UNIVERSITI TEKNOLOGI MALAYSIA

OBJECT ORIENTED PROGRAMMING (SECJ2154)

SEMESTER 2 2023/2024

GROUP PROJECT

Hajj and Umrah Booking System

Muhammad Faiz Bin Zakaria (A22EC0208)

Muhammad Ilman Bin Mohd Khairi (A22EC0215)

Alif Aiman Bin Mansor (A22EC0137)

Muhammad Iman Amier Bin Abu Bakar (A22EC0128)

2 SECRH

SECTION 04

Lecturer:

MADAM LIZAWATI MI YUSUF

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Table of Contents

SECTION A: PROJECT DESCRIPTION 4
Synopsis: 4
Objective and Scope: 4
Work flow: 6
OO Concepts: 17
SECTION B: CLASS DIAGRAMS 23
Class User 23
Class Jemaah 24
Class Doctor 25
Class Embassy 25
Class pelancongan_package 26
Class Kursus 26
Enum class Hajj_Umrah_Package 27
Class Flight 27
Class Hotel 27
Interface class userDisplay 28
SECTION C: SOURCE CODE AND USER MANUAL 29
Main Screen: 29
From Jemaah Point of View 29
Login page 29
Show Jemaah Info 31
Hajj and Umrah Package 31
Update Jemaah Info 35
Register new jemaah 35
From Doctor Point of View 38
Login page 38
Show Doctor Info 39
Medical Application Business 40
Update Doctor Information 43
From Embassy Point of View 44
Login Page 45
Show Embassy Info 46
Visa Application Business 47
Update Embassy Information 51

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Table of Figures

Figure 1 **6**
Figure 2 **9**
Figure 3 **11**
Figure 4 **12**
Figure 5 **13**
Figure 6 **16**
Figure 7 **17**
Figure 8 **18**
Figure 9 **19**
Figure 10 **19**
Figure 11 **20**
Figure 12 **21**
Figure 13 **21**
Figure 14 **21**
Figure 15 **22**
Figure 16 **23**
Figure 17 **29**
Figure 18 29
Figure 19 **29**
Figure 20 **30**
Figure 21 **30**
Figure 22 **31**
Figure 23 **32**
Figure 24 **32**
Figure 25 **32**
Figure 26 **33**
Figure 27 **33**
Figure 28 **34**
Figure 29 **35**
Figure 30 **35**
Figure 31 **35**
Figure 32 **36**
Figure 33 **36**

Figure 34 **36**
Figure 35 **37**
Figure 36 **37**
Figure 37 **37**
Figure 38 **38**
Figure 39 **38**
Figure 40 **38**
Figure 41 **39**
Figure 42 **39**
Figure 43 **40**
Figure 44 **40**
Figure 45 **41**
Figure 46 **42**
Figure 47 **43**
Figure 49 **44**
Figure 50 **44**
Figure 51 **45**
Figure 52 **45**
Figure 53 **45**
Figure 54 **45**
Figure 55 **45**
Figure 56 **46**
Figure 57 **46**
Figure 59 **47**
Figure 60 **47**
Figure 61 **48**
Figure 62 **48**
Figure 63 **49**
Figure 64 **50**
Figure 65 **50**
Figure 66 **51**
Figure 67 **52**
Figure 68 **52**
Figure 69 **53**

SECTION A: PROJECT DESCRIPTION

The work in this page has been done by: **MUHAMMAD ILMAN BIN MOHD KHAIRI** *Date:* **21/6/2024**

Synopsis:

We offer a project that will oversee the hajj and umrah system to make it run more smoothly and flawlessly. Our system offers a wide range of options, including those concerning pilgrims' health, visa approval, tour package selection, Hajj and Umrah package selection, and room size. In order to facilitate users' ability to perform prayer during the later Hajj and Umrah seasons, all of these alternatives are available. We also supply pilgrims who want to see fascinating historical sites travel packages at reasonable, dependable rates. This method intends to provide the public with the most convenient Umrah and Hajj worship packages while also making it easier for administrators to oversee pilgrims in terms of tourism, embassy relations, and health care.

Objective and Scope:

The purpose of the Umrah Hajj system is to make the physical package purchase procedure easier. Users of this system are limited to selecting the desired package online, along with a few supplementary features that encourage them to use the system. The following are the objectives of the proposed system.

1) allowing any individual to complete the Hajj and Umrah, while also raising the congregation's degree of faith and making sure participants are aware of both the practical and spiritual aspects of these pilgrimages.
2) providing pilgrims with aid in organizing their Hajj and Umrah, including help with travel, lodging, health, required documentation, and creating a clear schedule to ensure a hassle-free, well-planned experience.
3) providing pilgrims with assistance regarding the Hajj and Umrah procedures through seminars designed to meet their religious and emotional needs.

4) allowing the audience to experience learning about and getting to know the local culture while also highlighting the significance of the Hajj and Umrah to deepen their understanding of this act of worship.

Additionally, our program contains a number of modules that will assist us in implementing its features. Our system's reach is as follows:

1) Users can choose the Hajj/Umrah packages to choose from, update their personal information, and decide whether or not to travel.
2) This approach assigns a doctor to each congregation, enables the administrator to determine what ailment the pilgrims are facing, what medication is required, and if the pilgrims may resume worship or not.
3) Inform the congregation if their visa has been approved or denied, advising them to reapply to avoid any issues during immigration.
4) This program shows five different worship sites based on the date that has been established, along with an alluring tour package that can be chosen once the congregation has concluded their time in worship.
5) allowing the congregation to select the kind of accommodation that best suits their needs at a fair price.

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **21/6/2024**

Work flow:

The workflow for our Hajj Umrah Application:

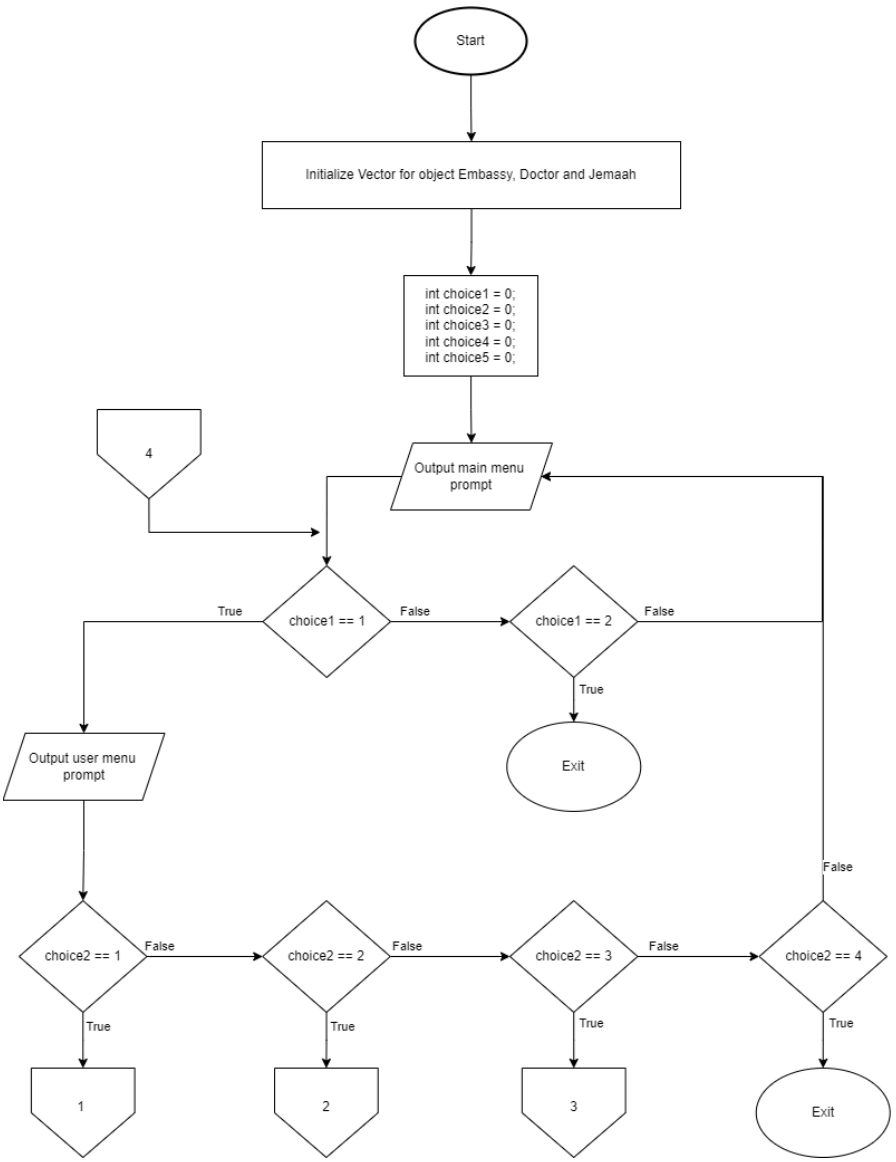


Figure 1

This is the workflow that all users will go through. The system starts with Login, If the user choose to exit instead, user will end the application. Then, user need to input the

choice2 to choose which role is the user. Its for authorization and authentication. (1) is for Jemaah, (2) Doctor, and (3) is for Embassy.

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Flow Chart Jemaah

After the user chooses (1) at the main user display, it will display the pre-jemaah menu and ask the user to press (1) for login and (2) for new jemaah registration. if user press (1), user need to enter identification card number and assigned jemaah ID to log in. The system will check whether the jemaah exists or not. if the jemaah does not exist. It will go back to the main display, if the jemaah exists, the system will display jemaah menu. system will ask the user to enter (1) to display jemaah info, (2) for choosing ibadah package and will invoke the chooseIbadahPackage() function. The explanation for this function will be explained in the next flowchart at figure 2. users need to enter (3) to update their credentials. If a user enters (2) for new jemaah registration, the system will ask to enter their new name, contact, identification card number, email and age. and the system will give their newly assigned jemaah ID and the new jemaah will be added to the jemaah vector.

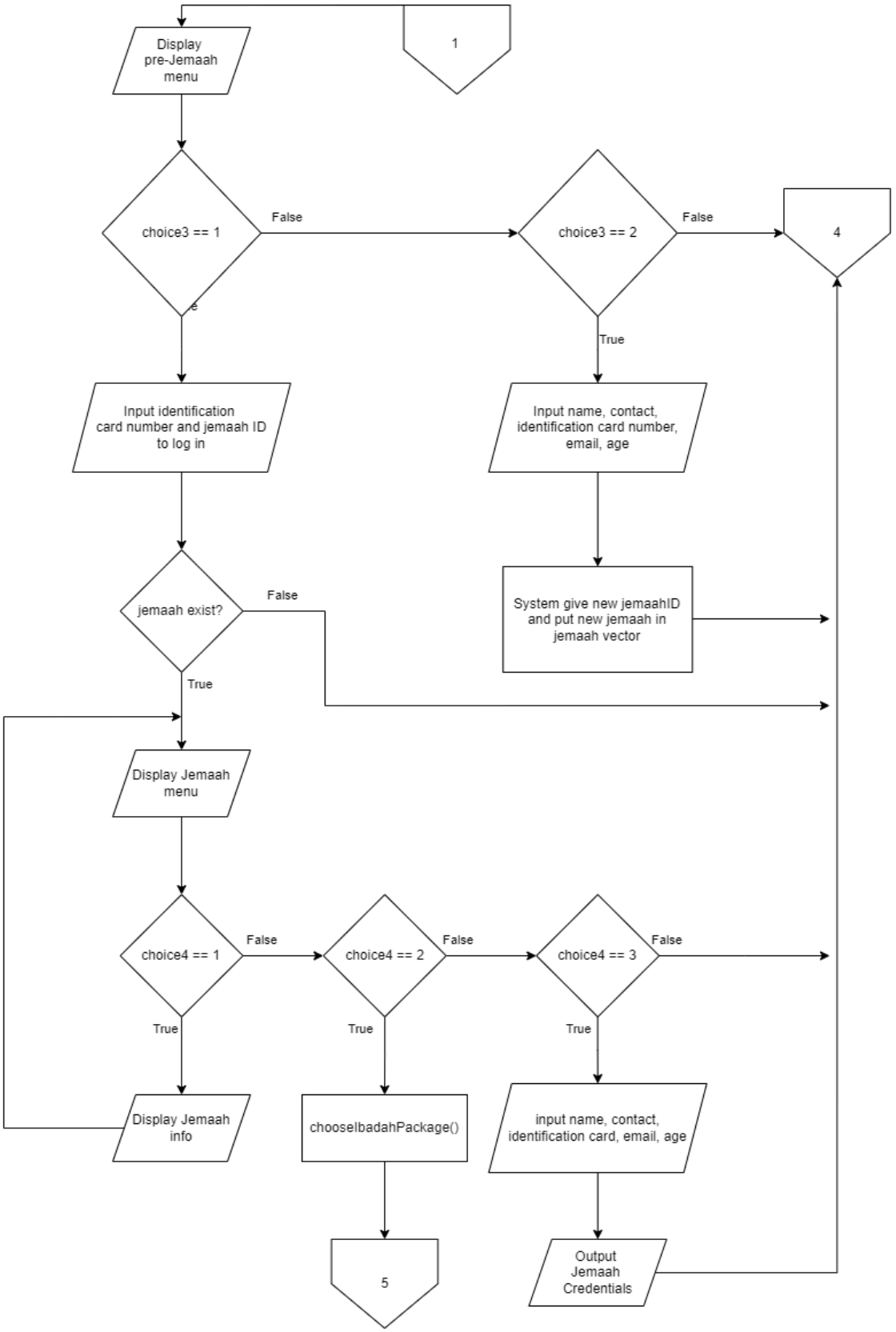

1
Display pre-Jemaah menu
choice3 == 1
False
choice3 == 2
False
4
True
Input identification card number and jemaah ID to log in
Input name, contact, identification card number, email, age
jemaah exist?
False
System give new jemaahID and put new jemaah in jemaah vector
True
Display Jemaah menu
choice4 == 1
False
choice4 == 2
False
choice4 == 3
False
True
True
True
Display Jemaah info
chooseIbadahPackage()
input name, contact, identification card, email, age
5
Output Jemaah Credentials

Figure 2

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Flowchart chooseIbadahPackage()

This flowchart is the continuity from the previous flowchart if the user enters (2) at the jemaah display. The system will ask the user to choose their preferred ibadah package. Once chosen, the system will initialize the respective package's ibadah, vector flight_ibadah, vector hotel_ibadah and kursus_ibadah thus will initialize a jemaah's pakejIbadah. The system will ask the user if the user wants to add a tour package after their Hajj or Umrah. if the user enters (1) it will ask the user to choose their desired tour package, if the user enters (2). it will take the user back to the login display. When the user enters the tour package they desire, the system will initialize the places, tour_Guide, price_Pelancongan, flight_Pelancongan, and hotel_Pelancongan following their respective tour package thus initializing the jemaah tour package and will remove the flight back. Lastly, the system will invoke the choose_Hotel_Room() function. The explanation of this function will be explained in the next flowchart at figure 3.

