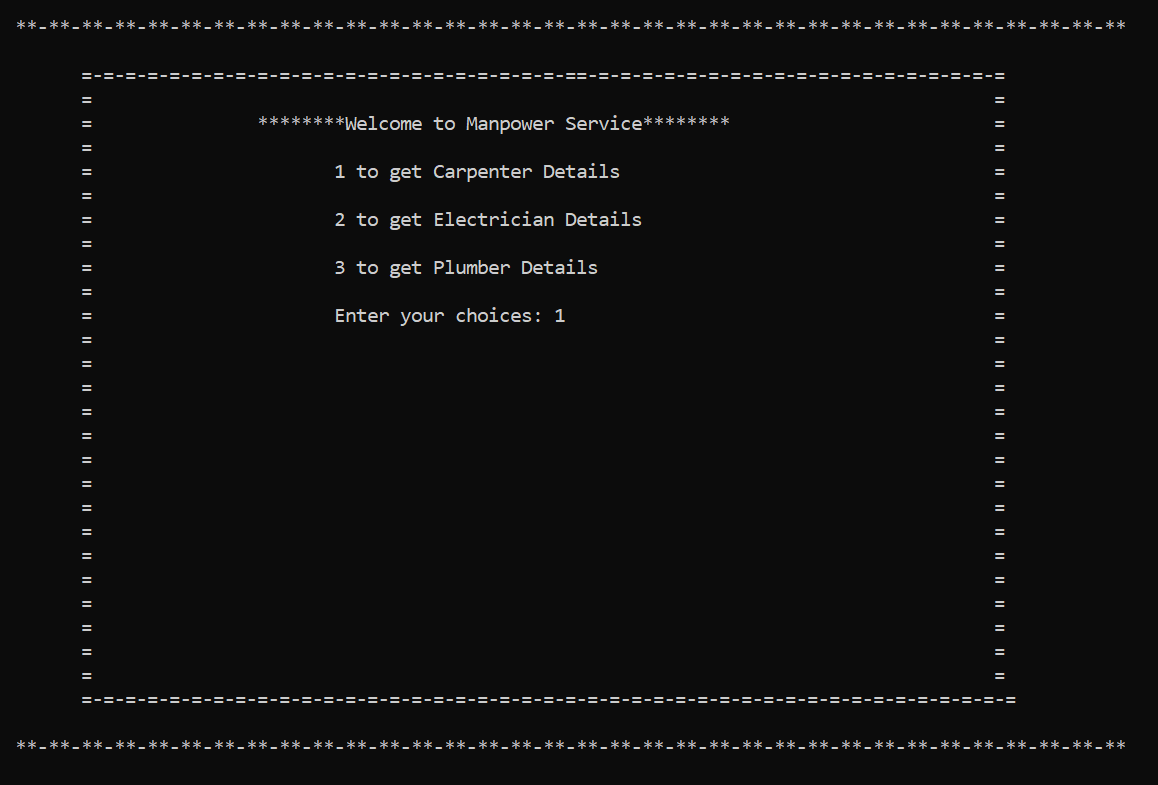
**Test Case 19**

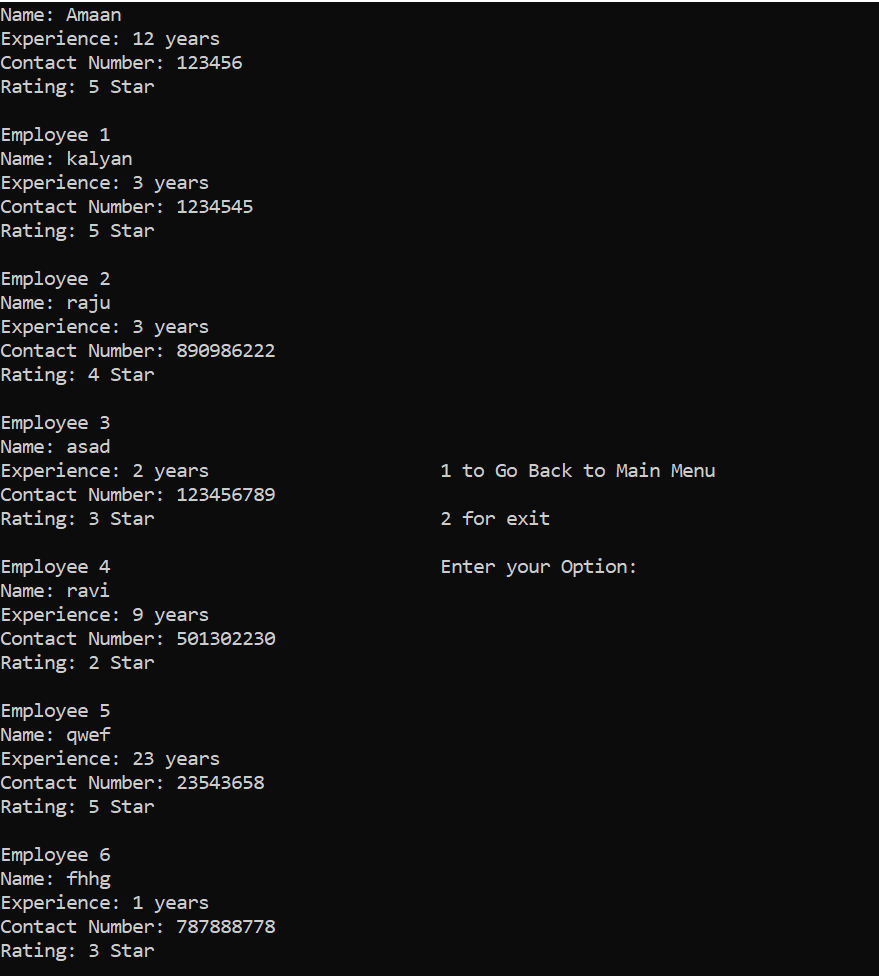
**Test Case 20**



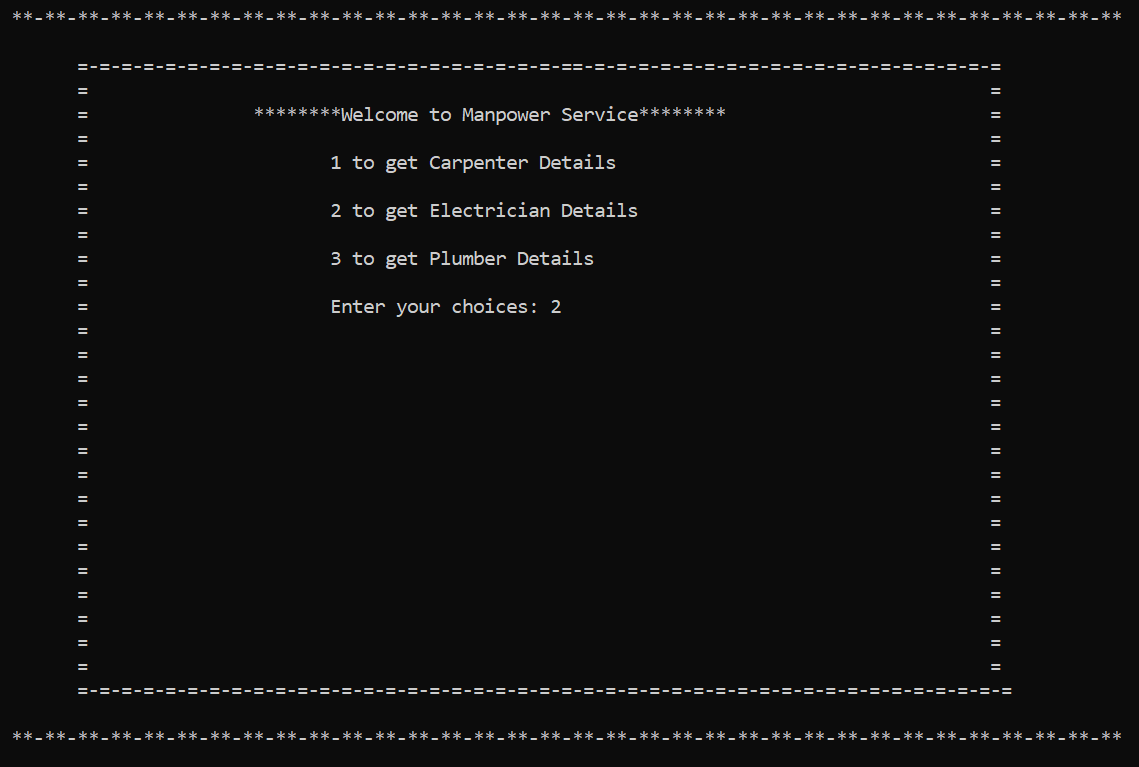


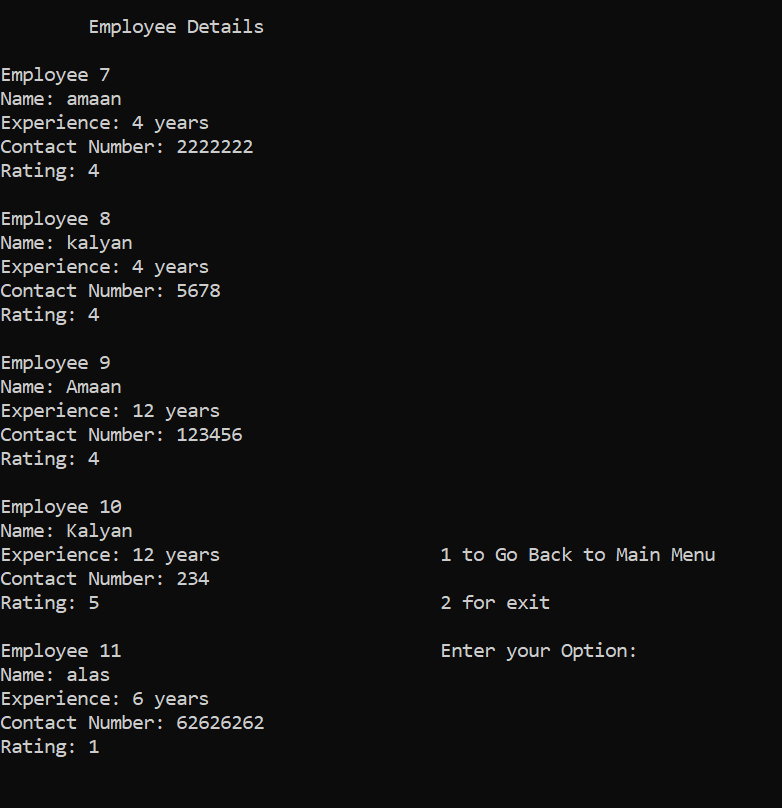
**Test Case 21**



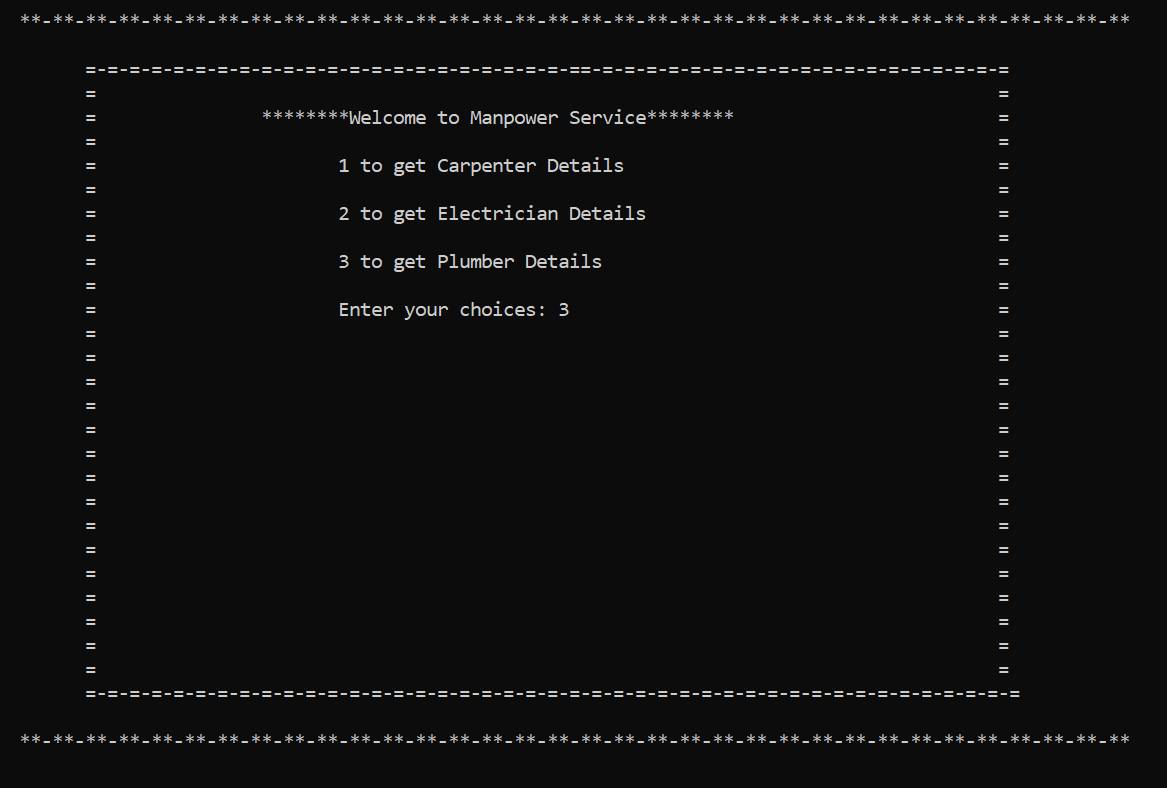


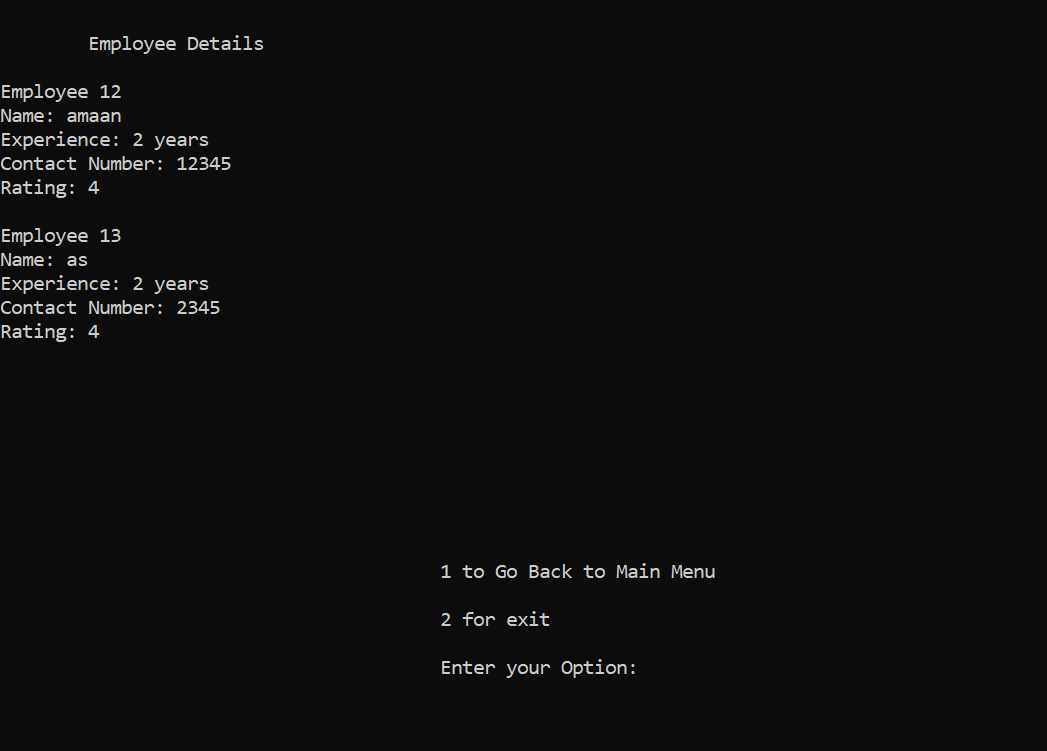
**Test Case 22**



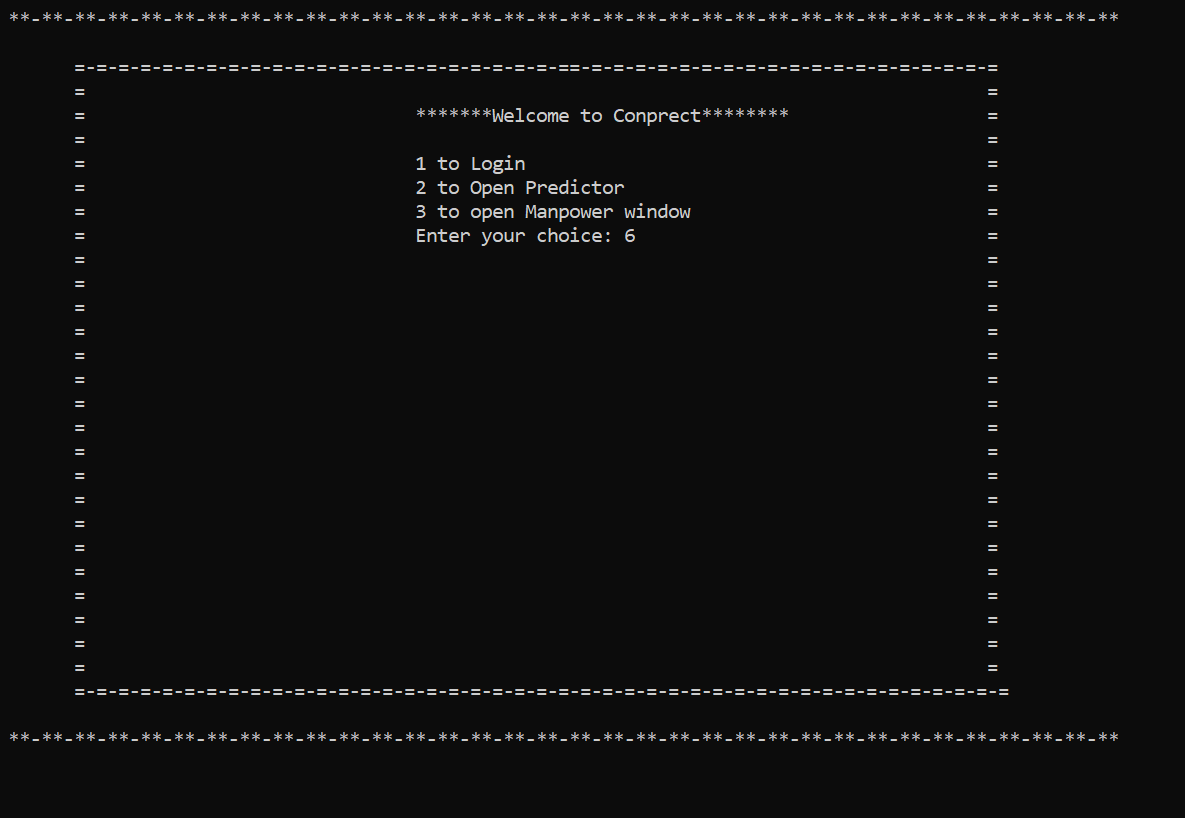


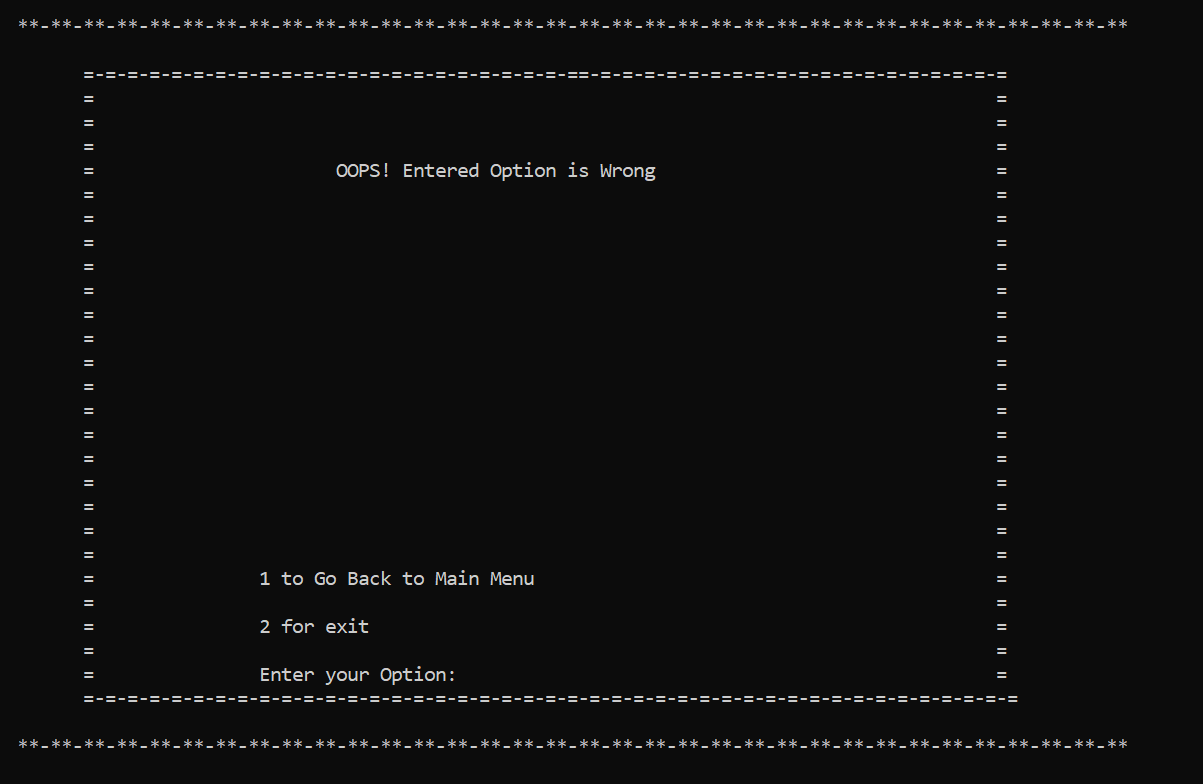
