STIA1123 PROGRAMMING 2

SESI 2020/2021 (SESSION A202)

GROUP C

PROJECT REPORT

SPORT BUSINESS / EVENT

PREPARED FOR:

Prof. Madya Dr. Azman B Yasin

PREPARED BY:

NAME	MATIC NO.
Muhamad Hazman Bin Haris	278632
Muhammad Hidayat Bin Ibrahim	279395
Pee Foo Ye	278632
Muhammad Syarif Bin Dollah	279271

DATE SUBMISSION:

2021

Company Background

Background of the Project

Our main project is based on the sport business event. There are various types of sport in Malaysia and not just a physical sport, indoor sport and e-sport for the person who not active in physical activities provided. Our group are managing bowling centre business, futsal centre business, gaming centre business and also racing car business. Each person from our group are assigned to a different sport and also a different location to make sure each place in our country have sport facilities. The main reason our group starting this business is to build a society that not just focus on academic but also focus on their health whether it is mental or physical health. Each of the sport have different types of rules and also different way of playing. We divide each of the sport in a different centre to make it easier to participant to choose the sport that they like and want to participate. Each of the sport centre have their own purpose for developing this system.

1. Futsal Centre (Muhamad Hazman Bin Haris 278632)

Our company is called Megah Futsal, established on August 2, 2017. Our company location is in Pekan Jitra 2, 06000, Jitra, Kedah. The location of our company also has some areas that are the focus of the public around our company such as Tesco supermarket and Jitra market.

Currently, our company has 5 employees who are able to manage futsal competitions well. The purpose of the establishment of our company is to run a great futsal competition, the winning prizes offered are also very lucrative and can attract the attention of customers. In addition, we also want to provide a place for futsal players to showcase their talents. Of course, our target customers are people who are interested in playing futsal competitively. Coinciding with our company slogan which is “Your TOP choice futsal tournament management”, therefore we warmly welcome you to participate in futsal competitions organized by our company.

For more information about our company or the competition, you can contact us by email at megahfutsal07@gmail.com, or call 04 - 0208007. You can also visit our website,

www.MegahFutsalEventManagement.com to see the activities that our company runs such as futsal competitions that we have done before.

2. Car Racing (Pee Foo Ye 278632)

Pro MotorSport Enterprise is a company which conducts car racing business. The company has been setup since 20 May 2019, the location is at 1760 Permatang, Simpang Ampat, 30300 Kedah and the business hour on Mondays to Friday are from 10.00 A.M. – 9.00 P.M. and the company will be closed at weekend. Every customer who wants to be contestant have to fulfill these requirements which are possess a Nation Competition License and good health situation. There are 4 types of racing will be held in the company which are Formula racing, Sports car racing, Stock car racing, and Drag car racing. Besides that, Pro MotorSport Enterprise will also provide several types of vehicles for each racing type. For Formula Racing, the company will provide "Alfa Romeo Ferrari C39", "Renault RS20 ", "Racing Point BWT Mercedes RP20" + "Williams Mercedes FW43". Next, the cars that provided by the company for the Sport car racing are "Mazda MX-5", "Porsche 911", "Alpine A110", "Porsche 718 Boxster/Cayman", "McLaren 570S", "Mercedes-AMG C 63 S". For "Declasse Burger Shot Stallion", "Bravado Redwood Gauntlet", "Vapid Pisswasser Dominator", "Bravado Sprunk Buffalo" are provided for Stock car racing. There are also several drag cars are provided, which are "Chevrolet Camaro ZL1", "Buick Grand National GNX", "Chevrolet Corvette C7 Z06", "Mustang GT500", "Chevrolet Nova". The company currently have 7 employees which consist of 1 manager, 1 assistant, 1 admin, 1line Stiper, 1 vehicle paint sprayer, 1 security, 1 maintenance manager.

3. Gaming (Muhammad Hidayat Bin Ibrahim 279395)

E-sports (also known as electronic sports, e-sports, or eSports) is a form of competition using video games. E-sports often takes the form of organized, multiplayer video game competitions, particularly between professional players, individually or as teams. Although organized competitions have long been a part of video game culture, these were largely between amateurs until the late 2000s, when participation by professional gamers and spectatorship in these events through live streaming saw a large surge in popularity. By the 2010s, e-sports was a significant factor in the video game industry, with many game developers actively designing and providing funding for tournaments and other events. Following the trend of the E-sports, Melio Enterprise was founded by

Hidayat. Our company provides customers the chance for them to compete with each other with E-sports to win the glory and prize.

The company is located at Lot G-10, Menara Amanah Ikhtiar,Jalan Cempaka SD 12/1A,Bandar Sri Damansara, PJU 9,52200 Kuala Lumpur . There are quite a few departments of employees in this company for example the manager, event host, technician, admin and more. And Melio Enterprise are currently still recruiting. For those who interested in participating the events or working as our partner, please kindly contact 03-46654432(Compony) or E-mail to melio@gmail.com for any inquiries.

Furthermore, the participants can kindly use the sport event business system to register themselves to participate the events organised by Melio Enterprise or surf the information about the recent events using the system. Meanwhile, the admin of Melio Enterprise can use the system to update the information about the events and employee. The admin can also use the system to update the finance details of the company and then calculate the cost and net profit of each event

4. Gaming (Muhammad Syarif Bin Dollah 279271)

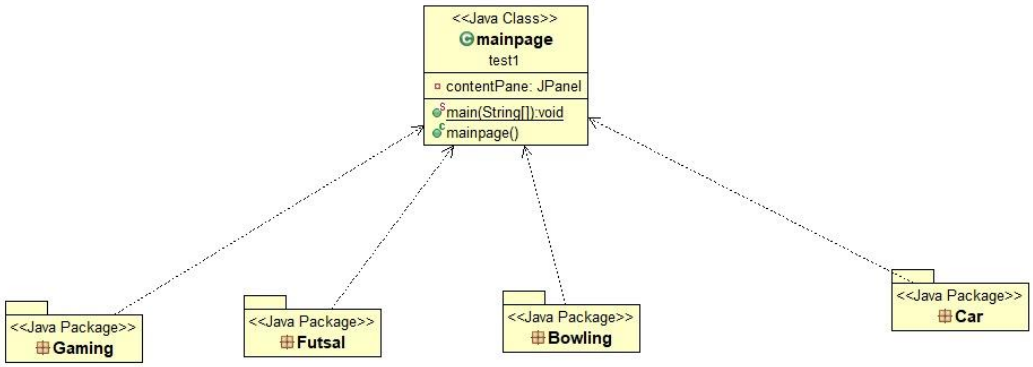
Bowling is a Team Sport Although bowling is usually seen on television as an individual sport, it is more common for bowling to be enjoyed as a team sport. Most bowling alleys have league competitions for teams every night of the week and leagues for children on the weekend. While individual bowlers compare their scores against other individuals, bowling teams simply add their scores together and compare that against the other teams cumulative score.

For this project, I make a system to develop bowling sport business. This bowling business is “Hunter Bowling Center” have been in business area for 3 years. Our sport center is more committed than ever to providing you with the best bowling experience possible. We hope our customer will enjoy everything with our facility and we look forward to helping our customer have a great time in future.

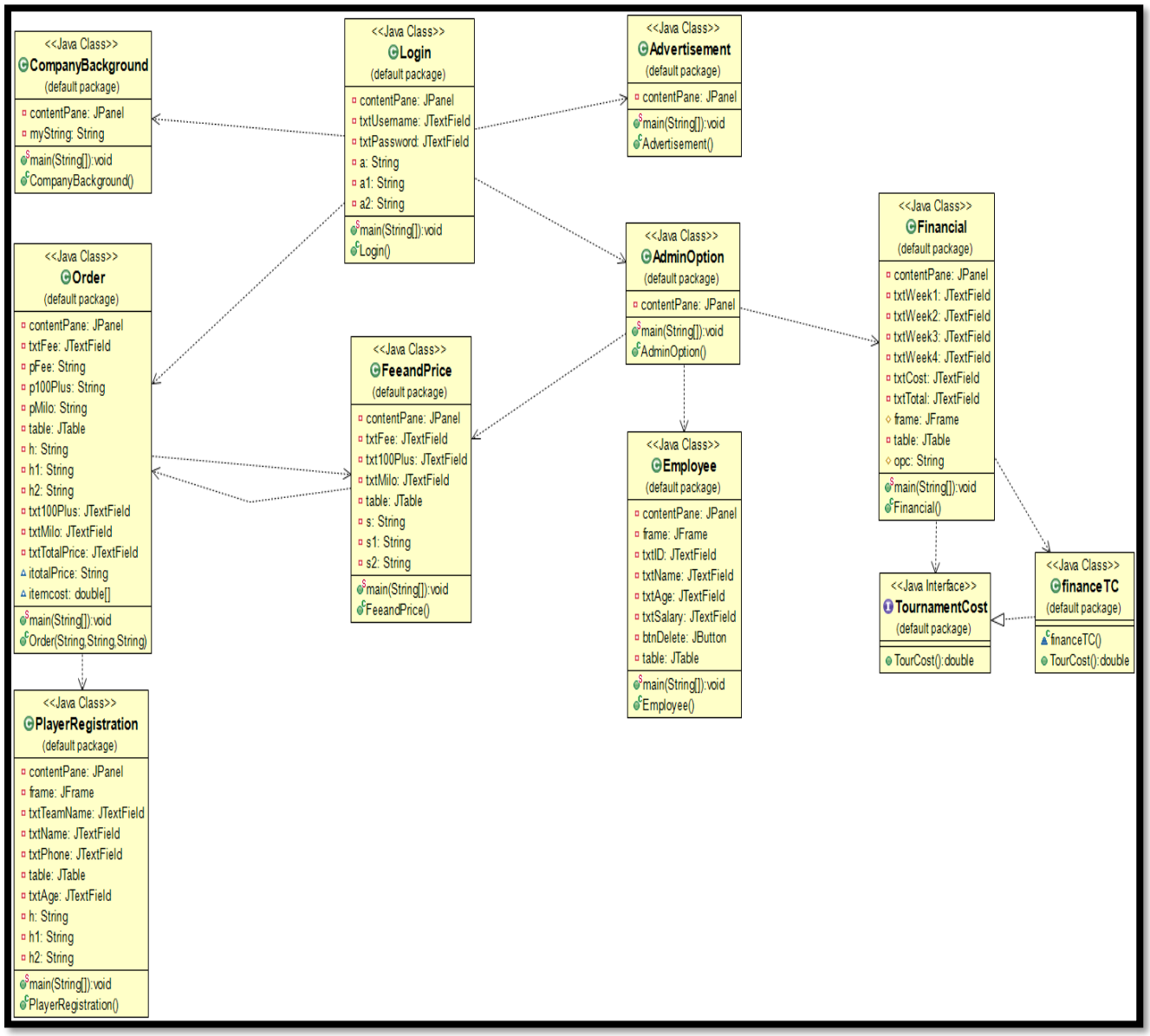
UML Diagram

Sports Event / Business

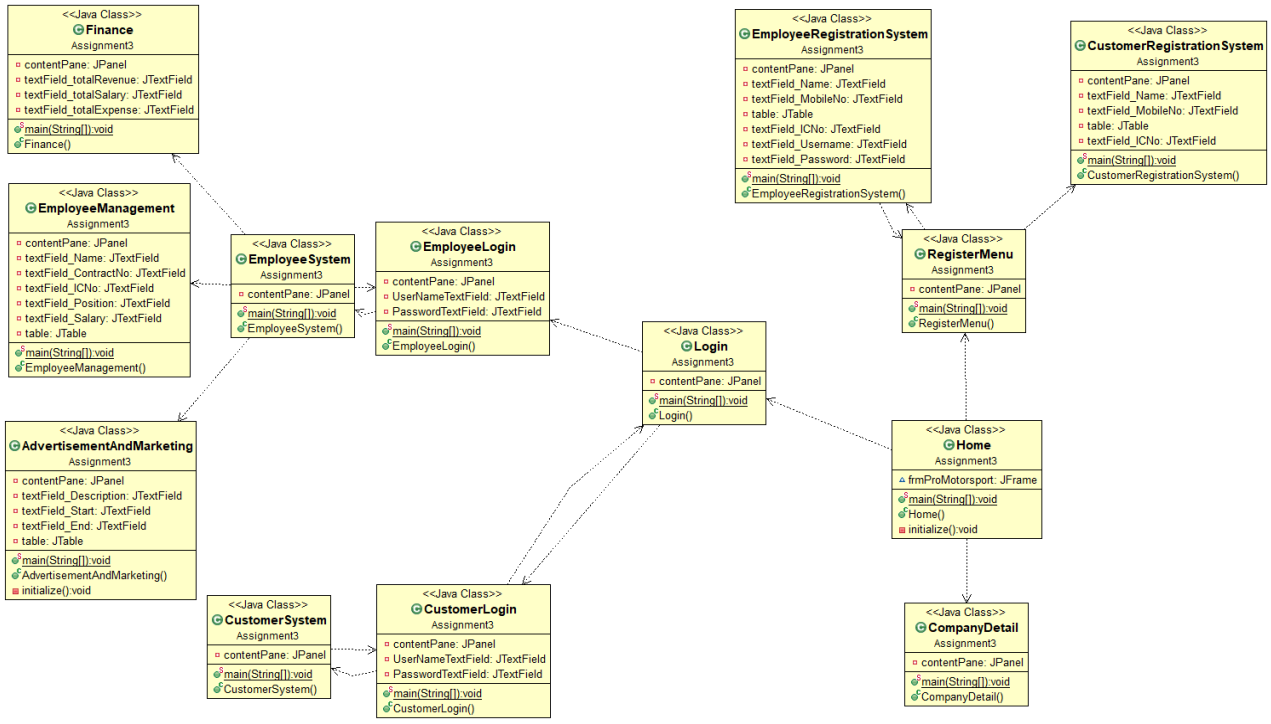
Main Page



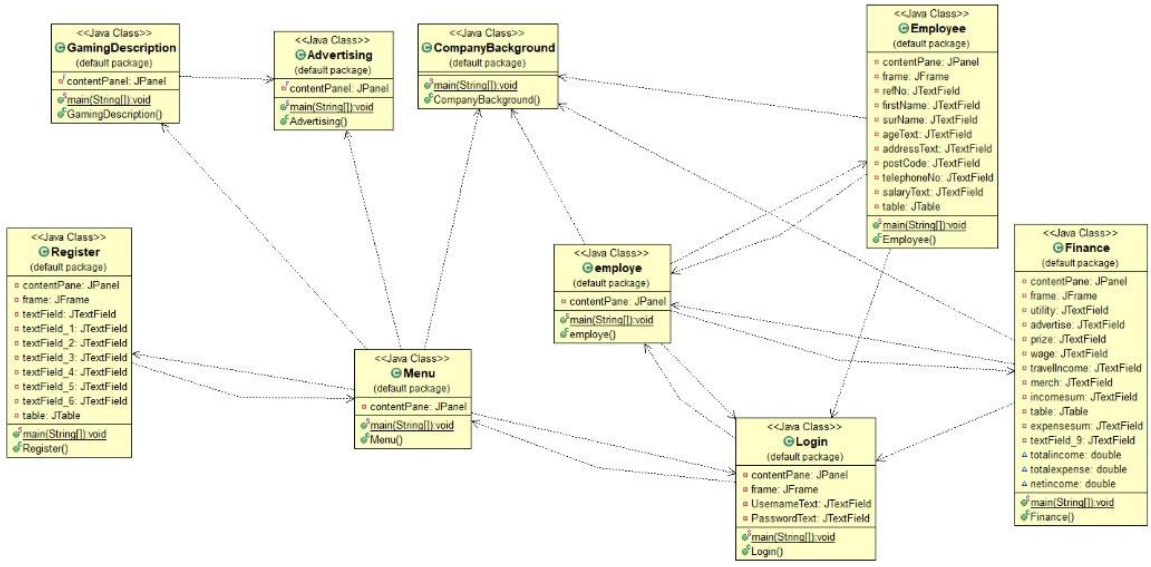
1. Futsal Centre (Muhamad Hazman Bin Haris 278632)



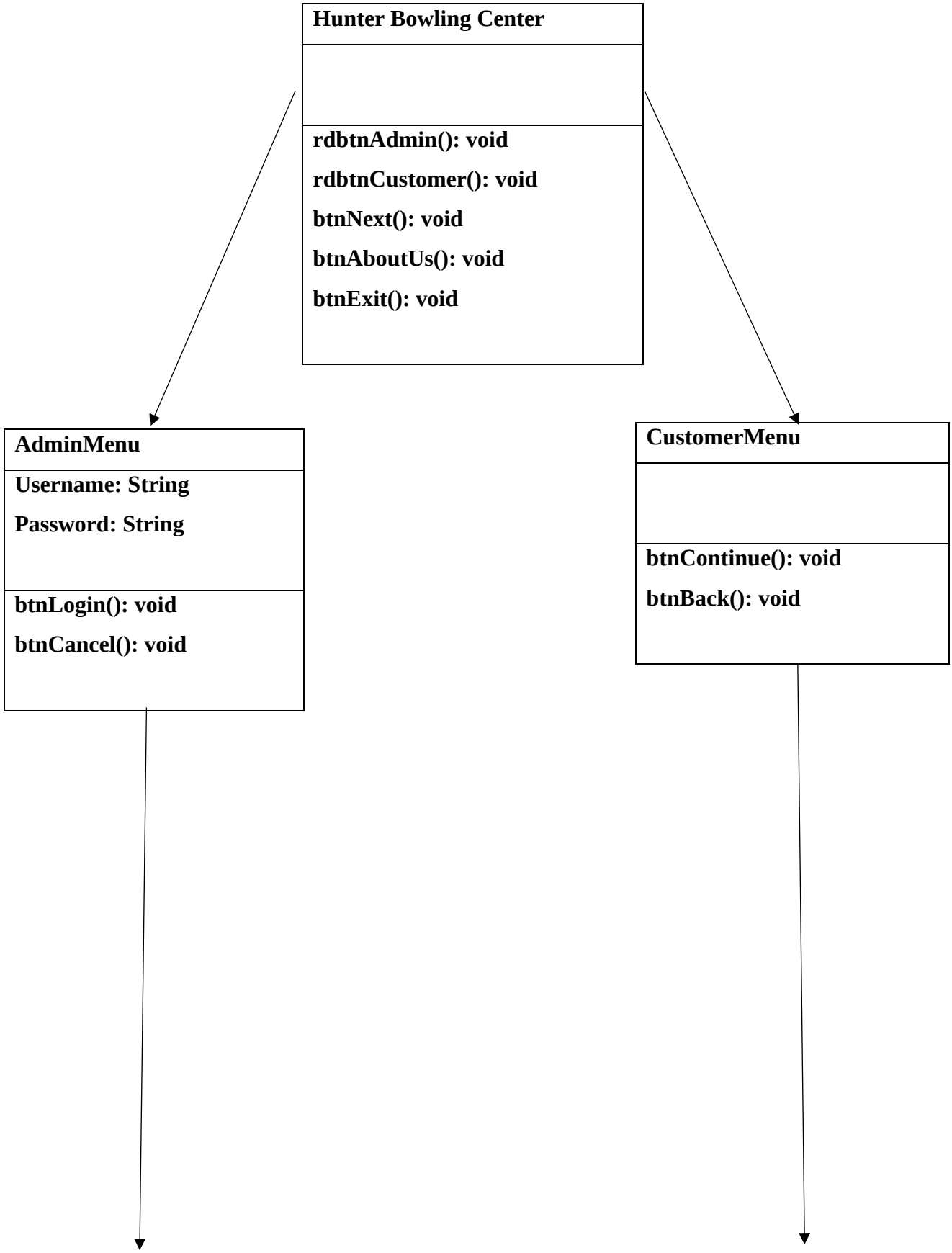
2. Car Racing (Pee Foo Ye 278632)