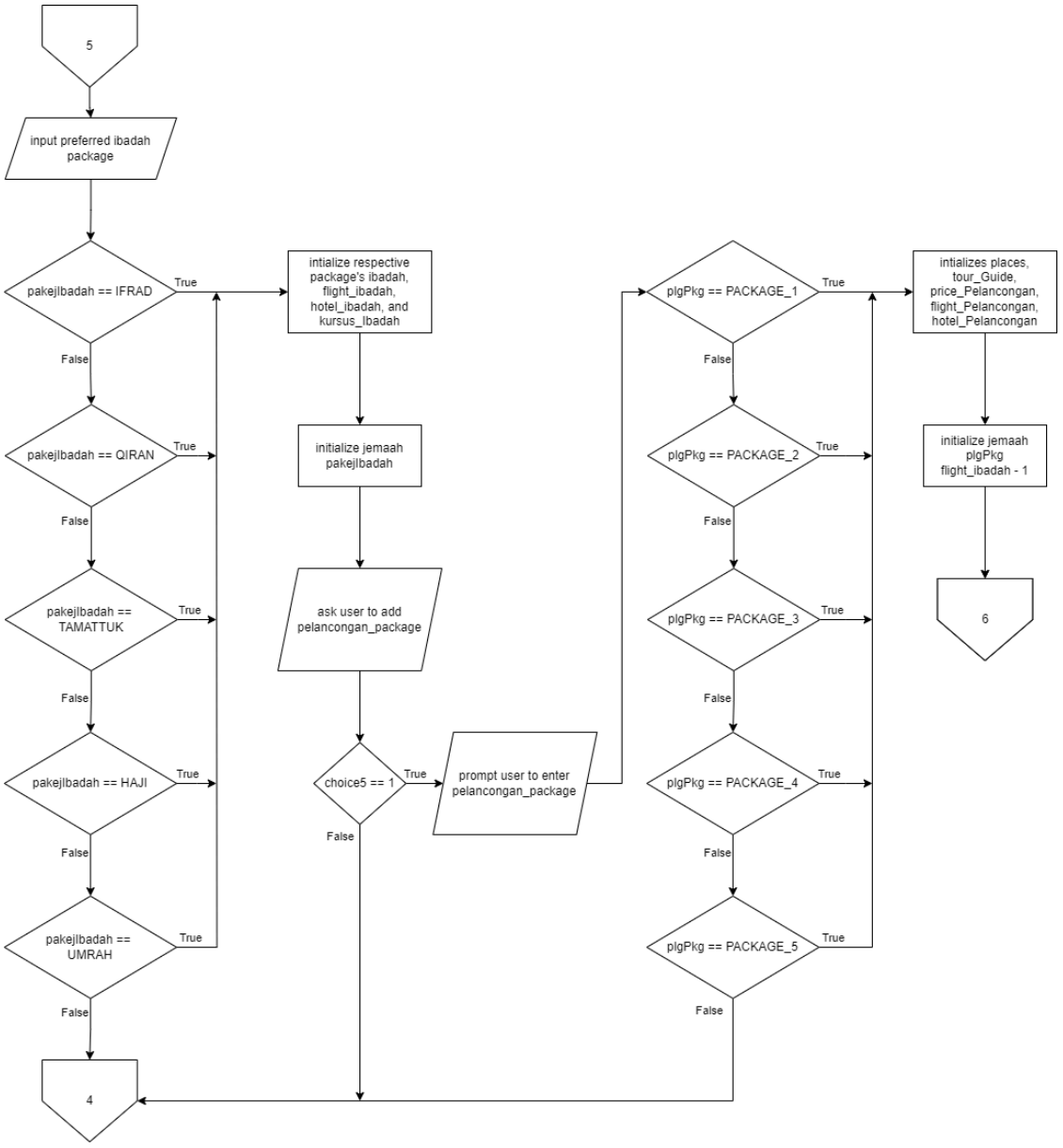

5
input preferred ibadah package
pakejIbadah == IFRAD
True
False
intialize respective package's ibadah, flight_ibadah, hotel_ibadah, and kursus_Ibadah
pakejIbadah == QIRAN
True
False
initialize jemaah pakejIbadah
pakejIbadah == TAMATTUK
True
False
ask user to add pelancongan_package
pakejIbadah == HAJI
True
False
choice5 == 1
True
False
prompt user to enter pelancongan_package
pakejIbadah == UMRAH
True
False
4
plgPkg == PACKAGE_1
True
False
intializes places, tour_Guide, price_Pelancongan, flight_Pelancongan, hotel_Pelancongan
plgPkg == PACKAGE_2
True
False
initialize jemaah plgPkg flight_ibadah - 1
plgPkg == PACKAGE_3
True
False
6
plgPkg == PACKAGE_4
True
False
plgPkg == PACKAGE_5
True
False

Figure 3

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Flowchart choose_Hotel_Room()

This flowchart is the continuity from the previous flowchart. The system will ask the user to enter the room size the user desired. if the user's choices are within range, it will assign the room size according to the user's choice and add the price of the room size to the price_pelancongan variable. it will ask the same question until you have chosen the room size for all hotels. After completion, it will display the tour packages info.

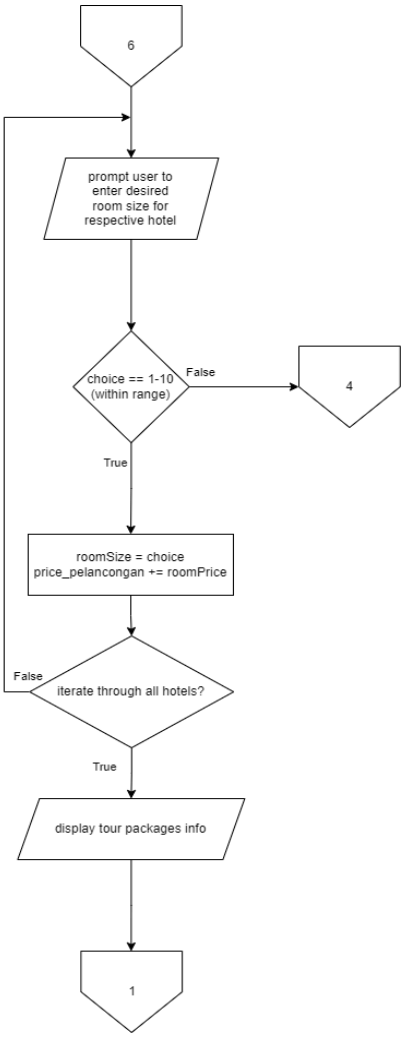


Figure 4

The work in this page has been done by: **MUHAMMAD ILMAN BIN MOHD KHAIRI** *Date:* **21/6/2024**

Doctor FlowChart

After the user chooses (2) as an option, a user prompt will pop up for inserting their card number and doctor ID. if the doctor exists, then it will proceed to display that specific doctor menu otherwise the system will repeat the previous step. User will be asked to choose either (1),(2),(3) and (4). (1) is for doctor info, (2) is for medical application business, (3) is for updating doctor information and 4 is for exit. if the user chooses (1), it will display the doctor credential and back to the menu of Doctor. If the user chooses (2,) the system puts jemaah assigned to that doctor in the vector of jemaahWithDoctor. If the same, then the system will display jemaah that needs to be updated. Users have been asked to choose which jemaah needs to be updated. After that, the Medical_Application method will be invoked. Then, the Jemaah medical application menu will be displayed. Here, there are 5 choices that users can choose. (1) is for checking medical application for that specific jemaah, (2) is for medical application approval, (3) is for stating the disease, (4) is for stating the medicine should be taken and (5) is back to the doctor display menu. When the user chooses (1), the system will display the jemaah credentials. For choosing number (2), users will be asked to choose (1) for approval and (2) for failure. For option (3), users will be asked to insert the disease that jemaah has. Then, the system will ask the user to add on more disease or not. If yes, then the insertion process will be done again. This concept is well same with insertion of medicine process. So, if the user chooses number (5), then it will exit the doctor display menu. That is what happens when the user chooses number (2) in the doctor display menu. Next, when the user chooses number (3) in the doctor display menu, updateUserInfo method will be invoked. The user was asked to insert the updated doctor credentials. After that, the system will display all the updated doctor credentials. Then, it will go back to the log in menu. This is the same when user choose number (4) in doctor display menu where it goes to log in menu.

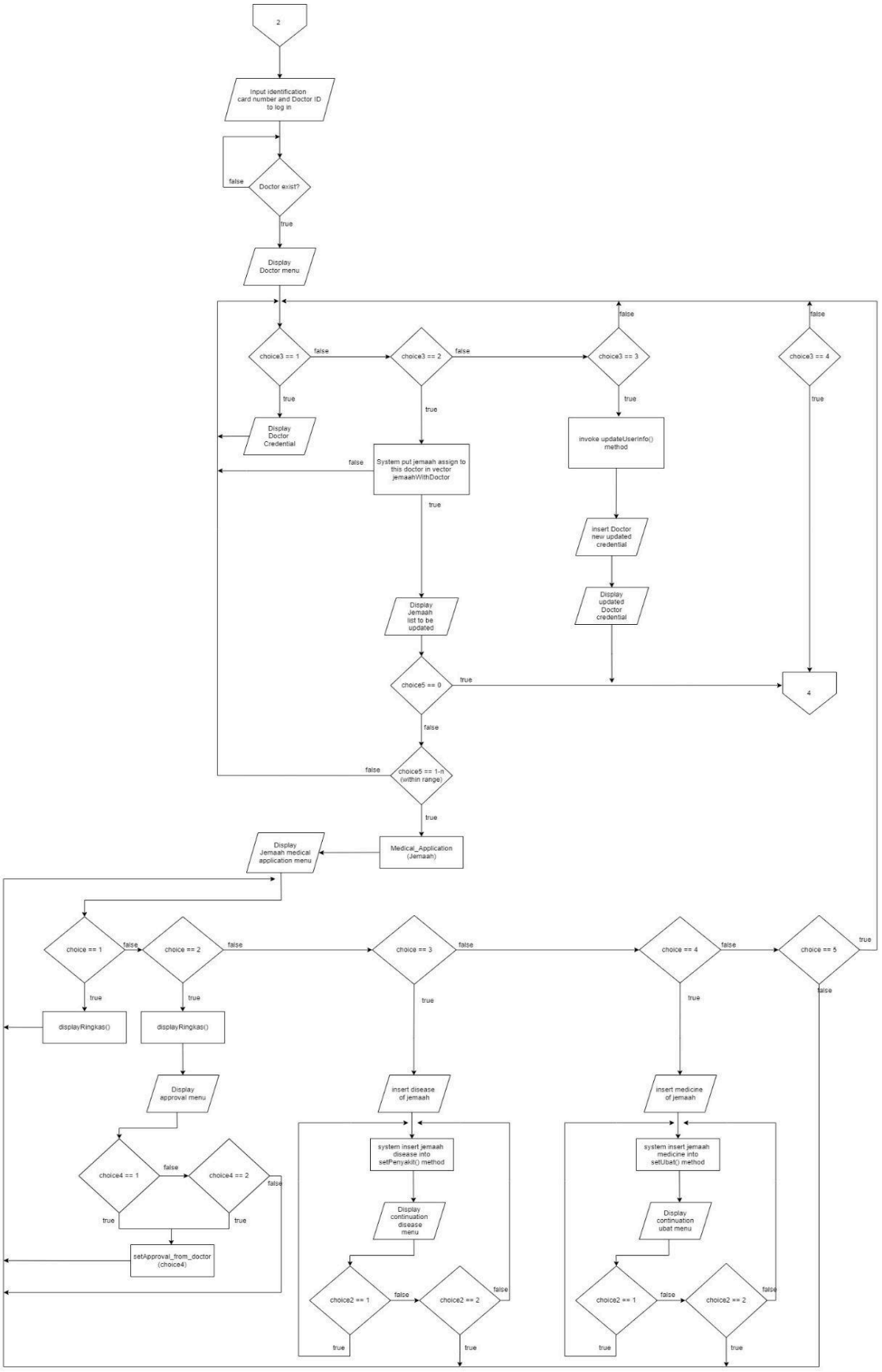

2
Input identification card number and Doctor ID to log in
Doctor exist?
false
true
Display Doctor menu
choice3 == 1
false
true
Display Doctor Credential
choice3 == 2
false
true
System put jemaah assign to this doctor in vector jemaahWithDoctor
false
true
Display Jemaah list to be updated
choice3 == 3
false
true
invoke updateUserInfo() method
insert Doctor new updated credential
Display updated Doctor credential
choice3 == 4
false
true
4
choice5 == 0
true
false
choice5 == 1-n (within range)
false
true
Medical_Application (Jemaah)
Display Jemaah medical application menu
choice == 1
false
true
displayRingkas()
choice == 2
false
true
displayRingkas()
Display approval menu
choice4 == 1
false
true
choice4 == 2
false
true
setApproval_from_doctor (choice4)
choice == 3
false
true
insert disease of jemaah
system insert jemaah disease into setPenyakit() method
Display continuation disease menu
choice2 == 1
false
true
choice2 == 2
false
true
choice == 4
false
true
insert medicine of jemaah
system insert jemaah medicine into setUbat() method
Display continuation ubat menu
choice2 == 1
false
true
choice2 == 2
false
true
choice == 5
true
false

Figure 5

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **21/6/2024**

Embassy Flow Chart

After User choose choice = 3. It shown that User is Embassy. to verify it, Embassy need to login by entering Identification Card and Embassy Number. If the Credentials is false it will show output and ask the Embassy login again. Else, It will show Embassy Menu and choice for Embassy to choose. If Embassy choose choice = 1 it will Display Embassy Info and redirect to Embassy Menu. Else If choice = 2, The system will check all Jemaah that with the Embassy, It will display all the Jemaah with the Embassy and ask the user to input choice5. If the choice5 = 0, It will redirect to Embassy Menu. Else if choice5 = (range of Jemaah With the Embassy size) it will display Visa Application Menu and ask Embassy to input Choice. Choice = 1 will Display Jemaah Credentials and redirect to Visa Application Menu. Meanwhile Choice = 2 will display Jemaah Credentials and ask Embassy to enter Choice2 input. Choice2 = 1 is to set Approval = “Approve”, Choice2 = 2 is to set Approval = “Failed” , Else is to set Approval = “Pending” and all if casses redirect to Visa Application Menu. Other Choice will redirect to Embassy Menu. Then, if choice3 = 3 it will update Embassy credentials. It will ask Embassy Input of Name, Contact, Identification Card, Email, Location, Position, Embassy Number. After that it will show the new embassy credentials and redirect to main menu. Other choice3 will redirect to main menu.

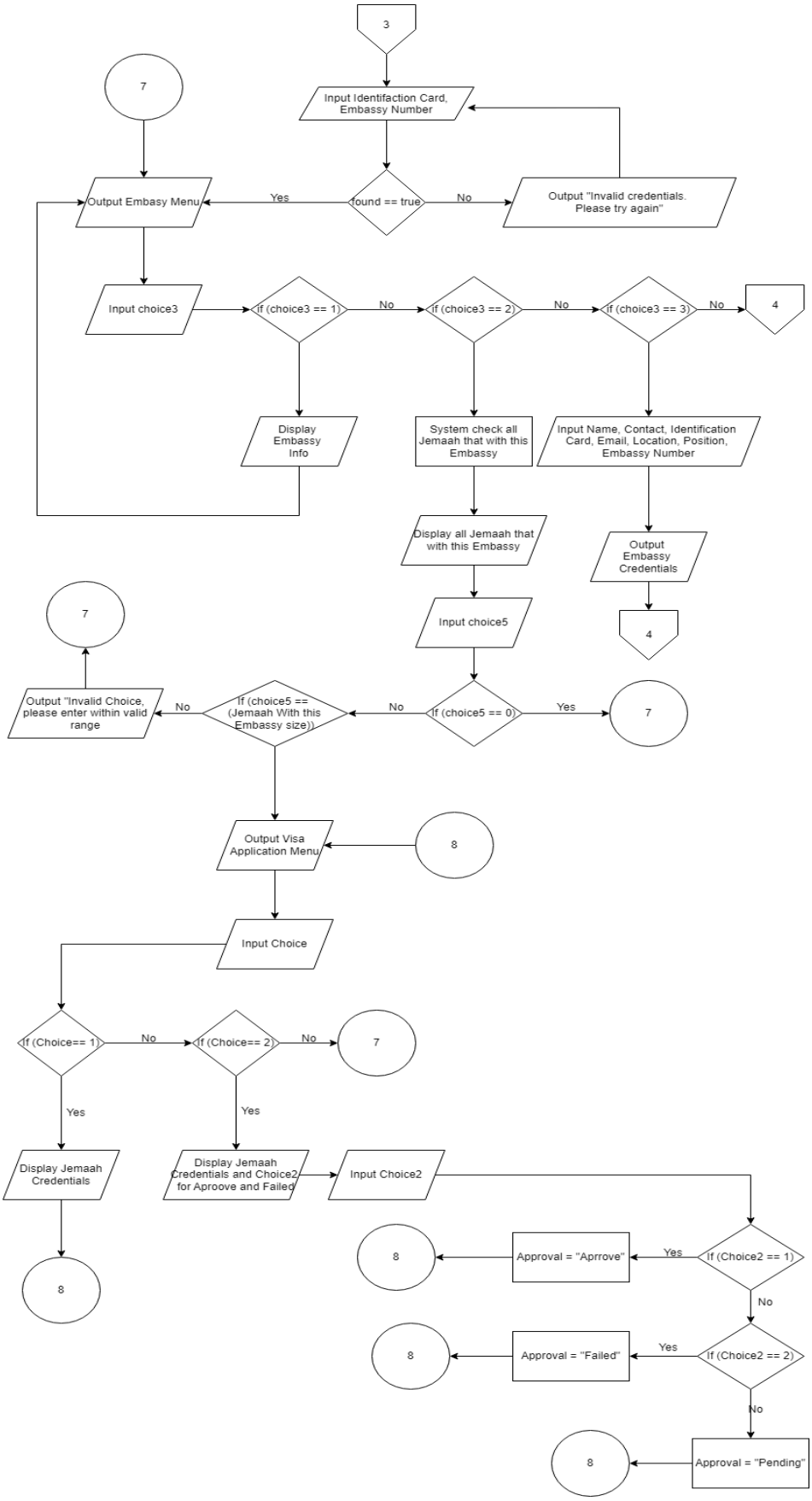

3
7
Input Identifaction Card, Embassy Number
Output Embasy Menu
Yes
found == true
No
Output "Invalid credentials. Please try again"
Input choice3
if (choice3 == 1)
No
if (choice3 == 2)
No
if (choice3 == 3)
No
4
Display Embassy Info
System check all Jemaah that with this Embassy
Input Name, Contact, Identification Card, Email, Location, Position, Embassy Number
Display all Jemaah that with this Embassy
Output Embassy Credentials
4
Input choice5
7
Output "Invalid Choice, please enter within valid range
No
If (choice5 == (Jemaah With this Embassy size))
No
If (choice5 == 0)
Yes
7
Output Visa Application Menu
8
Input Choice
If (Choice== 1)
No
If (Choice== 2)
No
7
Yes
Yes
Display Jemaah Credentials
Display Jemaah Credentials and Choice2 for Aproove and Failed
Input Choice2
8
8
Approval = "Aprrove"
Yes
If (Choice2 == 1)
No
8
Approval = "Failed"
Yes
If (Choice2 == 2)
No
8
Approval = "Pending"

