Implemented Methods Description

Our project is made up of 21 different methods that are divided into 7 classes. Firstly, the abstract Person class contains the signUp and login methods. The signUp method allows the person to enter personal information and create an account. Meanwhile, the login method allows the person to log in to his existing account. Secondly, the canLogIn interface class contains the validUserName and validPassword methods. Both of them are responsible for making sure that the username and password meet the conditions. Thirdly, the Employee class is made of four methods, the checkSalary, checkEmployeeType, and the implemented methods from the interface canLogIn. The checkSalary method checks the amount of salary for each employee depending on the employee's work hours. However, the checkEmployeeType method checks the type of the employee depending on the employee's department ID. Next comes the Customer and SpecialCustomer classes which contain the checkCustomerType, checkEmailType, increaseNumberOfVisits, decreaseNumberOfVisits, checkCustomer, and the implemented methods from the interface. The checkCustomerType and checkEmailType methods are responsible for checking if the customer is a student or a normal customer. In addition, the rest of the three methods are made for checking the number of visits and knowing whether the customer is special or not. Last but not least, the Booking class and Payment class methods work together for managing the booking and payment process. The Booking class methods are the reservationConfirmation, sendNotification, cancelBooking, and bookService methods. The first and second methods are for confirming the customer’s reservation(booking) and sending notification after that. Furthermore, the cancelBooking method cancels the customer’s booking after asking him/her, and the bookService method books any service chosen by the customer. On the other hand, the Payment class contains the checkService, displayPayment, checkPrice, calculateDiscount, refundPayment, and printRipt methods that check the price for each chosen service and then calculate the total by subtracting the discount amount if the customer was special or was a student. And finally printing the receipt or returning the paid amount after a customer cancels his/her booking.

- Please check the following tables for more details:

Class Name	*Method Name*	*Description*
Person	signUp	The signUp method allows the person to enter personal information and create an account.
Person	login	The login method allows the person to log in to his account.
canLogIn	validUserName	This method is implemented in the Employee class and Customer class.
canLogIn	validPassword	This method is implemented in the Employee class and Customer class.
Employee	checkSalary	This method checks the amount of salary for each employee depending on the employee's work hours.
Employee	checkEmployeeType	This method checks the type of the employee depending on the employee's department ID.
Employee	validUserName	This method is implemented from the interface canLogIn class. It checks if the user name is valid and then returns true if the name is valid and false if not.
Employee	validPassword	This method is implemented from the interface canLogIn class. It checks the password and then returns true if the password length is greater than 8 digits and false if it’s less than 8 digits.
Customer	checkCustomerType	This method checks the customer type.
Customer	checkEmailType	This method checks if the email is a student email or not.
Customer	validUserName	This method is implemented from the interface canLogIn class. It checks if the user name is valid and then returns true if the name is valid and false if not.
Customer	validPassword	This method is implemented from the interface canLogIn class. It checks the password and then returns true if the password length is greater than 8 digits and false if it’s less than 8 digits.

Class Name	*Method Name*	*Description*
SpecialCustomer	increaseNumberOfVisits	This method increases the number of customer visits by 1.
SpecialCustomer	decreaseNumberOfVisits	This method decreases the number of customer visits by 1.
SpecialCustomer	checkCustomer	This method to check if a customer is special or not by checking the number of visits.
Booking	reservationConfirmation	This method is for confirming the customer's reservation.
Booking	sendNotification	This method sends a notification to a customer after booking.
Booking	cancelBooking	This method cancels the booking.
Booking	bookService	This method allows the customer to choose and book a service.
Payment	checkService	This method checks the chosen services.
Payment	displayPayment	This method allows the customer to pay by credit card or cash.
Payment	checkPrice	This method returns the chosen services prices.
Payment	calculateDiscount	This method calculates the total discount for special customers and students.
Payment	refundPayment	This method displays the refund of the payment when the customer cancels the booking.
Payment	printRipt	This method prints the final receipt to a customer with all the needed details.

User Manual

Employee version

Welcome to our SPA!