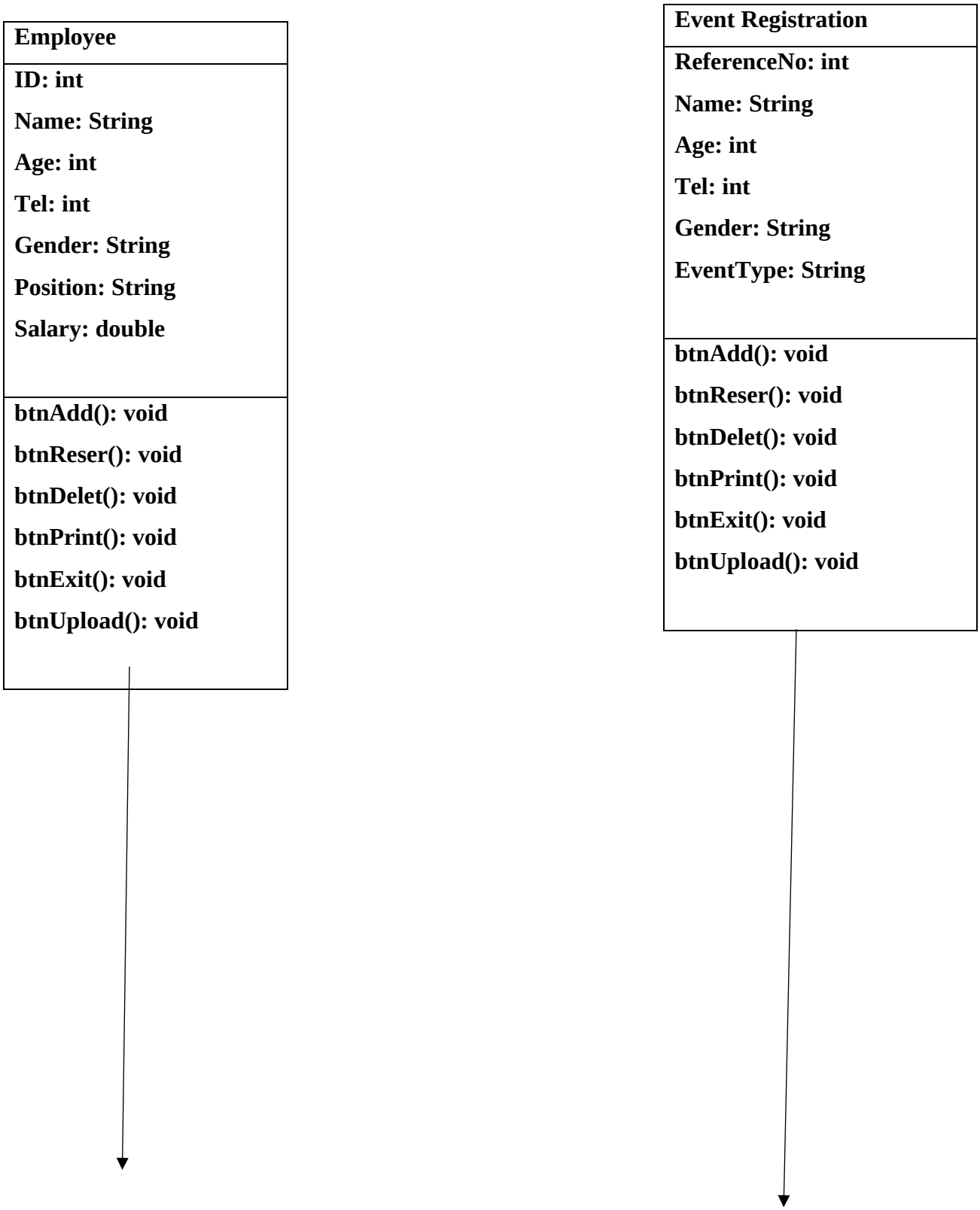


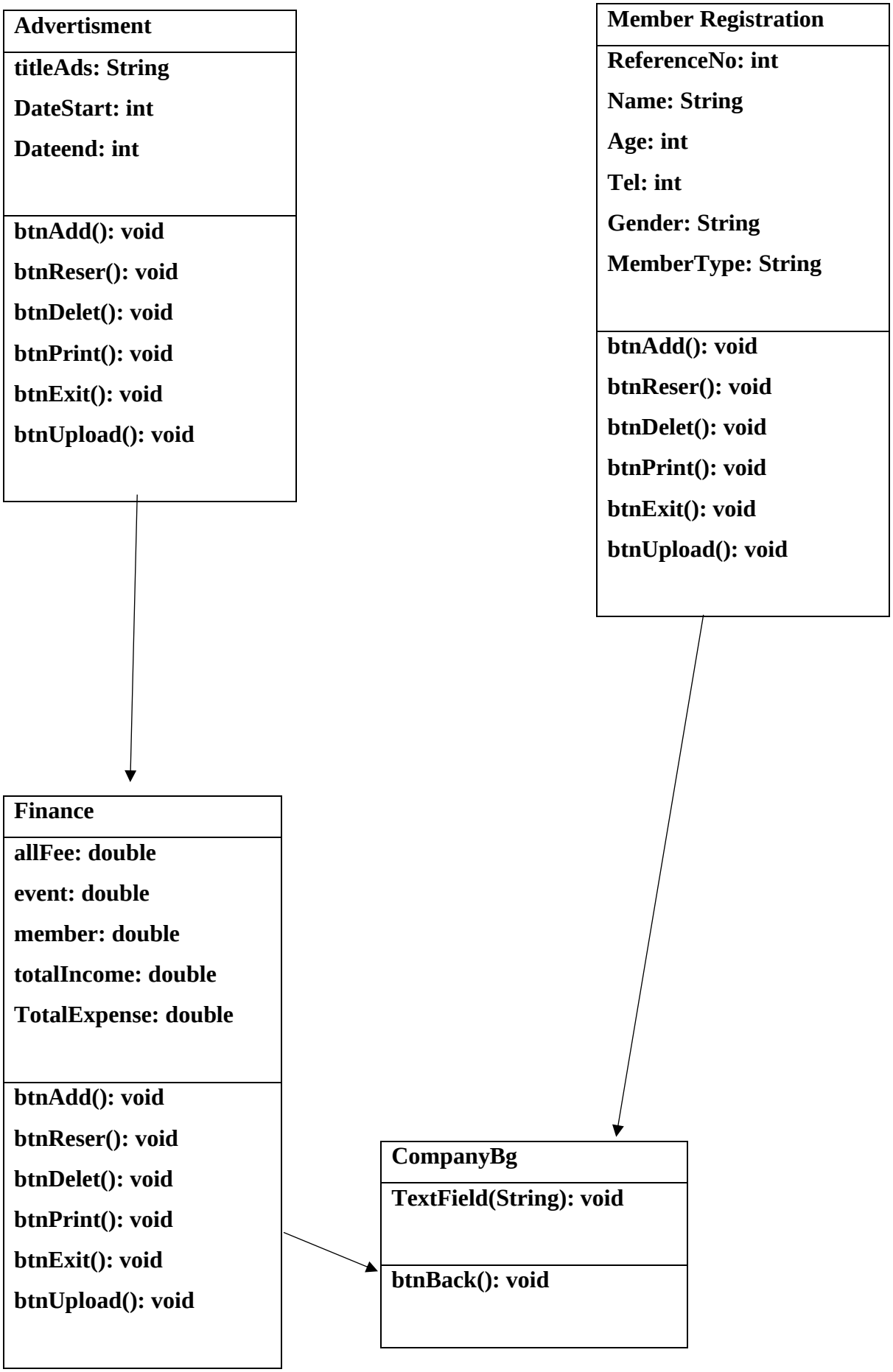
3. Gaming (Muhammad Hidayat Bin Ibrahim 279395)



4. Gaming (Muhammad Syarif Bin Dollah 279271)




Employee
ID: int
Name: String
Age: int
Tel: int
Gender: String
Position: String
Salary: double
btnAdd(): void
btnReser(): void
btnDelet(): void
btnPrint(): void
btnExit(): void
btnUpload(): void
Event Registration
ReferenceNo: int
Name: String
Age: int
Tel: int
Gender: String
EventType: String
btnAdd(): void
btnReser(): void
btnDelet(): void
btnPrint(): void
btnExit(): void
btnUpload(): void

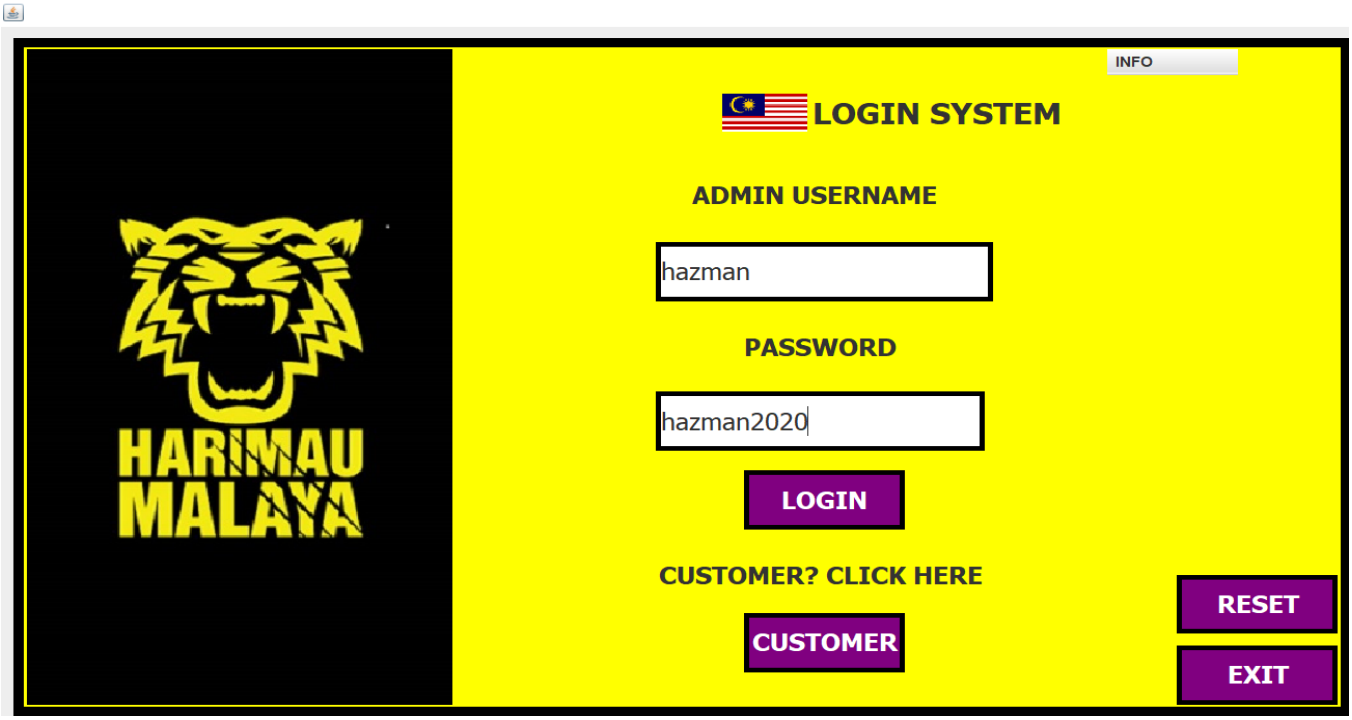

Advertisment
titleAds: String
DateStart: int
Dateend: int
btnAdd(): void
btnReser(): void
btnDelet(): void
btnPrint(): void
btnExit(): void
btnUpload(): void
Member Registration
ReferenceNo: int
Name: String
Age: int
Tel: int
Gender: String
MemberType: String
btnAdd(): void
btnReser(): void
btnDelet(): void
btnPrint(): void
btnExit(): void
btnUpload(): void
Finance
allFee: double
event: double
member: double
totalIncome: double
TotalExpense: double
btnAdd(): void
btnReser(): void
btnDelet(): void
btnPrint(): void
btnExit(): void
btnUpload(): void
CompanyBg
TextField(String): void
btnBack(): void

Program Description

1. **Futsal Centre (Muhamad Hazman Bin Haris 278632)**

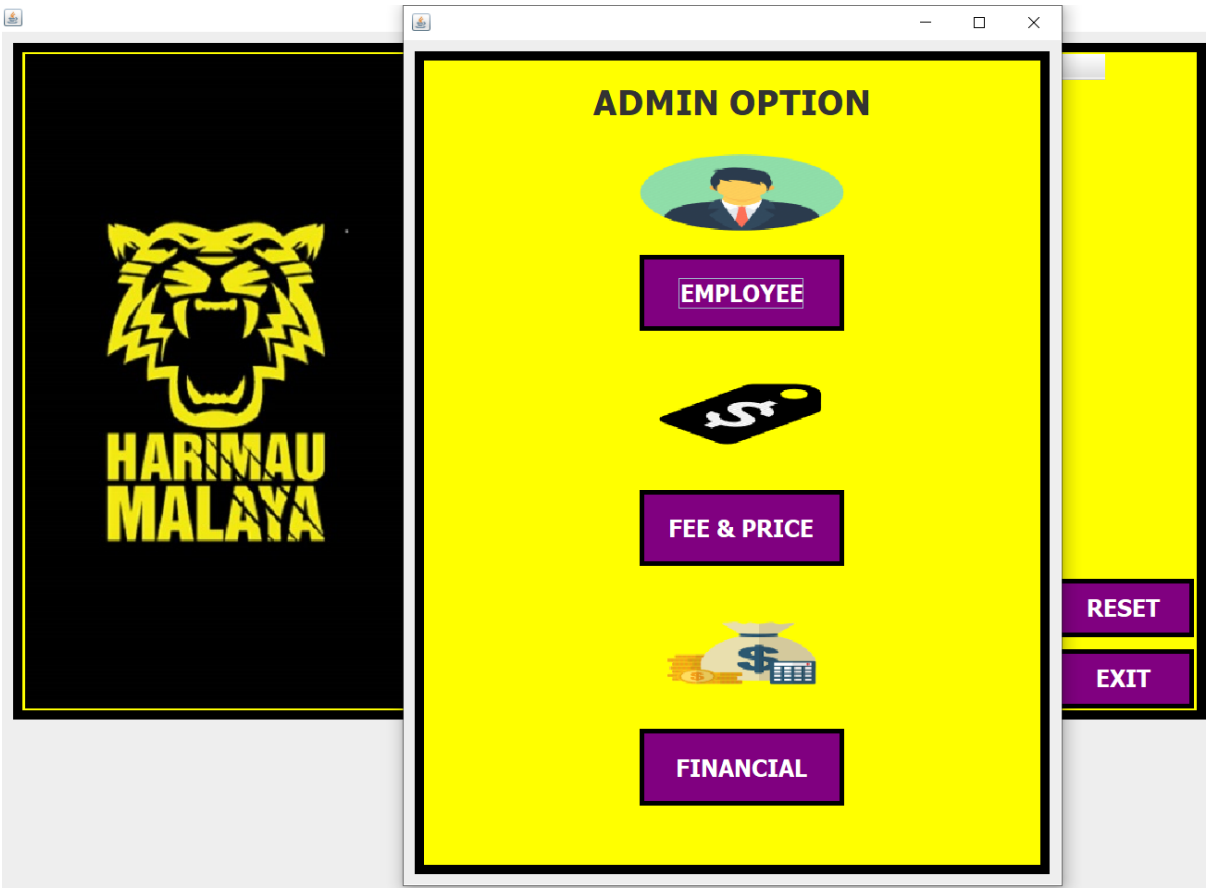
3.1 ADMIN

3.1.1 LOGIN



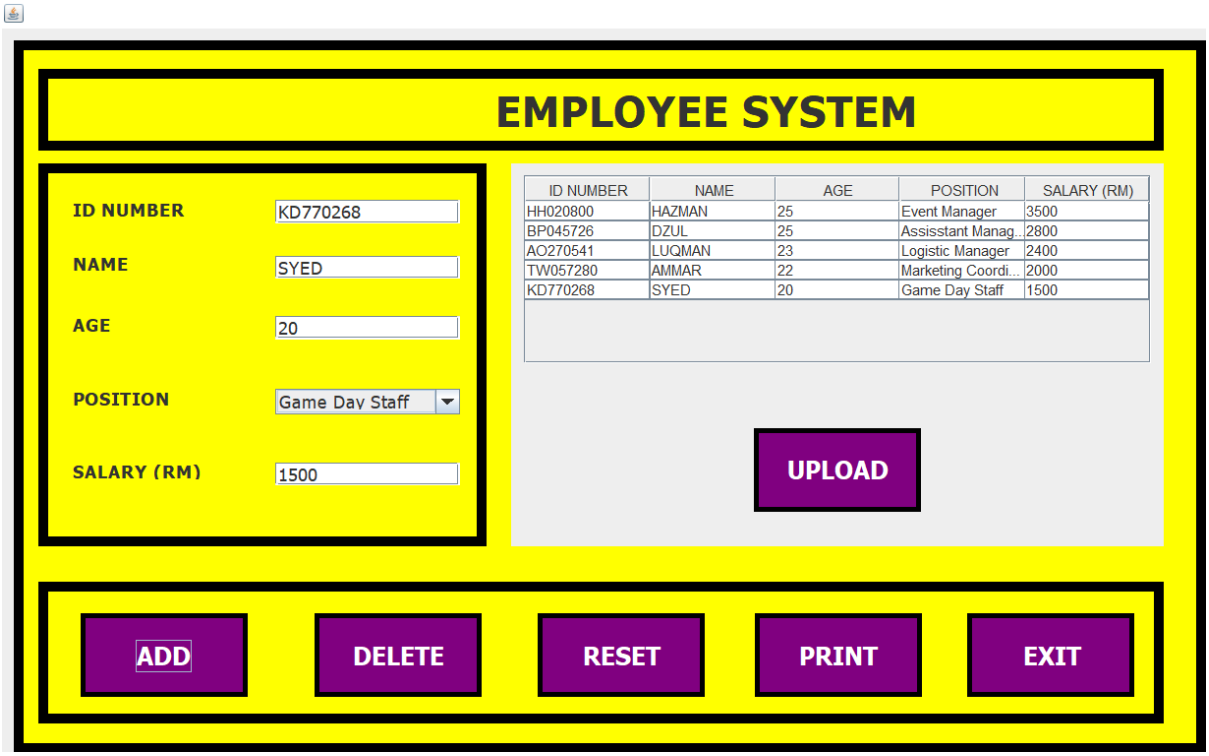
1. For employee who works as company admins, to access the admin section, you first need to enter a username and password.

3.1.2 ADMIN OPTION



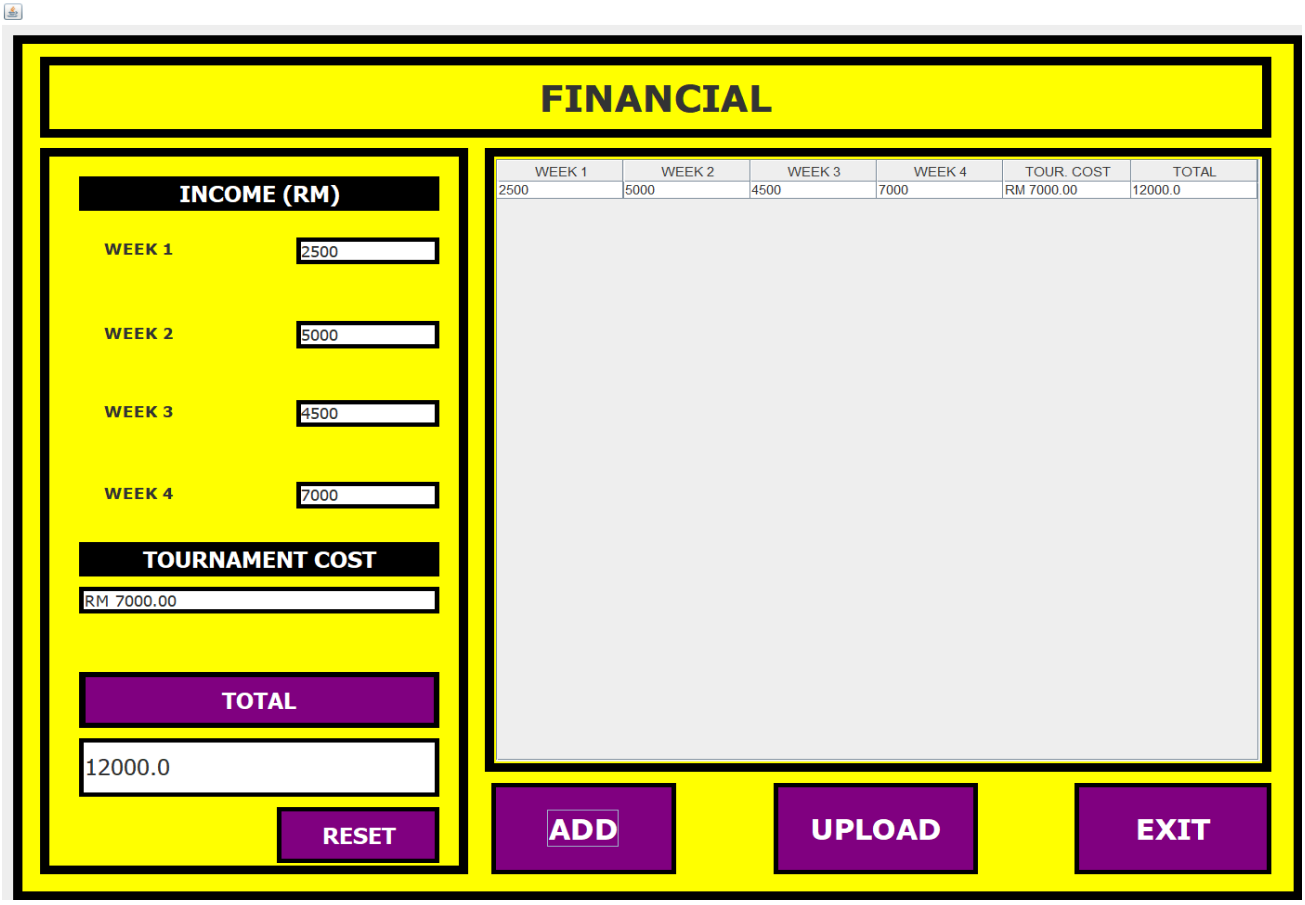
1. After successfully entering the username and password, the admin will be taken to the admin options section.
2. In this section, you can select the section you want to enter. For example, there are three options, which is employee, fee and price, and financial.

3.1.3 EMPLOYEE



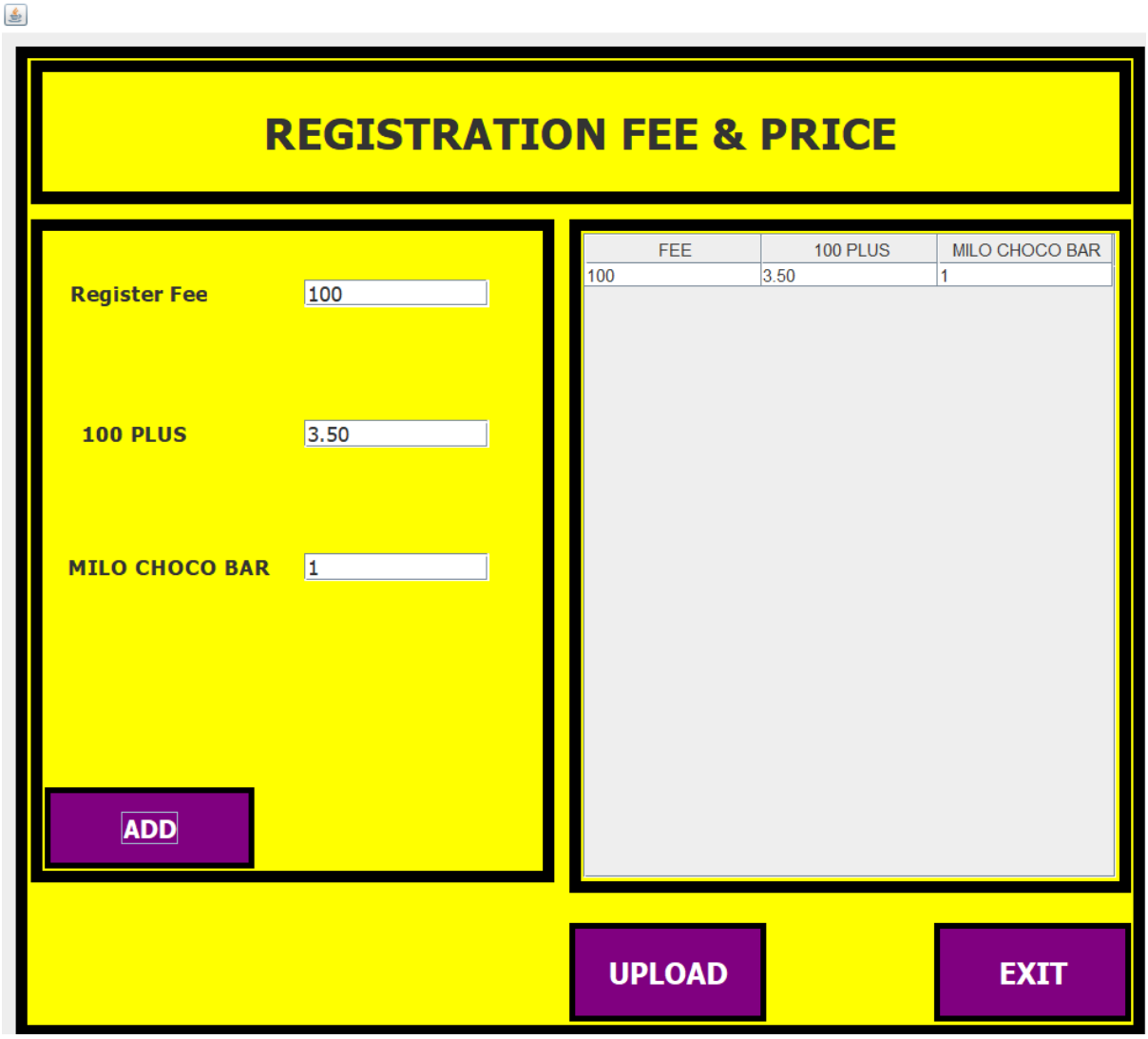
1. On the employee section, the admin can enter the employee's name, age, position, and salary.
2. Then, you (admin) can press the add button to add the details to the table provided.
3. If there is other employee information that needs to be added, you can repeat the same steps.
4. After entering and adding all the employee details, you can upload the information into a text document by pressing the upload button.