Figure 6

The work in this page has been done by: **ALIF AIMAN BIN MANSOR** *Date:* **20/6/2024**

OO Concepts:

1. Data hiding and Encapsulation

```
public class Kursus {
    private String kursusID;
    private String time;
    private String date;
    private String kursusPlace;
    private String nameUstaz;

    public Kursus(String kursusID, String time, String date, String kursusPlace, String nameUstaz) {
        this.kursusID = kursusID;
        this.time = time;
        this.date = date;
        this.kursusPlace = kursusPlace;
        this.nameUstaz = nameUstaz;
    }

    public void Display_Kursus_info() {
        System.out.printf(format:"%-37s: %22s%n", ...args:"Kursus ID", kursusID);
        System.out.printf(format:"%-37s: %22s%n", ...args:"Time", time);
        System.out.printf(format:"%-37s: %22s%n", ...args:"Date", date);
        System.out.printf(format:"%-37s: %22s%n", ...args:"Kursus Location", kursusPlace);
        System.out.printf(format:"%-37s: %22s%n", ...args:"Ustaz", nameUstaz);
    }
}
```

Figure 7

Figure 7 is one of the classes that are being implemented for the Hajj and Umrah Booking System. In this context, to show the concept of data hiding and encapsulation is where all attributes are being initialized as private in order to prevent outside objects from corrupting the attributes,that is called data hiding. Meanwhile, encapsulation is shown where attributes and methods are combined together in one class that proves it can maximize the reusability and minimize debugging time.

2. Aggregation and Composition

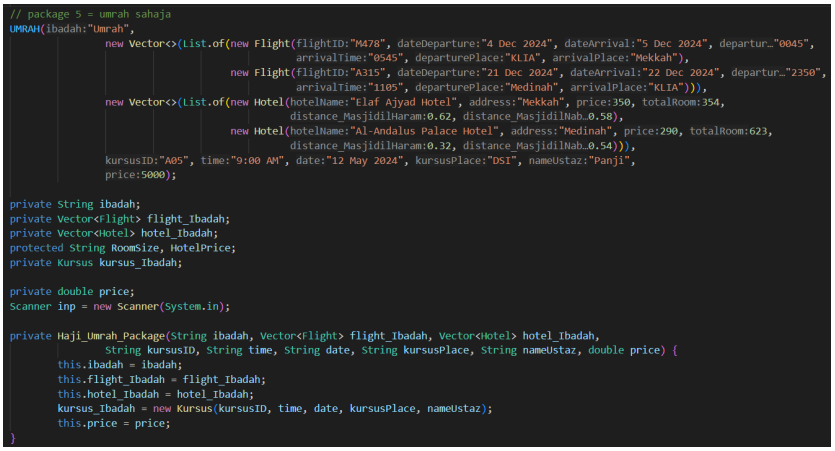
a) Aggregation between Pelancongan_package enum class and Flight class

```
PACKAGE_5(new Vector<>(List.of(e1:"Kuwait City", e2:"Manama")), tour_Guide:"Hafiz Haziq", price_Pelancongan:4000.00,
        new Vector<>(List.of(new Flight(flightID:"Q32", dateDeparture:"19/7/2024", dateArrival:"19/7/2024", departur…"1300", "1700",
                departurePlace:"King Abdul Aziz Airport", arrivalPlace:"Kuwait International Airport"),
                new Flight(flightID:"D78", dateDeparture:"26/7/2024", dateArrival:"26/7/2024", departur…"0900", "1700",
                departurePlace:"Kuwait International Airport", arrivalPlace:"KLIA"))),
        new Vector<>(List.of(new Hotel(hotelName:"Kuwait Palace", address:"Kuwait Avenue", totalRoom:350))));

private Vector<String> places;
private String tour_Guide;
private double price_Pelancongan;
private Vector<Flight> flight_Pelancongan;
private Vector<Hotel> hotel_Pelancongan;
Scanner inp = new Scanner(System.in);

private Pelancongan_Package(Vector<String> places, String tour_Guide, double price_Pelancongan,
            Vector<Flight> flight_Pelancongan, Vector<Hotel> hotel_Pelancongan) {
        this.places = places;
        this.tour_Guide = tour_Guide;
        this.price_Pelancongan = price_Pelancongan;
        this.flight_Pelancongan = flight_Pelancongan;
        this.hotel_Pelancongan = hotel_Pelancongan;
}
```

Figure 8

These two classes show the relationship between them which is aggregation. Aggregation depicts the “has-a” relation between the classes. In this case, Pelancongan_package has a Flight and Pelancongan_package enum class contains reference to Flight class. This shows that, Pelancongan_package enum class can exist independently without the Flight class.

b) Composition between Hajj_Umrah_Package enum class and Kursus class

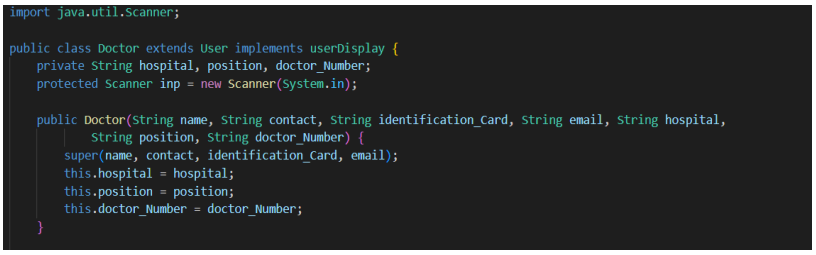


```
// package 5 = umrah sahaja
UMRAH(ibadah:"Umrah",
                new Vector<>(List.of(new Flight(flightID:"M478", dateDeparture:"4 Dec 2024", dateArrival:"5 Dec 2024", departur…"0045",
                                                arrivalTime:"0545", departurePlace:"KLIA", arrivalPlace:"Mekkah"),
                                        new Flight(flightID:"A315", dateDeparture:"21 Dec 2024", dateArrival:"22 Dec 2024", departur…"2350",
                                                arrivalTime:"1105", departurePlace:"Medinah", arrivalPlace:"KLIA"))),
                new Vector<>(List.of(new Hotel(hotelName:"Elaf Ajyad Hotel", address:"Mekkah", price:350, totalRoom:354,
                                        distance_MasjidilHaram:0.62, distance_MasjidilNab…0.58),
                                new Hotel(hotelName:"Al-Andalus Palace Hotel", address:"Medinah", price:290, totalRoom:623,
                                        distance_MasjidilHaram:0.32, distance_MasjidilNab…0.54))),
                kursusID:"A05", time:"9:00 AM", date:"12 May 2024", kursusPlace:"DSI", nameUstaz:"Panji",
                price:5000);

private String ibadah;
private Vector<Flight> flight_Ibadah;
private Vector<Hotel> hotel_Ibadah;
protected String RoomSize, HotelPrice;
private Kursus kursus_Ibadah;

private double price;
Scanner inp = new Scanner(System.in);

private Haji_Umrah_Package(String ibadah, Vector<Flight> flight_Ibadah, Vector<Hotel> hotel_Ibadah,
                String kursusID, String time, String date, String kursusPlace, String nameUstaz, double price) {
        this.ibadah = ibadah;
        this.flight_Ibadah = flight_Ibadah;
        this.hotel_Ibadah = hotel_Ibadah;
        kursus_Ibadah = new Kursus(kursusID, time, date, kursusPlace, nameUstaz);
        this.price = price;
}
```

Figure 9

These two classes show the relationship which is composition. Composition depicts the "contains a" relation between these two classes. In this case, Hajj_Umrah_Package contains a Kursus and Hajj_Umrah_Package enum class has strong ownership or exclusively owned Kursus class. Kursus class may be destroyed if Hajj_Umrah_Package enum class is destroyed.

3. Inheritance between class User and class Doctor



```
import java.util.Scanner;

public class Doctor extends User implements userDisplay {
    private String hospital, position, doctor_Number;
    protected Scanner inp = new Scanner(System.in);

    public Doctor(String name, String contact, String identification_Card, String email, String hospital,
            String position, String doctor_Number) {
        super(name, contact, identification_Card, email);
        this.hospital = hospital;
        this.position = position;
        this.doctor_Number = doctor_Number;
    }
```

Figure 10

The picture above shows the concept of inheritance of class User on class Doctor. Inheritance is the process in which a new class (derived class) is created from another class (base class) and it allows reusability on code. In this case, class Doctor is a derived class while class User is a base class. Doctor could access instance variables and methods from class User while it can have

its additional instance variables and methods. When Doctor instances is created, its constructor will invoke its base class constructor which class User.

4. **Polymorphism, Abstract Class and Interface Class between class User and its derived class (Jemaah, Doctor, Embassy)**

```java
import java.util.*;

public abstract class User { // using abstarct
    private String name, contact, identification_Card, email;
    protected Scanner inp = new Scanner(System.in);

    public User(String name, String contact, String identification_Card, String email) {
        this.name = name;
        this.contact = contact;
        this.identification_Card = identification_Card;
        this.email = email;
    }

    public String getName() {  return name;  }

    public void setName(String name) {  this.name = name;  }

    public String getContact() { return contact;    }

    public void setContact(String contact) {  this.contact = contact;  }

    public String getIdentification_Card() {  return identification_Card;  }

    public void setIdentification_Card(String identification_Card) {  this.identification_Card = identification_Card;  }

    public String getEmail() {  return email;  }

    public void setEmail(String email) {  this.email = email;  }

    public abstract void updateUserInfo();

}
```

Figure 11

```java
public interface userDisplay {
    public void display();
    public void displayRingkas(); // tanpa personal info
}
```

Figure 12

```
public void updateUserInfo() {

    System.out.println(x:"\n=======================================");
    System.out.printf(format:"%29s\n", ...args:"UPDATE CREDENTIALS");
    System.out.println(x:"=======================================");
    System.out.print(s:"Please enter your credentials: \nName: ");
    setName(inp.nextLine());
    System.out.print(s:"Contact :");
    setContact(inp.nextLine());
    System.out.print(s:"Identification Card : ");
    setIdentification_Card(inp.nextLine());
    System.out.print(s:"Email: ");
    setEmail(inp.nextLine());

    System.out.print(s:"Location: ");
    location = inp.nextLine();
    System.out.print(s:"Position: ");
    position = inp.nextLine();
    System.out.print(s:"Embassy Number: ");
    embassy_number = inp.nextLine();

    // new credentialsdsadasdasdas
    display();
}
```

Figure 13

```
// Interface
public void display() {
    System.out.println(x:"\n=======================================");
    System.out.printf(format:"%29s\n", ...args:"EMBASSY CREDENTIALS");
    System.out.println(x:"=======================================");
    System.out.println("Name : " + getName());
    System.out.println("Contact :" + getContact());
    System.out.println("Identification Card : " + getIdentification_Card());
    System.out.println("Email: " + getEmail());
    System.out.println("Location: " + location);
    System.out.println("Position: " + position);
    System.out.println("Embassy Number: " + embassy_number);
    System.out.println();
}

public void displayRingkas() {
    System.out.println(x:"\n=======================================");
    System.out.printf(format:"%29s\n", ...args:"EMBASSY CREDENTIALS");
    System.out.println(x:"=======================================");
    System.out.println("Name : " + getName());
    System.out.println("Contact :" + getContact());
    System.out.println("Email: " + getEmail());
    System.out.println("Location: " + location);
    System.out.println("Position: " + position);
    System.out.println();
}
```

Figure 14

Figure 11 and 12 shows the implementation of concept abstract class and interface class between class User and its derived class which are Jemaah, Doctor and Embassy class. Abstract and interface class is another way of implementing polymorphism concepts. Abstract class cannot be instantiated and it needs to be implemented in its derived classes. Inside an abstract class,

there are abstract methods with no implementation on it, so its derived class needs to override or implement all the abstract methods otherwise its derived class will become abstract. Meanwhile, interface class contain constant variables and abstract methods. The classes that used interface concepts need to include the "implements" keyword to show that the interface class is implemented at the classes. Similar to abstract class, derived class needs to override the variables and methods from interface class.

Figure 13 (Embassy) shows that the Embassy class is implementing the updateUserInfo() abstract method from the User class. In figure 14 (Embassy) shows that the Embassy class is implementing display() and displayRingkas() abstract methods from the User class.

5. Exception Handling

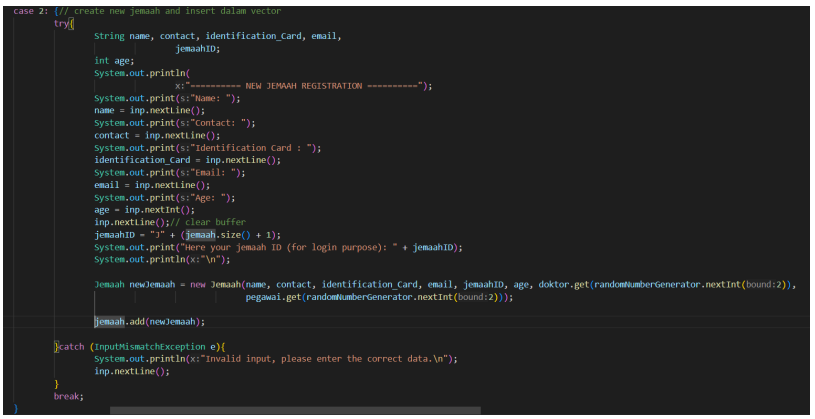


Figure 15

The picture above is one of the exception handling concepts that is being implemented. It will send the error message to the user when the user enters incorrect data. For example, users enter string values when the expected input is number.

