North South University

Project CSE 215

Hotel Management System

Submitted By:

semester: Fall2022

Course: CSE 215

Section:7

Date of Submission: 23/12/2022

Group number: 6

Group member: Mubasshir Sadat ID:2212468642
Abu Yousuf Neshad ID: 2212517042

Name: Mehedi Hassan Roktim ID: 2212497042

Name: Md. Hasan Emam ID: 2212478042

About Our Project:

We have created an online hotel Booking System. In this system, a user can open an individual account and then give a unique password. The user can then log in to the system. We have created a system where the user can book any room and all his transactions in this hotel will be stored and can be viewed. We have also implemented a system to store the dates on which a room is booked and that room cant is booked until the checkout date. In addition, the user can update or delete his account. We have used Swing to Design our GUI.

Features:

1. Signup/Create an account
2. Login
3. Viewing Information
4. Record saving and Viewing
5. Booking Room
6. Update Information
7. Delete Account
8. Doctor Appointment

ID : 2212468642
Name: Mubasshir Sadat

Main Class:

Main Method:

Create try and catch block for Exception Handling for the createDependecy() method. In the try block call createDependecy() method. and if any exception is found in the createDependecy() method then it's handled to catch block and print Error creating dependency. Outside the try-and-catch block Create an object for the main page and pass X and Y coordinates. then call the setVisible() method by MainPage object and set true.

createDependecy() Method:

This method first creates two files named usernamelist and Booking and also creates a directory. Then checks if we have 2 files “usernamelist.txt”, and “Booking.txt” and a directory named “users”. If the files don't exist then 2 files and a directory is created.

MainPage Class:

MainPage constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (140,140,200). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and set it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “Welcome”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call the setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

In the MainPage class, we have extended JFrame and have data fields Container c, ImageIcon icon, JButton zero, signpButton, loginButton, exitButton, employeeButton, aboutButtons. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. To Create Label First, we need to Create an Object of JLabel class for the Label and declare the label name then use the object dot setFortnt() method for text font and also use the setBounds() Method to set the boundary of the txt then add with

Container. All the other labels used in the class have been done by following this method. To Create Button First We need to create a button object and declare the button name then use the object name dot SetbackGround() method and declare the color of the button then call setForeground() and declare the color of the button text, setFornt() Method to use set text font then, at last, we add it to the container. All the other buttons used in the class have been done by following this method. We have 5 buttons for logging in, signup, employee button, about buttons, and exit button. We also have an action button for all 5 buttons. Pressing the signup button will create an object for the SignupPage class and call its constructor. Pressing the Login button will create an object for the LoginPage class and call its constructor. Pressing the Employee button will create an object for the EmployeePage class and call its constructor. Pressing the About button will create an object for the about page class and call its constructor. Pressing the exit button will exit the GUI.

ActionListener() Class:

This class enables us to use the addActionListener method and actionPerformed method which enables us to allow the signpButton to identify and perform an action upon clicking.

signpButton.addActionListener() method:

This is a method within the ActionListener class allowing us to make the signpButton a listener. That is the button can now be clicked or pressed.

actionPerformed() method:

This is a method within the actionListener Class that sets the signpButton to display the login page upon pressing the button.

Similarly, we have implemented the other buttons in the class.

AboutPage Class:

AboutPage Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and set it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the About Page Class, we have data fields Container c, ImageIcon icon, JButton zero, and backButton. Inside the AboutPage Constructor, We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use the object dot setFont() method for text font and also use the setBounds() Method for setting the boundary of the txt then add with Container. All the other labels used in the class have been done by following this method. To Create Button First We need to create a button object in the first Bracket declare the button name then use the object name dot SetbackGround in the First bracket create a color object and declare the color format then setForeground in the first bracket declare the color of the button name and use setFont Method to use set text font, use setBounds() Method for setting the boundary of the txt .then, at last, we add with the container. For performing the back button we need to call actionlistener() method in the method we need to call another method name actionPerformed() that method receives the ActionEvent class object in the actionPerformed() method to create a variable of point and calls getLocation() method the store x and y value into the x and y variable the pass x and y value into mainPage.If the user clicks the back button it will go to the main page.

Username Class:

Default Username constructor:

We have a default in this class.