3.1.3 FINANCIAL



1. On the financial section, the admin can enter the income on a weekly basis for the month.
2. For example, income for week 1, week 2, week 3, and week 4. Next, the operating cost for running a futsal event can also be included by the admin.
3. It will help to calculate the total income for the month by adding the income on a weekly basis and subtracting the costs used to run the contest.

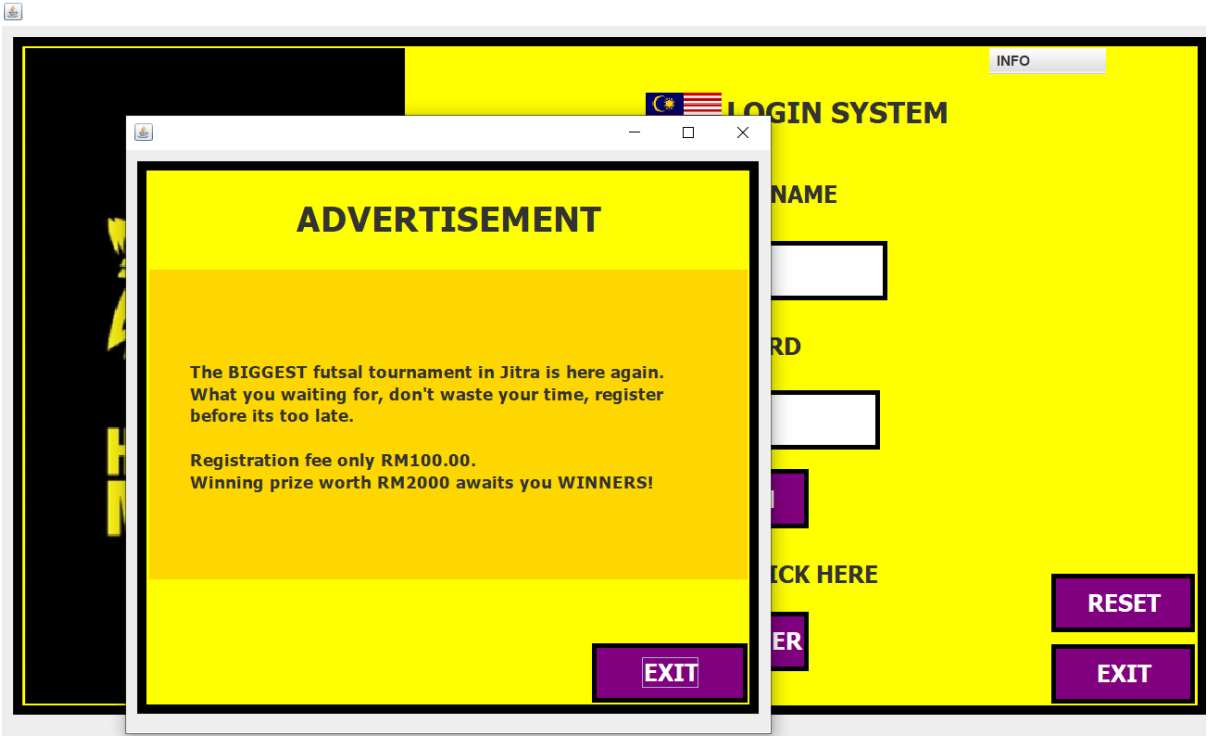
3.1.5 FEE AND PRICE



1. In the fee and price section, the admin can set the price. The fee for the registration of futsal competitions can be entered by the admin in the space provided.
2. This is to set the fees to be paid by the customer which is the participant. In addition, the admin can also set prices for drink and food sold such as 100 Plus and Milo Choco Bar.
3. The price entered is also intended to set the selling price of the item. This will be able to prevent the occurrence of mistakes or dissatisfaction among customers because the price for the registration fee and the price of goods for sale have been set.

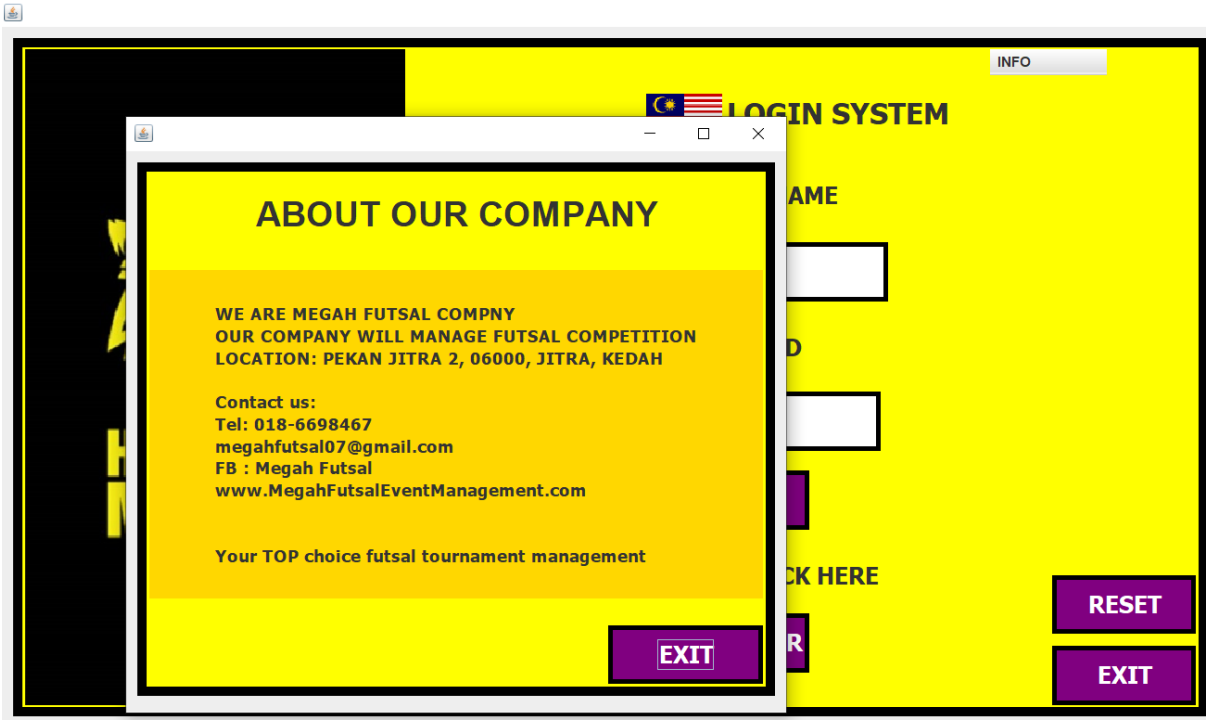
3.2 USER

3.2.1 ADVERTISEMENT



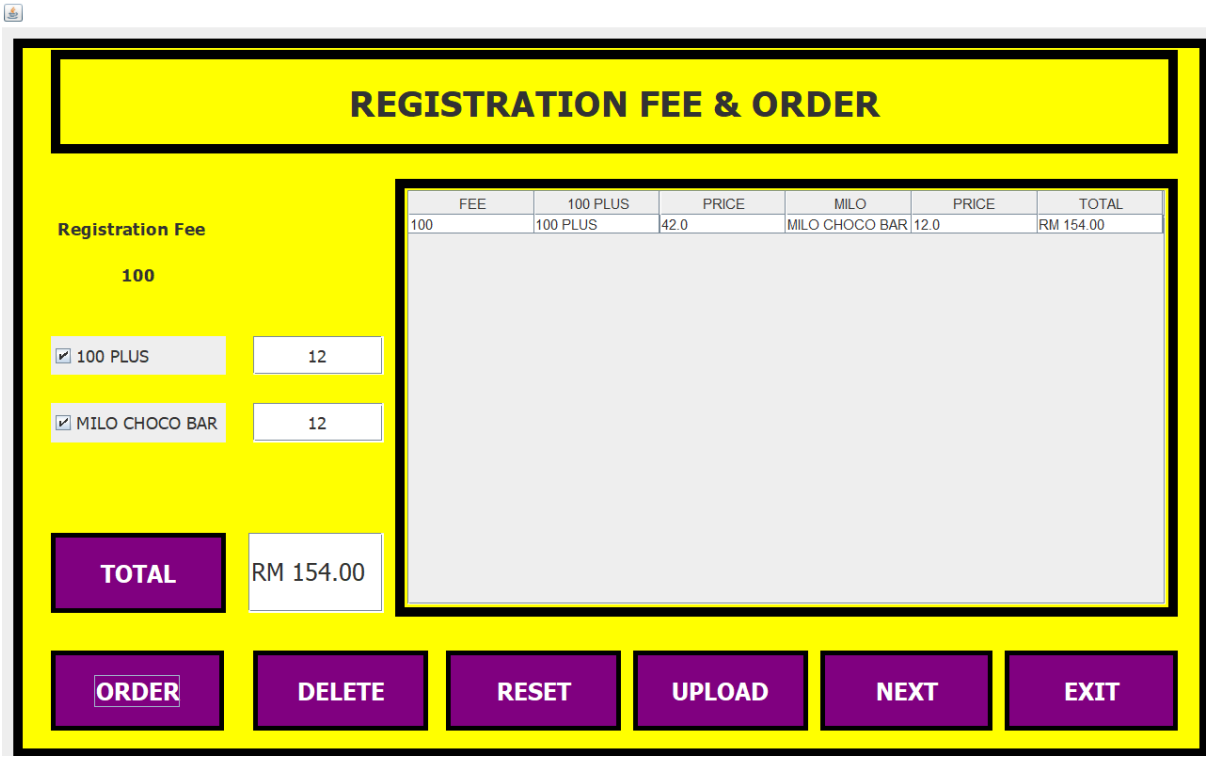
1. To find out more about the competition, customers can access the advertisement section on the login section.
2. Customers can press the info button located at the top of the login section.
3. Then, customers can click on the advertisement to find out information about the contest being conducted.
4. The relevant information will be displayed.

3.2.2 COMPANY BACKGROUND



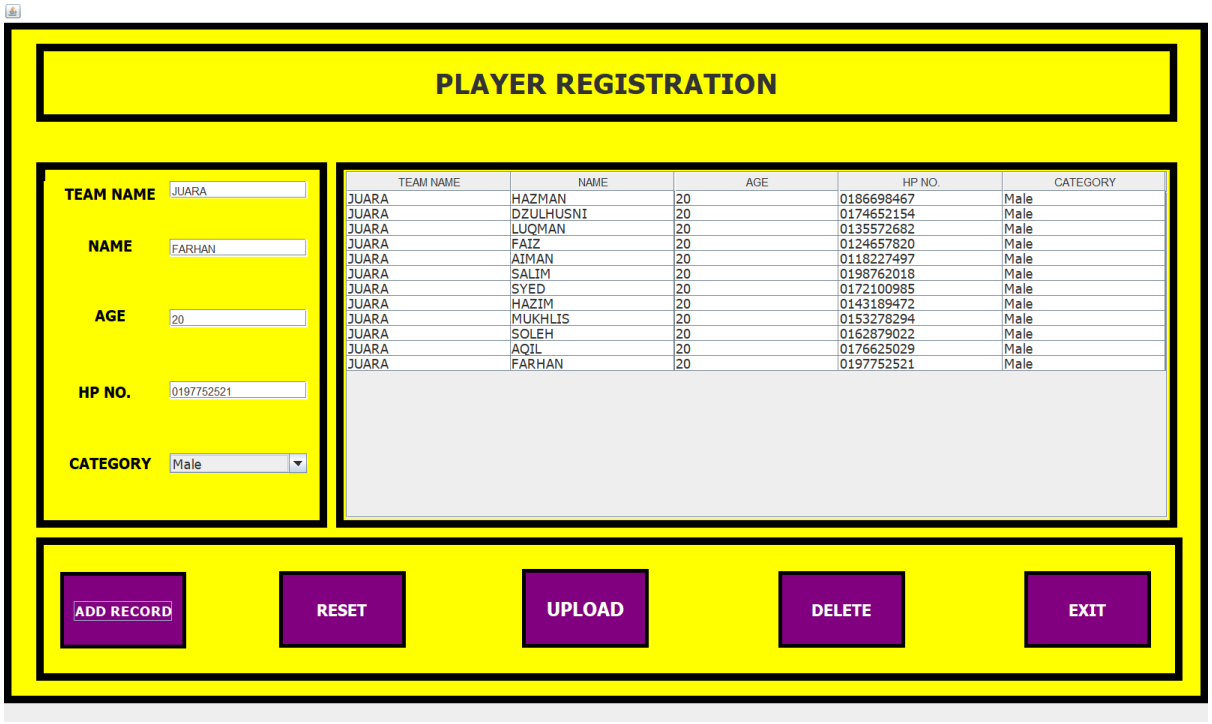
1. To find out information about the company's background, customers can access it on the login section.
2. Customers can press the info button located on the top of the login section. Then the customer can select the company background.
3. The relevant information will be displayed.

3.2.3 REGISTRATION FEE AND ORDER



1. On the registration and ordering fee section, customers can pay a fee and choose to purchase food and beverages sold or not.
2. The food sold is Milo Choco Bar, while the drinks sold are 100 Plus.
3. If you want to buy food or drink, you can click on the box to the left of the name of the drink or food.
4. Then, you can enter the quantity you want to buy.
5. Once you have finished entering the quantity, you need to press the Total button to get the total price.
6. You can press the delete button to discard the order, or press the reset button to refill from scratch about your order.
7. Finally, if you are satisfied with your order, you can press the order button so that the order information is entered in the table provided.

3.2.4 PLAYER REGISTRATION

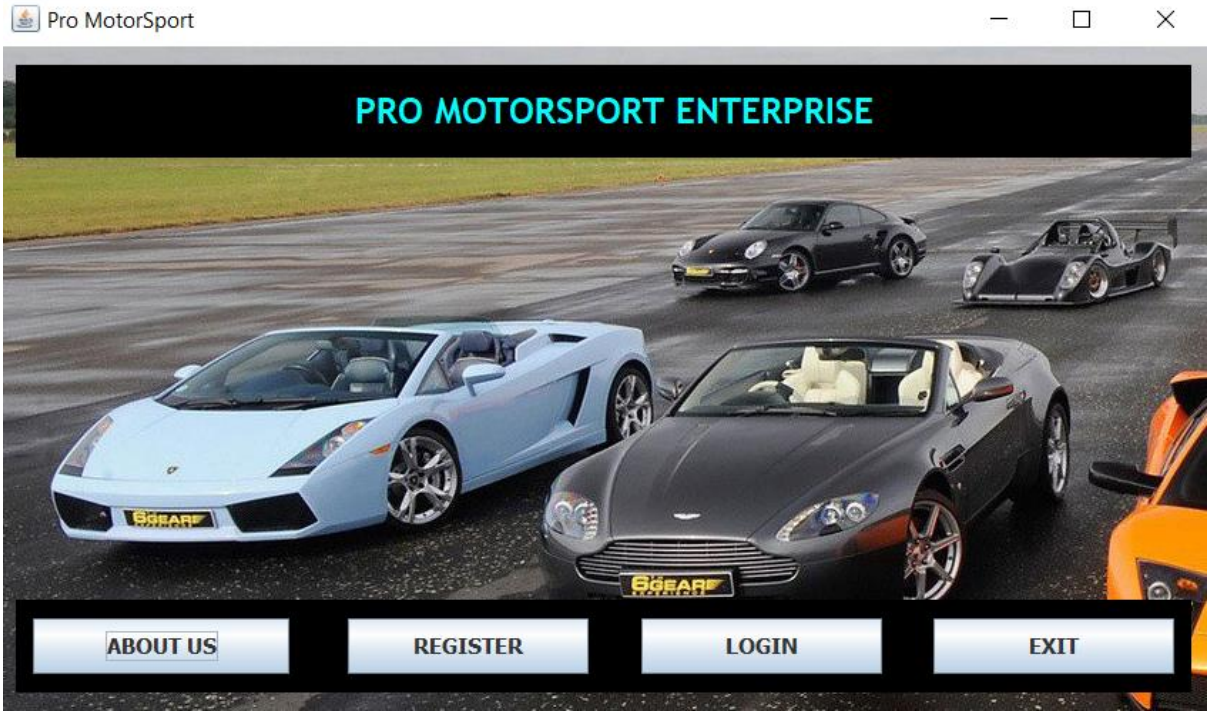


1. In the player registration section, customers can register the names of group members who will participate in futsal competitions.
2. You are required to fill in the team's name, player name, player age, phone number, and category to be joined.
3. Once completed, you can press the ADD button to add the player information that has been filled into the table provided.
4. Then you can use the RESET button to make it easier for you to delete the details of the previous player in the details box and fill in the details of the next player.
5. If there is an error in the information when you add it to the table, you can select the row of the table that has the incorrect information and press the DELETE button to delete it.

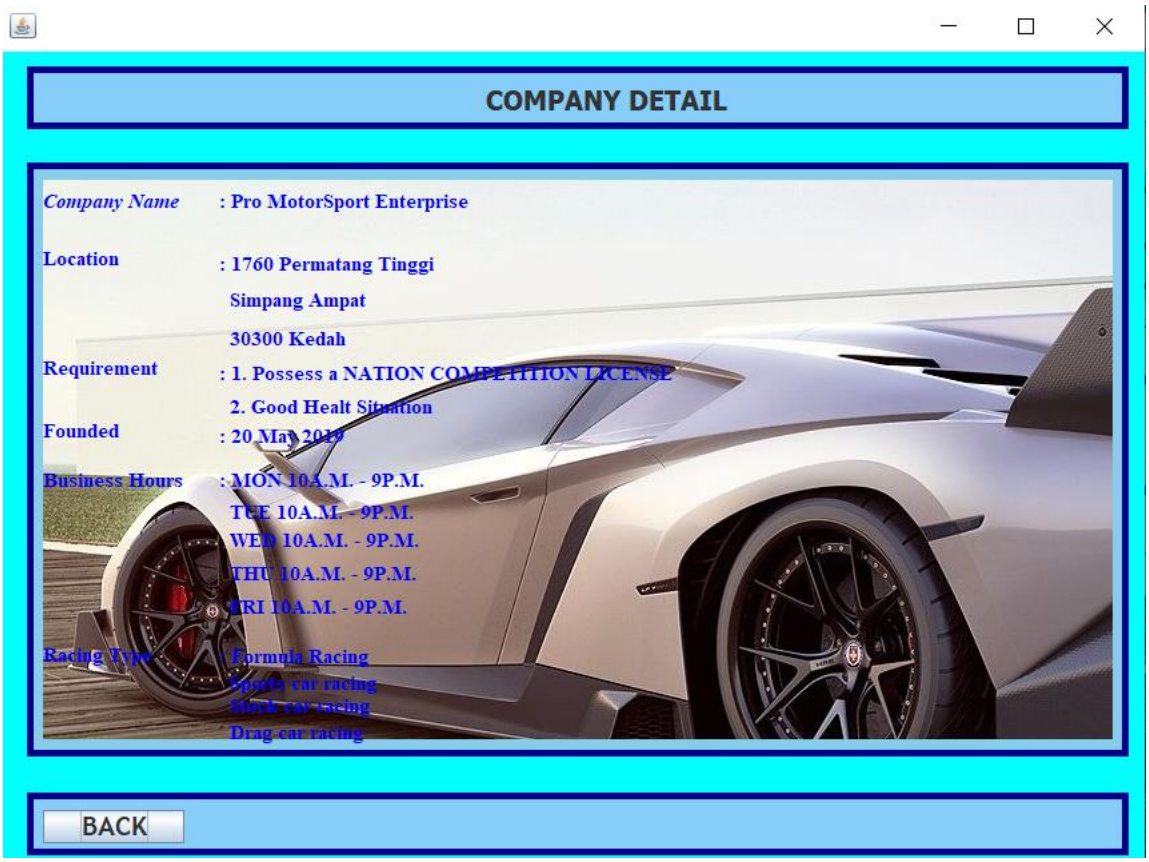
2. Car Racing (Pee Foo Ye 278632)

1. Home Page

The user may choose “ABOUT US”, “REGISTER”, “LOGIN” or “EXIT”.



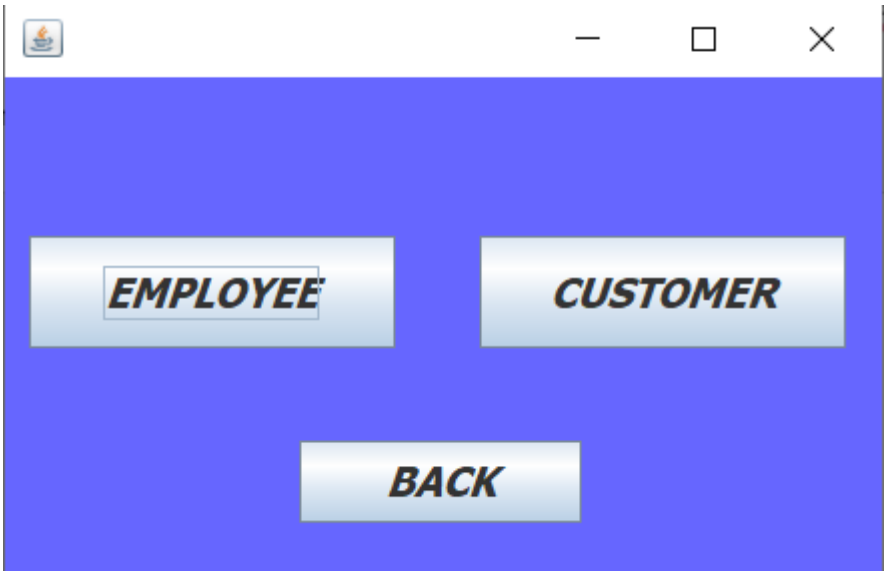
2. If the user clicks on “ABOUT US”, then the company detail will be displayed.