The work in this page has been done by: **MUHAMMAD ILMAN BIN MOHD KHAIRI** *Date:* **20/6/2024**

SECTION B: CLASS DIAGRAMS

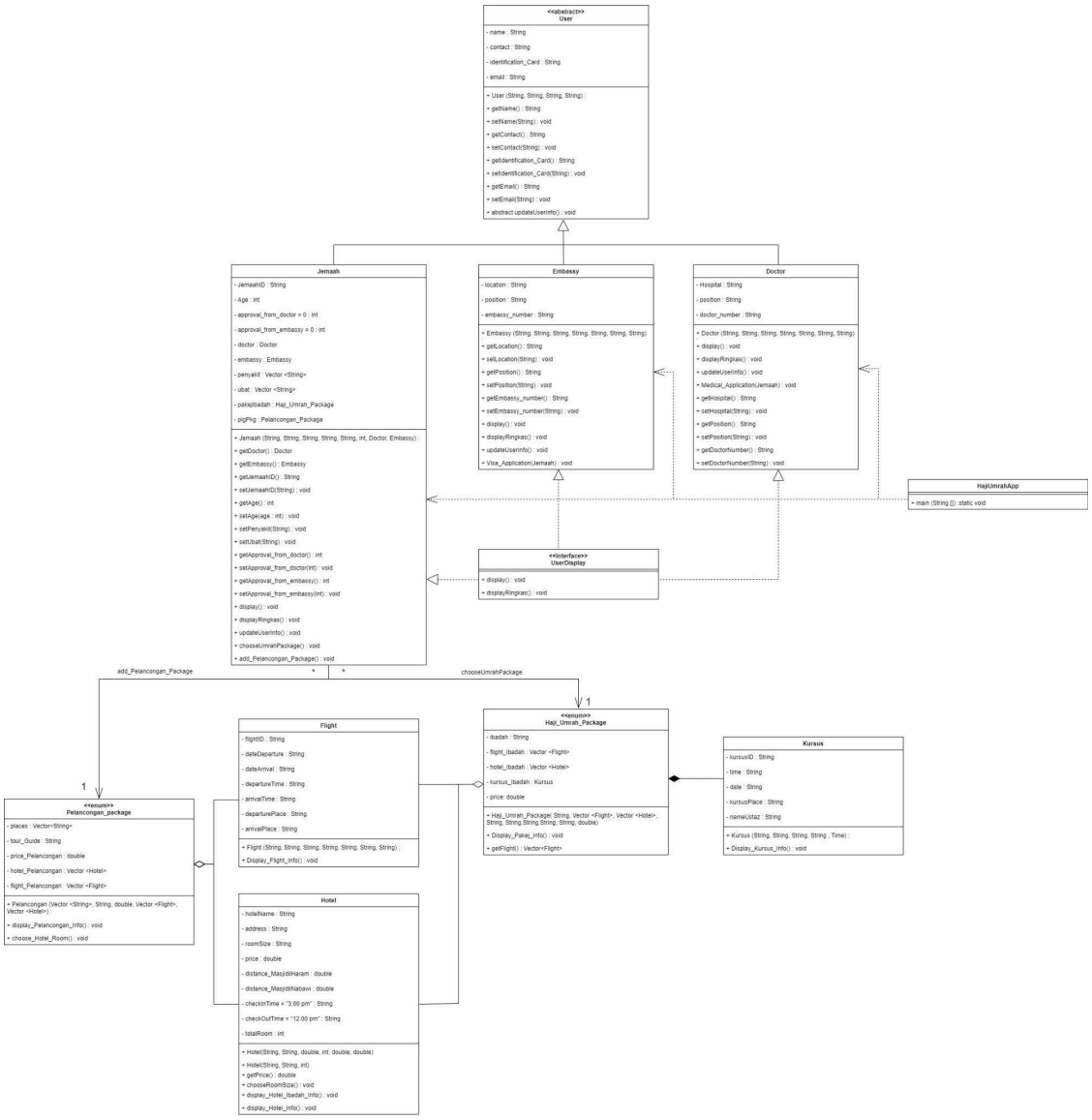


Figure 16

Class User

Attributes	Description
name	The name of the user
contact	The phone number of the user
identification_Card	The identification card of user
email	The email of user
Methods	**Description**
getName()	Return name
setName()	To set the user name
getContact()	Return contact number of the user
setContact()	To set the contact number of the user
getIdentification_Card	Return user's identification card
setIdentification_Card	To set user's identification card
getEmail()	Return email of the user
setEmail()	To set email of the user
(abstract) updateUserInfo()	Abstract method to update user information

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

Class Jemaah

Attributes	Description
JemaahID	The unique ID for the jemaah
Age	The age of Jemaah
approval_from_doctor	The approval from doctor
approval_from_embassy	The approval from embassy
doctor	object of doctor assigned to this jemaah
embassy	object of embassy assigned to this jemaah
penyakit	vector to store jemaah's disease
ubat	vector to store jemaah's prescription
pakejIbadah	object of Hajj_Umrah_Package
plgkg	object of Pelancongan_package
Methods	**Description**
getDoctor()	Return object Doctor
getEmbassy()	Return object Embassy
getJemaahID()	Return the JemaahID
setJemaahID()	Set the JemaahID
getAge()	Return age
setAge()	Set the age

setPenyakit()	Set a disease and insert it into the penyakit vector
setUbat()	Set a prescription and insert it into the ubat vector
getApproval_from_doctor()	Return approval_from_doctor
setApproval_from_doctor()	Set the approval_from_doctor
getApproval_from_embassy()	Return approval_from_embassy
setApproval_from_embassy()	Set the approval_from_embassy
display()	Display the Jemaah credentials and their respective disease, prescription, Hajj and Umrah package and tour package, doctor, embassy info
displayRingkas()	Display the Jemaah credentials only
updateUserInfo()	Update the Jemaah credentials
chooseUmrahPackage()	add object from Hajj_Umrah_Package
add_Pelancongan_Package()	add object from Pelancongan_package

Class Doctor

Attributes	**Description**
hospital	The name of the hospital
position	The position of the doctor in the hospital
doctor_Number	The unique ID for the doctor
Methods	**Description**
getHospital()	Return name of the hospital
setHospital()	To set the name of the hospital
getPosition()	Return the position of the doctor
setPosition()	To set the position of the doctor
getDoctor_Number	Return the doctor unique number
setDoctor_Number	To set the doctor unique number
display()	display the credential of doctor like name,contact,identification card,email, which hospital,position of doctor and its number
displayRingkas()	display simple credential of the doctor like name,contact of the doctor,email,hospital and position of doctor
updateUserInfo()	update doctor credential by prompting the new information
Medical_Application(Jemaah)	handles all congregational approval procedures, disease update and medicitaion administration

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **22/06/2024**

Class Embassy

Attributes	Description
location	To store Embassy location
position	To store Embassy position
embassy_number	To store Embassy number
Operations	**Description**
getLocation()	Return location of the embassy
setLocatio()	To set the location of the embassy
getPosition()	Return Position of the embassy
setPosition()	To set the Position of the embassy
getEmbassy_number()	Return Embassy number
setEmbassy_number()	To set the Embassy number
display()	To display Embassy Credentials
displayRingkas(()	To display Embassy Credentials without Identification Card number and Embassy Number
updateUserInfo()	To Update Embassy credential by prompting the new information
Visa_Application(Jemaah)	To set the Approval for Jemaah Visa

The work in this page has been done by: **MUHAMMAD ILMAN BIN MOHD KHAIRI** *Date:* **20/6/2024**

Enum class pelancongan_package

Attributes	Description
places	It is a vector that store various places
tour_guide	The name of the tour guide
price_Pelancongan	The price of the total tour
flight_Pelancongan	It is a vector that store various of flight
hotel_Pelancongan	It is a vector that store various of hotel
Methods	**Description**
display_Pelancongan_Info()	Display all the information of the pelancongan tour info, flight info and hotel info
choose_Hotel_Room()	prompt for user to choose what type of room

The work in this page has been done by: **ALIF AIMAN BIN MANSOR** *Date:* **21/6/2024**

Class Kursus

Attributes	**Description**
kursusID	The id for each course provided
time	Starting time for the courses held
date	Date for the courses held
kursusPlace	Place where the courses held
nameUstaz	Name of instructor that conduct the courses
Operations	**Description**
Display_Kursus_Info()	Display all the details informations of the courses

The work in this page has been done by: **ALIF AIMAN BIN MANSOR** *Date:* **21/6/2024**

Enum class Hajj_Umrah_Package

Attributes	**Description**
ibadah	The name of ibadah
flight_Ibadah	The vector that store various flights for Hajj and Umrah
hotel_Ibadah	The vector that store various hotels for Hajj and Umrah
kursus_Ibadah	The vector that store various courses for Hajj and Umrah
price	The price for each package ibadah
Operations	**Description**
Display_Pakej_Info()	Display all the details informations of the package ibadah
getFlight()	Retrieve flight information

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **22/06/2024**

Class Flight

Attributes	**Description**
flightID	To store flight ID
dateDeparture	To store flight date departure
dateArrival	To store flight date arrival
departureTime	To store flight departure time
arrivalTime	To store flight arrival time
departurePlace	To store flight departure place
arrivalPlace	To store flight arrival place
Operations	**Description**
display_Flight_Info()	To display all the flight info based on attributes

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **22/06/2024**

Class Hotel

Attributes	**Description**
hotelName	To store hotel name
address	To store hotel address
roomSize	To store hotel room size
price	To store hotel price
distance_MasjidilHaram	To store distance between hotel to Masjidil Haram
distance_MasjidilNabawi	To store distance between hotel to Masjidil Nabawi
checkInTime	To store hotel check-in time
checkOutTime	To store hotel check-out time
totalRoom	To store total room of the hotel
Operations	**Description**
getPrice()	Return price of the hotel
chooseRoomSize()	For user to choose their room size
display_Hotel_Ibadah_Info()	To display Hotel while doing ibadah
display_Hotel_Info()	to display Hotel on vacation

The work in this page has been done by: **ALIF AIMAN BIN MANSOR** *Date:* **21/6/2024**

Interface class userDisplay

Attributes	Description
Operations	**Description**
display()	Display all the details informations of the user (jemaah, doctor, embassy)
displayRingkas()	Display informations of the user briefly (jemaah, doctor, embassy)

The work in this page has been done by: **MUHAMMAD FAIZ BIN ZAKARIA** *Date:* **21/6/2024**

SECTION C: SOURCE CODE AND USER MANUAL

Main Screen:

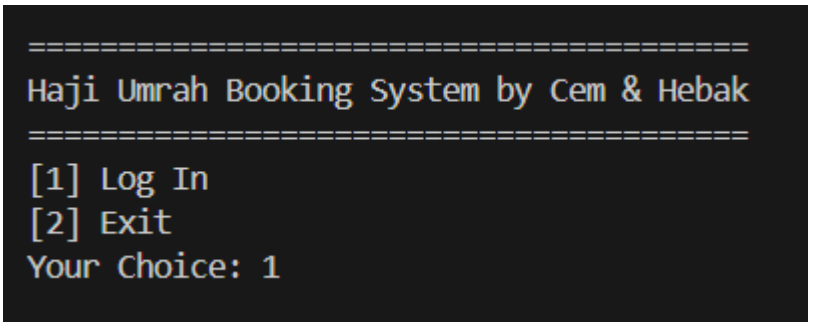


Figure 17

In the main screen, users will choose if they want to login or exit from the system

From Jemaah Point of View
Login page

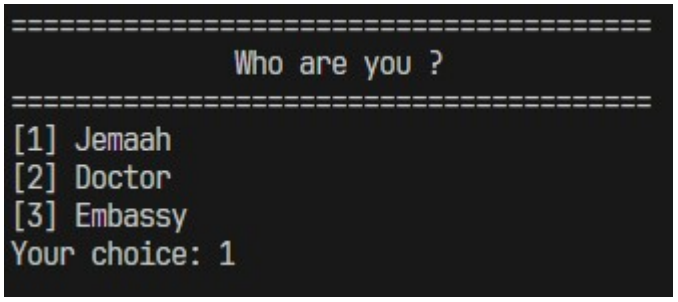


Figure 18

User need to enter 1 to indicate the user is a Jemaah

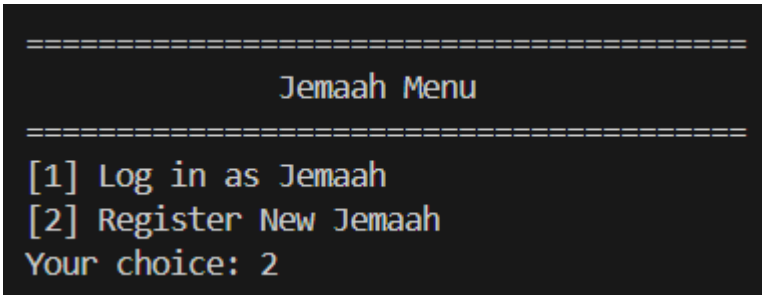


Figure 19

User need to choose 1 to Log in as Jemaah

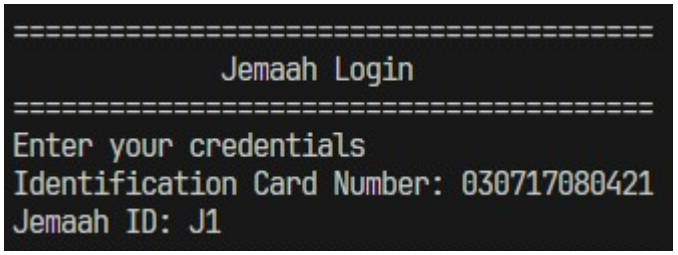


Figure 20

User will be able to enter their Identification card number and their Jemaah ID

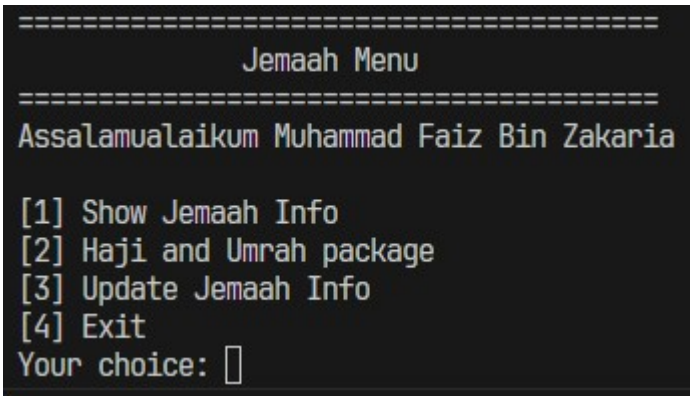


Figure 21

The screen will display the Jemaah Menu and the user will have 4 choices, 1 to show jemaah info, 2 to choose Hajj and Umrah package, 3 to update jemaah info and 4 to exit Show Jemaah Info

Show Jemaah Info

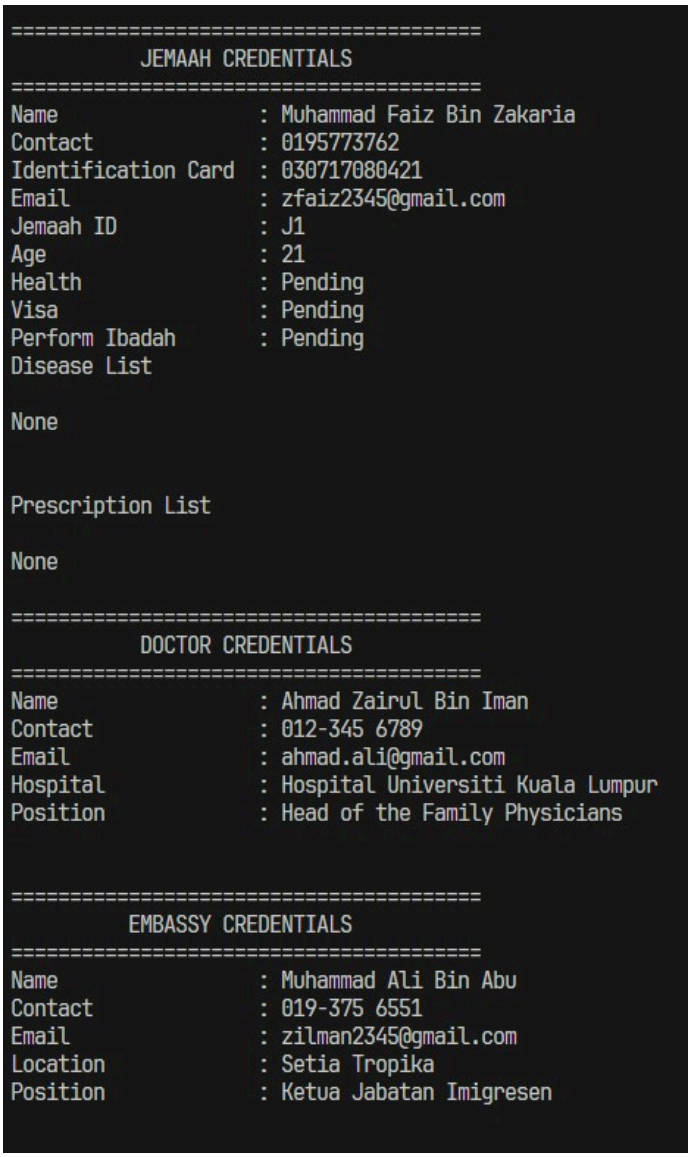


Figure 22

If the user choose 1, the system will display the jemaah info. if there are Hajj and Umrah package and tour package, it will display the info. But if there are none, it will not display it

Hajj and Umrah Package

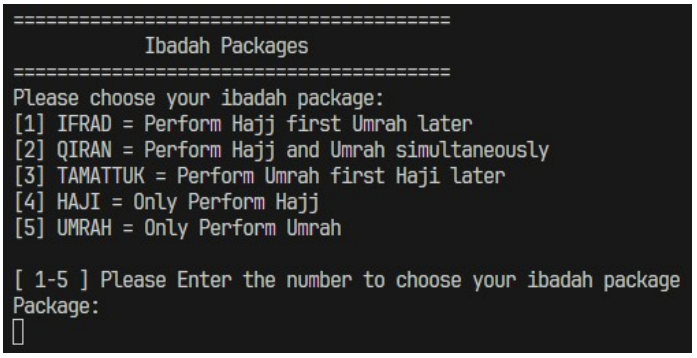


Figure 23

If the user choose 2, the system will display the Ibadah Package that are offered, the system will ask the user to choose between 1 until 5 to enter their preferred Ibadah packages.

