Logo, company name

Description automatically generated

**ASSIGNMENT**

**TECHNOLOGY PARK MALAYSIA**

CT038-3-2-OODJ

Object-Oriented Development with Java

HAND OUT DATE: 19th 2022

HAND IN DATE: 9th 2022

WEIGHTAGE:

STUDENT NAME : LAU JOO MING (TP065004)

: NG ZE-KANG (TP061488)

**INSTRUCTIONS TO CANDIDATES:**

**1 Submit your assignment at the administrative counter**

**2 Students are advised to underpin their answers with the use of**

**references (cited using the Harvard Name System of Referencing)**

1. **Late submission will be awarded zero (0) unless Extenuating**

**Circumstances (EC) are upheld**

**4 Cases of plagiarism will be penalized**

**5 The assignment should be bound in an appropriate style (comb bound or stapled).**

**6 You must obtain 50% overall to pass this module.**

Contents

[1.0 Assumption 4](#_Toc121500894)

[2.0 Diagram 6](#_Toc121500895)

[2.1 Use Case Diagram 6](#_Toc121500896)

[2.2 Use Case Specification 7](#_Toc121500897)

[Super admin 7](#_Toc121500898)

[Admin 9](#_Toc121500899)

[Customer 21](#_Toc121500900)

[2.3 Class Diagram 27](#_Toc121500901)

[3.0 User manual 28](#_Toc121500902)

[3.1 Login 28](#_Toc121500903)

[3.2 Customer 29](#_Toc121500904)

[Registration 29](#_Toc121500905)

[Customer home page 31](#_Toc121500906)

[Booking 34](#_Toc121500907)

[Check booking 36](#_Toc121500908)

[Payment 38](#_Toc121500909)

[Receipt 39](#_Toc121500910)

[Update profile 40](#_Toc121500911)

[Log out 42](#_Toc121500912)

[3.3 Admin 43](#_Toc121500913)

[Home Page 43](#_Toc121500914)

[Report/Analysis 44](#_Toc121500915)

[Customer Management Page 48](#_Toc121500916)

[Car Management Page 53](#_Toc121500917)

[Booking Management Page 58](#_Toc121500918)

[3.4 Super Admin 64](#_Toc121500919)

[Home Page (View) 64](#_Toc121500920)

[Add 64](#_Toc121500921)

[Edit (Edit and Delete) 66](#_Toc121500922)

[Search 69](#_Toc121500923)

[4.0 OO concept 71](#_Toc121500924)

[Class 71](#_Toc121500925)

[Object 72](#_Toc121500926)

[Abstraction 73](#_Toc121500927)

[Encapsulation 74](#_Toc121500928)

[Modularity 77](#_Toc121500929)

[Inheritance 78](#_Toc121500930)

[5.0 Additional features 81](#_Toc121500931)

[5.1 Date 81](#_Toc121500932)

[5.2 Show confirm dialogue 83](#_Toc121500933)

[5.3 Combo Box 84](#_Toc121500934)

[5.4 Table with extra functionalities 85](#_Toc121500935)

[5.5 Better Gui 85](#_Toc121500936)

[5.6 Pie Chart 86](#_Toc121500937)

[5.7 Auto generate ID (Extra fucntion) 86](#_Toc121500938)

[5.8 isNumeric (Extra fucntion) 87](#_Toc121500939)

[6.0 References 88](#_Toc121500940)

# Assumption

* The user must register as a customer and login to rent a car
* Customer ID will be generated automatically when users register an account, the ID will start with three characters ‘CUS’ followed by 6 digits.
* Admin ID will be generated automatically when super admin adds a new admin account, the ID will start from three character ‘ADM’ followed by 4 digits
* There will be only one super admin account created in the system, the super admin ID will start from three character ‘SUP’ followed by ‘0001’
* Car ID will be generated automatically when the admin adds a new car into the system, the ID will start with a character ‘C’ followed by 4 digits.
* Booking ID will be generated automatically when customer makes new booking, the ID will start with two characters ‘BK’ followed by 6 digits.
* Payment ID will be automatically generated when customer pay the booking, the ID will start with 2 characters ‘PY’ followed by 4 digits. Admin does not need to collect payment or generate receipt manually.
* The register button can only be used to create customer account but not admin account.
* Only customers can create a customer account, admin or super admin cannot create customer account.
* Super admins can only view admin accounts, while admins can only view customer accounts.
* Admin cannot edit, or view customer’s password.
* The address represents a specific area only, not the full address.
* Customers can only pick up or drop off the car at the store, they are not allowed to choose other locations.
* The total rental amount depends on the total number of days rented by a customer; therefore, customers need to input the rent date and return date while doing their booking.
* After a car is booked by the customer, the status of the car will change from "available" to "booked". In “booked” state, customers are allowed to make full payment or cancel the booking.
* If customers fail to make payment or cancel the booking; the status of the car will change from “booked” to “available”.
* Customer must pay the exact amount to make a successful payment to book the car. After that, the status of the car will change from “booked” to “not available”.
* Admin will change the car booking status manually from not available to available after customers return the car.