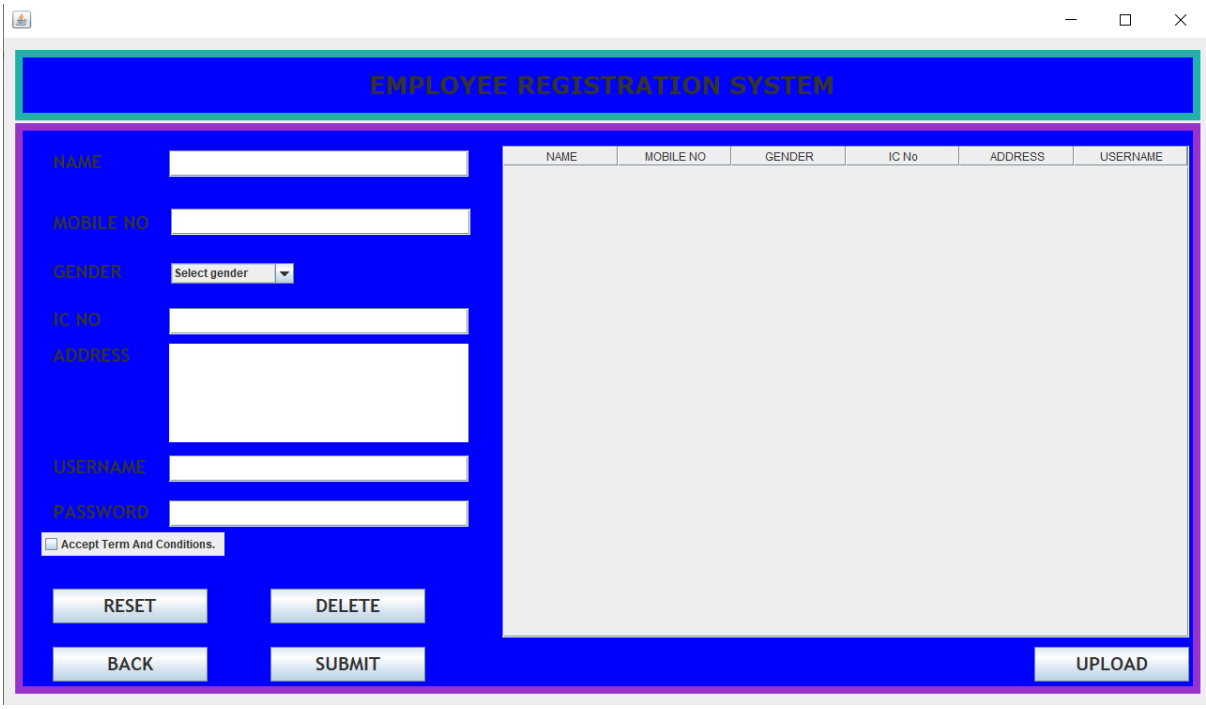
The user may click on “BACK” then the system will back to the home page.

3. Register

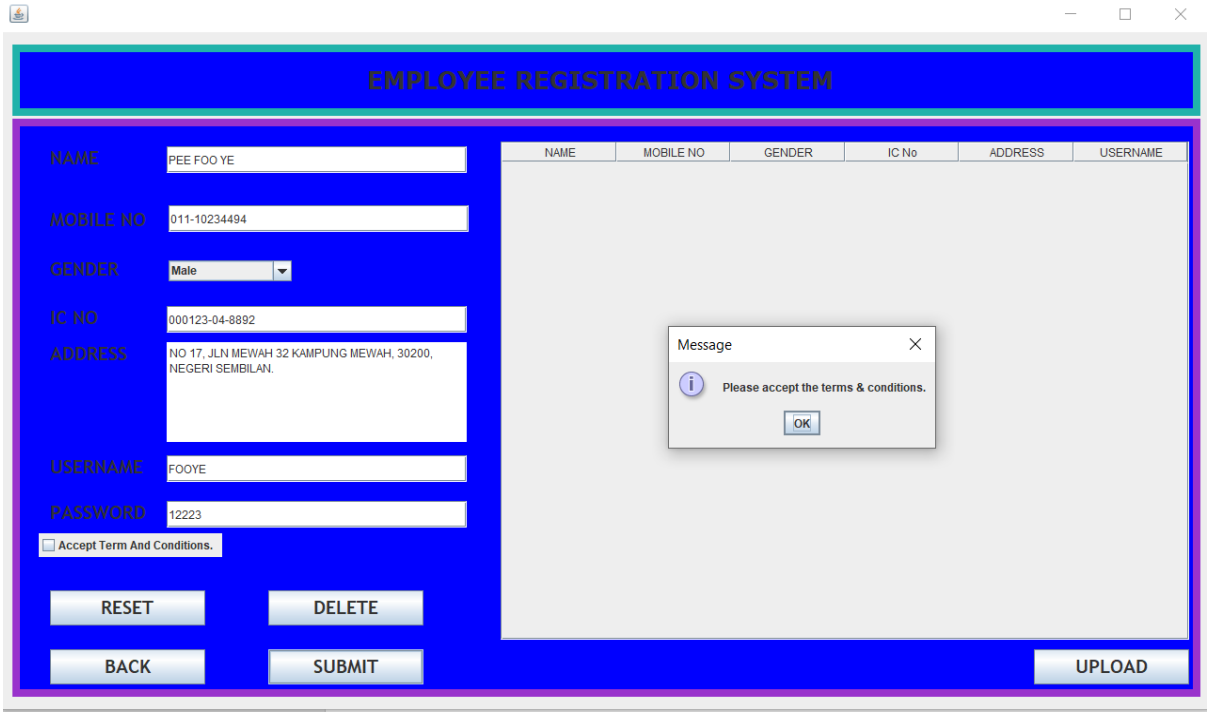
The user may clicks on “Register”. The system will ask for select your role whether you are employee or customer.



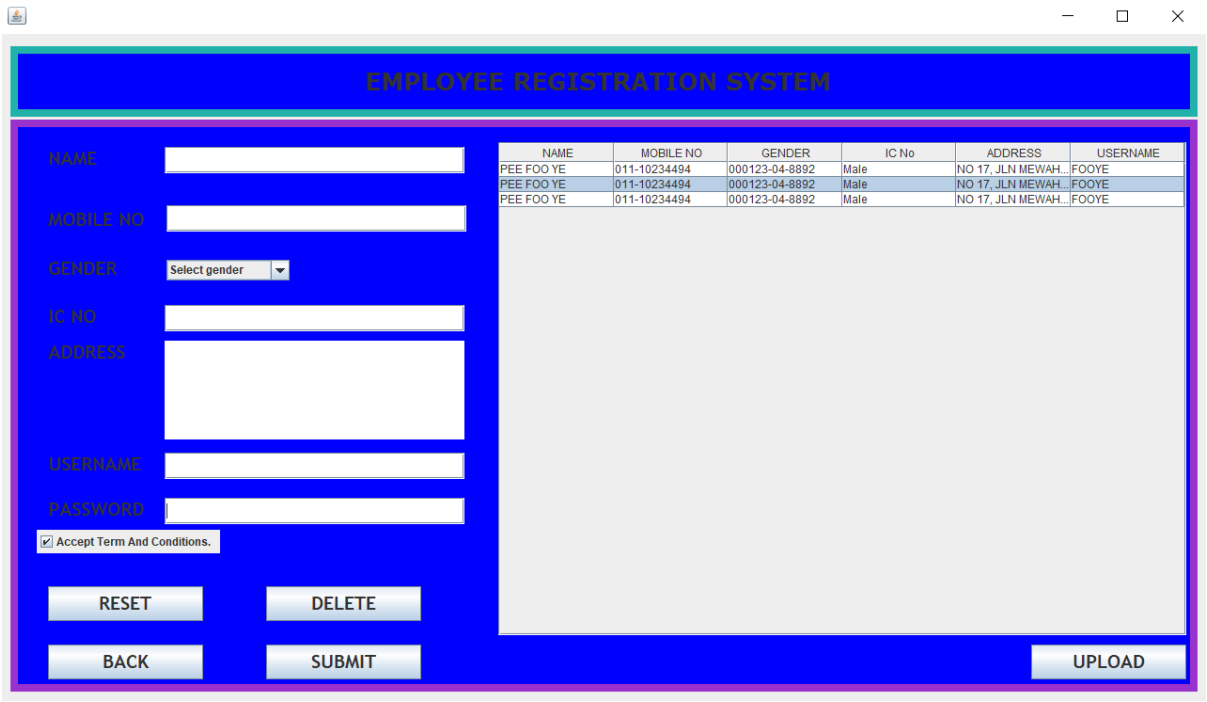
If the user choose “EMPLOYEE”, then the Employee Registration System will be shown out as below:



The user may fill in the name, mobile no, gender, identity card no, address, username and password. Then the user must select the “Accept Terms and Conditions” before submit. If the user failed to select, the system will remind the user.

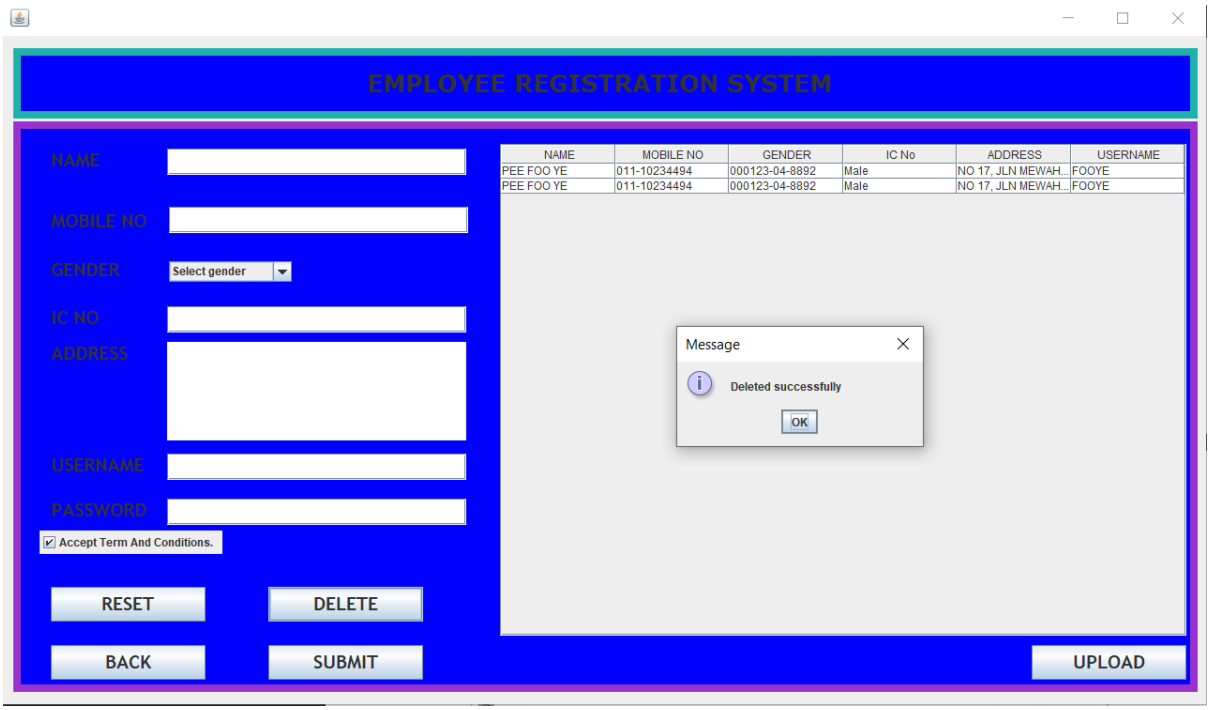


The user may click on “RESET” button, then the text field will be reset to empty.

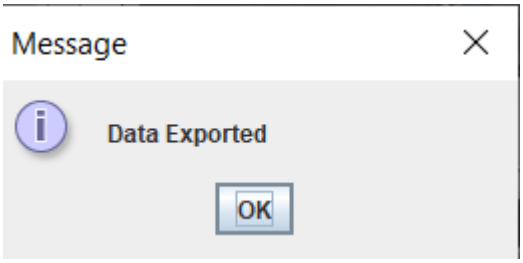


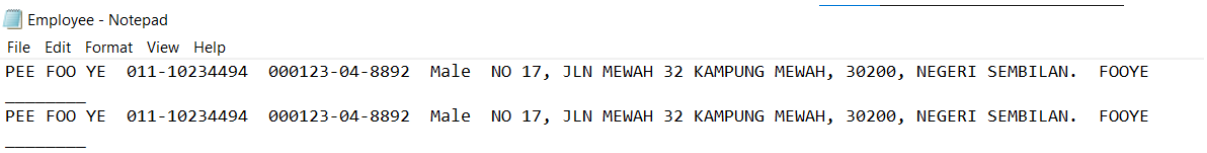
To delete data, the user may click on the data on the list and click on “DELETE”.

Then the data will be deleted.

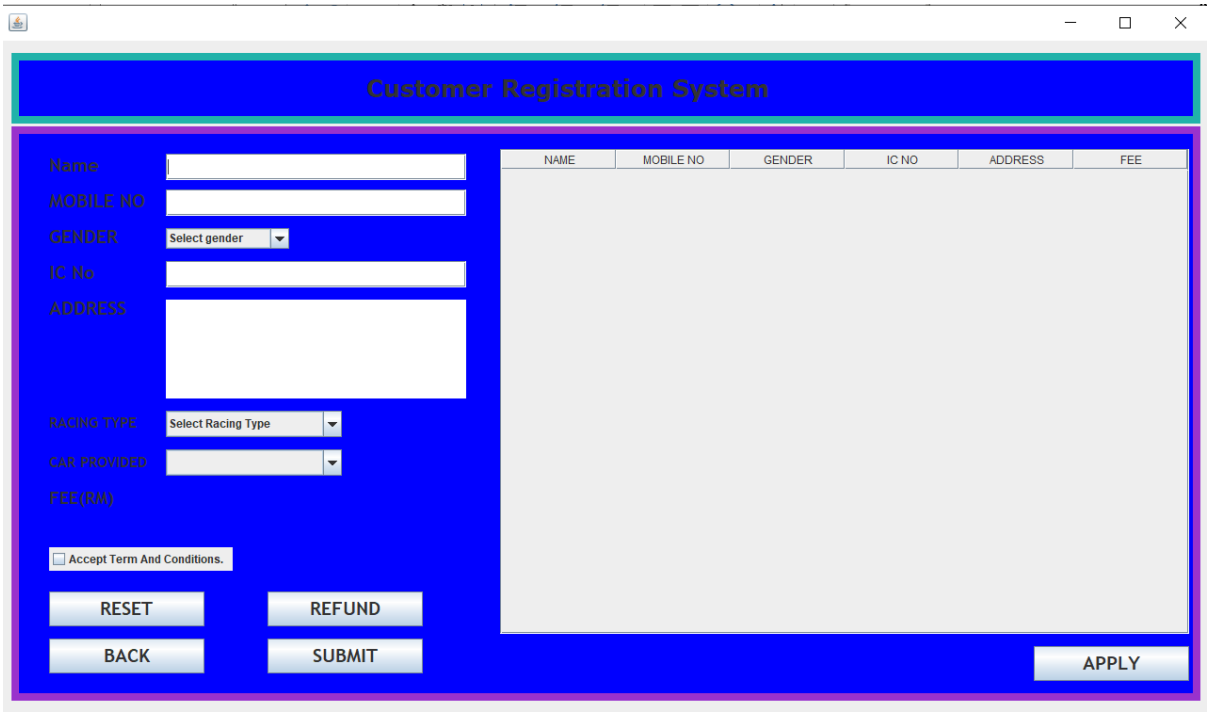


If the user clicks on “UPLOAD” button, the data will be export to the text file.

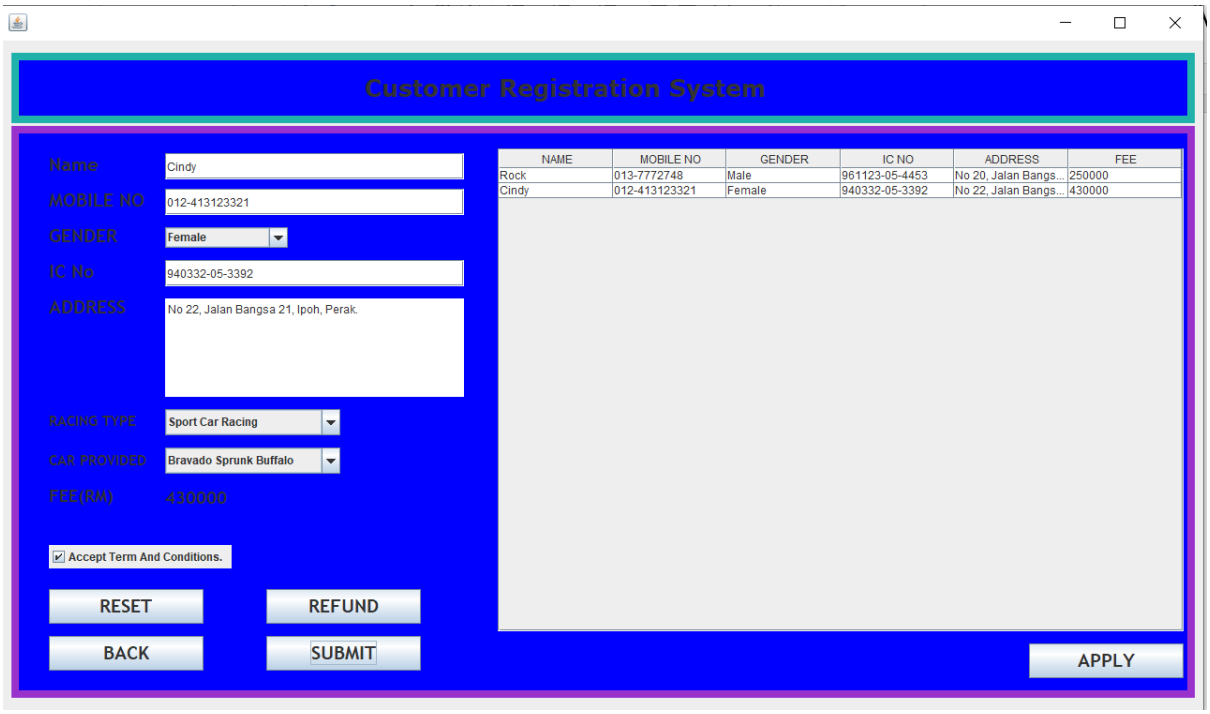




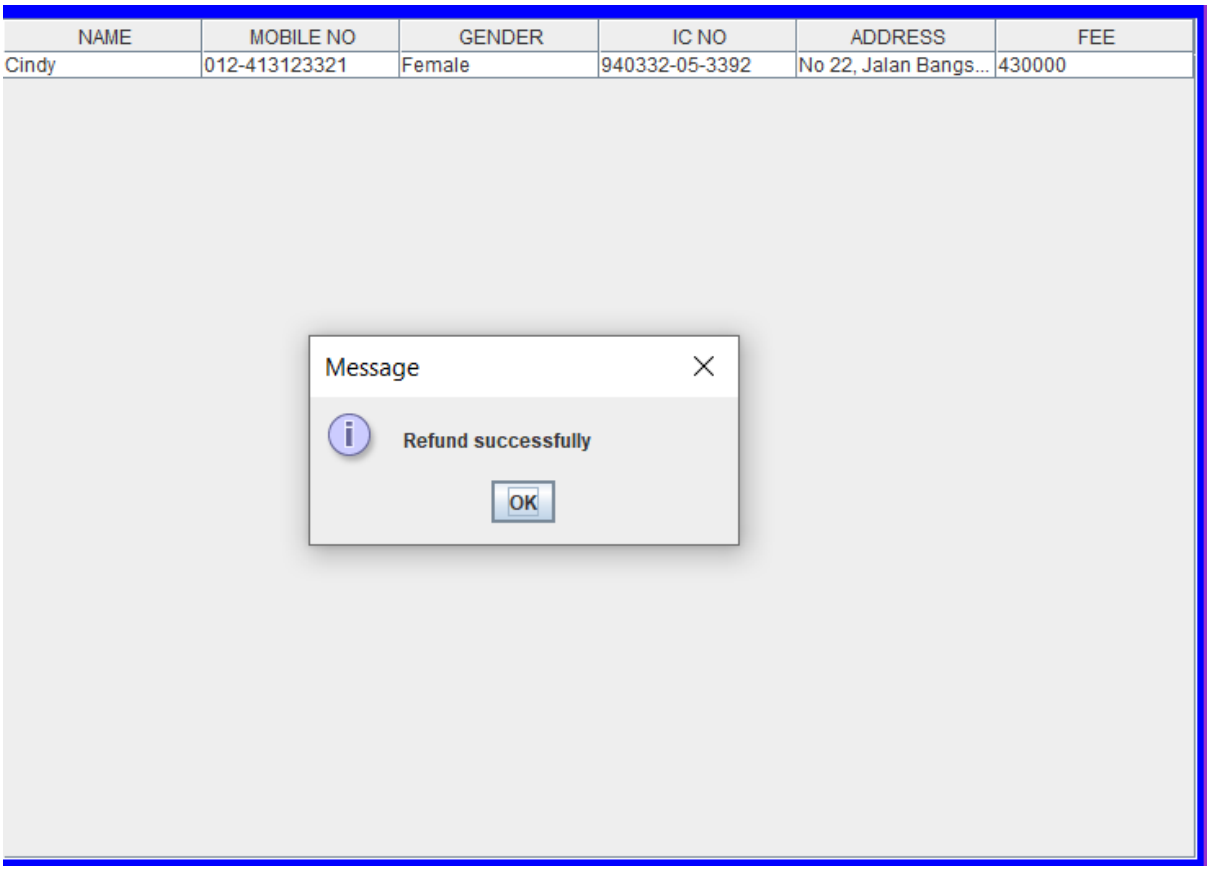
If the user is a customer, then he/she may clicks on “CUSTOMER” to enter the Customer Registration System.