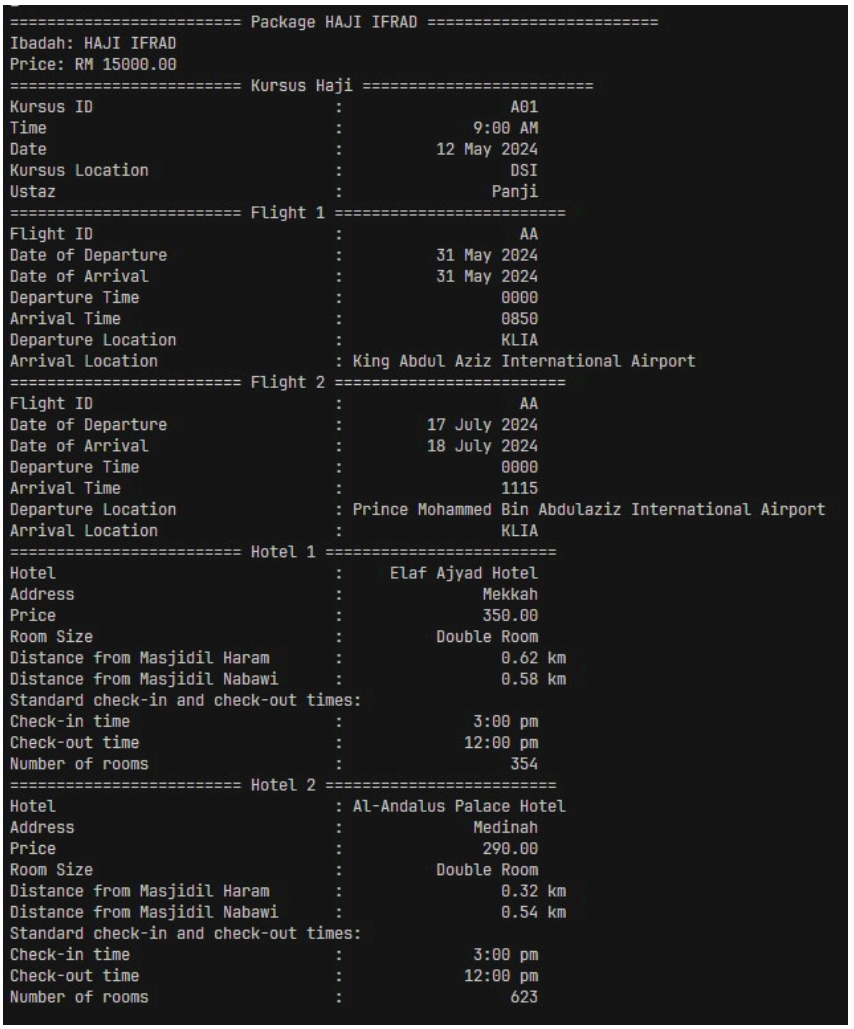


Figure 24

For example, the user choose 1. the system will display the package info and the system will assign this Ibadah Package to the jemaah.

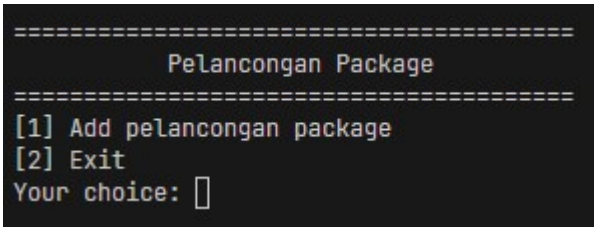


Figure 25

The system will ask the user as soon as the Ibadah Package are chosen if the user want to add Pelancongan Package.

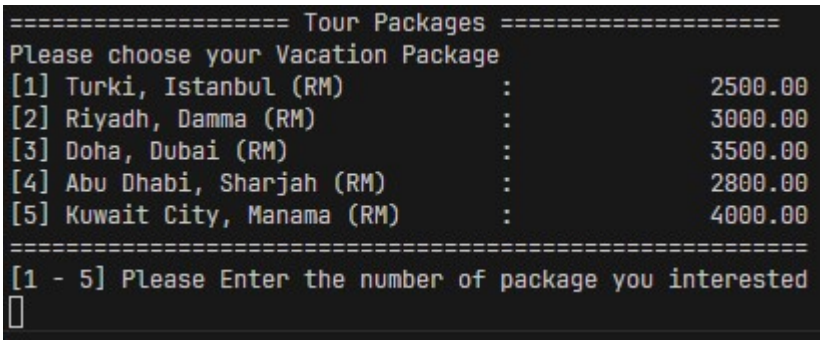


Figure 26

The system will display the tour packages that are offered and ask the user to enter which tour packages they desired.

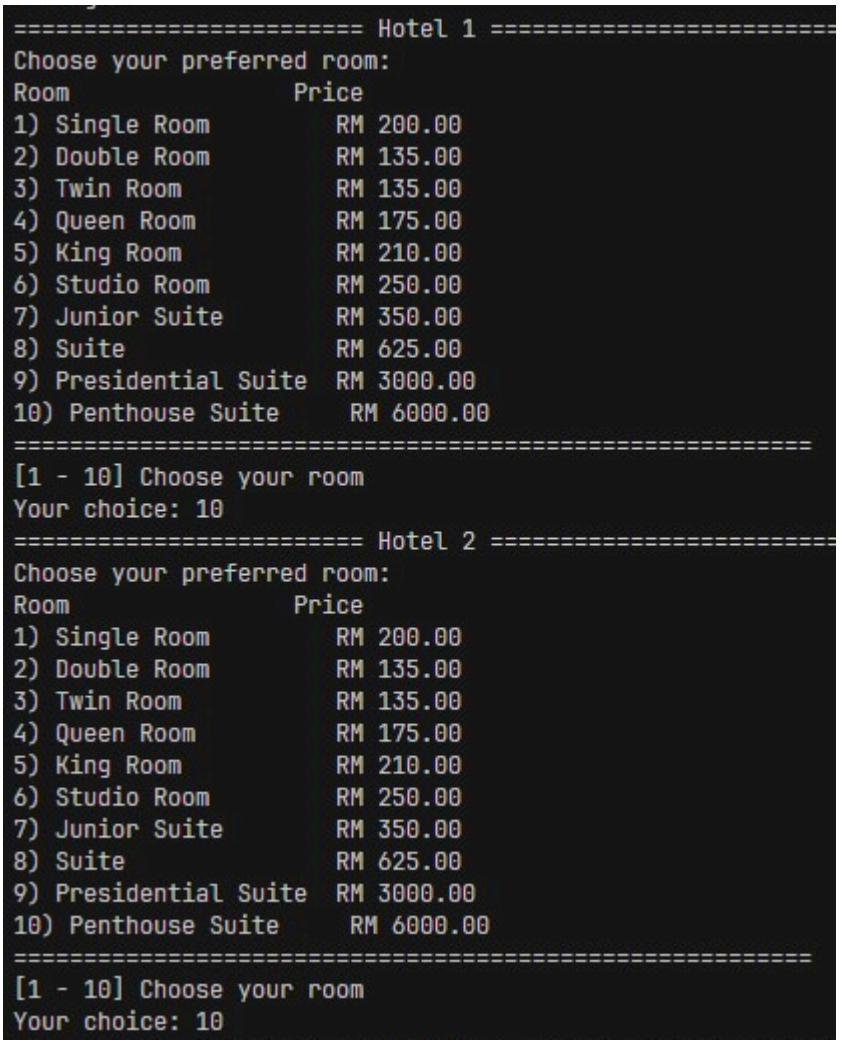


Figure 27

The system will display the hotel room size and ask the user to enter their room size for all of the tour's hotels.

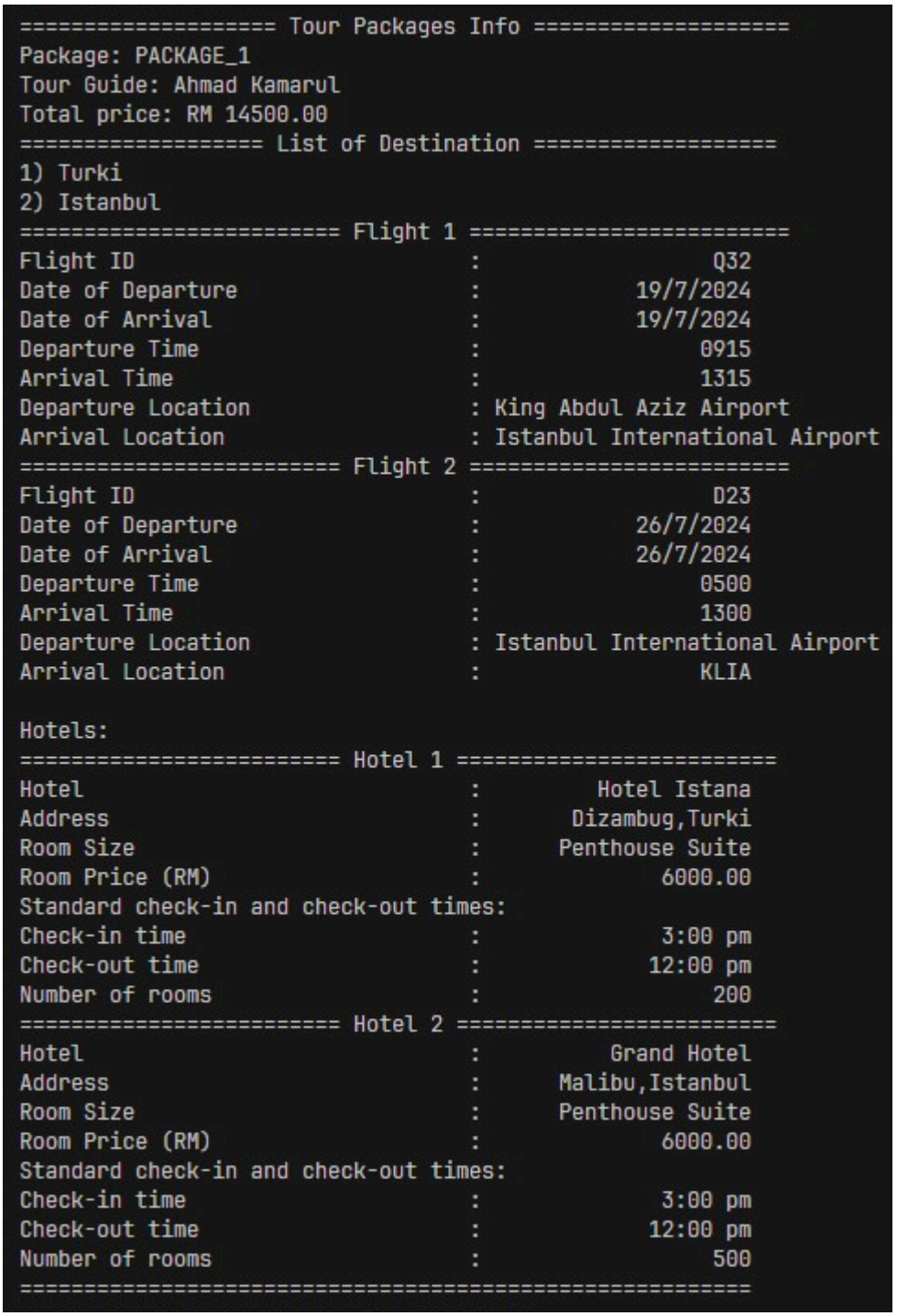


Figure 28

The system will display the tour packages info of the packages that the user choose Update Jemaah Info

Update Jemaah Info

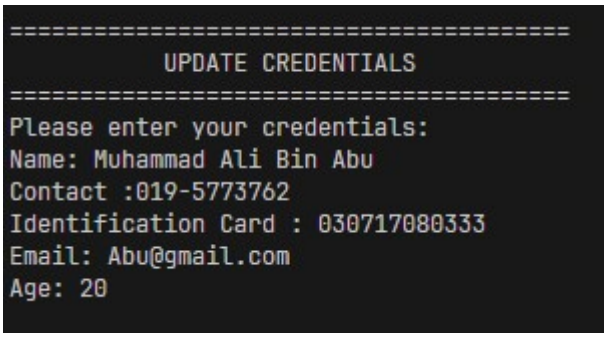


Figure 29

The user will be asked to update their name, contact, identification card, email and age

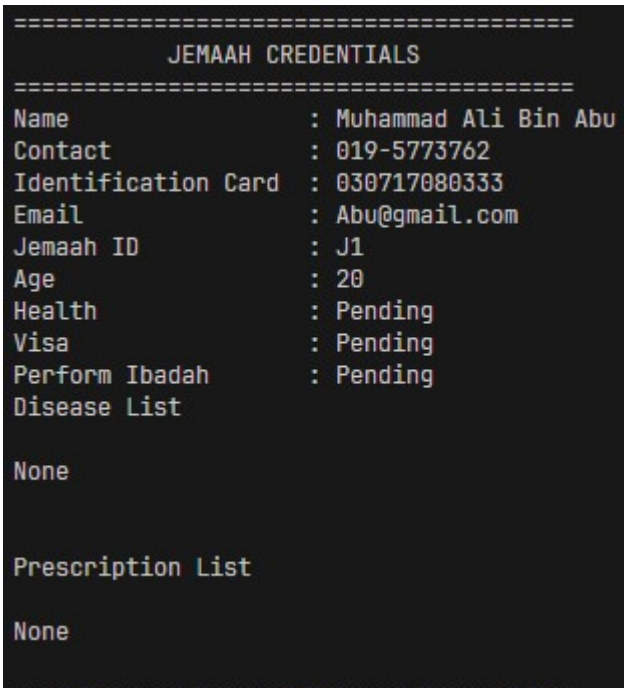


Figure 30

The system will display the new updated info.

The work in this page has been done by: **MUHAMMAD ILMAN BIN MOHD KHAIRI** *Date:* **22/6/2024**

Register new jemaah

```
========================================
              Jemaah Menu
========================================
[1] Log in as Jemaah
[2] Register New Jemaah
Your choice: 2
```

Figure 31

User will choose option 2 for register new Jemaah.

```
========== NEW JEMAAH REGISTRATION ==========
Name: Muhammad Abu Bin Salim
Contact: 0104643672
Identification Card : 303
Email: abu@gmail.com
Age: 45
Here your jemaah ID (for login purpose): J11
```

Figure 32

User will be able to insert their personal information base on what the system ask. by automatically, system will give the user new unique jemaah ID.

```
========================================
Haji Umrah Booking System by Cem & Hebak
========================================
[1] Log In
[2] Exit
Your Choice: 1
```

Figure 33

After new jemaah registration done, system will ask user to log in back.