# Diagram

## 2.1 Use Case Diagram

Diagram

Description automatically generated

Figure 2‑1: Use Case Diagram

## 2.2 Use Case Specification

### Super admin

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

### Admin

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

### Customer

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

## 2.3 Class Diagram

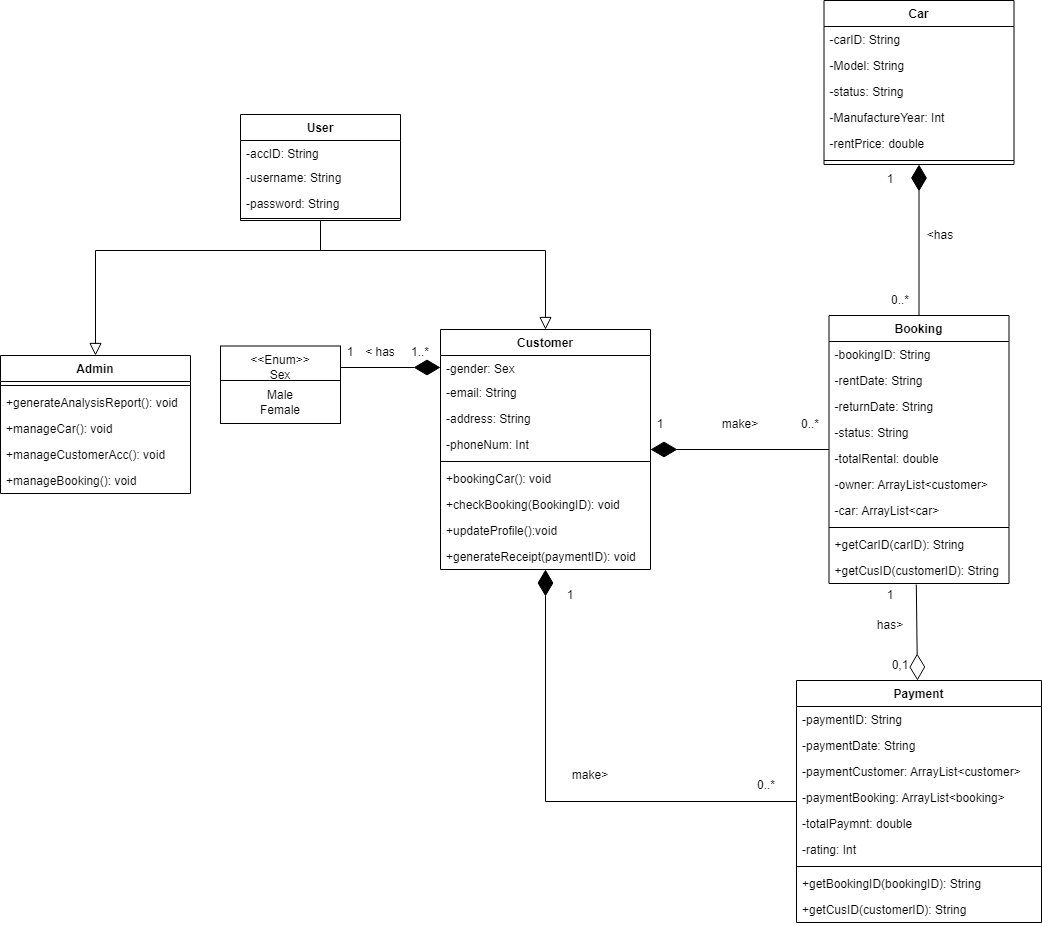


Figure ‑ Class Diagram

# User manual

## 3.1 Login

Graphical user interface, application, Word

Description automatically generated

Figure ‑

Figure above shows when the system is executed, an authentication login page is displayed for user to enter the account id and password. The user can choose the user type by using the drop list.

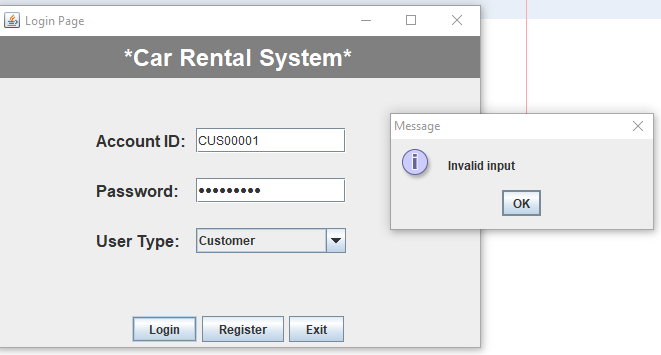


Figure ‑

If the user enters an invalid account ID or password, the system will pop out an error message to show that the input is not valid.

## 3.2 Customer

### Registration

Graphical user interface, application

Description automatically generated

Figure ‑

When the user clicks the register button on the login page, a registration page is displayed. The user is required to enter the username, select the gender, email, address, phone number and password.

Graphical user interface, application

Description automatically generated

Figure ‑

If the user enters the wrong information data type, such as a string in the phone number text field, an error message dialogue is shown, and the user needs to enter their personal information again. The user can click the reset button to set every text field to null.

Graphical user interface

Description automatically generated

Figure ‑

When the user enters valid input and clicks the register button, an auto-generated customer ID is generated, and a message dialogue is displayed. The message dialogue will show the customer ID and give instruction for the user to login again.

### Customer home page

Graphical user interface, diagram

Description automatically generated

Figure ‑

If the customer successfully logs into the system, the customer home page shown in the figure above will be displayed. A welcoming message with the customer’s name will also be displayed. There are 4 buttons that can be clicked by the customer, which are booking, receipt, check booking and update button. There is also logout button for customers to logout the system.

Graphical user interface, diagram

Description automatically generated

Figure ‑

Logically, if the customer does not make any bookings, he is not able to check the booking button. And a message dialogue will be displayed to inform the customer that no booking record was found.

Diagram

Description automatically generated

Figure ‑

If the customer clicks the "receipt" button without making any payment before, a message dialogue will be displayed that shows "no payment record found."

### Booking

Table

Description automatically generated

Figure ‑

When customer clicks on the booking button, a booking page appears, as shown in the figure above. A list of car information will also be displayed in a table for reference. The customer is required to enter the rent date, the return date of the car, and the car id.

Graphical user interface, table

Description automatically generated

Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

Customers are not allowed to enter an earlier return date compared with the rent date. Besides that, customer must follow the date format “dd/mm/yyyy”. Otherwise, an error message will be displayed as shown in figure above.

Graphical user interface, table

Description automatically generated

Figure ‑

Moreover, customers are not allowed to book a car with “Not available” or “Booked” status. If customer do so, an error message will be displayed to inform the customer selected car is not available.

Graphical user interface

Description automatically generated

Figure ‑

Finally, when the customer is successful in booking a car, a message dialogue will be shown so that the customer will know that the booking is successful, and the system will return back to the customer home page.

### Check booking

Table

Description automatically generated

Figure ‑

Customers are allowed to check their booking details once the booking is successful. Figure above shows the booking details page, which is able to display the booking id, rent date, return date, booking status, total rent, customer id, and the car id. The customer can pay for the booking, cancel booking, or back to home page.

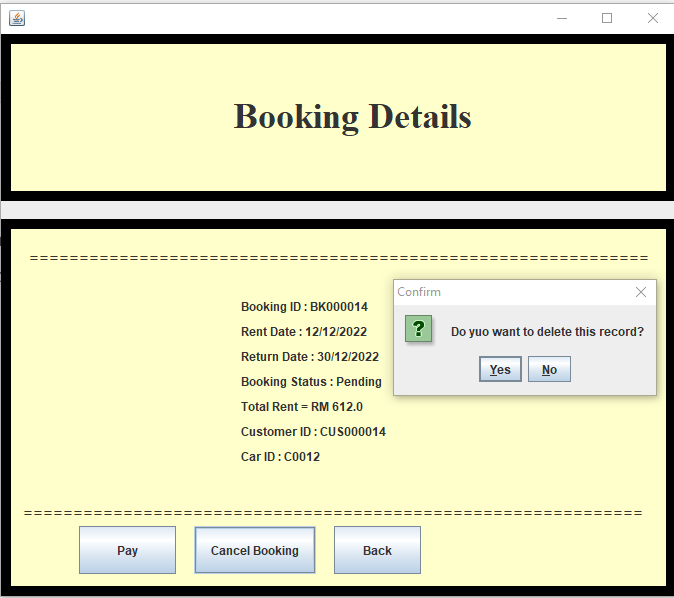


Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

If the customer clicks the "cancel booking" button, a confirmation dialogue will be displayed to confirm again with the customer whether they want to delete the record or not. When the customer selected yes, the booking record will be deleted.