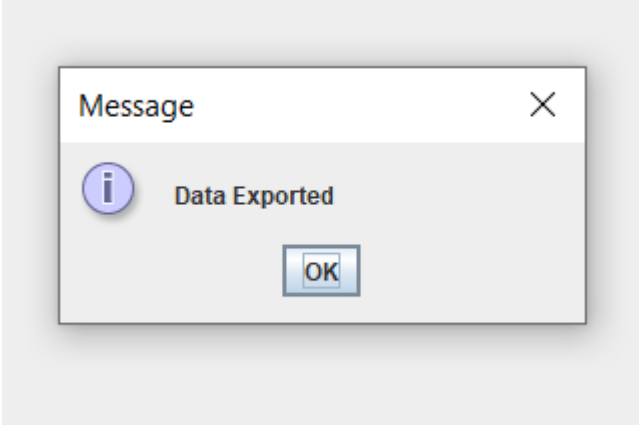
Then, the user has to fill in all the information and accept the term and conditions, then click on “SUBMIT” and the data will be displayed in the table. (Every racing type have different car provided)



If the user want to refund, then click on “REFUND”, the data will be deleted.

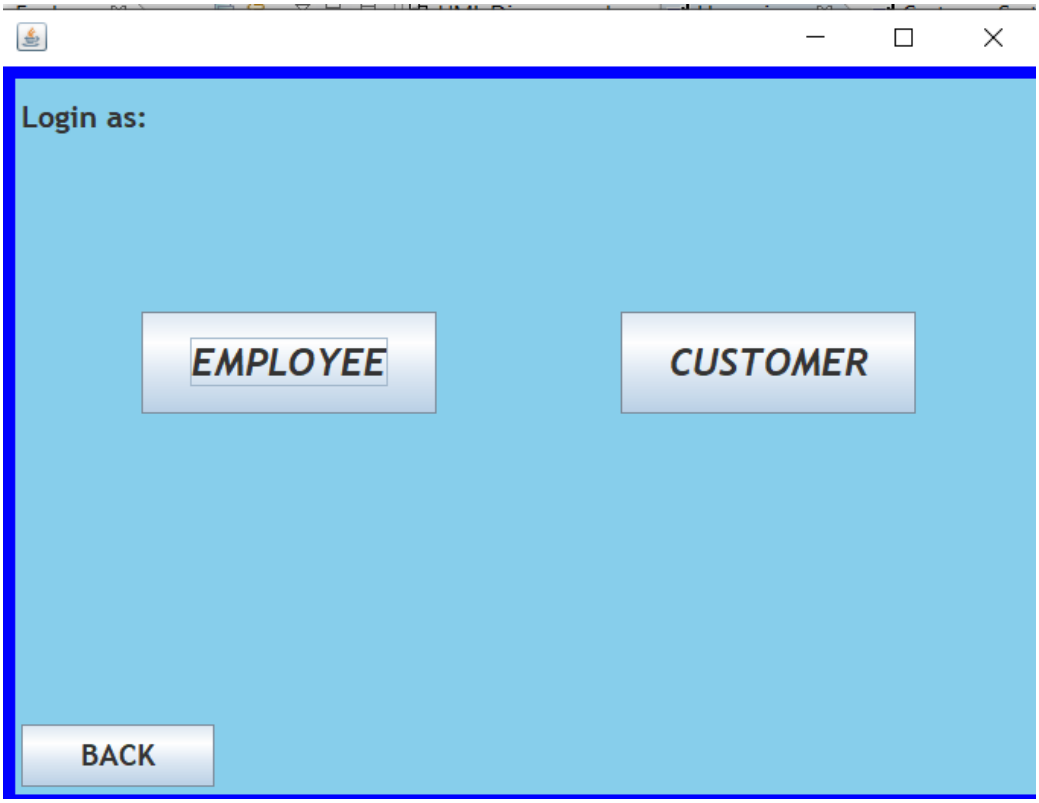


If the user click on “APPLY”, then the data will be exported to the text file.

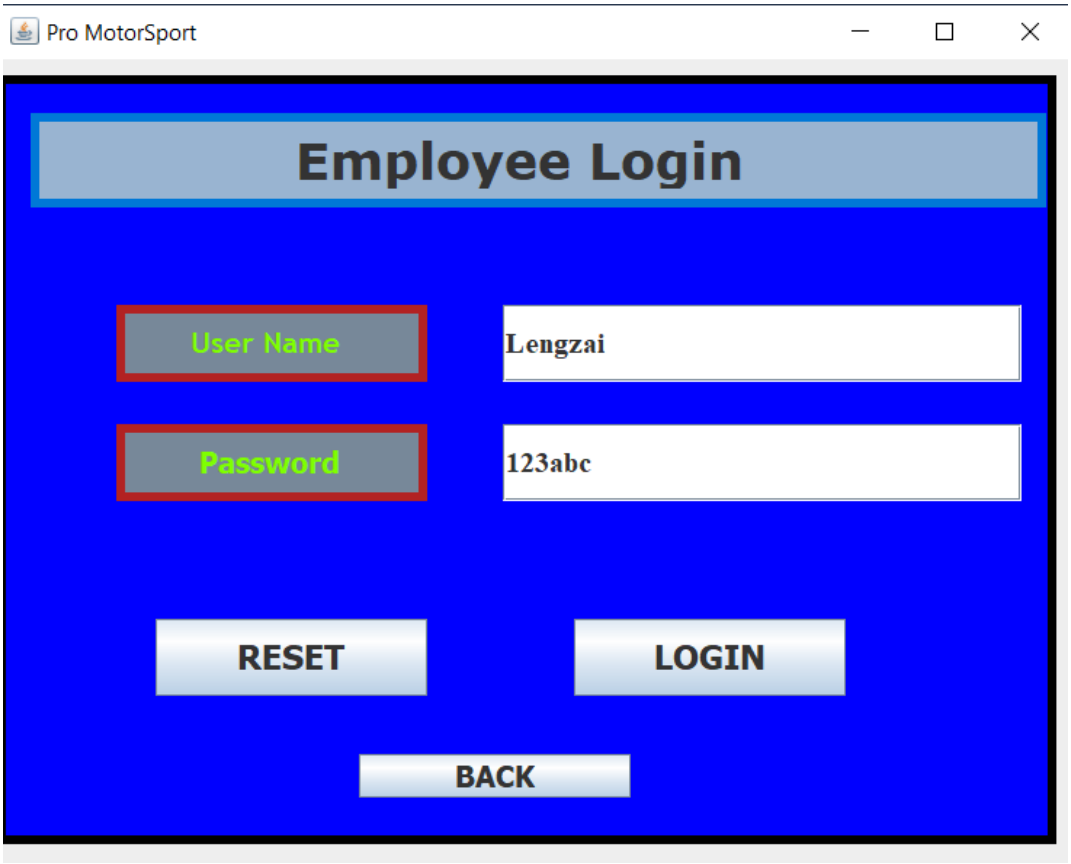


4. Login

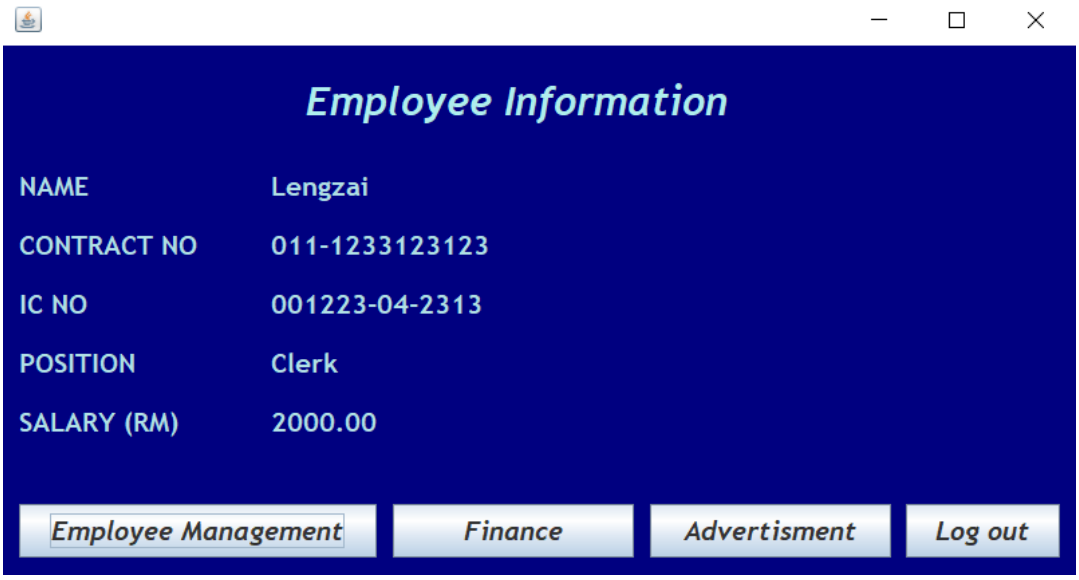
The user may click on “LOGIN” button in the home page to enter the Login menu, then the user may choose to login as employee or customer.



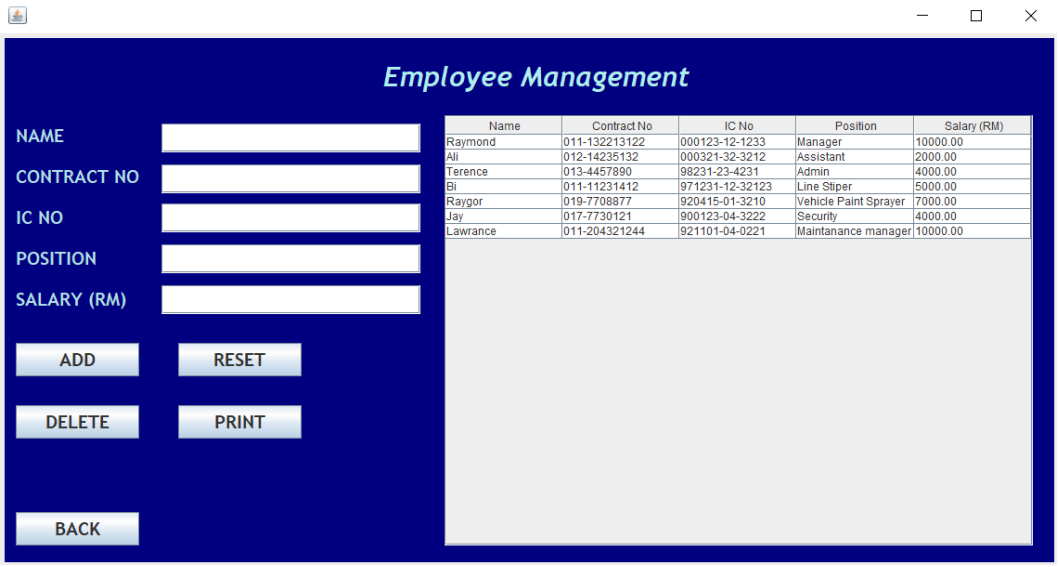
If the user click on “EMPLOYEE” button, the login page will be displayed. After that, the user can type in the username and the password, then click on “LOGIN”.



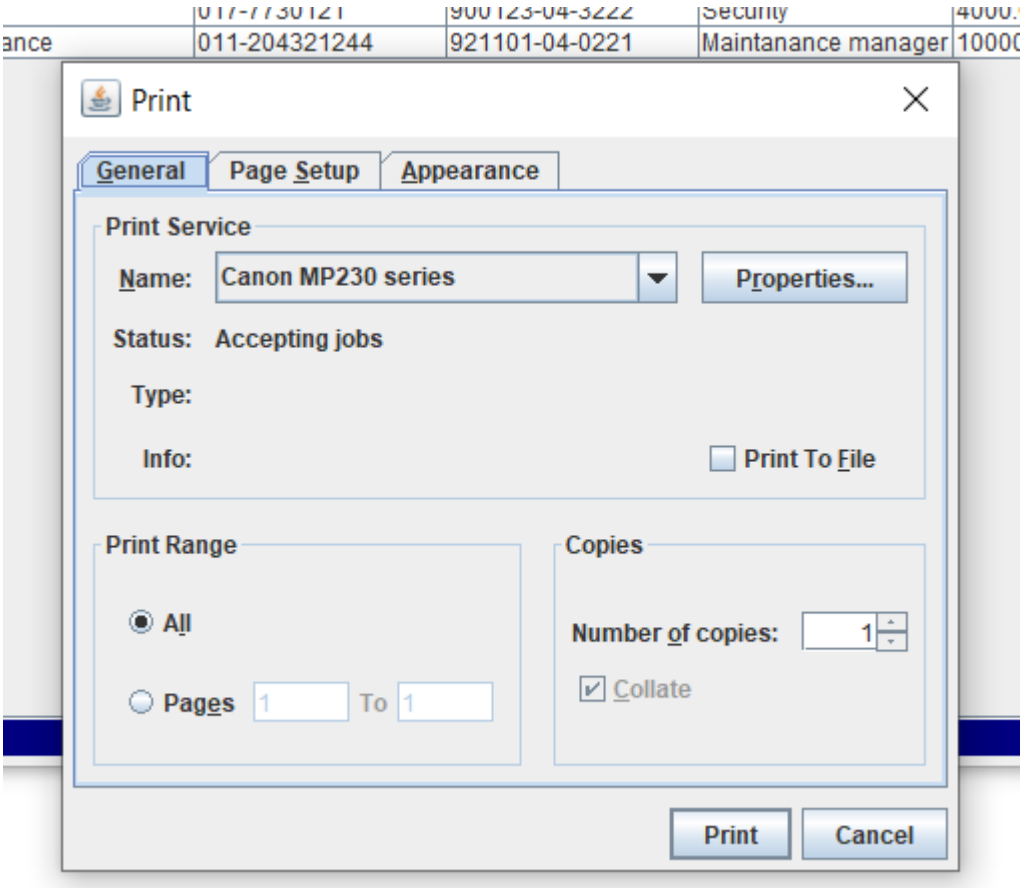
Then the employee information will be displayed. The user may choose Employee Management, Finance, Advertisement or Logout.



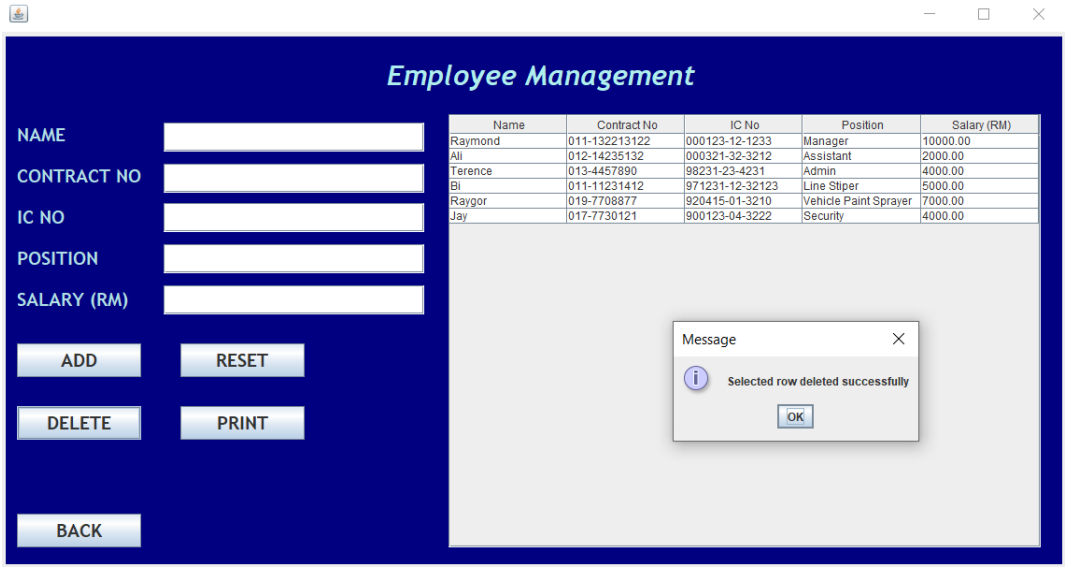
In the employee management system, if the user want to add employee, then the user may key in the information then press “ADD”.



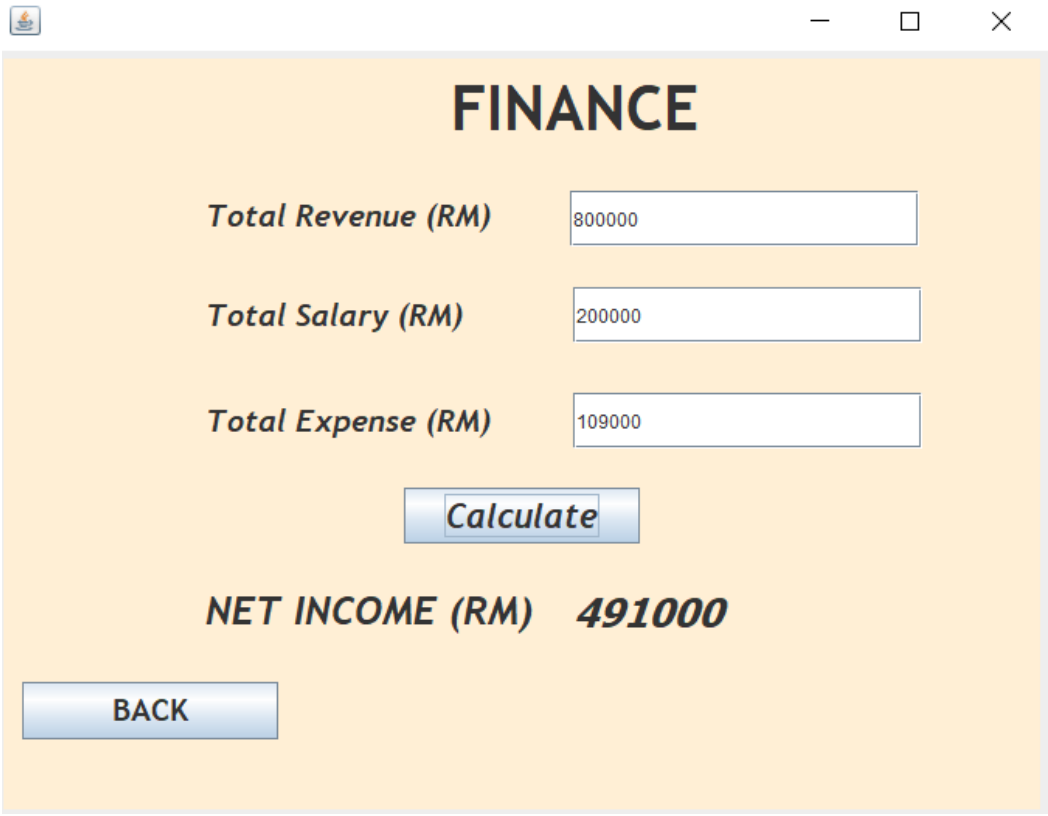
If the user wants to print the data. Then press “PRINT” button. Then the printer system will be shown as below.



The user also can delete the button by select the row in the table, then click on “DELETE”.

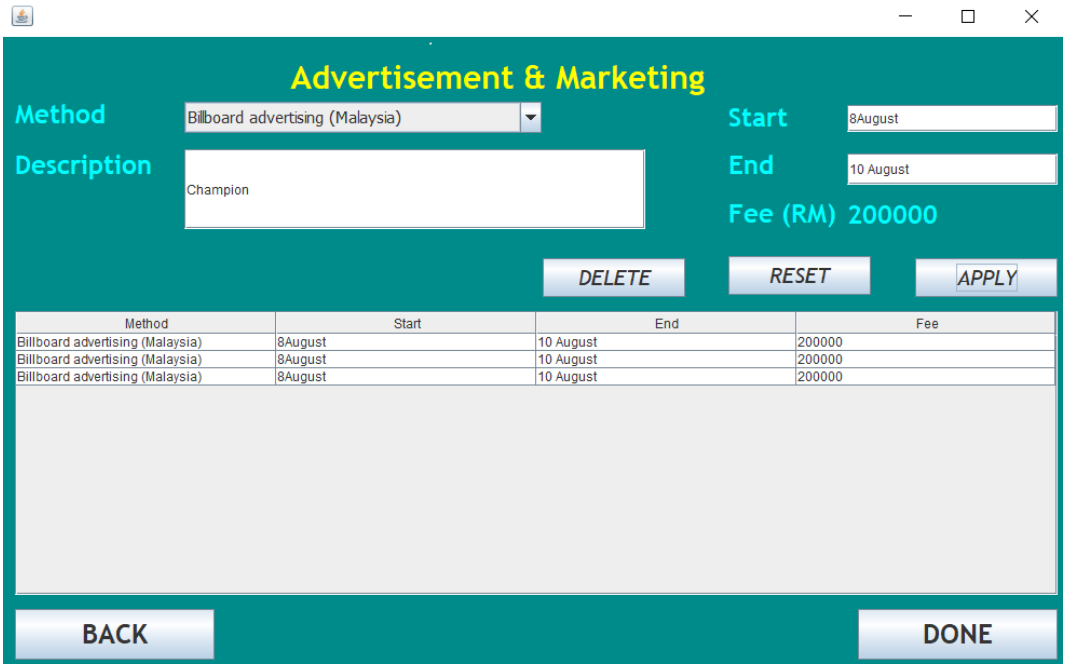


The user can click “BACK” to go back to the employee system page, then press “FINANCE”. Then key in the total revenue, total salary and total expenses, then press “Calculate”. After that the total net income will be shown.