Parameterized Username constructor:

We have a parameterized constructor which takes the string type variable and sets it to the username datafields.

write() method:

Inside the write method, we first open an object for the file for the usernamelist.txt file. If the file doesn't exist then we will have to open a new file with that name. Then we will create an object for PrintWriter Class and then create an object for BufferedWriter and FileWriter class. Then we will write the username inside the file and then close the file. We will open the File object inside a try-and-catch block to monitor and resolve any error.

exists() Method:

Inside this method, we will create an object for the BufferedReader class and then open the usernamelist.text file. We will have a temp variable and read a line from the text file and store it inside the temp variable. Then we will compare the given username with the temp variable by a while loop using a equals() method. If both are the same we will return true else we will return false.

get() method:

The getter method to access the private data member username.

User Class:

The user class implements the Serializable Interface. Inside we have
private String username;
private String firstName;
private String lastName;
private String email;
private String phone;
private String address;
private String dateOfBirth;
private String password;
private double balance;
private String price;
private String date;
private String duration;
private String view;
private Vector datafields.

Default User constructor:

We have aded a default constructor to call the class.

Parameterized User constructor:

WE ha a parameterized costructor to initialize all the datafields.

saveFile() Method:

The savifile method opens the users.sat file and writes the inormation of a user.

deleteAccount() Method:

We have a deleteAccount() Method to delete the user.dat file by calling the removeUsernameFromList() method. If the account cant be deleted than an exception is thrown.

removeUsernameFromList():

This method at first opens a new temp file and start reading the usernames from usernamelist.txt file and writes them in the temp file as long as the username of the account to be deleted doesn't match with the username from the txt file. Then tboth the files are closed. Then the usernamelist.txt file is deleted using userFile.delete() method.Then the temp file is renamed as usernamelist.txt file.

The Userclass also has getter setter methods.

ID : 2212517042
Name: Abu Yousuf Neshad.

SignupPage Class:

In the Signuppage class, we have Container Jlabel JtextField Jpasswordfield Jbutton and buttonGroup type data field.

SignupPage constructor:

We have a constructor Inside the constructor we design the Gui and create a form with labels and text fields. We have three buttons the signupbutton Clearbutton and the cancel button.

signpButton.addActionListener() method:

This is a method within the ActionListener class allowing us to make the signpButton a listener. That is the button can now be clicked or pressed. Inside the signup button addActonListener method, we have set all the entered info from the text field inside local datafields We have a pattern matcher for email. Email must have text before and after @ and also text before and after dot and dot will come after @. We also have a condition to check the date of birth with the current date to prevent the date of birth after or at the current date. If all the entered information is valid then we enter inside the if condition and then call the exists method to check if the username already exists inside the usernamelist.txt. If it doesn't exist then we call the constructor for the User class and send all the parameters and a showMessageDialog() method shows success. Then we create an obj for the main class and call it thus returning to the main page. If an error is in the information then an error label is shown. Inside the clearbutton addActionListener() method we set all textfields to empty.
CancelButton will send us to the mainpage.

clearButton.addActionListener() method:

if the user clicks a clear button then all the text filed set empty.

cancelButton.addActionListener() method:

This is a method within the ActionListener class allowing us to make the cancelButton a listener. That is the button can now be clicked or pressed. Inside the cancel button addActonListener method, then we create an object of the main page when user clicks cancel button the main page visible.

LoginPage Class:

We have various types of datafields name container, ImageIcon, Jlabel, JTextField, JPasswordField, and JButton.

LoginPage constructor:

Inside the login page constructor, we first design the GUI like previous classes. We create a login GUI with username and password textfields. We also have 2 buttons the login and cancelbutton.

loginButton.addActionListener() method:

Inside the loginbutton addActionListene() method we have a condition where if we press the wrong type of info 4 four times then it will create a method for the Holdpage and call its constructor. Inside it, we first check if the username exists inside the usernameliist.txt file and then match the password from the corresponding .dat file if both are ok then an object for Userpage is Cretard else if they are not matched then an errorlabel is shown and the value of count is increased by 1.

cancelButton.addActionListener() method:

This is a method within the ActionListener class allowing us to make the cancelButton a listener. That is the button can now be clicked or pressed. Inside the cancel button addActonListener method, then we create an object of the main page when the user clicks the cancel button the main page is visible.

UserPage Class:

Inside the userpage we have JTabbedpane JTable JScrollPane Container ImageIcon JPanel JLabel JPasswordField JButton boolean a type data fields.