### Payment

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface, application, Word, website

Description automatically generated

Figure ‑

Graphical user interface, application, Word

Description automatically generated

Figure ‑

Once the customer clicks the pay button, the payment page will be displayed, and the customer need to enter the exact amount payment. Error messages will be displayed if the customer make more or less than the exact payment.



Figure ‑

Graphical user interface, application, email

Description automatically generated

Figure ‑

Graphical user interface, application, Word

Description automatically generated

Figure ‑

After the customer enters the exact amount of payment, the customer must give a rating to the system. The customer can only rate between 1 to 5 stars. If the customer enters a rating that is not between 1 to 5, an error message is displayed, as shown in the figure above. Once the customer has made the rating for the system, a message dialogue will be displayed to inform the customer the full payment received.

### Receipt

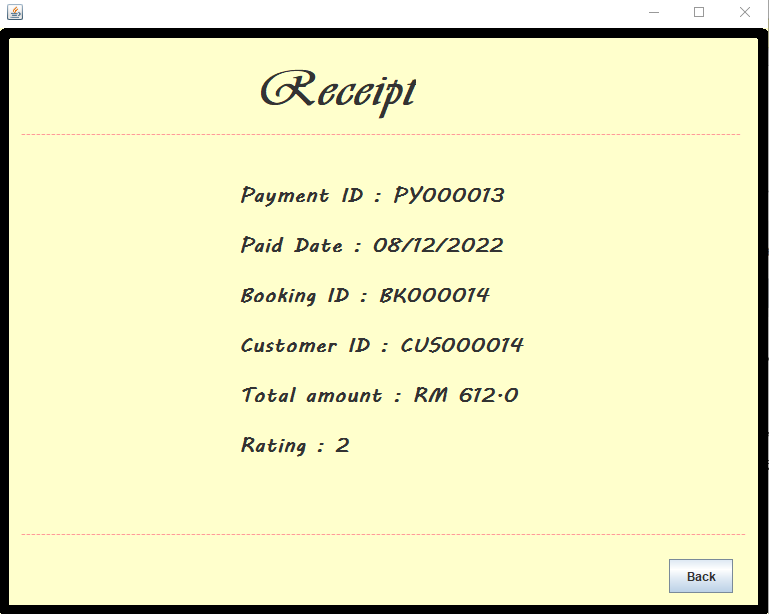


Figure ‑

After making the payment, the customer can always check the payment receipt by clicking the receipt button on the customer home page. The output of the receipt will be shown in Figure above, which will display the customer payment id, paid date, booking id, customer id, total amount, and rating.

### Update profile

Graphical user interface, application

Description automatically generated

Figure ‑

The car rental system allowed customers to change their personal information by clicking the update button on customer home page. Based on the figure above, customers are not able to edit when entering the customer profile page.

Graphical user interface

Description automatically generated

Figure ‑

The customer can click on the edit button, and all details except customer id will be editable.

Graphical user interface

Description automatically generated

Figure ‑

Once the customer has finished editing their personal details, they can press the save button, and the edited personal details will be stored.

Graphical user interface

Description automatically generated

Figure ‑

Same as registration, if the customer enters wrong information data type, the system will display an error message.

### Log out

Graphical user interface, diagram

Description automatically generated

Figure ‑

By clicking the log out button, the customer is able to exit, and the system will be back to the login page.

## 3.3 Admin

### Home Page

Graphical user interface, application

Description automatically generated

Figure ‑

For admin, the user type selected must be “Admin” in order to get into the admin home page with valid AccountID and password.

Graphical user interface

Description automatically generated

Figure ‑

If the admin successfully logs into the system, the admin home page shown in the figure above will be displayed. A welcoming message with the admin’s name will also be displayed. There are 4 buttons that can be clicked by the admins, which are “Report/ Analysis”, “Customer Management Page”, “Car Management Page”, “Booking Management Page”

### Report/Analysis

Graphical user interface

Description automatically generated

Figure ‑

After clicking “Report/ Analysis” button, there are 6 more button for admin to generate 6 different report or analysis.

Graphical user interface, chart, application, pie chart

Description automatically generated

Figure ‑

Based on Figure above, Pie Chart of gender analysis of customer will be displayed after clicking ‘Gender Analysis’ button. Pie Chart of current car status analysis will be displayed after clicking ‘Car Status Analysis’. Pie Chart of current booking status analysis will be displayed after clicking ‘Booking Status Analysis’. Pie Chart of feedback analysis will be displayed after clicking ‘Feedback Analysis’. All of the pie charts have labels, and legends displayed.

Table

Description automatically generated

Figure ‑

After clicking Average Feedback Analysis button, the feedback analysis page will display as shown as Figure above. The rating for each payment is displayed in a table with other information. Then, the overall average rating is displayed below the table with 2 decimal places. Back button can be pressed to go back to previous page.

Table

Description automatically generated

Figure ‑

After clicking Finance Report button, the finance report will display as shown as Figure above. The amount paid for each payment is displayed in a table. Then, the overall profit is displayed below the table. Back button can be pressed to go back to previous page.

### Customer Management Page

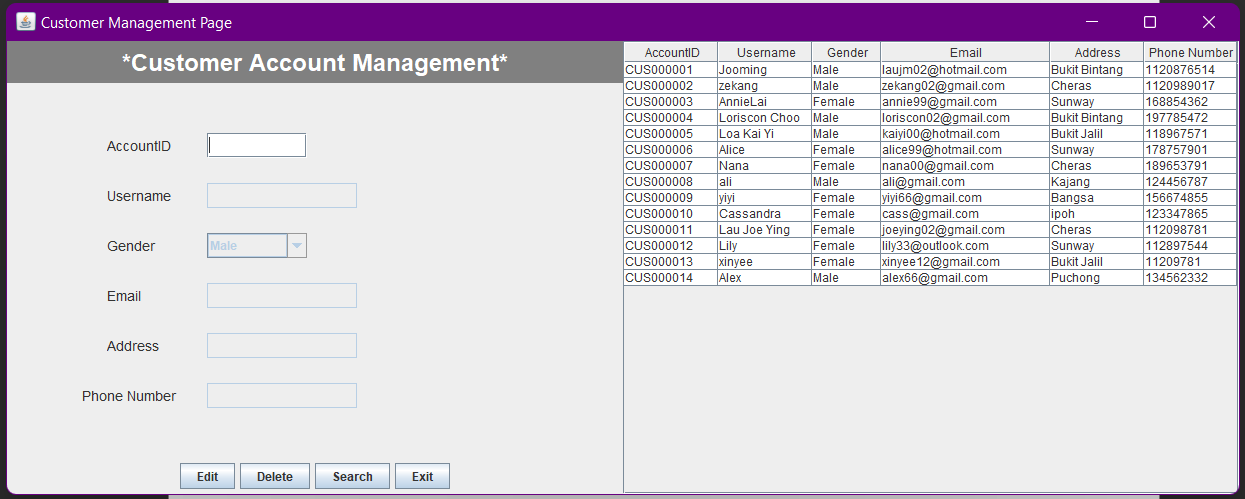


Figure ‑

After admins clicks the ‘Customer Management Page’ button, the page as shown in Figure above will be displayed.