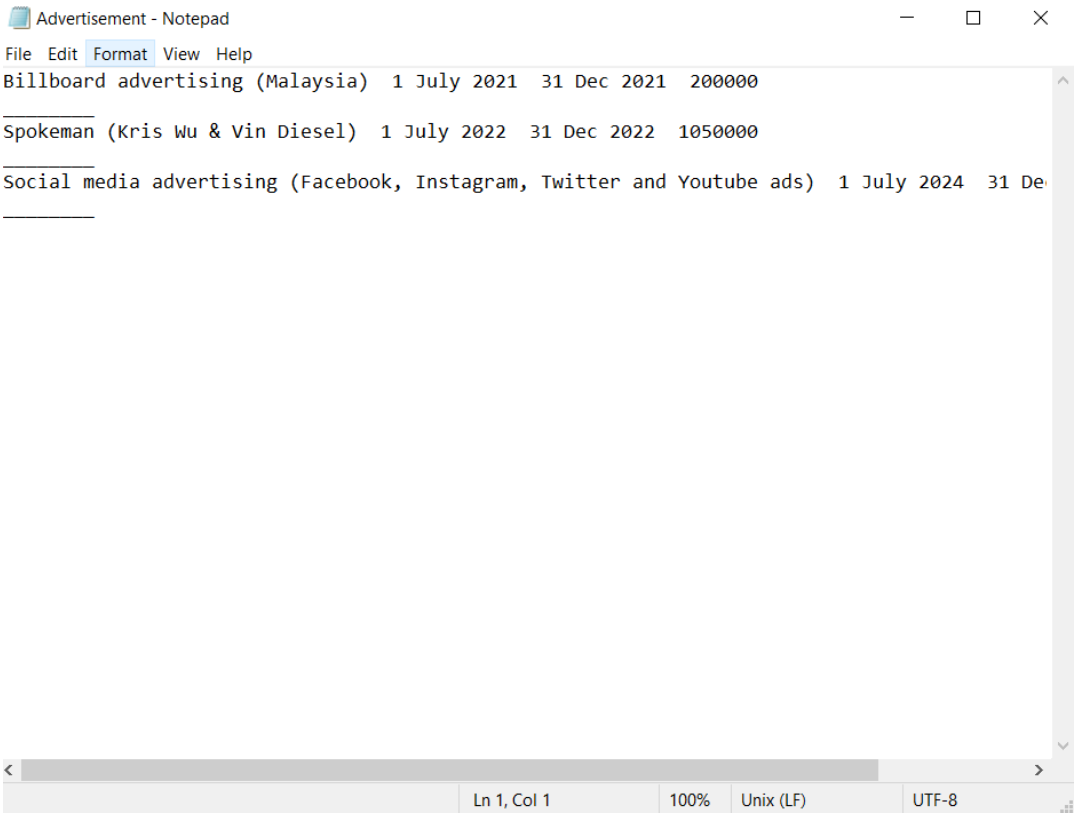
After that, click on “BACK” and select “ADVERTISEMENT” to enter the advertisement and marketing system.

Then the user may choose the advertisement and marketing method, type in the description, start date, end date. The fee will be auto generated. Click on “APPLY”, the information will be displayed in the table.



The user also can delete the data or reset by clicking the button.

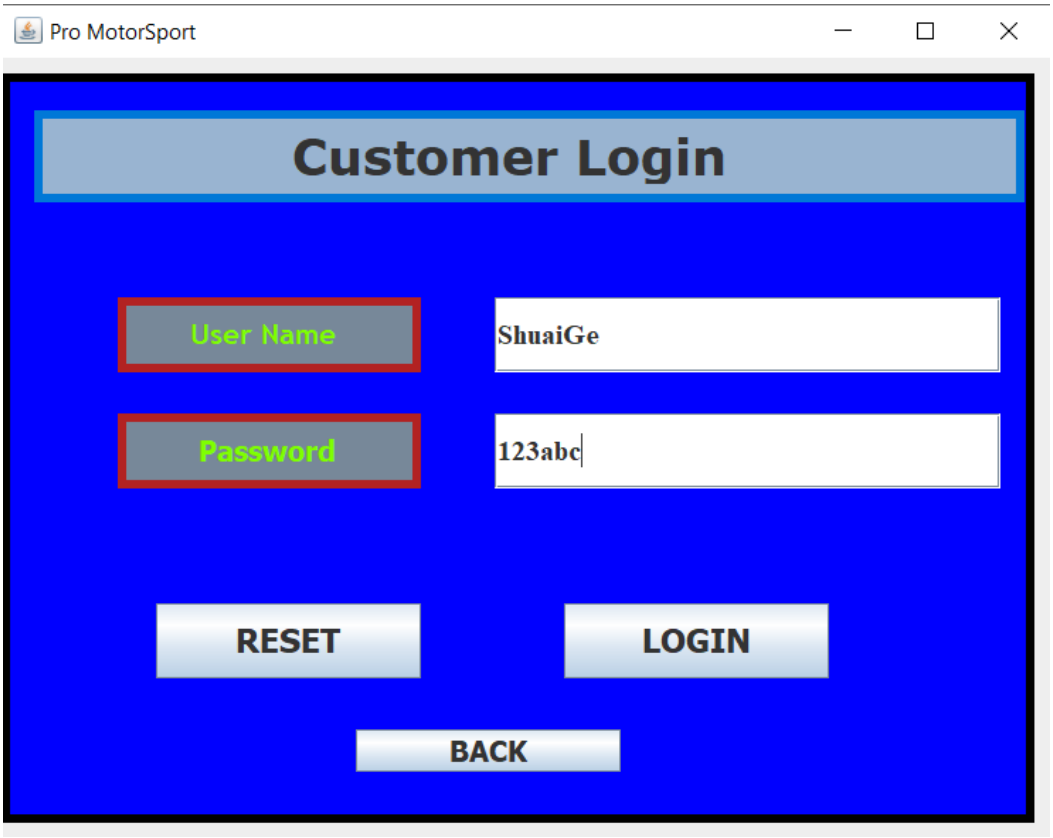
After that, click on “DONE” and the data will be exported in the text file.



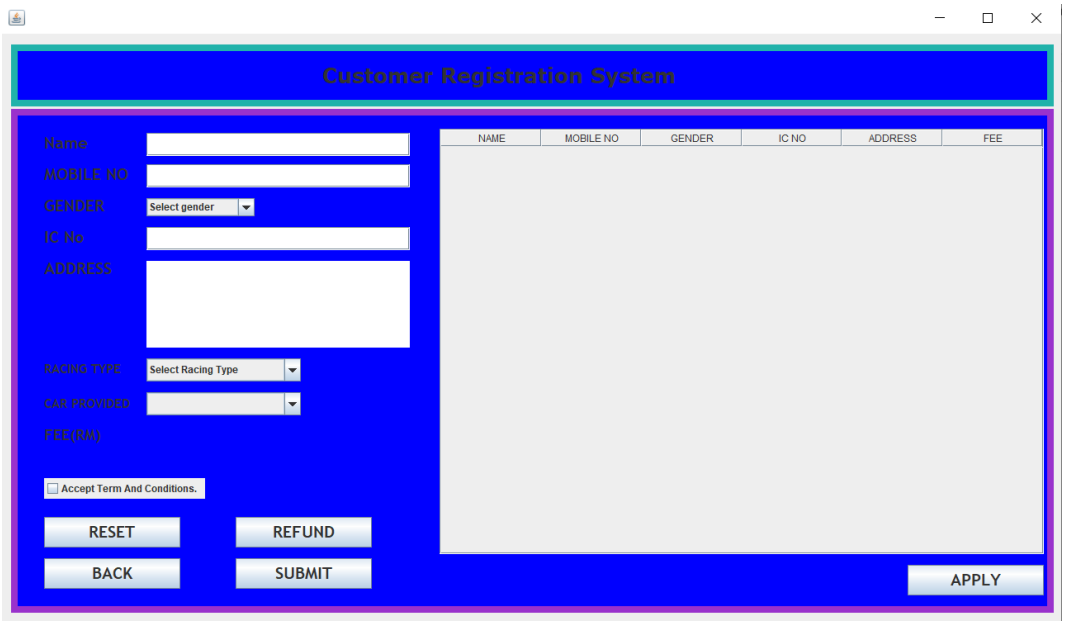
Then the user can click on “BACK” and Press “LOG OUT” to log out and Click on “Back” to the login menu.

If the user wants to login as a customer, then select “CUSTOMER” to enter the customer login system.

After that, enter the username and password, then login.

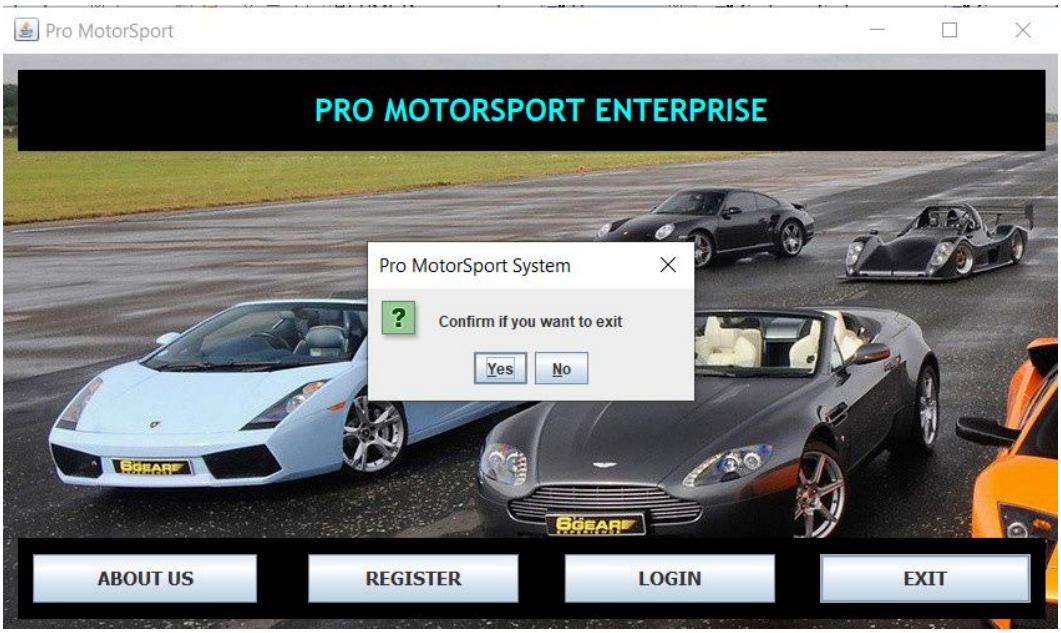


Then the customer information will be shown. In the system, the user can click on "REGISTER" to enter the customer registration system, or log out.



5. Exit

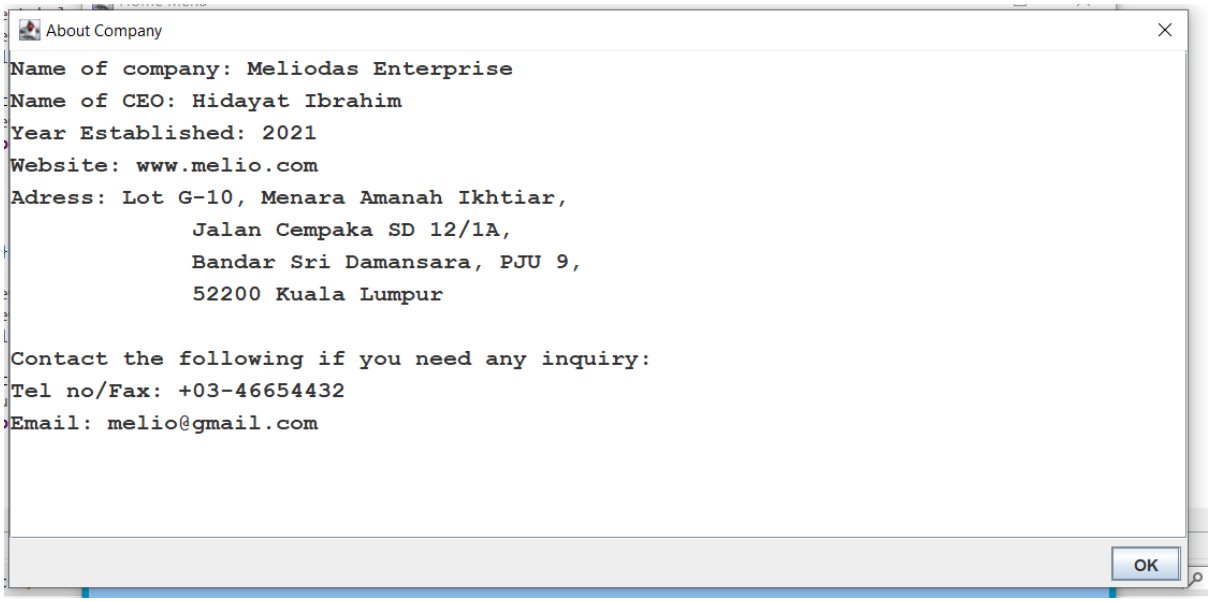
The user may click on “EXIT” and click on “Yes” if the user confirms to exit the system.



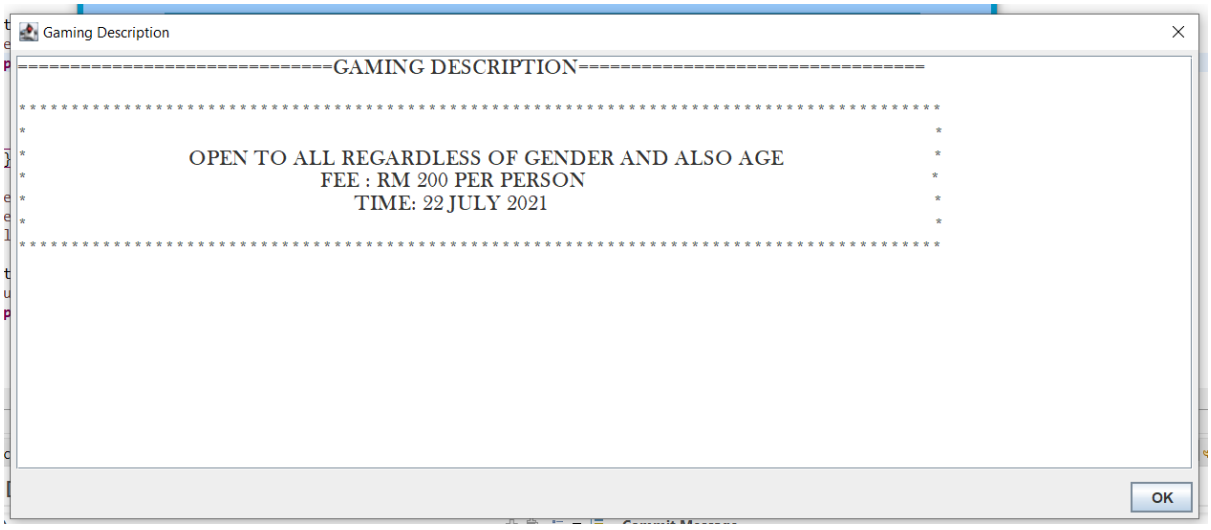
3. Gaming (Muhammad Hidayat Bin Ibrahim 279395)

Step 1: first page of the system is the home page of the system of Melio Enterprise. User can click the “Employee” button or “Customer” to go a specific page of the system. There is a menu bar on the top-left corner for users to go to a specific page of the system. In the menu, user can choose whether home which contain button “Exit” or “About Compony” at home or use can click “advertisement” or “Gaming Description” at product button.

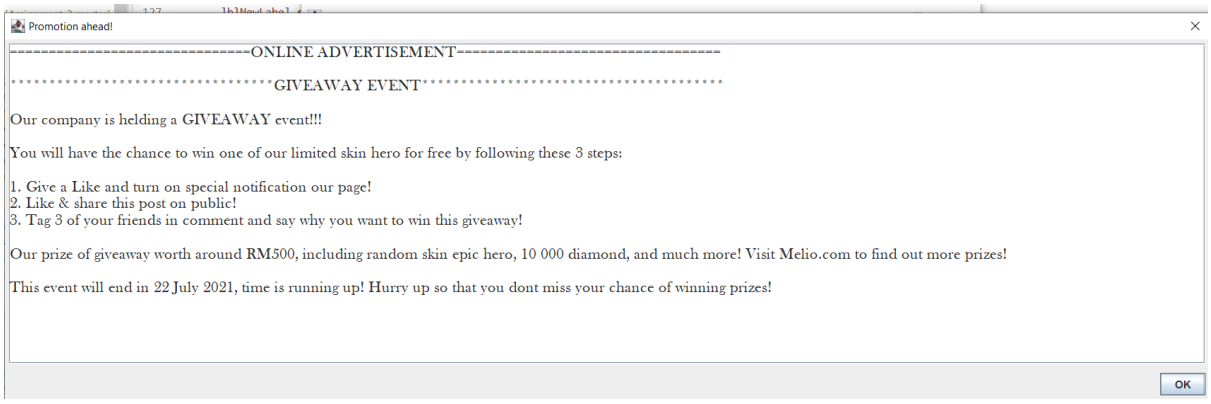
Step 2: if the user click home and choose “about compony”, the program will bring to specific page.Then user just click ok to turn to menu page.



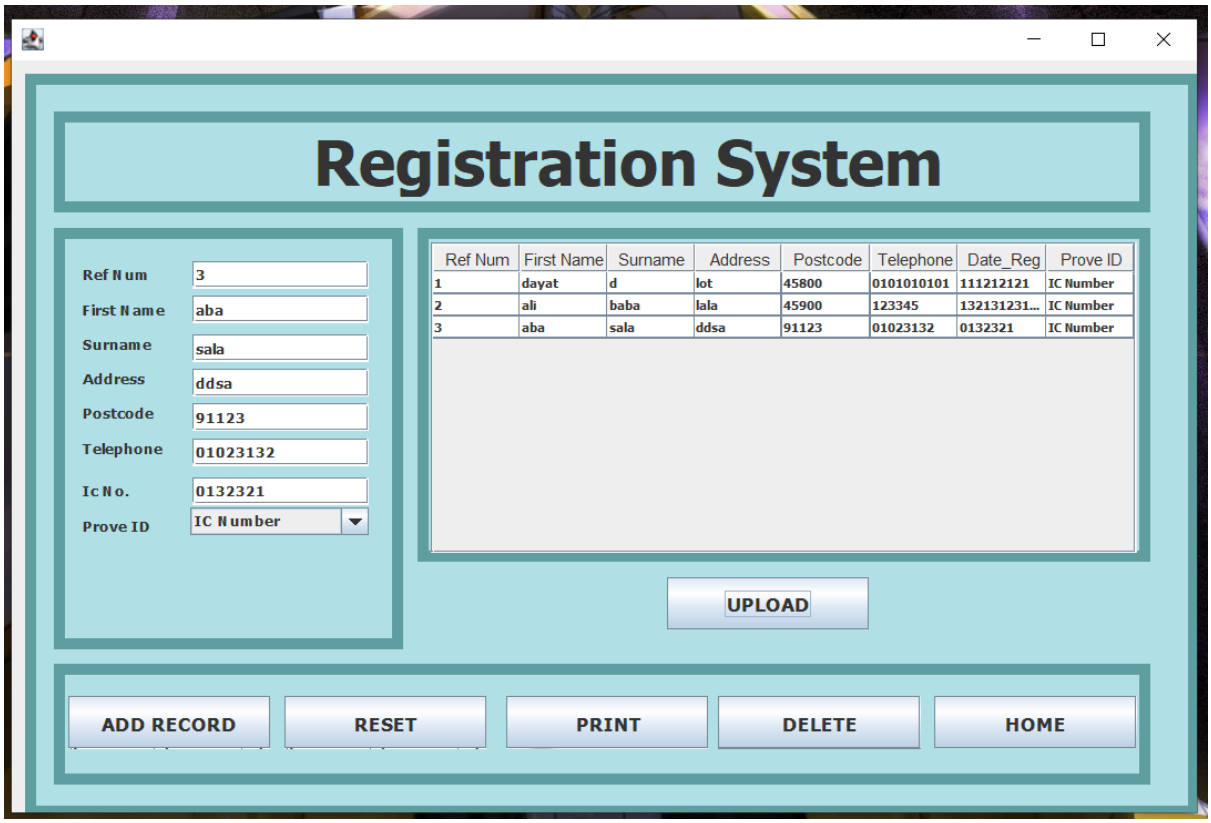
Step 3: if the user click product and choose “gaming decription”, the program will bring to specific page.Then user just click ok to turn to menu page.



Step 4: if the user click product and choose “Advertisement”, the program will bring to specific page.Then user just click ok to turn to menu page.

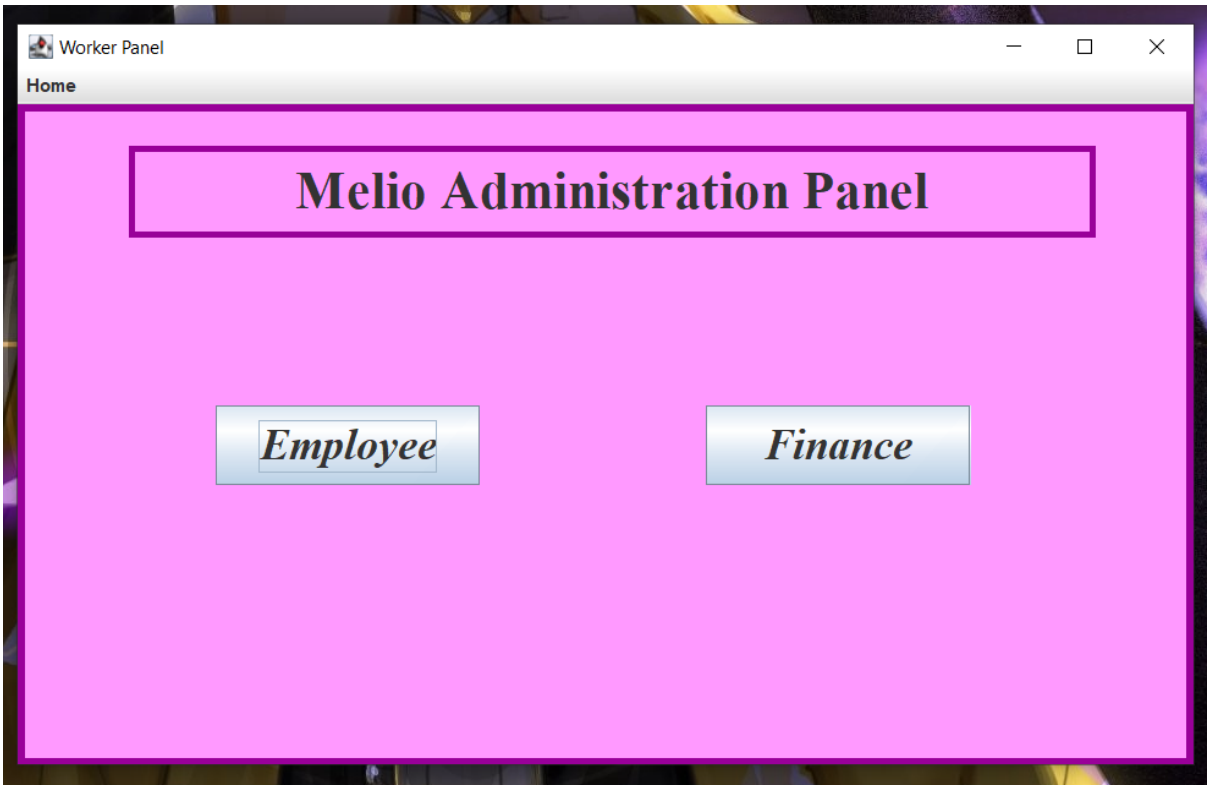


Step 5: If you want to register to this competition, you can just click to customer button for the registration. Then the program will bring user to Registration system.In this page, user can choose wheter to click “Add Record” to add name, “Reset” to enter new name, “Delete” if the user want to delete name, “Print” if user want the hardcopy, “Upload” for save data in soft copy and “Home” to return in home page.