UserPage constructor:

we have designed a GUI as we did in other classes. We have created an object for the tabbed pane. We have created panels and added icons to them.
Inside the **home panel**, we show all the info of the account creator. Inside the **history panel**, we create a table, and inside the table all the actions performed will be shown.

roompanel we have created 3 buttons named SeaviewOk, HillviweOk, Regularok.

SeaviewOk.addActionListener(new ActionListener():

pressing rooomseaview button will create an object for SeaView Class and call its constructor.

HillviewOk.addActionListener(new ActionListener():

pressing the rooomHillview button will create an object for Hillview Class and call its constructor.

RegularOk.addActionListener(new ActionListener():

pressing rooomregularview button will create an object for RegularView Class and call its constructor.

Inside the food panel, we have created a button named food court.

foodCourtOk.addActionListener(new ActionListener():

pressing the food court button will create an object for Foodcourt and call its constructor.

Medical panel:

We have created a button named Doctor appointment, pharmacy, and Ambulance.

ambulanceok.addActionListener(new ActionListener():

pressing the ambulance button will create an object for doctor Class and call its constructor.

farmacyok.addActionListener(new ActionListener():

Pressing the pharmacy button will create an object for Pharamavcy class and call its constructor.

DoctorOk.addActionListener(new ActionListener():

The doctor button pressing will create an object for the Doctor class and call its constructor.

The update Date panel has been designed like a form with labels and textfields and a updatebutton and a refresh button.

updateButton.addActionListener(new ActionListener():

Pressing the update button will first store all the data from the text field inside the variable and if the given info doesn't match with the previous then we will call the change() methods inside the use CLass and update them. Else if there are errors then an errorlabel will be shown.

refreshButton.addActionListener(new ActionListener():

The refresh button will call the Userpage class again.

delete panel we take the password and have a delta button that will check if the password matches. If the password matches then it will call the deleteAccount() method and then return to the mainPage

BookingDate Class

Inside the class, we have integer-type data field and create two objects for the vector class. one object name is vc_Checkin and another one is vc_room.

availableRoom() method:

A method whose return type is boolean named availableRoom and this method receives check-in date and Checkout Date. We Create a try-and-catch blocks for exception handling for file reading. Inside the Try block create an object for BufferedReader class in the first bracket we create another object of fileReader class and declare the name of the text file. We Declare a String type variable name temp. we run a while loop to read line by line from the file and the lines are stored inside the temp variable this loop run till it finds null. In the loop, we check if the line equals to Checkin date or Checkout Date and return false otherwise, return true. that is check if the user-selected room is available or not. If any exception occured Inside the try block, it goes to Catch Block, and that catch block prints Error Occured reading Booking! And exception Object.

BookingWrite Class

BookingWrite constructor:

The constructor receives as an argument String type variable name check-in date, long type variable name diff, and String Type Variable name room. Create a try and Catch block for Exception handling for the file. Inside the try, the block creates a file object to create a new file name Booking. Then We check whether creating the file already exists or not. If the file does not exist, then create a new file. Create an object for the printWriter class to write into the file in the first bracket of the class we create a FileWriter class and the First Bracket of the File Writer class declares the text file name and that file opens in the true mode because of what we write anything into the file, write must add into the last line of the file, and previous data does not remove into the file. we run a while loop that loops this loop run till daysdiff. Inside the while loop, we create an object of SimpleDateFormat class into the First Bracket of the class and declare the "yyyy-MM-dd" format of a date. Create a variable of Calendar and call calendar dot getInstance() method. Inside the try and catch block, call the setTime() method to set CheckIn date. then declare cal. add() method to add dates one by one check in date to checkout date.at last, we close the file.

ID:2212478042
Name:Md. Hasan Emam

SeaView Class:

SeaView Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the Seaview room we have some datafields. datafields are Container, ImageIcon, zero, backButton, bookingButton. inside the page also have a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create an Object of ImageIcon class in the First Bracket we call getclass() method dot getResource("images/SeaView3.png") method for add icon location. To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot SetbackGround() method for Button color then setForeground() method to set the button text color and use setFont Method for set text font then call setBounds() method to set the boundary of a button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use the object dot setFortnt() method for text font and setBounds() Method for setting the boundary of the txt then add with Container. For the performing button we need to call actionlistener() method in the method we need to call another method name actionPerformed() that method receives the ActionEvent class object in the actionPerformed() method to create a variable of point and call getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the back button it will go to the user page. In the same way to create a booking button and the values are passed into the booking Page if the user clicks the Booking button go to the booking page.