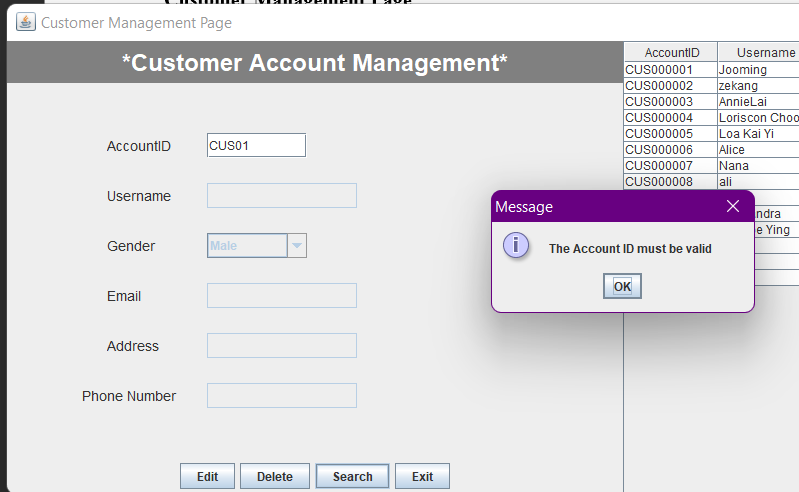


Figure ‑

An error message will pop up once the account ID is not valid, this applied to Delete button and Search button.

Graphical user interface

Description automatically generated

Figure ‑

For delete button, after entering the valid accountID, a message will pop up, and the record in the table will be removed.

Graphical user interface, application

Description automatically generated

Figure ‑

For Search button, after entering valid accountID, all the other information will be displayed in the textbox, but the textbox is still not editable.

Graphical user interface

Description automatically generated

Figure ‑

After clicking the Edit button, all the text field become editable.

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

Based on Figure shown above, validation take place at username, email, and address to ensure the values entered are in correct data type. Not only that, the text field also cannot be left empty.

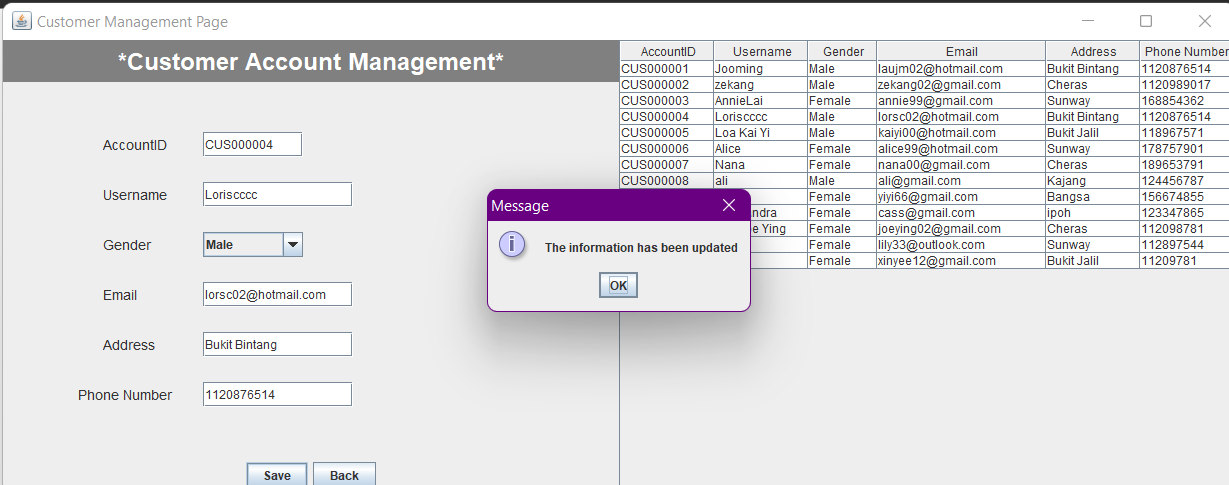


Figure ‑

When all the information entered are valid, a message will be displayed, and the table will be automatically with the latest information. If users do not want to make the changes, they can also click Go back button to go back to previous page. Then, Exit button can be clicked to go back to admin home page.

### Car Management Page

Graphical user interface, table

Description automatically generated with medium confidence

Figure ‑

After admins clicks the ‘Car Management Page’ button, the page as shown in Figure above will be displayed. The car information is displayed on the table on the right. CarID is autogenerated and it is set uneditable. The status of the car by default is set to Available, and it is also not editable.

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface, text, application, email

Description automatically generated

Figure ‑

Graphical user interface, text, application, email

Description automatically generated

Figure ‑

Based on Figure shown above, validation take place at car model, manufacture year, and rent price to ensure the values entered are in correct data type. Not only that, the text field also cannot be left empty. After valid information are entered, and add button is clicked, the table will be updated with new data, and the text field will be clear, to be ready for next input. The Car ID will also autogenerated again.

Graphical user interface, table

Description automatically generated

Figure ‑

After clicking “Search/Delete/Edit” button, The Car ID will set to editable, and other text field will become not editable. 4 new buttons are shown which are Edit, Delete, Search, and Go back buttons.

Graphical user interface

Description automatically generated

Figure ‑

An error message will pop up once the car ID is not valid, this applied to Delete button and Search button.

Graphical user interface, application

Description automatically generated

Figure ‑

When the car ID is valid and Delete button is clicked, a message will pop up to notify the car has been deleted. At the same time, the car information will be removed from the table.

Table

Description automatically generated

Figure ‑

For the Search button, after entering valid Car ID, other information will be displayed, but the text fields are still not editable.

Table

Description automatically generated

Figure ‑

After clicking the Edit button, all the text fields will become editable. The validation is also same as when adding car information, the data type for model, manufacture year, and rent price are checked.

Graphical user interface

Description automatically generated

Figure ‑

After the Save button is clicked, a message will pop up. The information inside table will be updated. Then users can click Go back button to go back to the previous page. Then users can click the Exit button to go back to admin page.

### Booking Management Page

Graphical user interface, text, application

Description automatically generated

Figure ‑

After admins clicks the ‘Booking Management Page’ button, the page as shown in Figure above will be displayed.