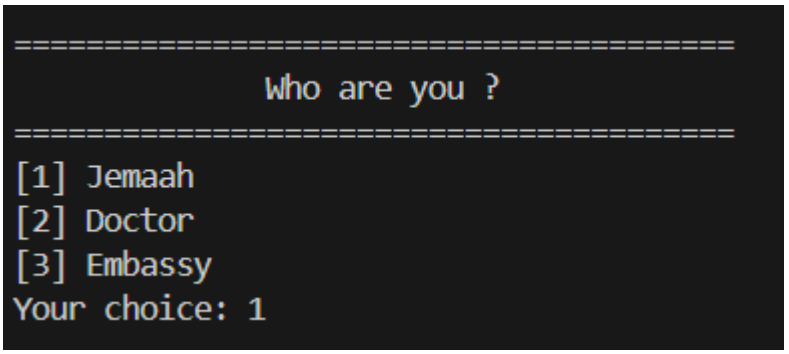


Figure 34

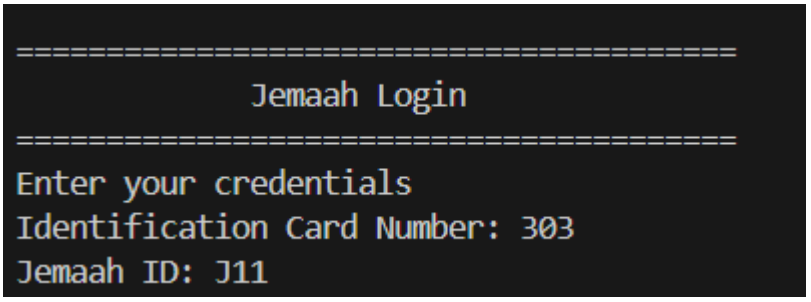


Figure 35

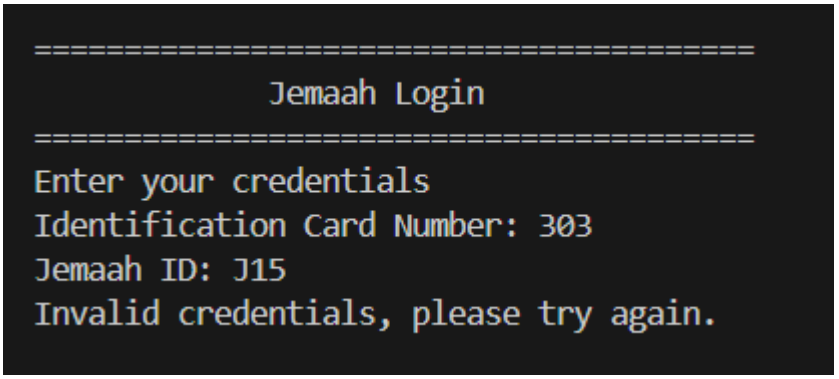


Figure 36

In figure 34 user will select option 1 to log in for jemaah part. Then, in figure 35 jemaah will insert their identification card number with their jemaah ID. if either one if it is false, then system will show error message like in figure 36.

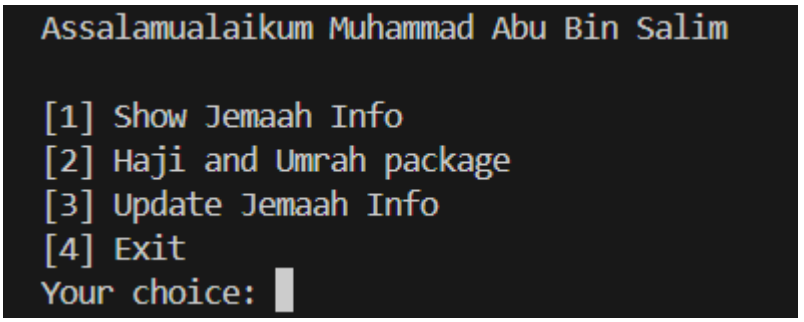


Figure 37

if the identification card number and jemaah ID is correct, the system will proceed to show the jemaah menu. So, Jemaah can choose which option they want and what will happen next is the same like the previous one.

The work in this page has been done by: **ALIF AIMAN BIN MANSOR** *Date:* **21/06/2024**

From Doctor Point of View

Selection role

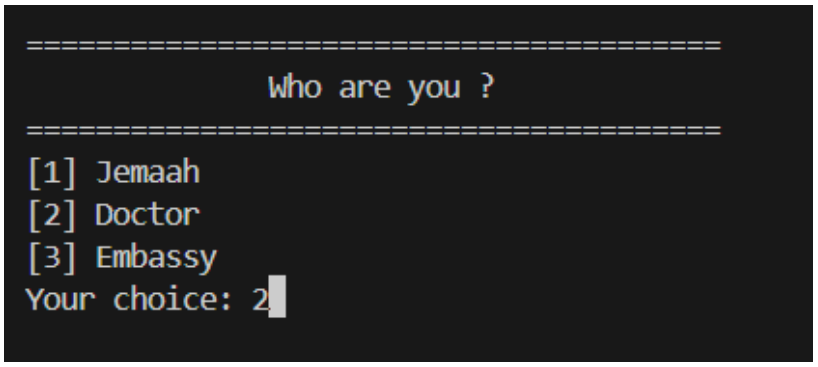


Figure 38

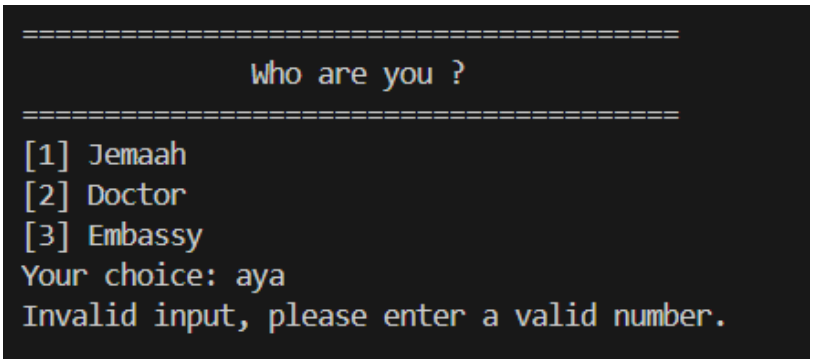


Figure 39

Doctor needs to choose their role in the selection role menu.. Doctor cannot enter string value, otherwise system will display error message to them like in Figure 39.

Login page

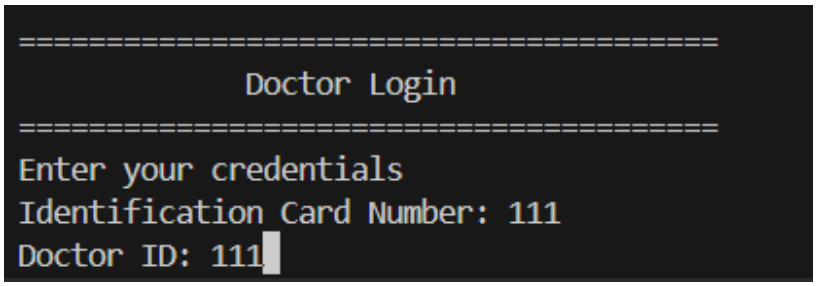


Figure 40

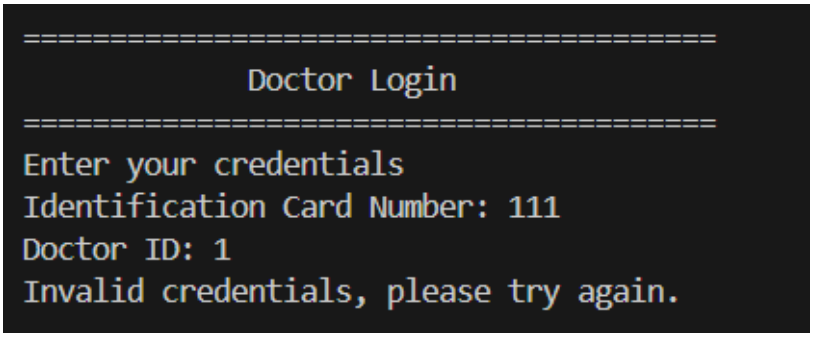


Figure 41

Doctors need to authenticate themselves by entering their number ID and doctor ID. If they enter incorrect authentication information, the system will display an error message like in Figure 41 and they need to start again from the beginning.

Doctor main menu

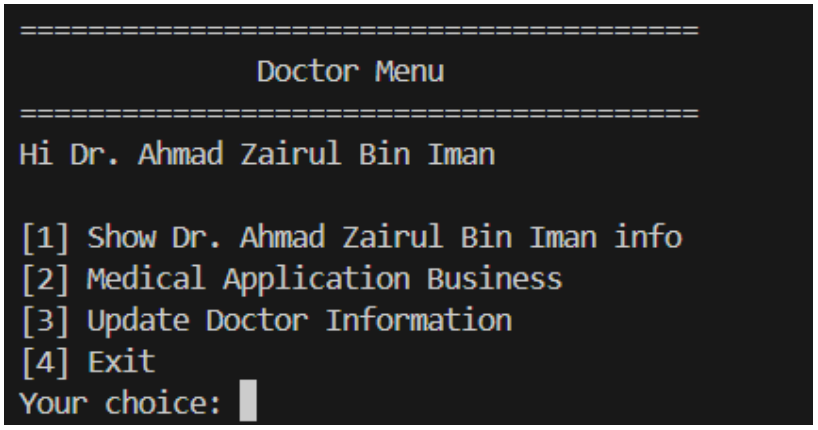


Figure 42

Now, the doctor has logged in and at the doctor main menu page. They have four choices provided which are view their information, view application medication from jemaah, update their information and exit to the main screen,

Show Doctor Info

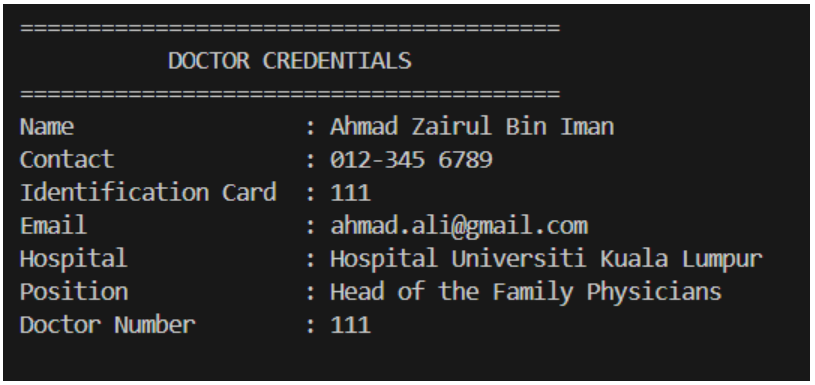


Figure 43

This page will show the doctor's detailed information that they had registered early.

Medical Application Business

```
==========================================
MEDICAL APPLICATION MUHAMMAD FAIZ BIN ZAKARIA
==========================================
[1] Check Medical Application
[2] Medical Application Approval
[3] State Disease
[4] Prescribe Medication
[5] Exit
Your Choice: 2

======================================
          JEMAAH CREDENTIALS
======================================
Name : Muhammad Faiz Bin Zakaria
Contact :0195773762
Identification Card : 030717080421
Email: zfaiz2345@gmail.com
Jemaah ID: J1
Age: 21
Health: Pending
Visa: Pending
Pending to perform ibadah

         Disease List

None
         Prescription List

None

[1] Approve
[2] Failed
Your Choice: 1
```

Figure 44

Doctors have five choices on this page, they will get a random number of jemaah Hajj and Umrah applications. Doctors need to choose whether to approve or fail the application in choice number 2.

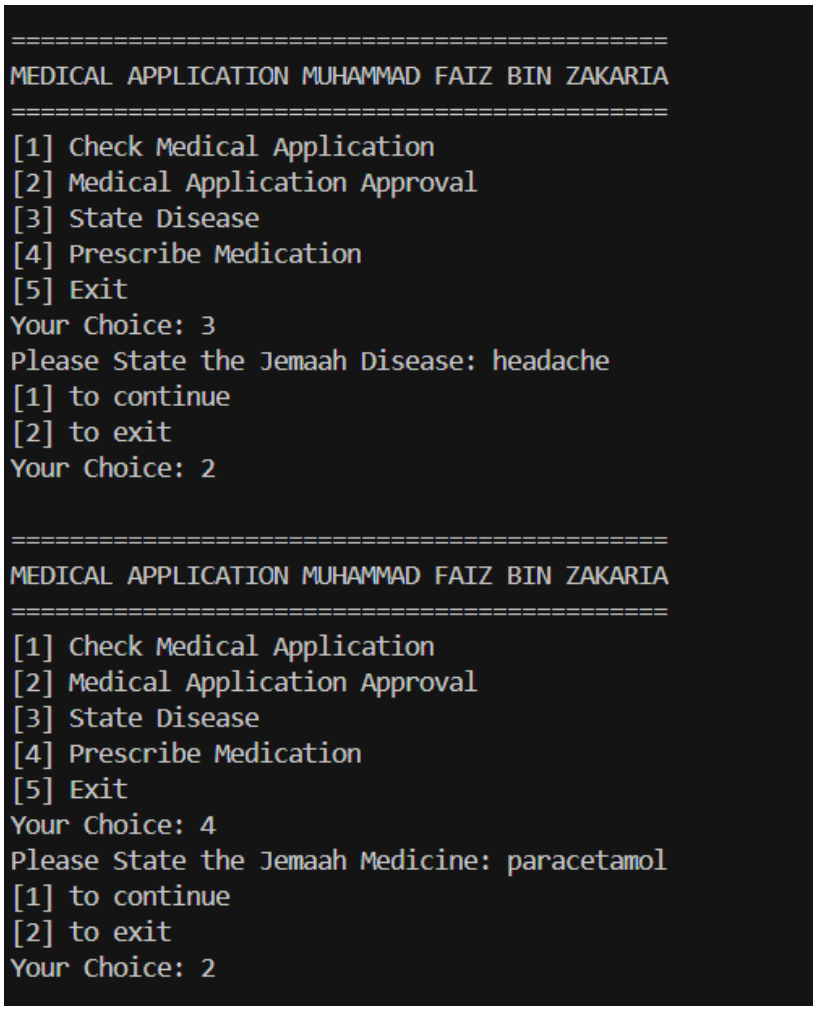


Figure 45

Doctor could list the diseases at choice 3 and prescribe medication at choice 4 for the jemaah as shown at figure 45. It will save the record in jemaah medical records.