Hillview Class:

Hillview Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call the setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the Hillview room, we have some data fields,datafields are Container, ImageIcon, zero, backButton, bookingButton. inside the page also have a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create an Object of ImageIcon class in the First Bracket we call getclass() method dot getResource("images/SeaView2.png") method to add icon location. To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot SetbackGround() method for Button color then setForeground() method to set the button text color and use setFornt Method for set text font then call setBounds() method to set boundary of button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use the object dot setFortnt() method for text front and also use the setBounds() Method for setting the boundary of the txt then add with

Container. For the performing button we need to call actionlistener () method in the method we need to call another method name actionPerformed() that method receives the ActionEvent class object in the actionPerformed() method to create a variable of point and calls the getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the back button it will go to the UserPage.In the same way create a booking buttonbutton and the values are passed to the booking Page if the user clicks the Booking button go to the booking page.

RegularView Class:

RegularView Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the **RegularView** room, we have some data fields, data fields are:Container, Image**Iular**V**i**con, zero, backButton, and bookingbutton. inside the page also have a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create an Object of ImageIcon class in the First Bracket we call getclass() method dot getResource("images/SeaView2.png") method for add icon location. To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot SetbackGround() method for the Button color then setForeground() method to set button text color and use setFont Method for set text font then call setBounds() method to set the boundary of a button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use the object dot setFortnt() method for text font and also use the setBounds() Method for setting the boundary of the txt then add with Container. For the performing button we need to call actionlistener () method in the method we need to call another method name actionPerformed() that method receives the ActionEvent class object in the actionPerformed() method creates a variable of point and calls the getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the back button it will go to the UserPage.In the same way to create a bookingButton button and the values passed into the booking Page if the user clicks the Booking button go to the booking page.

Pharmacy Class:

Farmacy Conostractor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and error front.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the Pharmacy room, we have some data fields, data fields are:Container, ImageIularVicon, zero, back button, and booking button. inside the page also have a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create an Object of ImageIcon class in the First Bracket we call getclass() method dot getResource("images/SeaView2.png") method for add icon location. To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot SetbackGround() method for Button color then setForeground() method to set button text color and use setFont Method for set text font then call setBounds() method to set the boundary of a button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use the object dot setFortnt() method for text font and also use the setBounds() Method for setting the boundary of the txt then add with Container. For the performing button we need to call actionlistener() method in the method we need to call another method name actionPerformed() that method receives ActionEvent class object in the actionPerformed() method create a variable of point and call getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the backbutton it will go to the UserPage.

Doctor Class:

Doctor Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null. we call the setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the Doctor class, we have some datafields,datafields
are Container, ImageIcon, zero, backButton, ConfirmButton, PriceLabel, PriceError, PriceText, and a string answer. inside the page also has a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create a Object of ImageIcon class in the First Bracket we call getclass() method dot getResource("images.png") method for add icon location.To the same way we add all ImageIcon.To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot

SetbackGround() method for Button color then setForeground() method for set the button text color and use setFont Method for set text font then call setBounds() method to set the boundary of a button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for Label in the first bracket declare the label name then use the object dot setFortnt() method for text font and the setBounds() Method for setting the boundary of the txt then add with Container. To the same way create all labels.For the performing button we need to call actionlistener() method in the method we need to call another method name actionPerformed() that method receives the ActionEvent class object in the actionPerformed() method to create a variable of point and calls getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the back button it will go to the UserPage.In the same way create a confirmButton button and the values pass into the booking Page if the user clicks the confirm button go to the userpage.