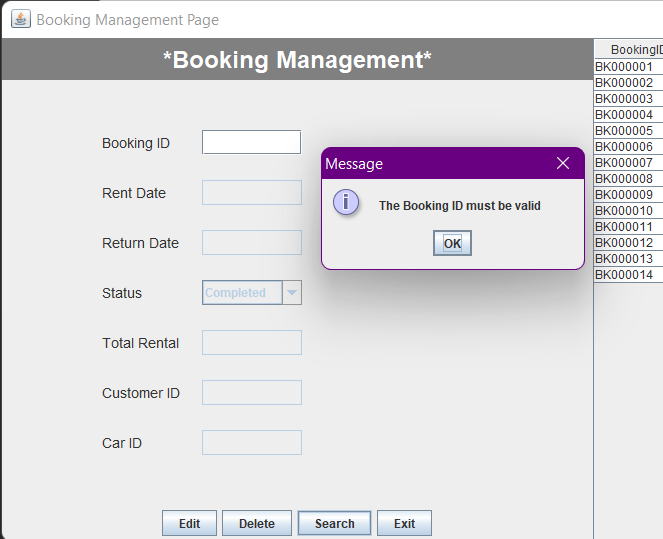


Figure ‑

An error message will pop up once the booking ID is not valid, this applied to Delete button and Search button.

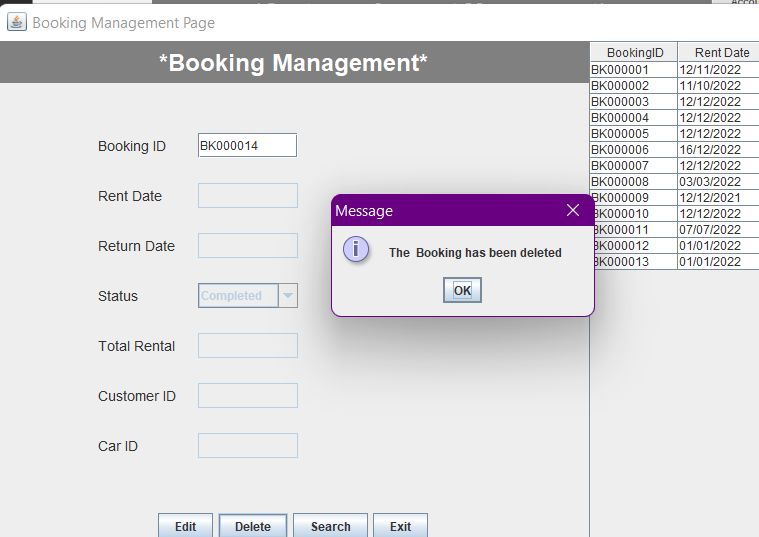


Figure ‑

For Delete button, after entering the valid booking ID, a message will pop up, and the record in the table will be removed.

Graphical user interface

Description automatically generated

Figure ‑

For Search button, after entering valid booking ID, all the other information will be displayed in the textbox, but the textbox is still not editable.

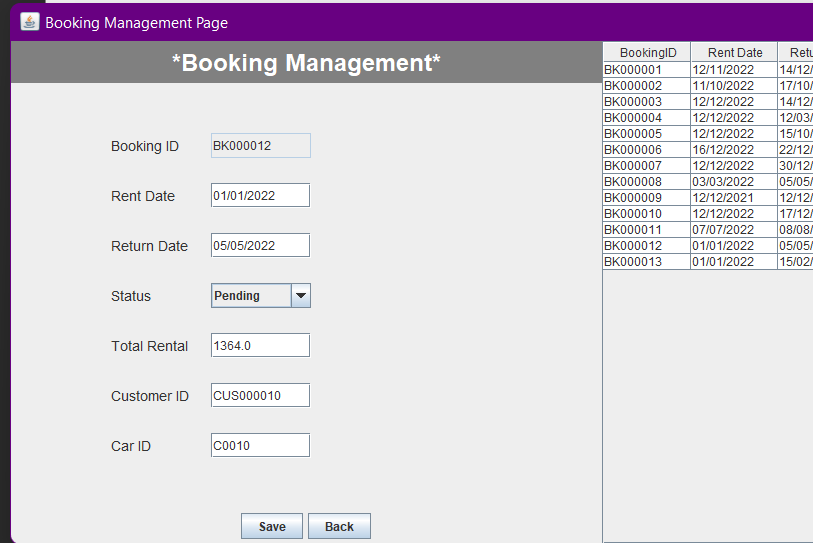


Figure ‑

After clicking the Edit button, all the text field become editable except Booking ID.

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface, application

Description automatically generated

Figure ‑

Graphical user interface, application

Description automatically generated

Figure ‑

Based on Figure shown above, validation take place at customer ID and Car ID to ensure the data entered is valid. Not only that, the text field also cannot be left empty.

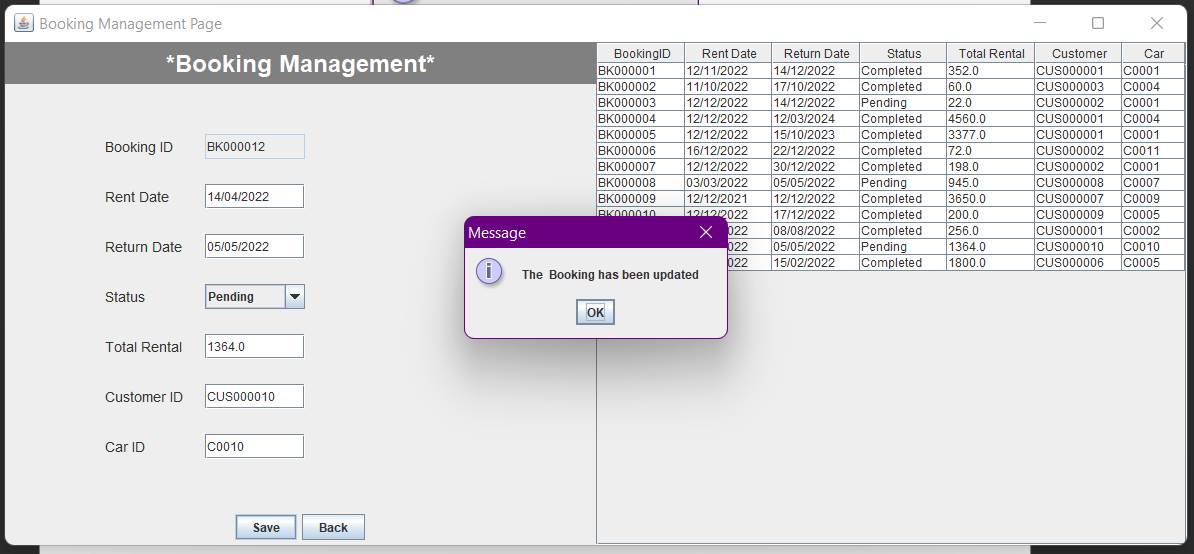


Figure ‑

When all the information entered are valid, a message will be displayed after clicking Save button, and the table will be automatically with the latest information. If users do not want to make the changes, they can also click Go back button to go back to previous page. Then, Exit button can be clicked to go back to admin home page.