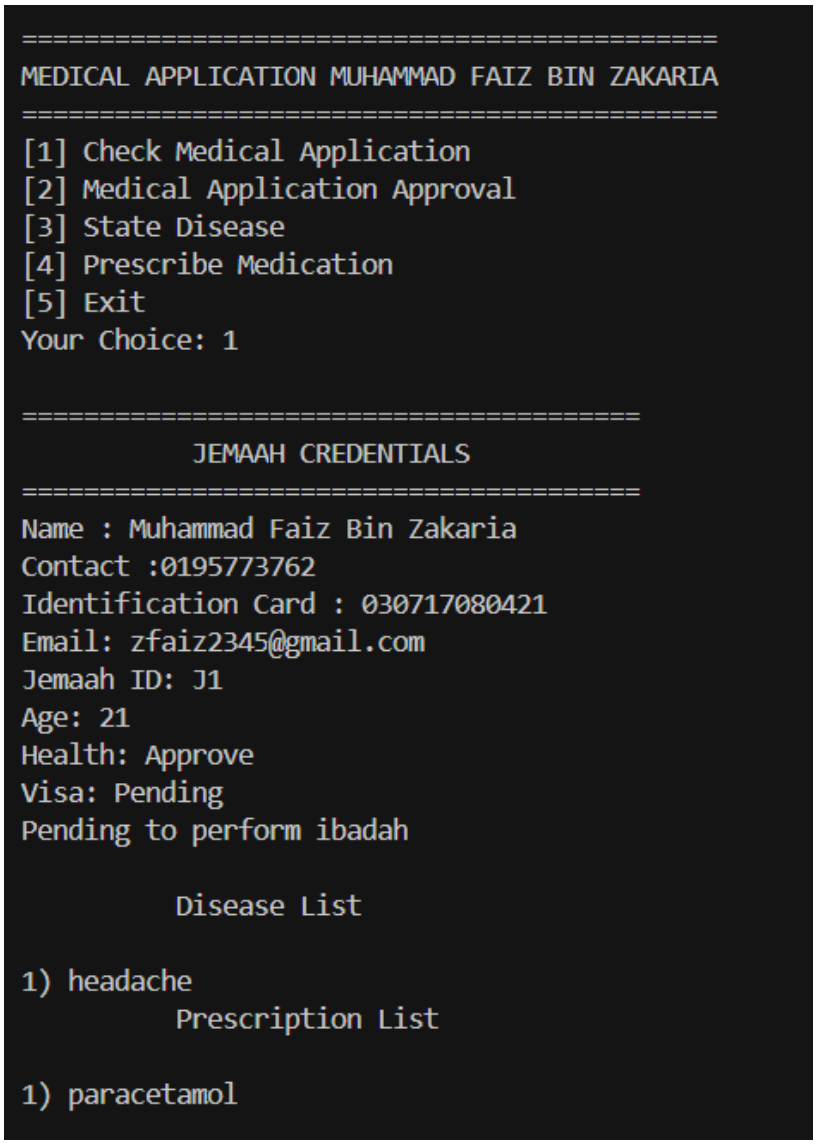


Figure 46

At this page choice 1, doctors could check their patient's latest medical information including disease list and prescription list that doctor updated previously.

Update Doctor Information

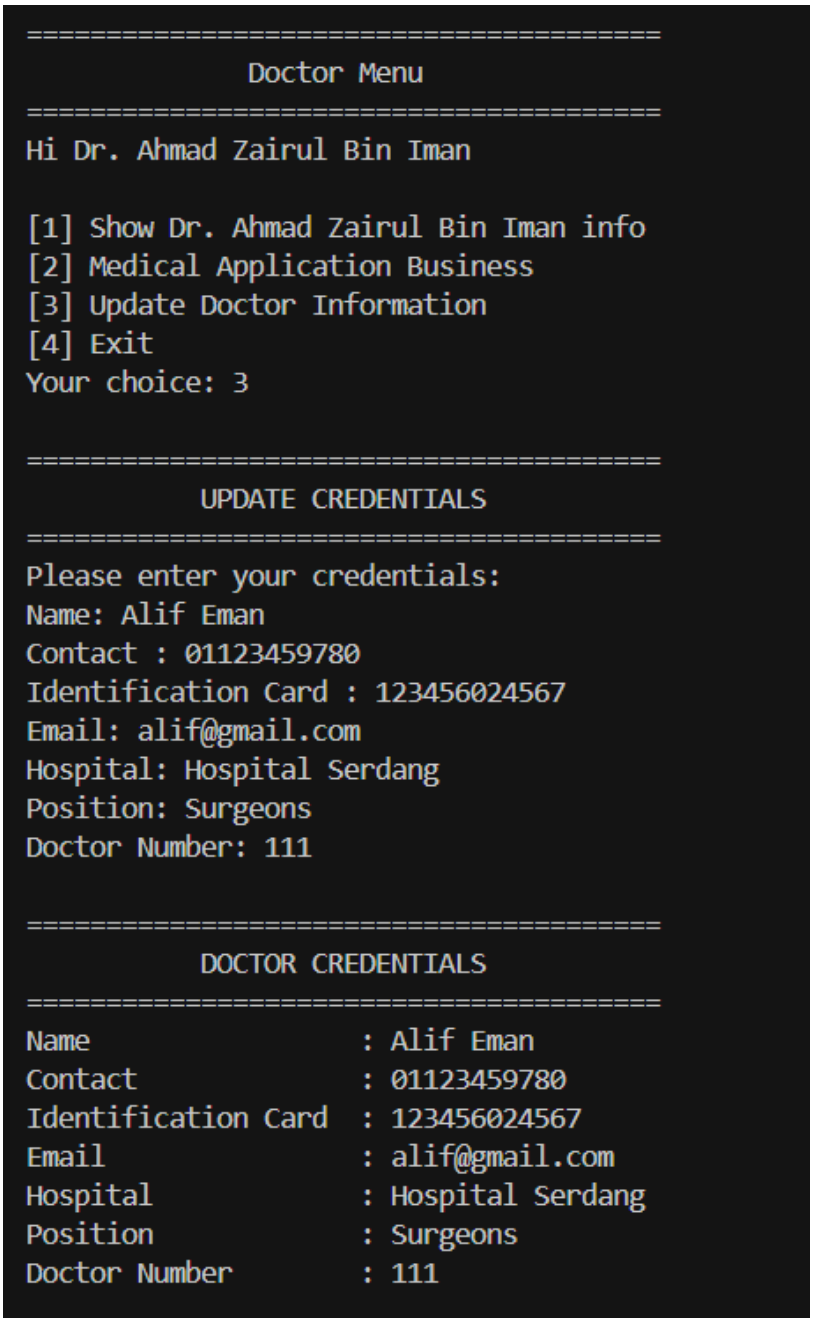


Figure 47

Doctors could update their current information by selecting choice 3. Then, doctors need to fill up the details form and lastly the system will display their latest information.

The work in this page has been done by: **MUHAMMAD IMAN AMIER BIN ABU BAKAR** *Date:* **22/06/2024**

From Embassy Point of View

Main Menu

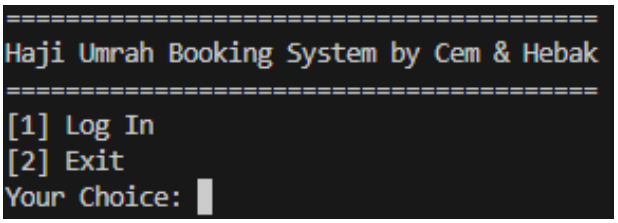


Figure 48

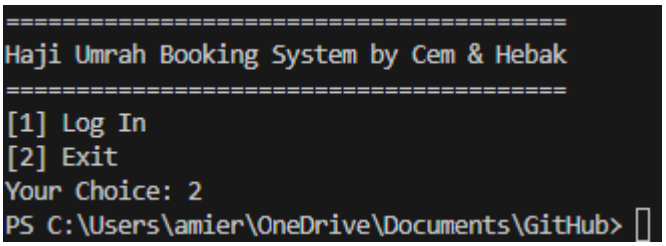


Figure 49

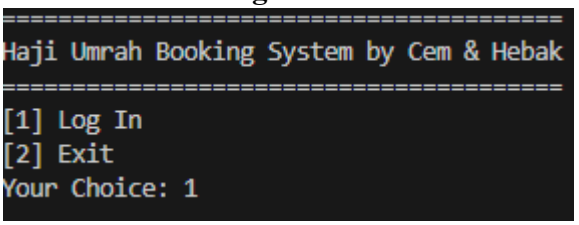


Figure 50

Embassy will direct to main menu and type the choice (1) to proceed with login. Otherwise, it will end the program

Selection Role

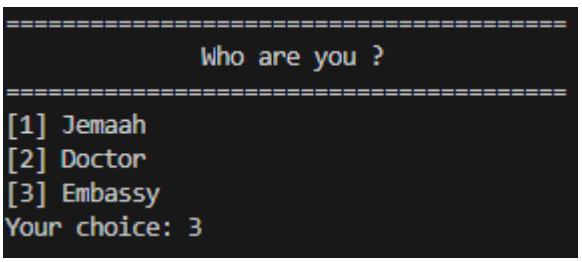


Figure 51

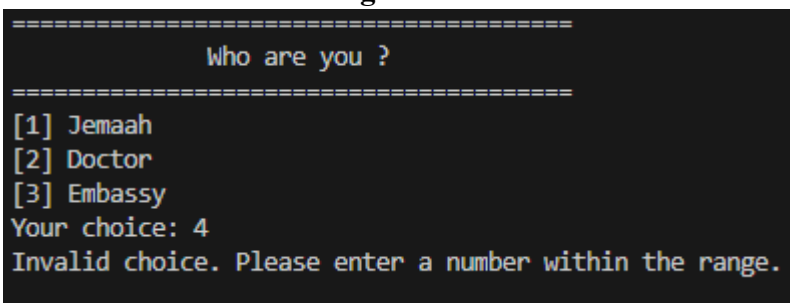


Figure 52

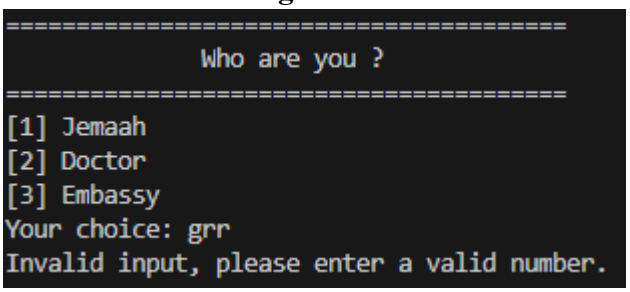


Figure 53

Embassy should enter choice number 3 to proceed with Embassy main menu. If the choice number is other than choice given or string type it will display error message like Figure 53

Login Page

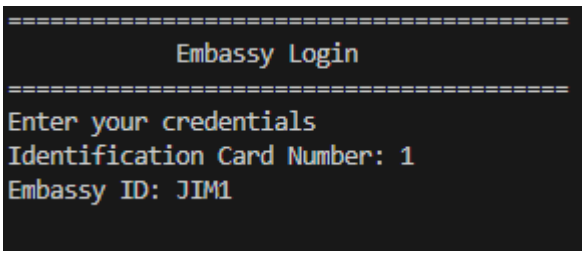


Figure 54

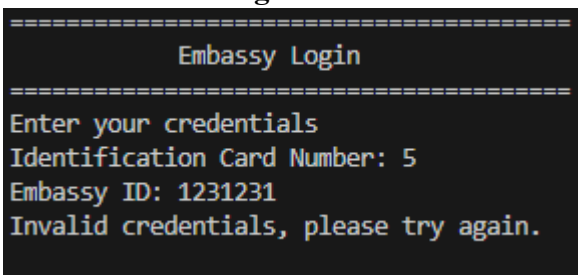


Figure 55

Embassy need to enter their Identification Card Number and Embassy ID. if the

Identification Card Number and Embassy ID are valid and correct, it will proceed to Embassy menu. Else it will show error message like Figure 55

Embassy Menu

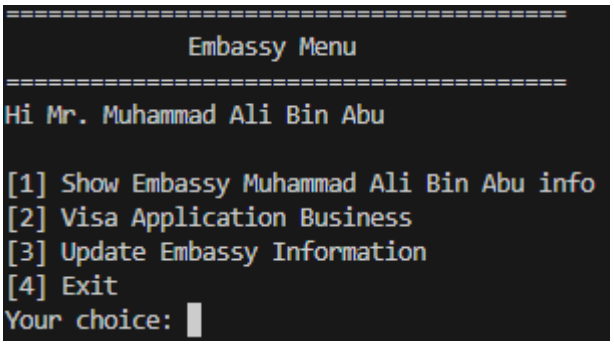


Figure 56

Embassy can show their own information, Go to Visa Application Business, Update their own information or Exit the Embassy Menu

Show Embassy Info

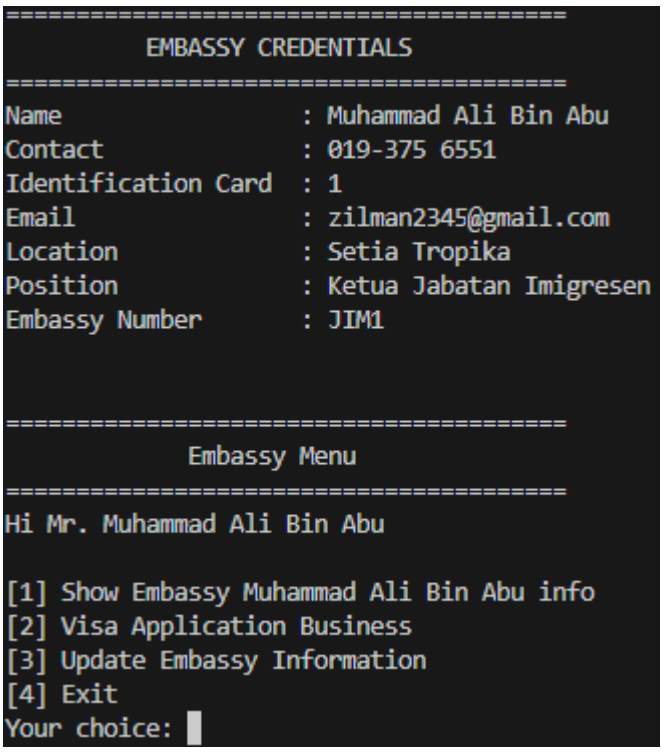


Figure 57

It will show their own information like Name, Contact, Identification Card, Email, location, Position, and Embassy Number. It will redirect to Embassy Menu.

Visa Application Business

```
=====================================
            List Of Jemaah
=====================================
1) Muhammad Faiz Bin Zakaria
2) Siti Khadijah Binti Ismail
3) Zulhilmi Bin Yusof
4) Ahmad Bin Samad
5) Siti Aminah Binti Abdul Rahman

[a] Press 0 for Exit
[b] Press number of jemaah [1 - 5]
Jemaah Number:
```

Figure 59

It will display List of Jemaah that attach with the Embassy.

Jemaah Number 0

```
=====================================
            List Of Jemaah
=====================================
1) Muhammad Faiz Bin Zakaria
2) Siti Khadijah Binti Ismail
3) Zulhilmi Bin Yusof
4) Ahmad Bin Samad
5) Siti Aminah Binti Abdul Rahman

[a] Press 0 for Exit
[b] Press number of jemaah [1 - 5]
Jemaah Number: 0

=====================================
            Embassy Menu
=====================================
Hi Mr. Muhammad Ali Bin Abu

[1] Show Embassy Muhammad Ali Bin Abu info
[2] Visa Application Business
[3] Update Embassy Information
[4] Exit
Your choice:
```

Figure 60

If the Embassy enter the Jemaah Number 0 it will redirect to Embassy Menu.

Jemaah Number not in range

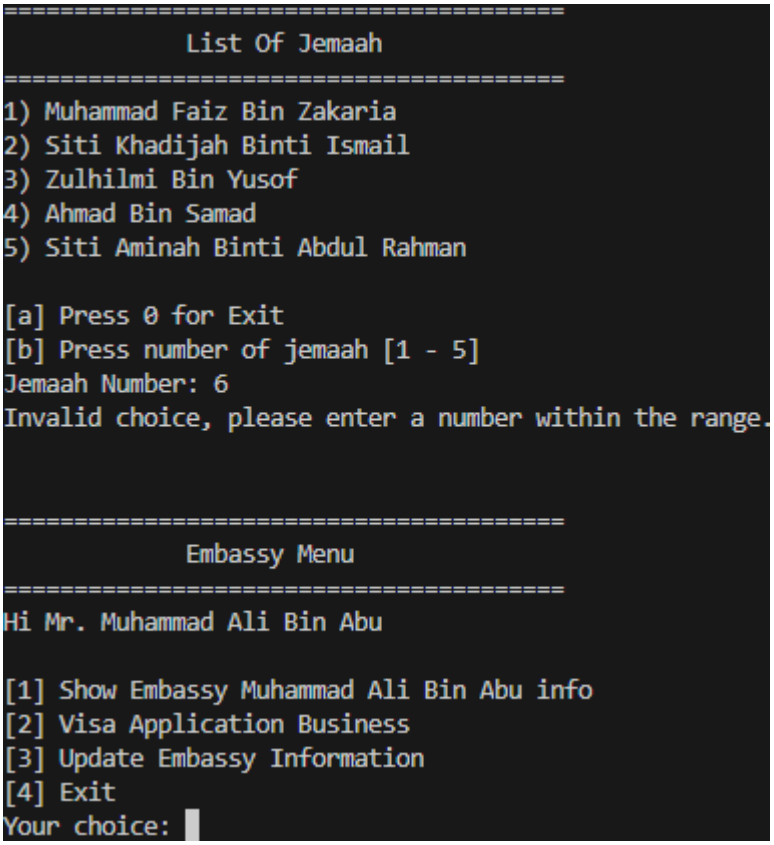


Figure 61

It will show error message like shown in Figure 61 and redirect to Embassy Menu.