Using our spa management system as an employee is very simple. All you need to do is log in with your name, username, email, and password given to you on your first day at work and then enter your employee ID to access your department. Next, enter the hours of work that you want to work for. Finally, enter how many years have you been working with us. Your data should be saved now and your amount of salary will be displayed on the screen for you to see. Thank you for working with Calm Wind Spa!

- Please read the following instructions carefully for the best experience:

- **Stage1: Pick your account type:**

  1- Employee Account

  Log in using your account information given at work.

- **Stage 2: Department ID:**

  Enter your department ID number.

- **Stage 3: Hours of work:**

  You can choose to work for 4 - 8 - 12 hours.

- **Stage 4: Experience Years:**

  Enter your years of experience (for how many years have you been working in this job).

- **Stage 5: Final output:**

  Your personal information and salary will be displayed on the screen.

User Manual

Customer version

Welcome to our SPA!

Using our spa management system as a customer is very simple. All you need to do is sign up for a new account or log in to your existing account. After that, you will get full access to our services menu and you can choose the services that you prefer or want to book. Next, enter the day, month, and year you would like to book in. Finally, you could choose the credit card payment method and pay smoothly through our program or the cash payment method and pay to our reception directly. However, if you change your mind and decide to cancel your booking, no worries! You can easily do that by typing "Yes" to our cancel booking method. But if you confirm your booking by typing "No" you will receive a notification that confirms your booking with the reservation ID. Now your data and booking details should be saved and your reception with all necessary information will be displayed on the screen for you to see. Thank you for using the Calm Wind Spa management system!

- Please read the following instructions carefully for the best experience:

- **Stage1: Pick your account type:**

  2- Customer Account

  Log in or sign up for a new account.

- **Stage 2: Check Services:**

  After accessing successfully, The services menu will be displayed on the screen.

  You can choose an unlimited number of services!

- **Stage 3: Book an appointment:**

  Enter the day, month, and year that you wish to book in.

- **Stage 4: Payment methods:**

  We offer payment by credit card or cash.

  Choose the payment method that suits you most.

- **Stage 5: Cancel booking:**

  You will be asked if you want to cancel your booking or not.

  1- Yes, the refund of your payment will be displayed on the screen.

  2- No, you will receive a notification that confirms your booking with a reservation ID.

- **Stage 6: Final output:**

  Your personal information and reception will be displayed on the screen.

Team Work

We worked on this entire project together, by keeping up with each other's work and arranging about 3-4 meetings each week. First of all, we worked on the UML as a team and we exchanged ideas and opinions. Next, we divided the classes between us and started to write the first version of the code. Then we gathered them together and coded the main and modified it together. At last, we tried to learn as much as we could about the Graphic User Interface (GUI) and implemented it in our project as extra work from outside the course.

Graphic User Interface (GUI)

First, we created the Frame from the JFrame class, and then we designed the interface according to our needs using the tools in the same user interface class. We created a basic interface class called JF1 that contains two JButton tools, one for the customer class and the other for the employee class. Next, we used the properties of each tool to change colors and add texts and fonts. Then we started creating another window and we linked it to the first main window by defining a method of type setVisible. After that, we named the second window FEmployee where the employee can enter his/her login information. This window contains 3 basic tools: type TextFiled to enter the name, username, and email, type Password Filed to enter the password and the last one is type JButton to move to the next page. In the last step, we created the last window for the employee class and linked it to the previous window. This window contains some questions to determine the employee's salary according to his experience and working hours. Therefore, the information for this employee is displayed on the screen. We used 4 tools in this window, the TextFiled, combo box, text area to show the final information, and the spinner. Then we linked them all to the source code. Also, we modified the source code for the same interface according to our needs. In the last window, we added a switch that works If the employee enters his department ID number, then his information will be displayed. and we have added an if statement to specify the employee's salary according to his job. One of the problems that we faced, was the Combobox tool, which shows us the option that we have chosen as an Object and we need it as a String, so we had to use Casting to solve this situation.