Graphical user interface, application

Description automatically generated

Figure ‑

After clicking Logout button, a message will display, then user will be prompted to login page.

## 3.4 Super Admin

### Home Page (View)

Graphical user interface, application

Description automatically generated

Figure ‑

For super admin, the accountID is pre-set in the system, The accoundID must be SUP0001 in order to get into super admin page. The user type selected must be “Admin”.

### Add

Table

Description automatically generated with medium confidence

Figure ‑

After logging to the system, the super admin home page will be displayed as shown above. A list of admin account information will be displayed in a table. The accoundID for admin will be automatically generated and displayed in the textbox and this textbox is not editable. Then, super admin can fill in the text box for username and password to add the admin account by clicking the Add button.

Graphical user interface

Description automatically generated

Figure ‑

Validation happen here whereby the value enter in username textbox must be a string.

Table

Description automatically generated

Figure ‑

After valid username and password are entered, the table will be updated with new admin account information, and AccountID at the text field will automatically updated again.

### Edit (Edit and Delete)

Table

Description automatically generated

Figure ‑

After clicking Edit button, there are 3 new button shows up, which are Delete, Save and Go back buttons. The textbox of AccountID is also become editable now. The Go back button will return to previous page.

Graphical user interface

Description automatically generated

Figure ‑

An error message pop up if user enter non-existing accountID. These validation applied for Delete button and Save button.

Graphical user interface, application

Description automatically generated

Figure ‑

In order to delete account, only accountID text field is required to be filled up with valid AccountID. After clicking the Delete button, a pop up message will display, and the record will be removed from the table automatically.

Graphical user interface, timeline

Description automatically generated

Figure ‑

Graphical user interface, application

Description automatically generated

Figure ‑

For the Save button, the text field for username and password cannot left empty or else a pop up message will be displayed. Next, the system will reject digit or number in username text field.

Graphical user interface, application

Description automatically generated

Figure ‑

After entering valid username and password, a pop up message will show up, and the data inside table is updated.

### Search

sTable

Description automatically generated

Figure ‑

Graphical user interface

Description automatically generated

Figure ‑

After clicking Search button, the accountID will become editable while username, and password will be not editable. Validation also applied here to make sure the AccountID entered is valid.

Table

Description automatically generated

Figure ‑

After entering valid accountID and click the Search Now button, the username and password will be displayed at the text field. Then, Go Back button can be clicked to go back to previous page.

Graphical user interface, application

Description automatically generated

Figure ‑

After clicking Logout button, the system will go back to login page and pop-up message will show up. Lastly, at the login page, there is a ‘Exit’ button to exit the system.

# OO concept

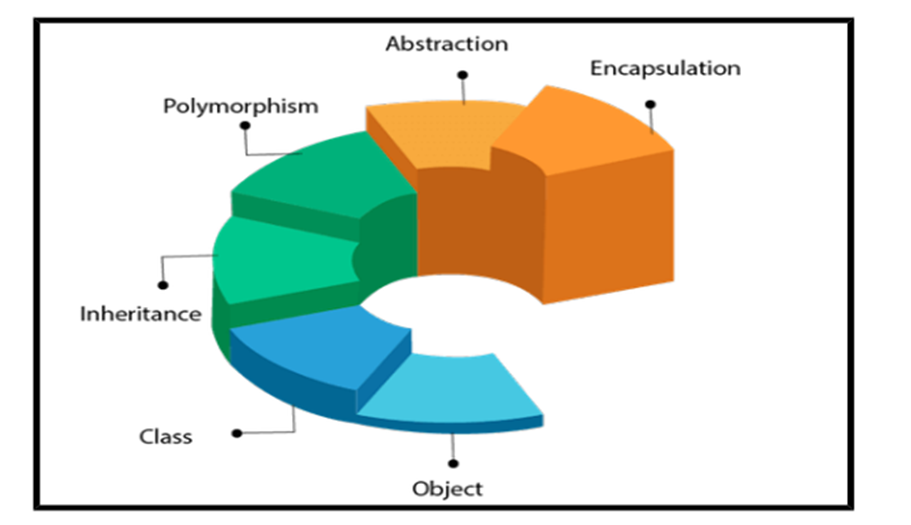


Figure ‑

The approach of object-oriented programming offers various types of notions, for example, abstraction, modularity, encapsulation, inheritance and so on. These ideas seek to incorporate real-world objects into programmes and develop operational procedures and variables to repeat them without jeopardising security. The foundation of Java programming, which is used to create programmes using classes and objects, is object-oriented programming. Data control for code access is another way to define OOPs. In this method, programmers specify the operations that will be performed on a data structure as well as its data type.

## Class

A class is a design or a set of guidelines for creating a particular kind of object. It is a fundamental idea in object-oriented programming that centres on actual living things. In Java, a class determines an object's behaviour and what it will include. Classes improve code reusability. One class can include common code that is shared by multiple classes with similar functionality. When necessary, the related classes can offer unique behaviour to increase both the functionality of the starting class and that of other related classes (Progress, 2017).

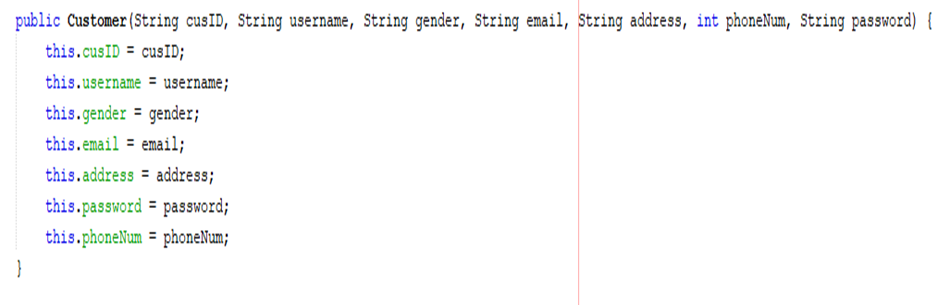


Figure ‑

Figure above shows a customer class with cusID, username, gender, email, address, password, phoneNum attributes.

## Object

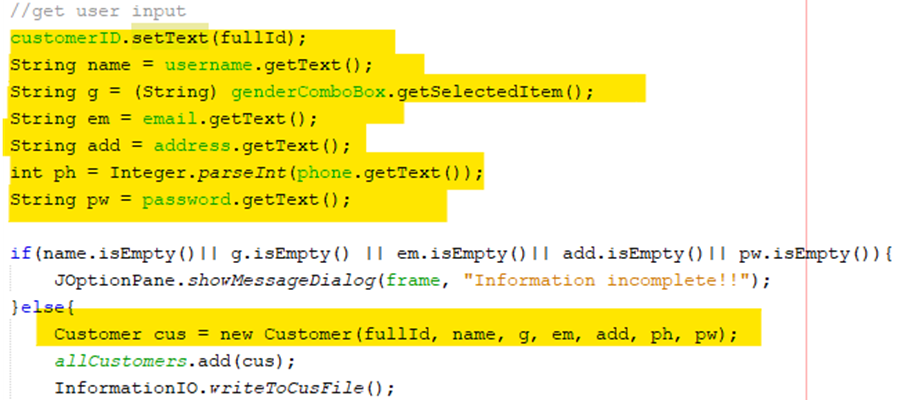


Figure ‑

When a user registers as a customer, a new object called "cus" is created. It is clearly stated that the new object "cus" has the same attributes as the customer class. In addition, every object will have different values of attributes that compare with each other.