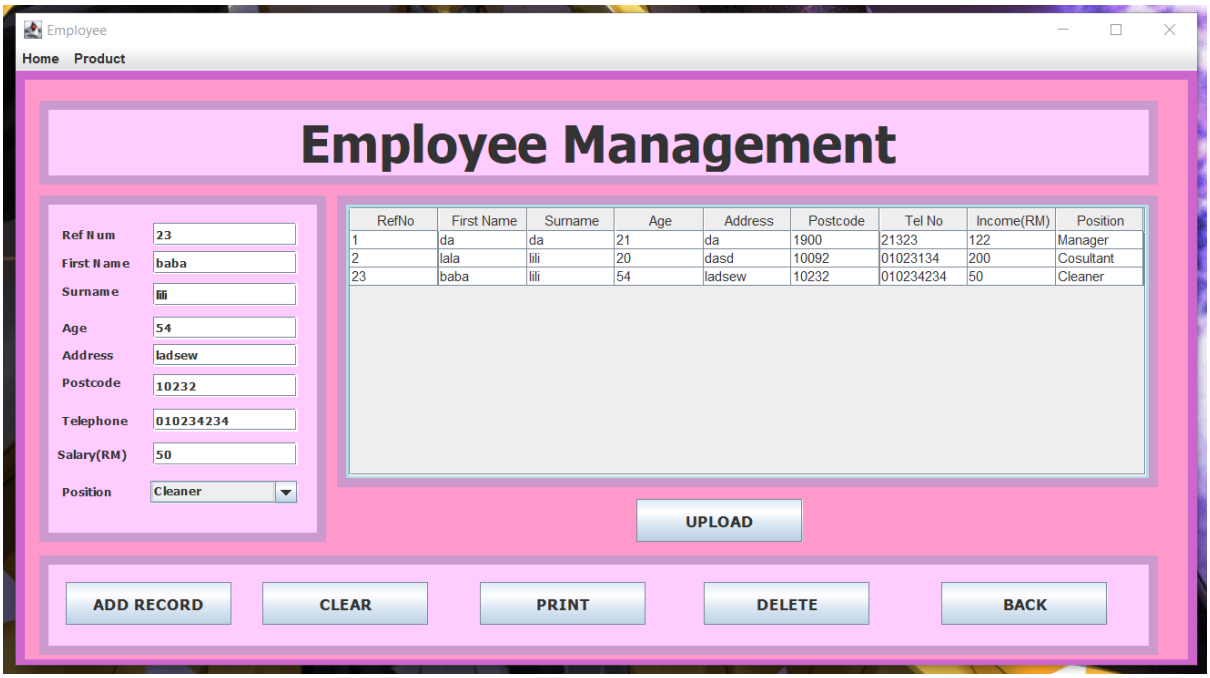


Step 6: If you are a worker for melio enterprise, you can just click to wmployee button for the enter to compony program Then the program will bring user to login to make sure you are employee of melio enterprise. This page contains 3 buttons, which first button “Login” if you enter the password correct you can go to the nect page, if you enter wrong password, you can click “Reset”, And if you want go to main page, just click “Home”

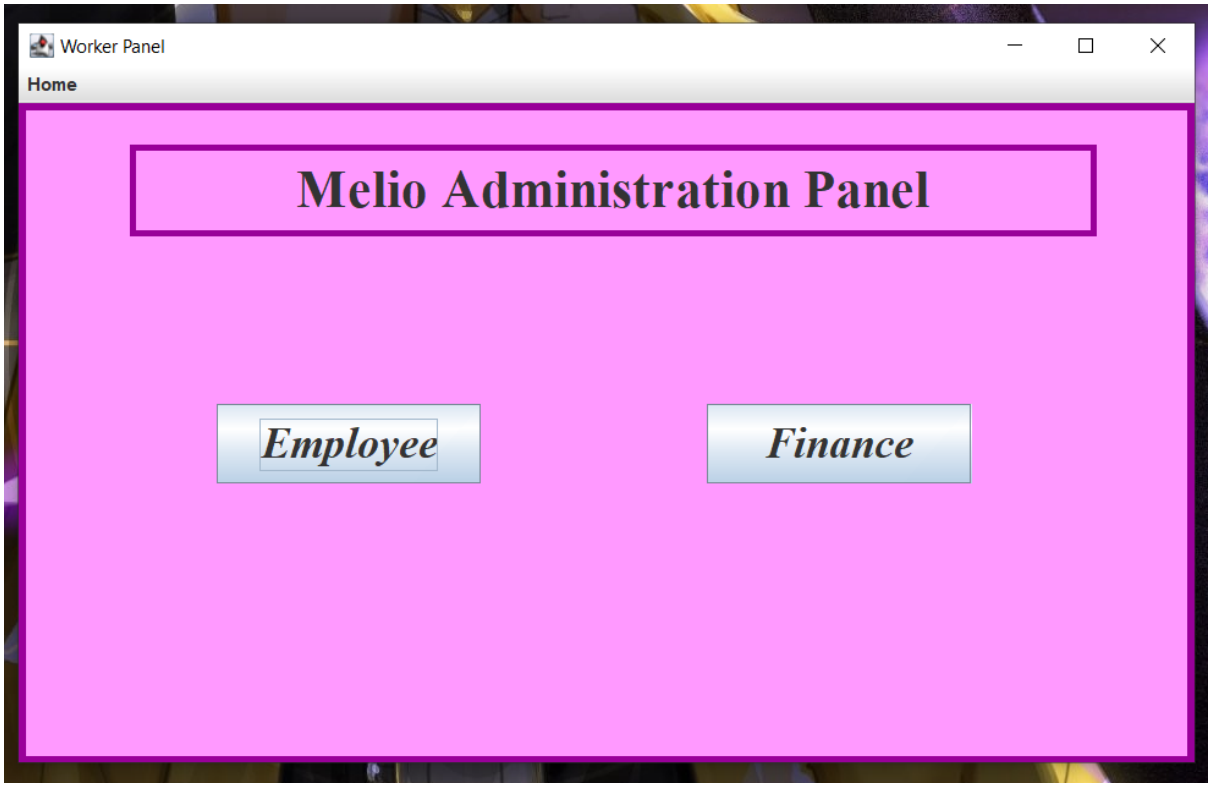
Step 7: If you enter correct password you can go to next page which employee page



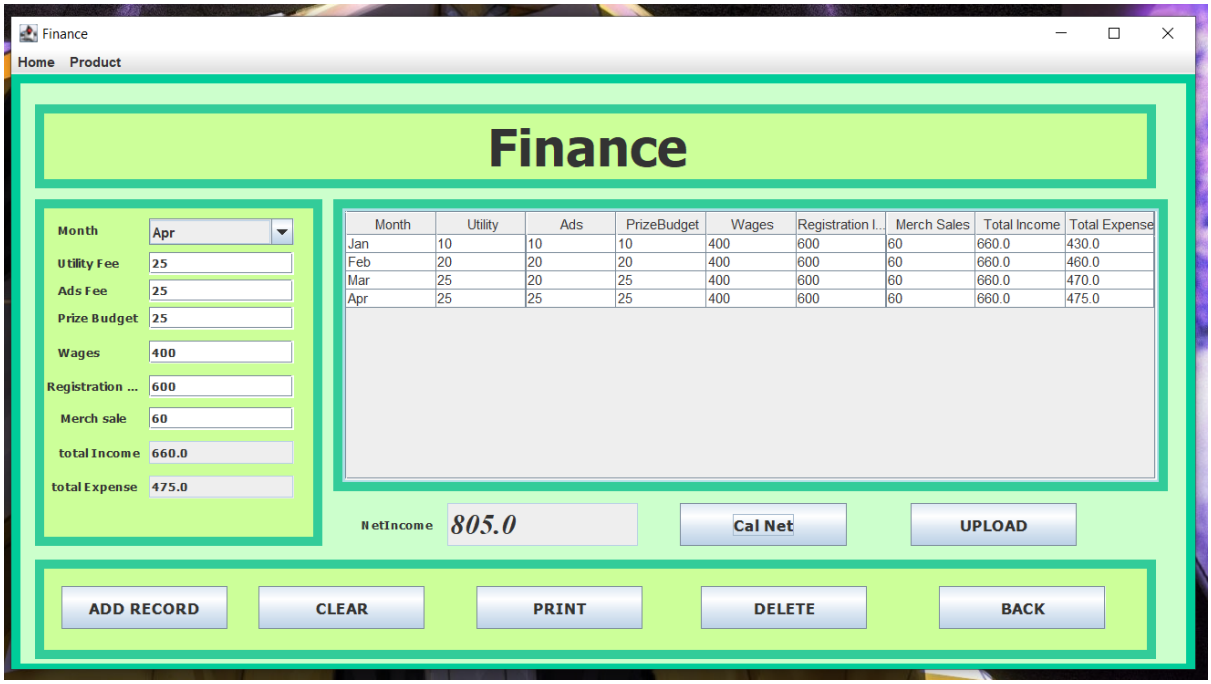
Step 8: In this page, you can choose whether to choce employee or finance. And at the left corner, it has home button for return to login page if the user settle their work. If user click employee button, it will display employee management. In this page, user can choose wheter to click “Add Record” to add name, “Reset” to enter new name, “Delete” if the user want to delete name, “Print” if user want the hardcopy, “Upload” for save data in soft copy and “Home” to return in home page.



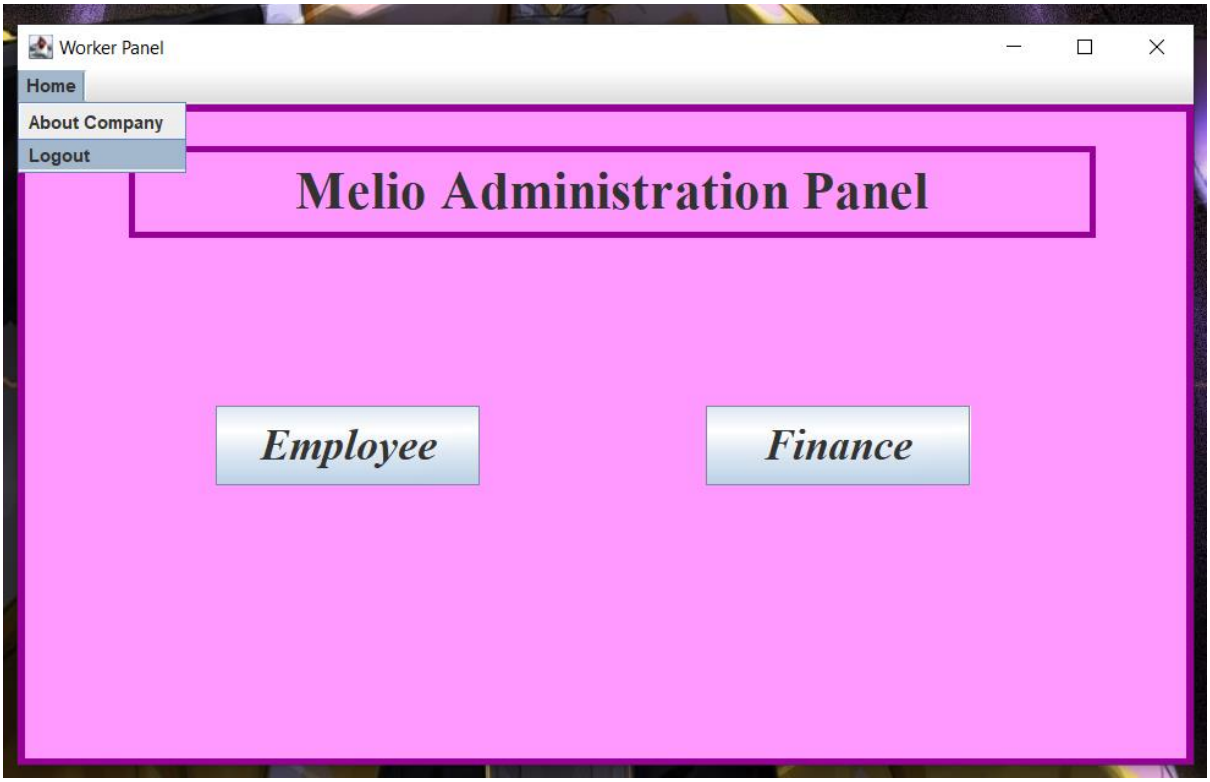
Step 9; from melio administration you can choose finance to check finance for this company.



Step 10; In this page, user can choose wheter to click “Add Record” to add name, “Reset” to enter new name, “Delete” if the user want to delete name, “Print” if user want the hardcopy, “Upload” for save data in soft copy and “Home” to return in home page.



Step 11; If user finish with their work, they can click button home and choose logout to back to login page.So it will bring you to login page.



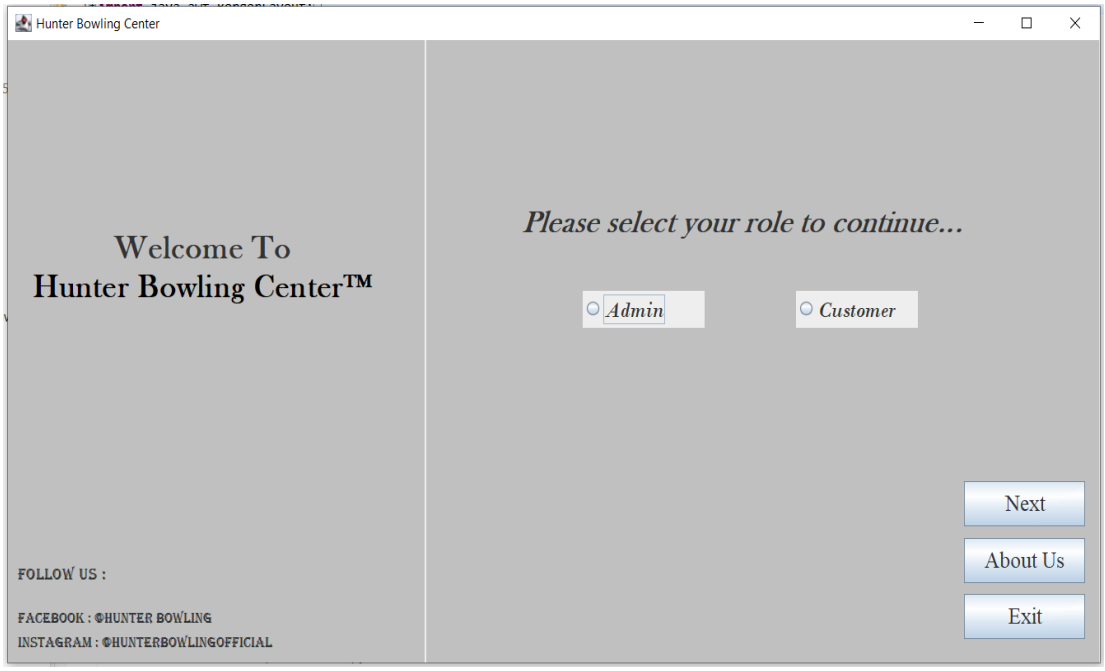
Step 12; User just need to click Home button to go main page.

Step 13: If user want to exit from this program just click Home at menu bar and just click exit to close this program.