ID :2212497042
Name: Mehedi Hassan Roktim

Ambulance Class:

Ambulance Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont.We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name “About this project and us”.setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call the setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

Inside the Ambulance class, we have some datafields, datafields are Container, ImageIcon, zero, backButton, bookingButton. inside the page also have a constructor. Inside the constructor, We have created a GUI by following using swing. We have used methods such as setTitle, setBounds, setResizable, setBackground, etc to design the interface. We have created objects for various image icons and used them. Labels have also been used inside the GUI. Create an Object of ImageIcon class in the First Bracket we call getclass () method dot getResource("images/SeaView2.png") method for add icon location. To Create Button First We need to create a button object in the first Bracket of JButton class declare the button name then use the object name dot SetbackGround() method for Button color then setForeground() method to set the button text color and use setFont Method for set text font then call setBounds() method to set the boundary of a button and at last we add with the container. All the other buttons used in the class have been done by following this method. To Create Label First, we need to Create an Object of JLabel class for the Label in the first bracket declare the label name then use object dot setFortnt() method for text fornt and setBounds() Method for setting the boundary of the txt then add with Container.For performing button we need to call actionlistener() method in the method we need to call another method name actionPerformed() that method receive ActionEvent class object in the actionPerformed() method creates a variable of point and call getLocation() method the store x and y value into the x and y variable the pass x and y value into UserPage.If the user clicks the back button it will go to the UserPage.

BookingPage Class:

BookingPage class, we extend the JFrame class because of the swing. we have container icon JButton, JPanel JTextfields JLabel JRadioButton, ButtonGroup, and some String type data fields.

BookingPage Constructor:

For the constructor, we first design the GUI like other classes. We have the labels text fields and the radio button.

totalamoutButton.addActionListener(new ActionListener():

Create 3 types of String data fields view,checkindate,checkoutdate. When the user selected any room type then it will store it into the room variable and check-in date and check-out date store the check-in date and check-out date from its text fields by using the getText() method. To find the duration of the day by using ChronoUnit.DAYS.between() method. calculate the price by checking the room type the user has given.

ConfirmButton.addActionListener(new ActionListener():

Create 5 types of String data fields room, check-in date,checkoutdate, price, view, room, and two types of boolean data filed when the user selected any room then it will store in the room variable. In using the Concat function, we will join the checkindate with room number and checkout date with room number and then call availableroom() method which will return a boolean type to show if the room is available or not. We use the after() method to check if the given check-in date or checkout date is after the current date. Then we check if all the entered data are valid or not from the if else condition. If the room is available and data are valid, we call the BookingWrite class by creating an object and calling its constructor and then send the check-in date,daysDiff, and room and create an object for the user page class and call its constructor. If any data is invalid or the room is unavailable, then we Show the corresponding error label.

clearButton.addActionListener() method:

if the user clicks a clear button then all the text filed set empty.

EmployeePage Class:

EmployeePage Constructor:

We received x and y coordinates as parameters then we create color objects and set the RGB color format (133,172,233). Two font variables to use labelFont and errorFont. We call setDefaultCloseOperation() and put it to EXIT_ON_CLOSE which causes the program to terminate when the close button at the top is clicked. Then we call the setTitle() method to set the title name "About this project and us".setBounds() method to set the size of the container. We call the setResizable() method to ensure that a graphical interface looks how you intend to prevent the user from re-sizing it.getContentPane() method sets the contentPane Property and stores its container object. Set the container layout manager to null.we call setBackground() method to set the color of the Background for the set Image Icon we call the ImageIcon() method

We have only designed an Employee page Class and designed a GUI and have yet to add any functionality to the page.

Application of the project:

The project is an online hotel management system. This project can be used to open a user account with a unique password and then book a room. The records are stored and can be viewed later. There are also other things that can be done in the hotel management software such as updating and deleting accounts. There are also food and medical panel that can be used.

Limitations:

Our project has some limitations such as we don't have any administrative classes so the number of rooms and the price of those rooms cant is changed. The owner acnt also check which rooms are booked or free or for how to long the room are booked. In addition we have implemented a system to check the validity of the date of birth by checking if the date of birth given is before the current date so if any user gives any date of birth such which is less than the current date by 1 day then it will be valid which is not right since a 1 day old should not be able to open an account. In addition, our account delete option has a problem too since we only check the password so if more than 1 account has the same password then the account cant is successfully deleted. We also don't have a system to cancel a given booking.

Future Plan:

Due to time limitations, we were not able to fulfill all our plans regarding this project. We have some unimplemented empty classes such as the EmployeePage class Ambulance Class, and Farmacy Class which we want to implement in the future. We were only able to design the Foodcourt and only show the menu and were not able to implement the order system which we want to finish. We also want fully implement an administrative section in the system where the owners and employees can add or remove rooms, change the price of rooms, and also check which room is booked and for how long and which rooms have no booking.