## Abstraction

The object-oriented programming notion of abstraction "shows" only necessary properties and "hides" extraneous data. Abstraction's fundamental goal is to shield people from pointless information. Abstraction is the process of choosing information from a bigger pool such that the user only sees pertinent details of the item. It aids in lowering programming effort and complexity. It is one of the most crucial OOPs ideas (Hartman, 2022).

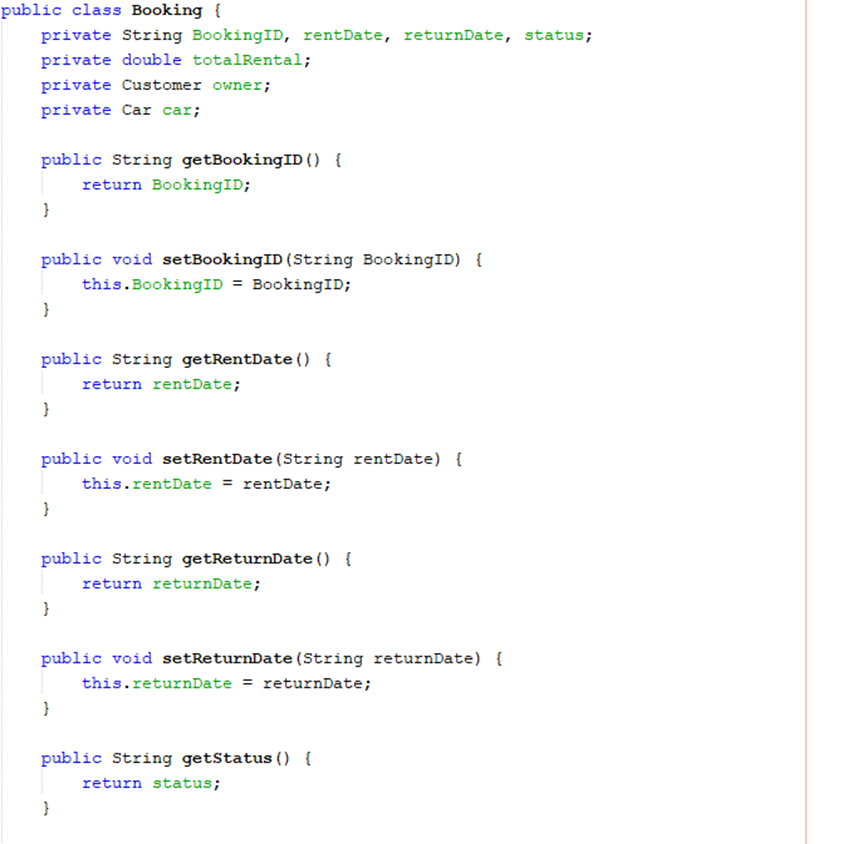


Figure ‑

Using abstraction concepts, the essential entities for a booking class, such as bookingID, rentDate, returnDate, status, totalrental, CustomerID, and carID, are depicted in the figure above. Moreover, the abstraction allows the system to get the customer id and car id only, not every attribute in the customer class and car class. Furthermore, figure above shows several methods and data are used to generate booking object which is one of the significant concepts of abstration.

## Encapsulation

Java's encapsulation feature is a potent tool for storing a class's data type and data methods together. It is done by creating a secure field that is only available to other people in the same class. In Java, encapsulation refers to the practise of combining data (variables) as well as the code that affects them (methods) into a single entity. The variables of a class are protected by an encapsulation so that only the methods of the class can use them.

The main goal of using encapsulation is for security purposes. Any external classes cannot access data members due to encapsulation. The encapsulation procedure increases the security of the data that has been encased. Moreover, encapsulation provides more flexibility on code, this is due to it is possible to successfully implement changes to one section of the code without having an impact on any other sections (S, R. A., 2022).

Graphical user interface, text

Description automatically generated

Figure ‑

In this project, the car’s attributes are set to private to protect the data.

Graphical user interface, text, application

Description automatically generated

Figure ‑

Since the attributes in the class Car are private and protected, getters and setters are designed so that other functions in this project are able to get the car attributes.

Chart

Description automatically generated with low confidence

Figure ‑

Text

Description automatically generated with medium confidence

Figure ‑

The CarID attribute from the car class is shown in the figure above as one of the attributes from the booking class. Even though the carID attribute is set as private data, by using the getter and setter functions, the booking class still manages to get the carID. If there is a case that the car class attributes are public instead of private, the CarID is not safe because it is editable while inside the booking class.

## Modularity

An autonomous software segment that communicates over an interface is more like a module. When designing a software, modularity looks at using multiple modules rather than a single old architecture. By dividing the software environment into various modules, modularity improves process efficiency and lessens the coupling impact. Functional testing on the fly while concurrent development is taking place is helpful for Java developers.

Graphical user interface, text, application

Description automatically generated

Figure ‑

Based on the figure above, a method called “autogenerateCusID” is created under “InformationIO” class. “autogenerateCusID" method is used to generate customer id in this system.

Graphical user interface, text, application

Description automatically generated

Figure ‑

Based on this project, whenever the user registers as a customer, the customer ID will be automatically generated. By utilising modularity concepts, the auto-generate-id method is able to be reused many times, which helps to sustain the stability of the project.

As mentioned before, modularity helps reusability. Utilizing the same code to create a newer version of the program or system can save developers' time. Java's modularity allows programmer to reuse the modules outside of the originally designed system. Moreover, while making modifications to specific lines of code, modularity aids in maintaining the stability of the system. It implies that the new changes can be easily tested without altering the current programme and remain concealed from users.

## Inheritance

One of the fundamental concepts of OOP is inheritance or hierarchy, which enables us to derive a new class with an existing one. The existing class from which the child class is derived is called as the superclass (parent or base class), and the newly produced class is known as the subclass (child or derived class). This type of relationship can be shown in class diagram. Subclass will inherit all variables and methods from superclass, but subclass is allowed to have new variables and methods different from superclass. The use of inheritance is to standardise the variables and methods name, so that it is easy to be invoked in client program.

Graphical user interface, text, application

Description automatically generated

Figure ‑

Figure above shows a superclass called “User” is created with attributes, accID, username, and password. There are also 6 methods in the superclass which can be inherited to subclass.

