Tamanna Sharma and Joy Jeong: Title Page

Island Hotel

**Programmers:**

* Thapana Berpan
* Juhee Kim
* To Yin Yu
* Joy Jeong
* Tamanna Sharma
* Francesco Ward

**Course Number and Name:**

CS& 141: COMPUTER SCIENCE I JAVA

**Date:**

8/25/2016

Tamanna Sharma and Joy Jeong: Analysis

Project Description

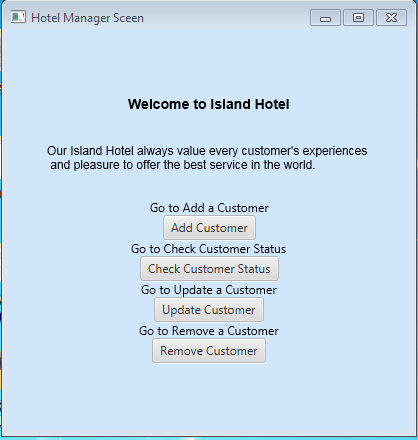
This program will allow the manager of Island Hotel to manage the information of customers in the Hotel. He/she will be able to add customers, and delete customers along with their information from the database. The manager can see the information of each customer or all the customers. The manager can also see the total amount the customer had to spend. The manager can also update information of the customers in the Hotel database. In general, the manager can update or see the name of the customer, the room number customer is staying, the number of nights the customer stayed, and the cell-phone number.

User Stories

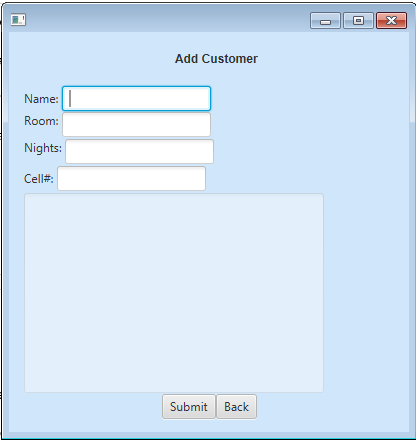
* The user will be able to add, delete, update and check on the status of up to 10 customers (10 rooms) in the hotel.
* By going to the add customer screen, the manager can add a person with their name, room assignment, cellphone number and number of nights they wish to spend.
* The manager can enter a room number and get all customers associated with that room or enter a name and room number to calculate the cost of the person or display all customers in the hotel.
* The manager can remove a specific customer or all customers in the hotel. They can also update the person’s info such as cellphone number, room assignment and nights booked.
* The manager can also see the information of the customer by clicking on check customer status. Here, if the manage clicks on display all button, the manager can see the name of the customer, room in which customer stayed, number of nights he or she spent, and their respective cell-phone number in the form of a list. The manager can also see the number of total customers, and the number of rooms still available

StoryBoards

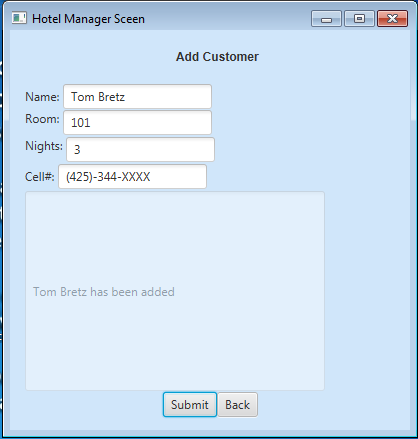
When the manager opens the program, he will see this window, which gives him the name of the Hotel that is with a welcome. Also, the manager will see the hotels official quote. The manager can see different buttons that are Check Customer Status, Add Customer, Remove Customer, and Update Customer. The manager can click on anyone of them according to task he or she wants to do.



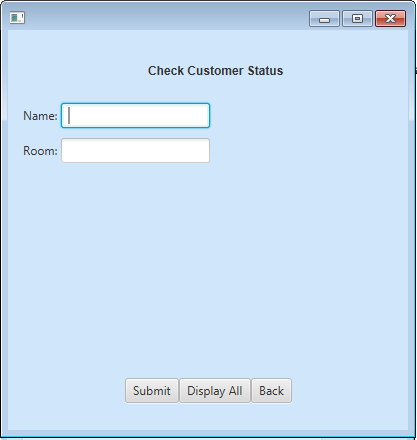
When the manager clicks on Add Customer button, he or she can input the customer’s name, the room number he or she stayed, the number of nights spent in the hotel, and the cell phone number of the customer. The manager can also see Submit button to submit the information, and the Back button to go back to the previous screen.