4. Gaming (Muhammad Syarif Bin Dollah 279271)

GUI PROGRAM EXECUTION

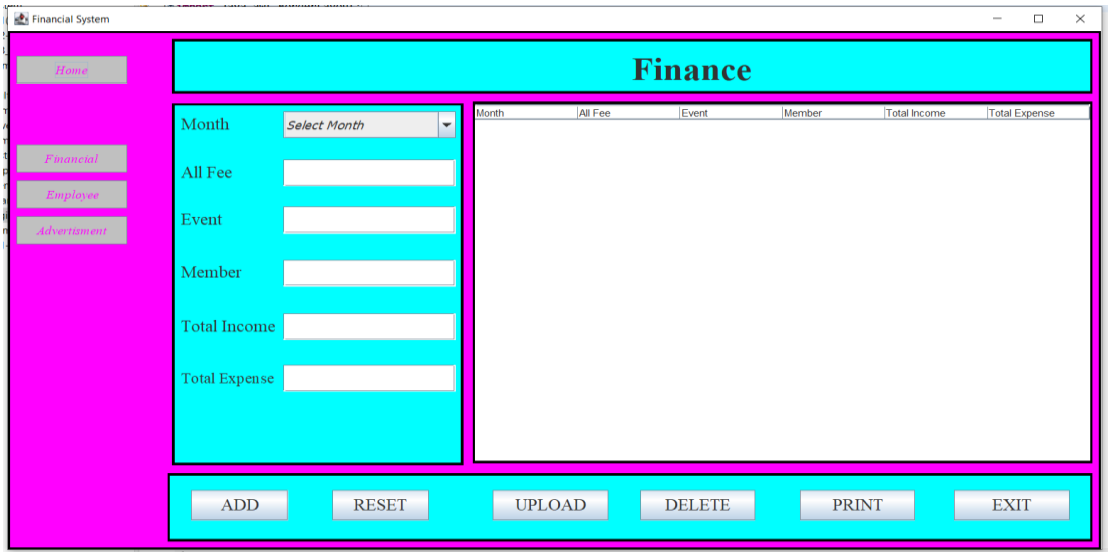
1. Login page



2. AdminMenu page

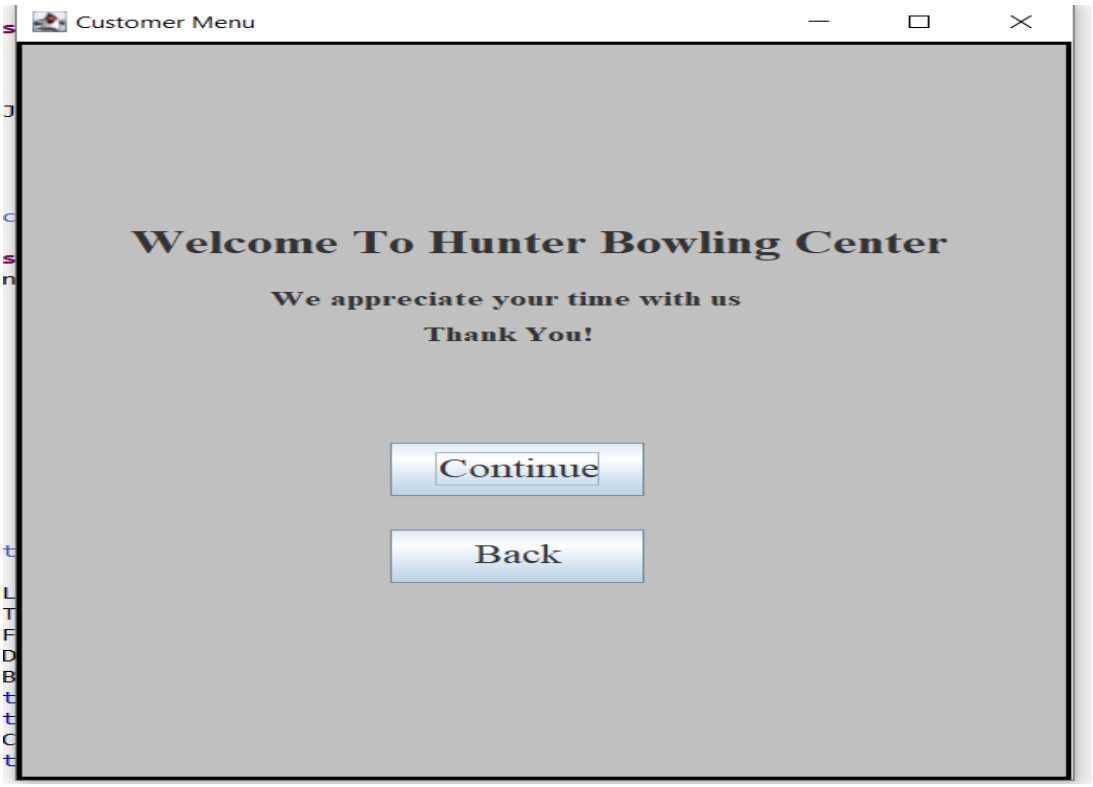
3. Employee page

4. Finance

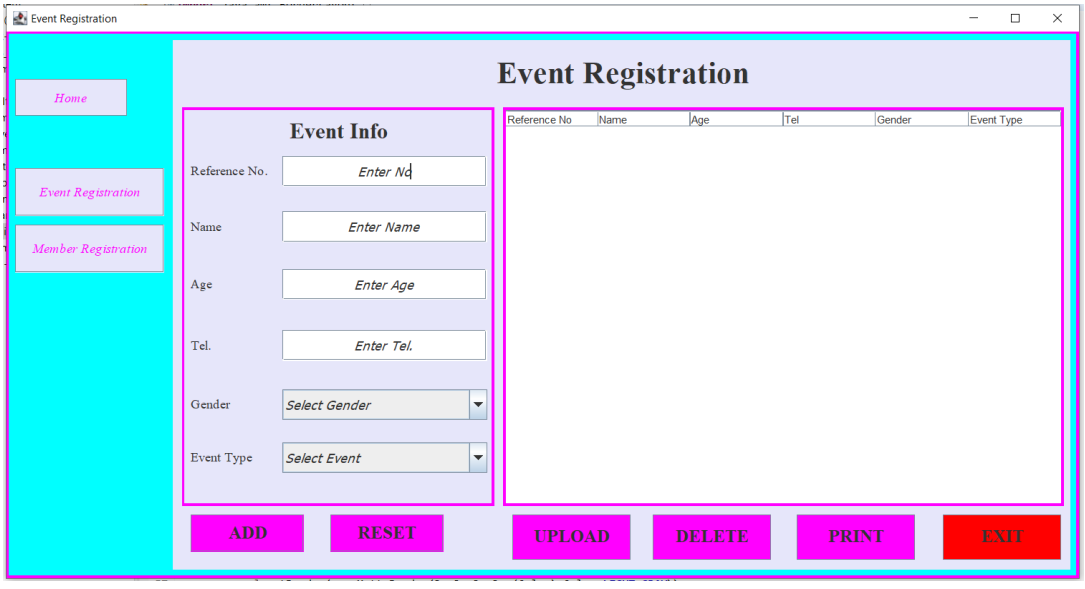


5. Advertisement

6. CustomerMenu page

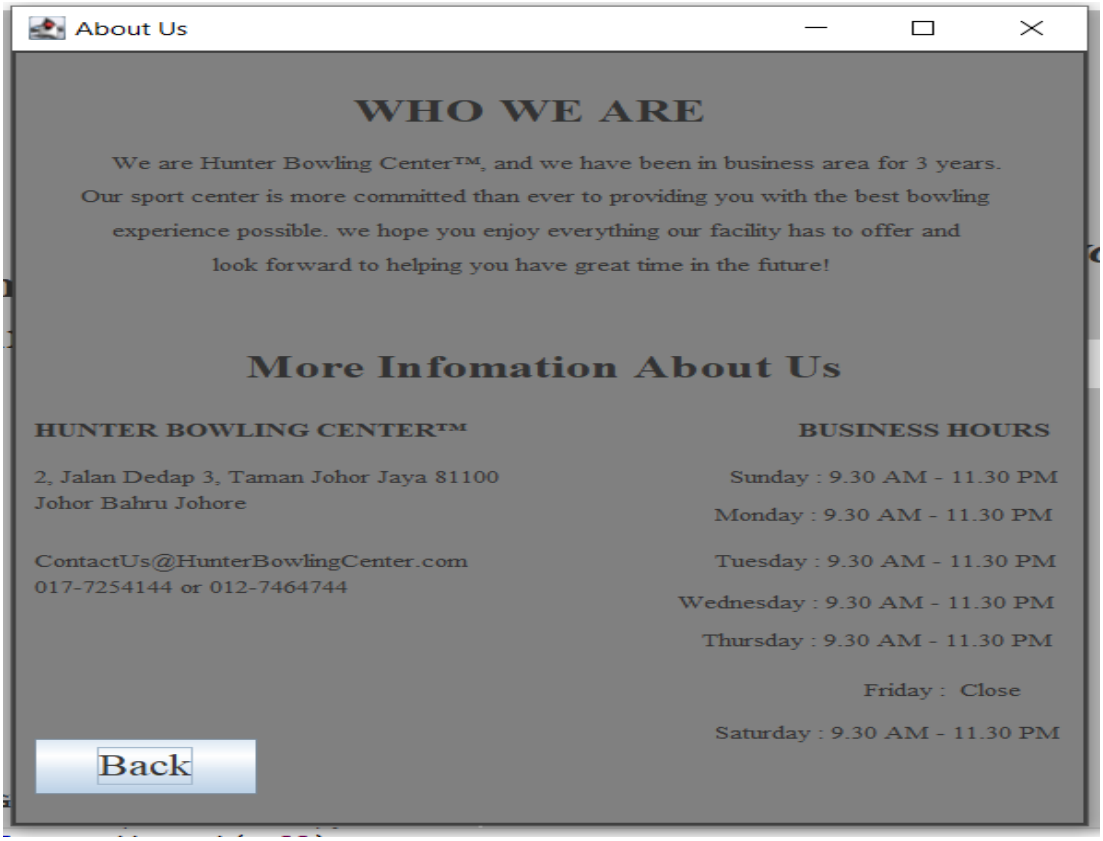


7. Event Registration page



8. Member Registration page

9. About us page



Screenshots of the application's running

1. Futsal Centre (Muhamad Hazman Bin Haris 278632)

CONSTRUCTOR

```
49      public Login() { //constructor
50          setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
51          setBounds(100, 100, 1065, 642);
52          contentPane = new JPanel();
53          contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
54          setContentPane(contentPane);
55          contentPane.setLayout(null);
56
```

```
64      public Employee() {
65          setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
66          setBounds(100, 100, 1000, 636);
67          contentPane = new JPanel();
68          contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
69          setContentPane(contentPane);
70          contentPane.setLayout(null);
71
```

ENCAPSULATION

```
36  public class PlayerRegistration extends JFrame {
37
38      private JPanel contentPane;
39      private JFrame frame;
40      private JTextField txtTeamName;
41      private JTextField txtName;
42      private JTextField txtPhone;
43      private JTable table;
44      private JTextField txtAge;
45
```

Getter

```
170
171         JButton addButton = new JButton("ADD RECORD");
172         addButton.addActionListener(new ActionListener() {
173             public void actionPerformed(ActionEvent e) {
174                 DefaultTableModel model = (DefaultTableModel) table.getModel();
175
176                 model.addRow(new Object[]{
177                         txtTeamName.getText(),
178                         txtName.getText(),
179                         txtAge.getText(),
180                         txtPhone.getText(),
181                         Categorycombobox.getSelectedItem(),
182
183
184                 });
185
```

Setter

```
201
202         JButton resetButton = new JButton("RESET");
203         resetButton.addActionListener(new ActionListener() {
204             public void actionPerformed(ActionEvent e) {
205                 txtTeamName.setText("");
206                 txtName.setText("");
207                 txtAge.setText("");
208                 txtPhone.setText("");
209                 Categorycombobox.setSelectedItem("Make a selection");
210
211
212
213             }
214         });
```

FILE I/O

```
290         JButton btnUpload = new JButton("UPLOAD");
291         btnUpload.addActionListener(new ActionListener() {
292             public void actionPerformed(ActionEvent e) {
293
294                 try {
295                     File file = new File("C:\\Users\\user\\Desktop\\Semester 2\\Programming 2\\Assigment 3 Text File\\Employee.txt");
296                     if(!file.exists()) {
297                         file.createNewFile();
298                         }
299                     FileWriter fw = new FileWriter(file.getAbsoluteFile());
300                     BufferedWriter bw = new BufferedWriter(fw);
301
302                     for (int i=0; i<table.getRowCount(); i++) {
303                         for (int j=0; j<table.getColumnCount(); j++) {
304                             bw.write(table.getModel().getValueAt(i, j) + "              ");
305                             }
306                         bw.write("\n_____\n");
307                         }
308                         bw.close();
309                         fw.close();
310                         JOptionPane.showMessageDialog(null, "Data Exported");
311                     }
312                     catch(Exception ex) {
313                         ex.printStackTrace();
314                     }
315             }
316         });
317         btnUpload.setForeground(Color.WHITE);
318         btnUpload.setFont(new Font("Tahoma", Font.BOLD, 20));
319         btnUpload.setBorder(new LineBorder(new Color(0, 0, 0), 4));
320         btnUpload.setBackground(new Color(128, 0, 128));
321         btnUpload.setBounds(197, 215, 136, 68);
322         panel_2.add(btnUpload);
323     }
324 }
325
```

EXCEPTION HANDLING

```
236         JButton btnPrint = new JButton("PRINT");
237         btnPrint.setForeground(Color.WHITE);
238         btnPrint.setBackground(new Color(128, 0, 128));
239         btnPrint.addActionListener(new ActionListener() {
240             public void actionPerformed(ActionEvent arg0) {
241                 try {
242                     table.print();
243                 } catch (java.awt.print.PrinterException e) {
244                     System.err.format("No printer found", e.getMessage());
245                 }
246             }
247         });
248         btnPrint.setFont(new Font("Tahoma", Font.BOLD, 20));
249         btnPrint.setBounds(583, 26, 136, 68);
250         btnPrint.setBorder(new LineBorder(new Color(0, 0, 0), 4));
251         panel_1_2.add(btnPrint);
```

POLYMORPHISM

```
269         @Override
270         public void actionPerformed(ActionEvent e) {
271             try {
272             File file = new File("C:\\Users\\user\\Desktop\\Semester 2\\Programming 2\\Assigment 3 Text File\\Player Registration.txt");
273             if(!file.exists()) {
274                 file.createNewFile();
275                 }
276             FileWriter fw = new FileWriter(file.getAbsoluteFile());
277             BufferedWriter bw = new BufferedWriter(fw);
278 
279             for (int i=0; i<table.getRowCount(); i++) {
280                 for (int j=0; j<table.getColumnCount(); j++) {
281                     bw.write(table.getModel().getValueAt(i, j) + "    ");
282                     }
283                 bw.write("\n_____\n");
284                 }
285                 bw.close();
286                 fw.close();
287                 JOptionPane.showMessageDialog(null, "Data Exported");
288             }
289             catch(Exception ex) {
290                 ex.printStackTrace();
291             }
292         }
293     });
```

2. Car Racing (Pee Foo Ye 278632)

1. **Constructor**

```
public Login() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 699, 529);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);
    contentPane.setLayout(null);
```

2. **Encapsulation**

```
private JPanel contentPane;
private JTextField textField_Name;
private JTextField textField_MobileNo;
private JTable table;
private JTextField textField_ICNo;
private JTextField textField_Username;
private JTextField textField_Password;
```

3. **Inheritance**

```
public class EmployeeManagement extends JFrame{
```

Setter

```
public void actionPerformed(ActionEvent e) {
    DefaultTableModel model = (DefaultTableModel) table.getModel();
    model.addRow(new Object[]{
                    textField_Name.getText(),
                    textField_ContractNo.getText(),
                    textField_ICNo.getText(),
                    textField_Position.getText(),
                    textField_Salary.getText(),
```

Getter

```
public void actionPerformed(ActionEvent e) {
                    textField_Name.setText("");
                    textField_ContractNo.setText("");
                    textField_ICNo.setText("");
                    textField_Position.setText("");
                    textField_Salary.setText("");
```

4. Polymorphism (Overriding)

EmployeeRegistrationSystem

```
JButton btnUpload = new JButton("UPLOAD");
btnUpload.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File f = new File("C:\\Users\\raymo\\OneDrive\\Desktop\\Sem 2\\Programming 2\\Project\\Employee.txt");
        if(!f.exists()) {
            f.createNewFile();
            }
        FileWriter fw = new FileWriter(f.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "   ");
                }
            bw.write("\n________\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

CustomerRegistrationSystem

```
JButton btnApply = new JButton("APPLY");
btnApply.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File f = new File("C:\\Users\\raymo\\OneDrive\\Desktop\\Sem 2\\Programming 2\\Project\\Customer.txt");
        if(!f.exists()) {
            f.createNewFile();
            }
        FileWriter fw = new FileWriter(f.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "   ");
                }
            bw.write("\n________\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

5. File I/O

AdvertisementAndMarketing

```
JButton btnDone = new JButton("DONE");
btnDone.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {

                if (table.getRowCount() != 0) {
                        JOptionPane.showMessageDialog(null, "Advertisement and Marketing has been set!");
                        try {
                            File f = new File("C:\\Users\\raymo\\OneDrive\\Desktop\\Sem 2\\Programming 2\\Project\\Advertisement.txt");
                            if(!f.exists()) {
                                f.createNewFile();
                                }
                            FileWriter fw = new FileWriter(f.getAbsoluteFile());
                            BufferedWriter bw = new BufferedWriter(fw);

                            for (int i=0; i<table.getRowCount(); i++) {
                                for (int j=0; j<table.getColumnCount(); j++) {
                                    bw.write(table.getModel().getValueAt(i, j) + "   ");
                                    }
                                bw.write("\n________\n");
                                }
                                bw.close();
                                fw.close();
                                JOptionPane.showMessageDialog(null, "Data Exported");
                            }
                            catch(Exception ex) {
                                ex.printStackTrace();
                            }
                } else if (table.getRowCount() == 0) {
                    JOptionPane.showMessageDialog(null, "No data has been set!");
                }

        }
});
```

EmployeeRegistrationSystem

```
JButton btnUpload = new JButton("UPLOAD");
btnUpload.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File f = new File("C:\\Users\\raymo\\OneDrive\\Desktop\\Sem 2\\Programming 2\\Project\\Employee.txt");
        if(!f.exists()) {
            f.createNewFile();
            }
        FileWriter fw = new FileWriter(f.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "   ");
                }
            bw.write("\n________\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

CustomerRegistrationSystem

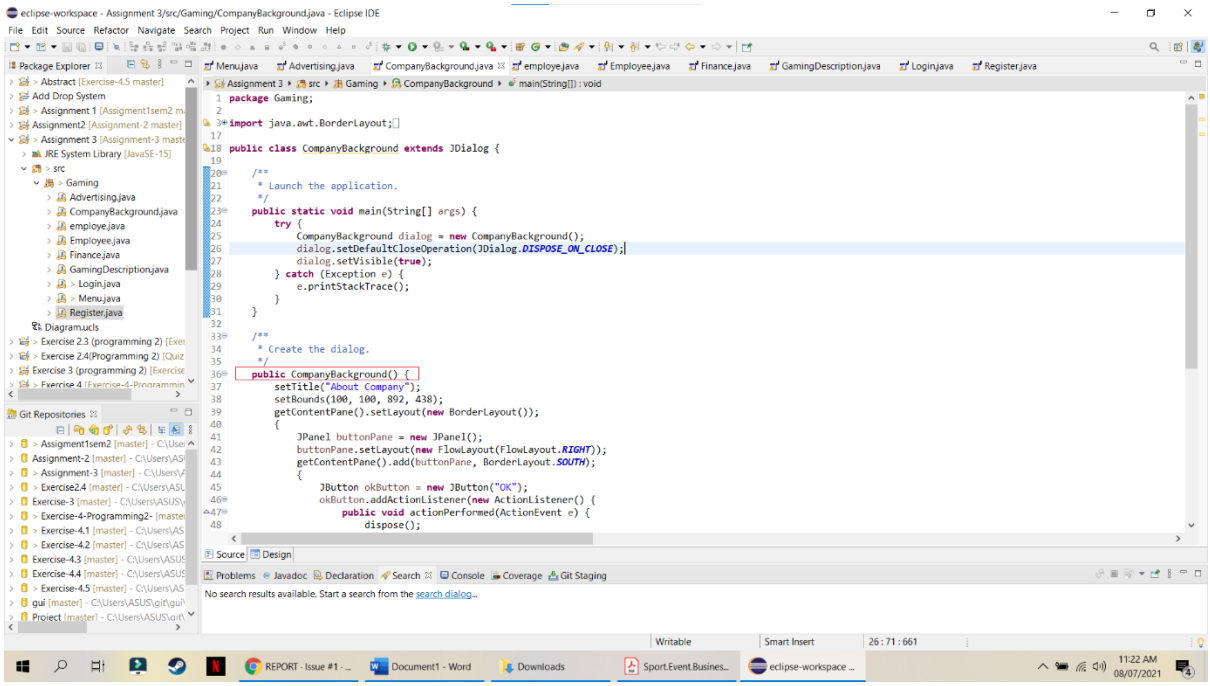
```
JButton btnApply = new JButton("APPLY");
btnApply.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File f = new File("C:\\Users\\raymo\\OneDrive\\Desktop\\Sem 2\\Programming 2\\Project\\Customer.txt");
        if(!f.exists()) {
            f.createNewFile();
            }
        FileWriter fw = new FileWriter(f.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "  ");
                }
            bw.write("\n________\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

6. Exception Handling

```
btn_Print.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        try {
            table.print();
        } catch (java.awt.print.PrinterException ee) {
            System.err.format("No printer found", ee.getMessage());
        }
    }
});
```

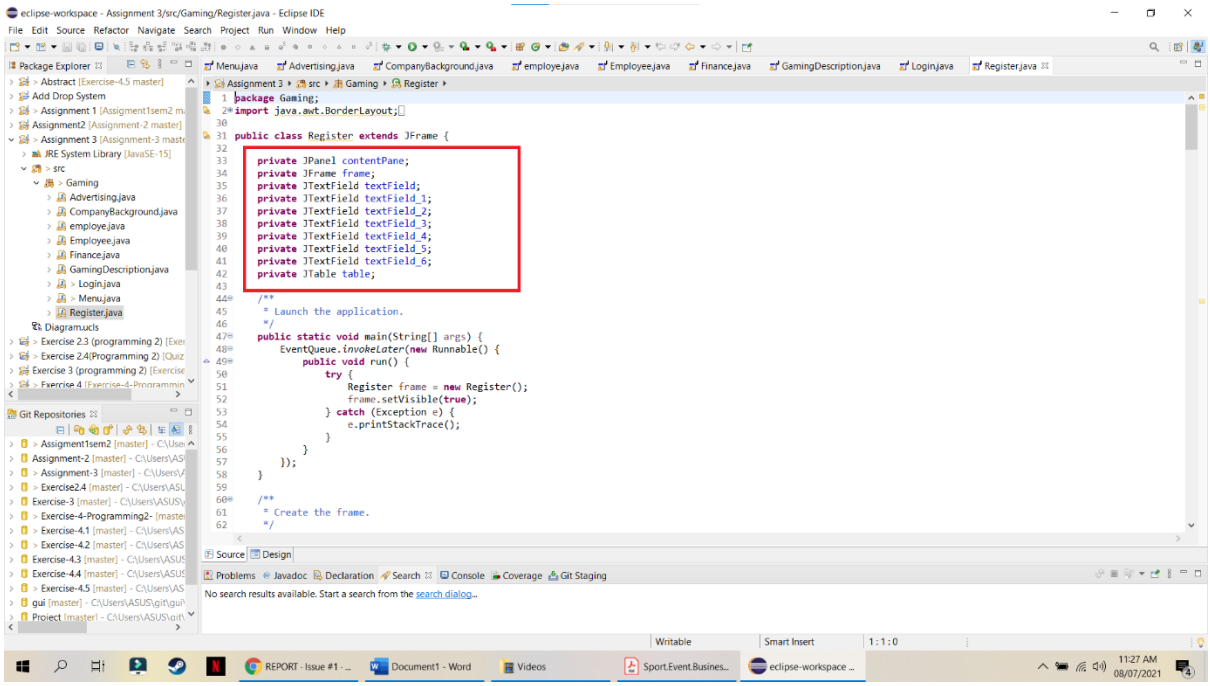
3. Gaming (Muhammad Hidayat Bin Ibrahim 279395)

Constructor



The constructor inside this screenshot is default constructor, that is, the constructor with no arguments. All the codes of the design will be in this constructor. The constructor is called when an object of a class is created. It can be used to set initial values in the constructor.

Encapsulation



In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class. We usually declare the variables of a class as private. If they want to excess the code, they need to provide public setter and getter methods to modify and view the variables values. From this example, all the variables of Register class are declared as private.

Inheritance

```
1  package Gaming;
2
3  import java.awt.BorderLayout;
22
23 public class Login extends JFrame {
24     private JPanel contentPane;
25     private JFrame frame;
26     private JTextField UsernameText;
27     private JTextField PasswordText;
28
29     /**
30      * Launch the application.
31      */
```

Inheritance is a mechanism wherein a new class is derived from an existing class. classes may inherit or acquire the properties and methods of other classes. A class derived from another

class is called a subclass, whereas the class from which a subclass is derived is called a superclass. As the picture show, the Login class extends JFrame. It means Login class inherits JFrame.

Polymorphism (overloading and overriding)

a. Overloading

```
JButton btnHome = new JButton("Home");
btnHome.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        frame = new JFrame();
        if (JOptionPane.showConfirmDialog(frame, "Confirm if you want to back", "Login System",
                JOptionPane.YES_NO_OPTION) == JOptionPane.YES_NO_OPTION) {
            Menu frame = new Menu();
            frame.setModalExclusionType(null);
            frame.setVisible(true);
            dispose();
        }
    }
});
```

Method, addActionListener() is used many time in this project. Subclass employee overrides a method inherit from a superclass Abstract Button. The overriding method has the same name, number and type of parameters, and return type as the overriding method

b. Overriding

```
JButton btnUpload = new JButton("UPLOAD");
btnUpload.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File file = new File("C:\\\\Users\\\\ASUS\\\\eclipse-workspace\\\\Employee");
        if(!file.exists()) {
            file.createNewFile();
            }
        FileWriter fw = new FileWriter(file.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "  ");
                }
            bw.write("\n_______\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

Overriding is a feature that allows a subclass or child class to provide a specific implementation of a method that is already provided by one of its super-classes or parent classes. When a method in a subclass has the same name, same parameters or signature, and same return type as a method in its super class, then the method in the subclass is said to

override the method in the super class. For example, in this project, the mouseClicked method is an overriding because it also appears in other JFrame.

FILE I/O

```
JButton btnUpload = new JButton("UPLOAD");
btnUpload.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
        File file = new File("C:\\\\Users\\\\ASUS\\\\eclipse-workspace\\\\Employee");
        if(!file.exists()) {
            file.createNewFile();
            }
        FileWriter fw = new FileWriter(file.getAbsoluteFile());
        BufferedWriter bw = new BufferedWriter(fw);

        for (int i=0; i<table.getRowCount(); i++) {
            for (int j=0; j<table.getColumnCount(); j++) {
                bw.write(table.getModel().getValueAt(i, j) + "  ");
                }
            bw.write("\n_______\n");
            }
            bw.close();
            fw.close();
            JOptionPane.showMessageDialog(null, "Data Exported");
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

There is a file input/output here which is the participant .txt file. When the user c licks the upload button it will save all the information on the table to this employee .txt file

Exception Handling

```
        catch(Exception ex) {
            ex.printStackTrace();
        }
    }
});
```

The exception handling is exception ex. The exception handling is through try and catch. If there is error within the try code it will throw an exception.

4. Gaming (Muhammad Syarif Bin Dollah 279271)

1. Constructor Without Argument

```
55
56    /**
57     * Create the frame.
58     */
59    public Employee() {
60        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
61        setBounds(100, 100, 1239, 673);
62        contentPane = new JPanel();
63        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
64        setContentPane(contentPane);
65        contentPane.setLayout(null);
66
67        JPanel panel = new JPanel();
68        panel.setBackground(Color.MAGENTA);
69        panel.setBorder(new MatteBorder(3, 3, 3, 3, (Color) Color.PINK));
70        panel.setBounds(0, 0, 1225, 636);
71        contentPane.add(panel);
72        panel.setLayout(null);
```

2. Inheritance

```
23 public class Finance extends JFrame {
24
```

3. File I/O

```
File file = new File("C:\\Users\\User\\OUTPUT\\WORKERS.txt");
      if(!file.exists()) {
            file.createNewFile();
            }
FileWriter fw = new FileWriter(file.getAbsoluteFile());
BufferedWriter bw = new BufferedWriter(fw);

for (int i=0; i<table.getRowCount(); i++) {
      for (int j=0; j<table.getColumnCount(); j++) {
      bw.write(table.getModel().getValueAt(i, j) + "   ");
      }
```

4. Exception Handling

```
try {
    table.print();
} catch (java.awt.print.PrinterException e1) {
    System.err.format("No printer found", e1.getMessage());
}
```