Graphical user interface, text, application

Description automatically generated

Figure ‑

Figure above shows Admin class inherits User class using extends keyword. There are no variables created in this class because all attributes are inherited from User. The constructor is created at Admin class because constructor cannot be pass to subclass from superclass.

Text

Description automatically generated

Figure ‑

Next, User class is also inherited by Customer class. There are new attributes (gender, email, address, phoneNum, myBooking, myPayment) and new methods created in this class.

Graphical user interface, application, Word

Description automatically generated

Figure ‑

Figure above shows accID, username and password are passed from superclass and they are used to create constructor in Customer class.

# Additional features

## 5.1 Date

Text

Description automatically generated

Figure ‑

In order to get the local date, “java.util.Date” is imported and “java.text.SimpleDateFormat” is to set the sequence for the date.

Text

Description automatically generated

Figure ‑

Figure above shows the code for getting a local date for variable “paydate” and the sequence of the date is is “dd/MM/yyyy”.

Text, timeline

Description automatically generated

Figure ‑

The code shown above calculates the difference between two dates, the rent date and the return date entered by the customer. Firstly, rent date and return date were assigned in the same date format. After that, a method “ChronoUnit.DAYS.between()” is used to shows the difference between 2 date.

## 5.2 Show confirm dialogue

Graphical user interface, text, application, email

Description automatically generated

Figure ‑

The figure above shows the code for creating a confirmation dialogue on the booking details page using ‘showConfirmDialog()’ method. IF else statement will also be used to control the flow of the system.

Graphical user interface, application

Description automatically generated

Figure ‑

The confirmation dialogue will ask the customer whether they want to delete the record or not. If the customer clicks the yes button, the booking record will be deleted, while if the customer clicks the no button, the system will return to the customer's home page.

## 5.3 Combo Box

Text

Description automatically generated

Figure ‑



Figure ‑



Figure ‑



Figure ‑

In order to use combo box, javax.swing.JComboBox is needed to be imported. An array called gender is created to store the string displayed by the combo box. The Combo box can be clickable or unclickable using the ‘setEnabled’ method. True parameter allows users to access the combo box, but False parameter will lock the combo box. Then, the values in the combo box can be obtained using ‘getSelectedItem’ method while the values can be set to the combo box using ‘setSelectedItem’ method.

## 5.4 Table with extra functionalities

Text

Description automatically generated

Figure ‑

Based on Figure above, table can be locked using ‘setEnabled(false)’ method. This is because the table is only for display purpose but cannot be directly edited by users. Next, the width of each columns can be adjusted using ‘setPreferredWidth()’ method. On the other hand, the vertical and horizontal scrollbar will be displayed on the table only when necessary. The scrollbar will only show up when there is data in table exceed the window. In order to create scroll pane, javax.swing.JScrollPane must be imported.

## 5.5 Better Gui

Text, letter

Description automatically generated

Figure ‑

In order to have a better GUI, a header is designed for each page. The background of the header is set with gray colour using ‘setBackground()’ method. New font is created with bold and larger font size. This font is implemented in headerLabel using ‘setFont()’ method. The colour of the label is also changed to white using ‘setForeground()’ method.

## 5.6 Pie Chart

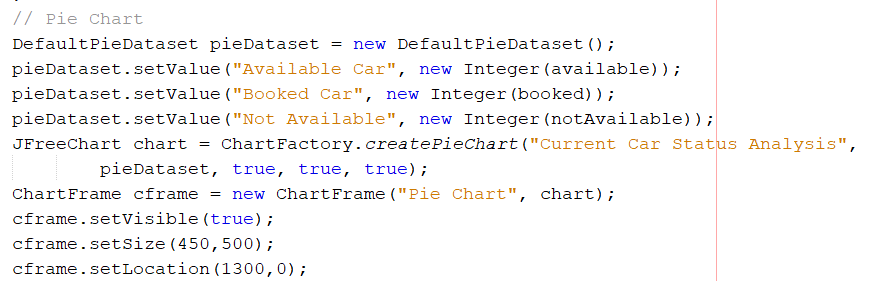


Figure ‑

Based on Figure above, DefaultPieDataset is created and stored in a variable called ‘pieDataset’. This dataset is set with values with its name. Then, JFreeChart is created with title. A frame will also be created to store pie chart to be displayed. It is set as visible, with specific size and location.

## 5.7 Auto generate ID (Extra fucntion)

Text

Description automatically generated

Figure ‑

Autogenerate ID method is created so that this method can be reused in other class. The size of an array list is obtained. If it is 0, it means no data in the text file, so the newID will set as ‘0001’ because it is the first data in the file. If there is data in the text file, the digits of the previous ID is obtained, and it is added by 1. For example, ‘C0099’ returns ‘99’ and it is added by 1 to get 100. The value of digits is checked to add corresponding number of ‘0’ in front of the digits. For example, values 100 will be added with one 0 in front of it, to get ‘0100’ and this value is stored in newID and being converted to string data type. This method will return newID with a character ‘C’ in front of it.

## 5.8 isNumeric (Extra fucntion)

Graphical user interface, text

Description automatically generated

Figure ‑

isNumeric method is created so that this method can be reused in other class. This method is used to check if the string is numeric values. If it is numeric value, it returns true else it return false.

# References

Biniasz, K. (2021, November 24). What are OOP concepts in Java? 4 primary concepts. Stackify. Retrieved December 6, 2022, from <https://stackify.com/oops-concepts-in-java/>

Hartman, J. (2022, November 5). Java Swing Tutorial: How to create a GUI application in Java. Guru99. Retrieved December 6, 2022, from <https://www.guru99.com/java-swing-gui.html>

Java Classes and Objects. Java classes and objects. (n.d.). Retrieved December 6, 2022, from <https://www.w3schools.com/java/java_classes.asp>

S, R. A. (2022, October 28). What is encapsulation in Java and how to implement it. Simplilearn.com. Retrieved December 6, 2022, from <https://www.simplilearn.com/tutorials/java-tutorial/java-encapsulation#:~:text=Encapsulation%20in%20Java%20is%20the,the%20class%20can%20access%20them>.

Team, G. L. (2022, November 16). Oops concepts in java with examples: 2023. Great Learning Blog: Free Resources what Matters to shape your Career! Retrieved December 6, 2022, from <https://www.mygreatlearning.com/blog/oops-concepts-in-java/>

Team, G. L. (2022, November 16). Oops concepts in java with examples: 2023. Great Learning Blog: Free Resources what Matters to shape your Career! Retrieved December 2, 2022, from <https://www.mygreatlearning.com/blog/oops-concepts-in-java/#:~:text=Object%2Doriented%20programming%20is%20a,reuse%20them%20without%20compromising%20security>.

vats, rohan. (2022, November 22). Modularity in Java explained with step by step example [2023]. Modularity in Java Explained With Step by Step Example [2023]. Retrieved December 5, 2022, from <https://www.upgrad.com/blog/modularity-in-java/>