Jemaah Number valid

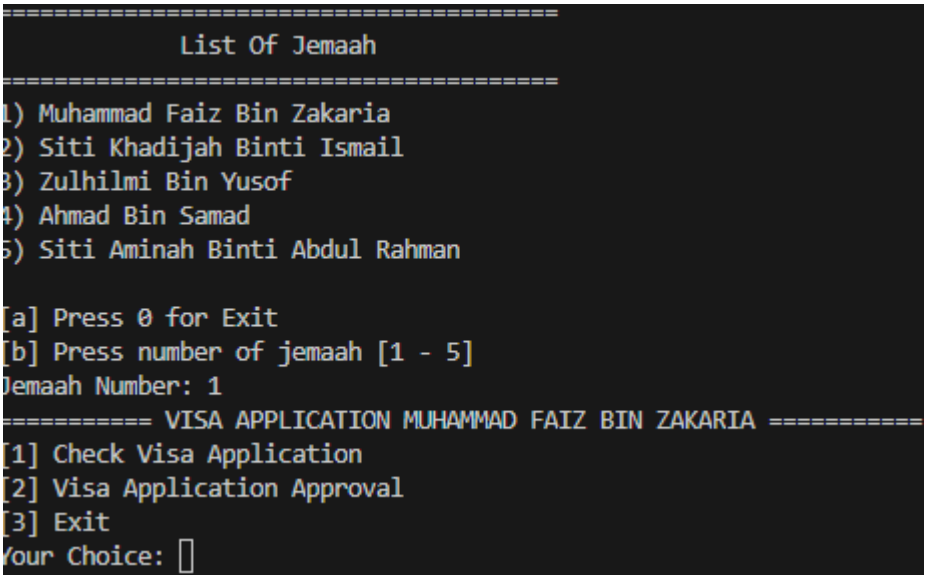


Figure 62

Embassy will enter to Visa Application for Jemaah selected. Embassy need to enter input choice.

Check Visa Application (choice 1)

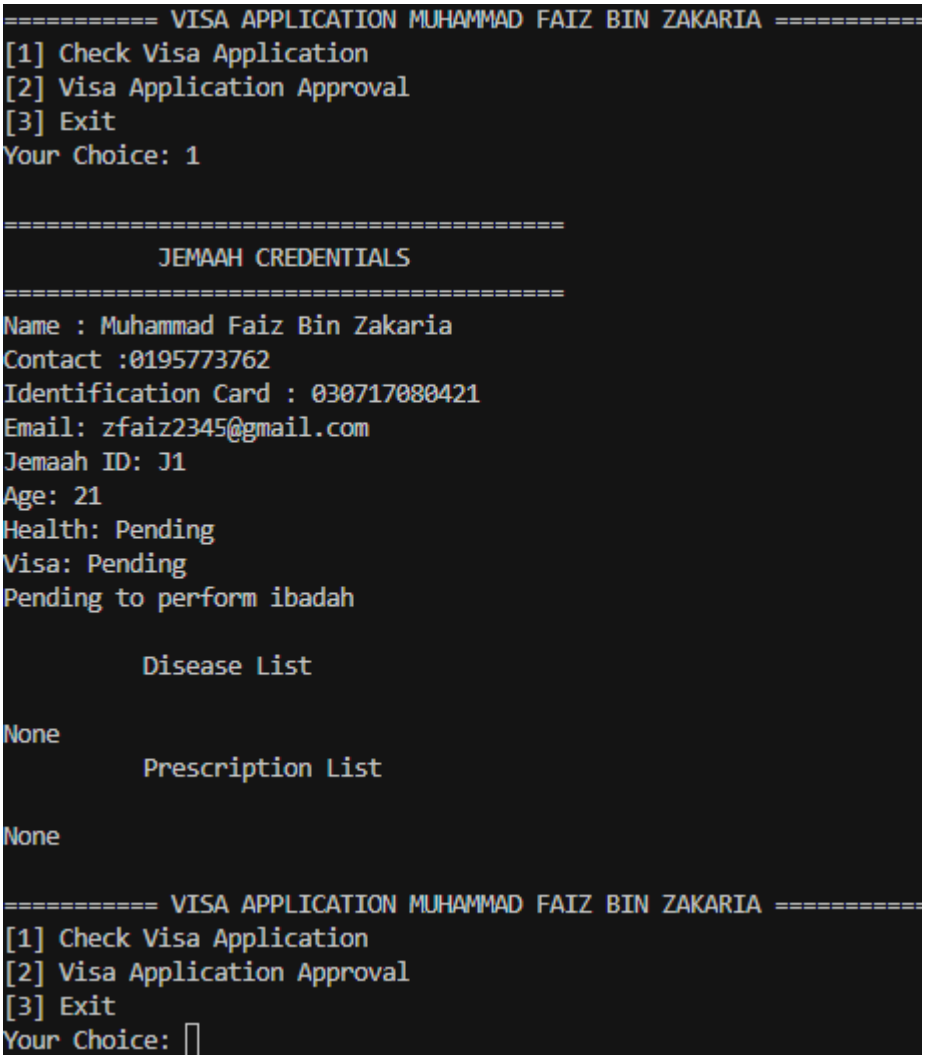


Figure 63

It will display Jemaah Credentials without Identification Number and Jemaah ID.

After that it will redirect to Visa Application Menu

Visa Application Approval (choice 2)

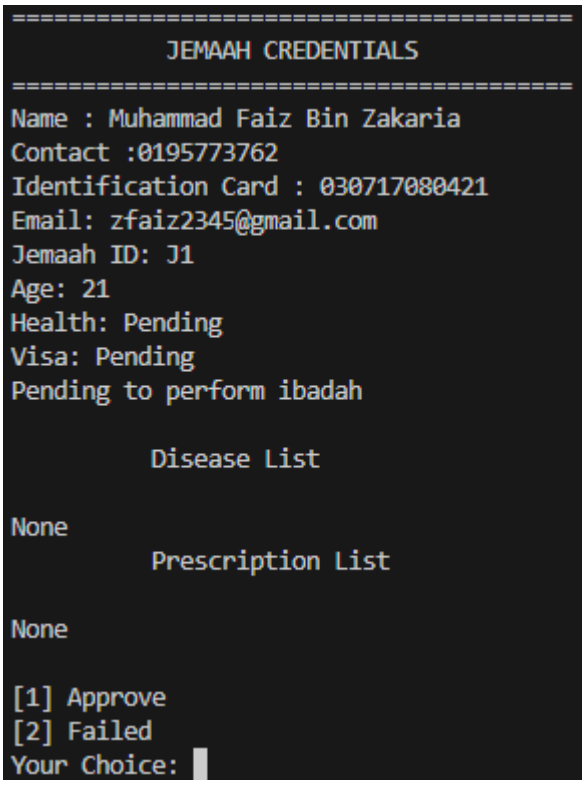


Figure 64

Embassy will be shown like Figure 64 and need to enter input of choice. Choice number 1 will approve the Visa, number 2 will Failed the Visa, and others will keep the visa stay on Pending status. It will redirect to Visa Application Menu.

Exit In Visa Application Menu (choice 3)

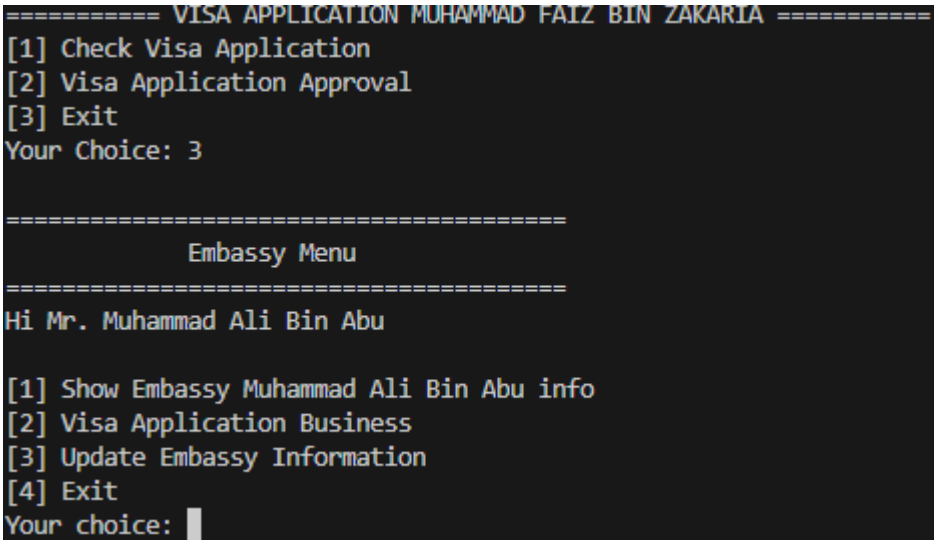


Figure 65

Choice number 3 will redirect to the Embassy Menu.

Update Embassy Information

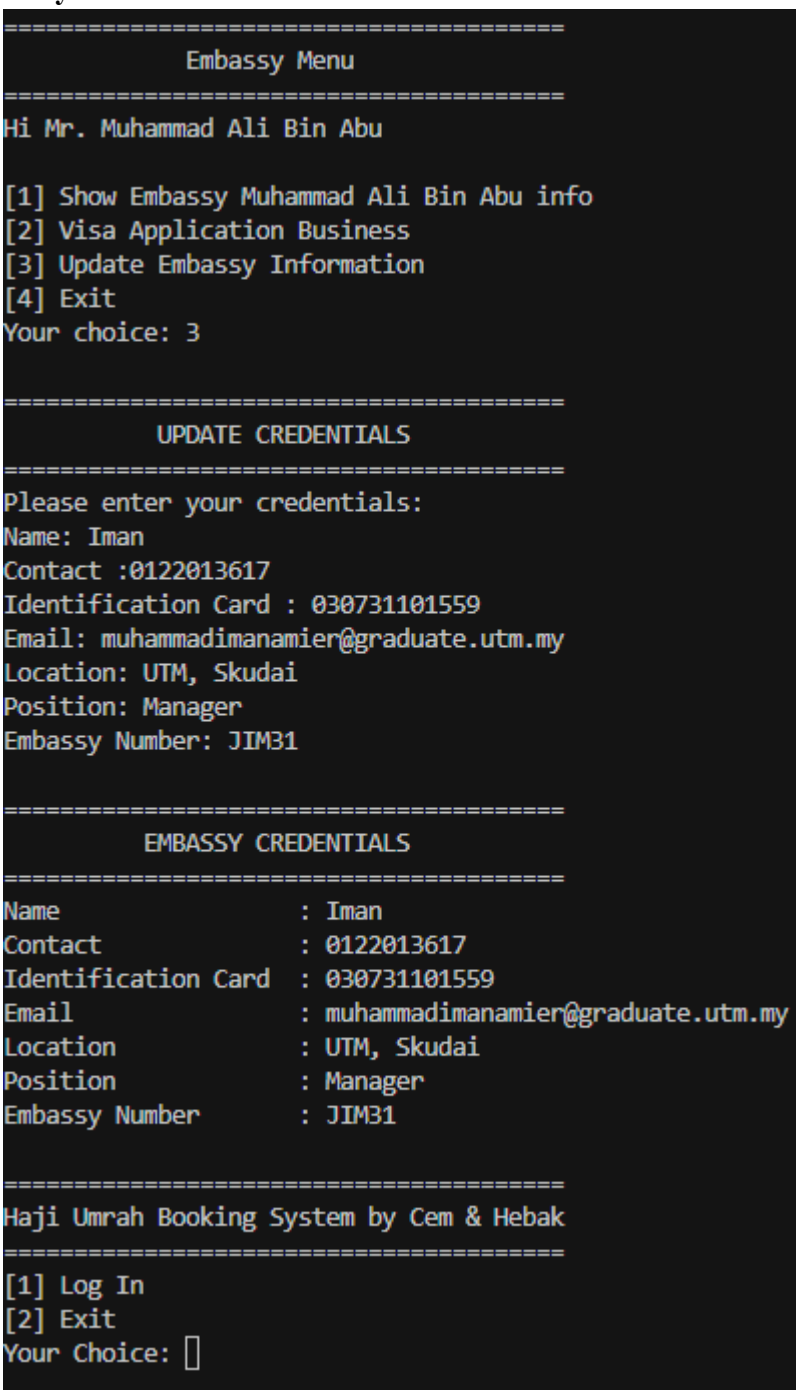


Figure 66

Embassy need to enter input of their new credentials like Name, Contact, Identification Card, Email, location, Position, and Embassy Number. It will show the new Embassy Credentials and needed to re-log in with new Identification card and Embassy Number that just updated.

Embassy Login with new Identification Card and Embassy Number

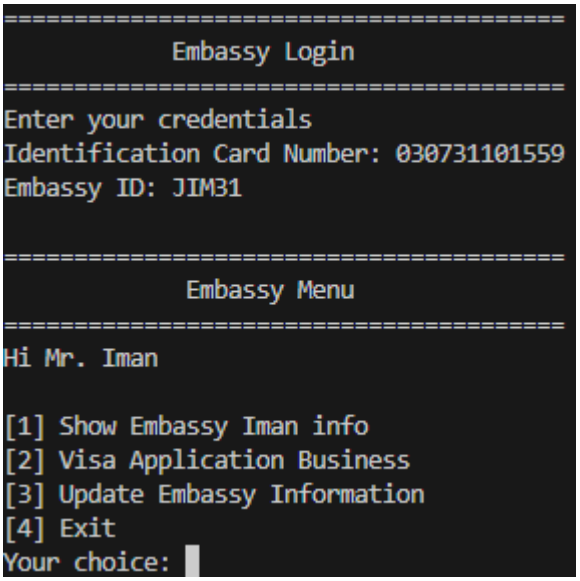


Figure 67

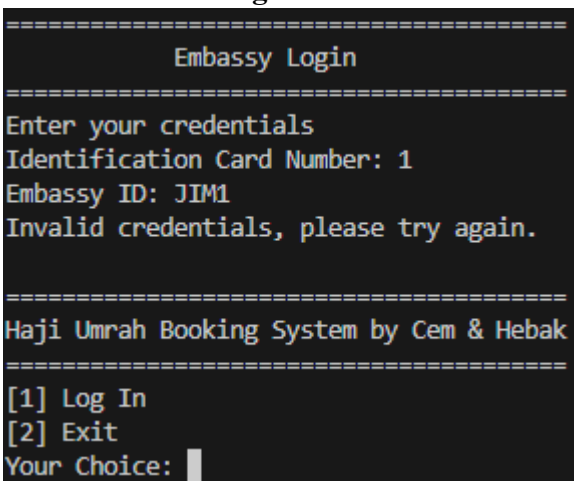


Figure 68

Embassy should login with their new Identification Card and Embassy Number to proceed to Embassy Menu. If the Embassy enter their old Identification Card and Embassy Number it will show error message like Figure 68

Exit in Embassy Menu (choice 4)

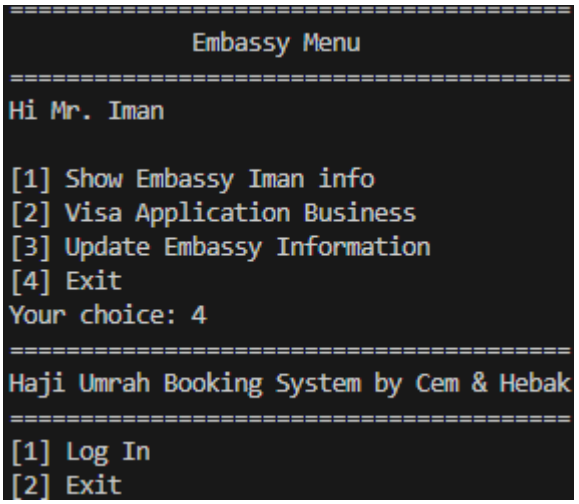


Figure 69

It will exit and redirect to Main Menu.