**Test Case 23**





**Test Case 24**





# 5. ADDITIONAL KNOWLEDGE ACQUIRED

Implementing this project in C Language has introduced us to different libraries such as: ‘conio.h’ and ‘windows.h’. We were able to use the knowledge we have on the File Handling and execute it in real-time application. We used the ‘windows.h’ library for controlling the display in a controlled manner. We explored the ‘conio.h’ libraries for achieving a very refreshing look to our application.

Also, we have further improved our knowledge in structures and pointers. Due to which we got to know the real power of Pointers.

Other than this, we have learnt the value of Java and have understood the intention of java as a Object Oriented Language.

# 6. CONCLUSION AND FUTURE WORK

To conclude, we have built a platform in which users can improve their skill-set in important Computer Science-based concepts such as C, Python and OOPs. The intention behind this project was to enhance our knowledge in these crucial subjects and further provide the same to all users of our platform. We also wanted to inculcate the practise of Self-Learning, that is, without guidance of professors or institutions. Our motive is for students to take ownership of their learning and additionally build independence.

Our future work includes by using AI to get the best output by teaching the system using Machine Learning and then adding features like Interior Designs which is generated when we give the required details with all required Construction Drawings with the help of the same ML. We can get the best result by training with different test cases possible. Then this would then become the complete portfolio of construction predictions.

This project can be further improved by changing this console-based application to web and mobile based application using Python and the Django Framework and React Native respectively. Due to which many could connect and use our application.

# 7. REFERENCES

1. YouTube: [YouTube](https://www.youtube.com/?gl=IN)
2. C Language notes
3. Greeks for Greeks: [GeeksforGeeks | A computer science portal for geeks](https://www.geeksforgeeks.org/)
4. Stack Overflow: [Stack Overflow - Where Developers Learn, Share, & Build Careers](https://stackoverflow.com/)
5. Tutorials Point: [C Tutorial - Tutorialspoint](https://www.tutorialspoint.com/cprogramming/index.htm)