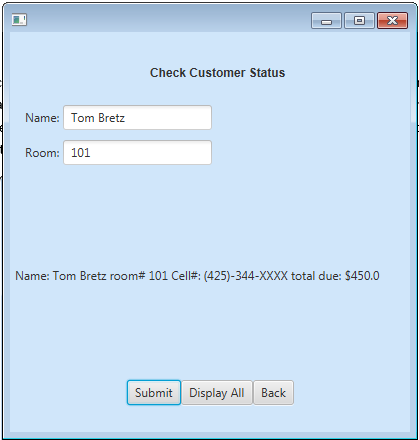
When the manager fills out all the information and clicks on Submit button, he can see a confirmation that the customer is added to the list.



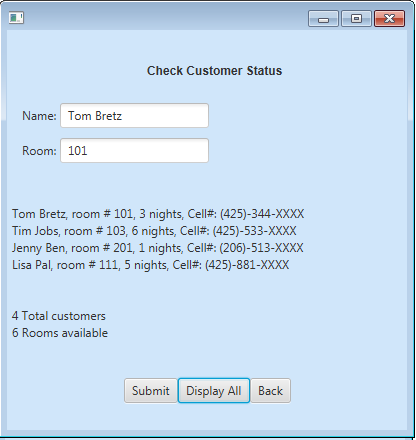
If the manager clicks on Check Customer Status, he or she have to enter both name and room number they want to find. The manager can also use other buttons such as Submit button to submit the name and room number he or she wants to find, Display All button to see all the customers in database, and the Back button to go back to previous screen.



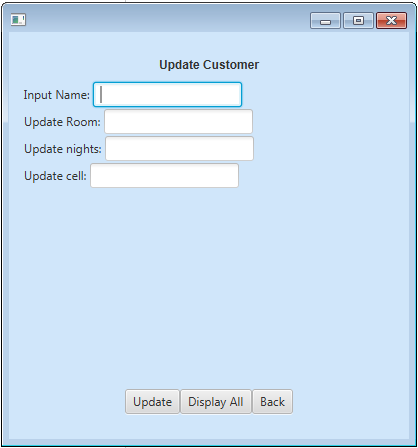
This is how the screen will look if the manager clicks on submit button after putting the name and the room number he or she wants to find. The information includes the name of the customer, the room he or she stayed, the cell phone number and the amount that was due for the customer.



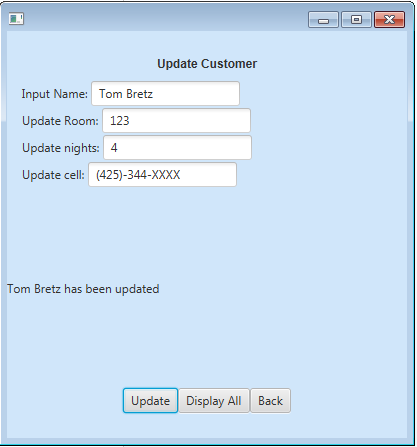
If the manager clicks on Display All button, he or she can see the information of all the customers along with numbers of customer in the list and number or rooms still available in the hotel.



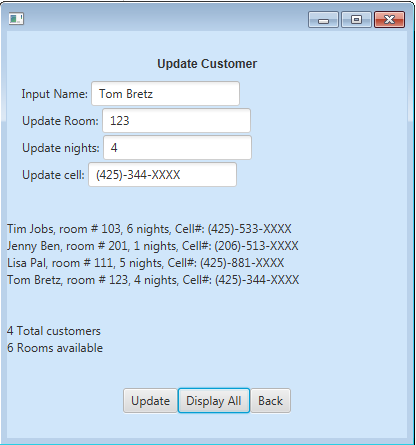
When the manager clicks update customer button, he or she can see this screen where he or she have to input the name, room number, number of nights, and update cell. He or she can use the Update button, Display All button and Back button



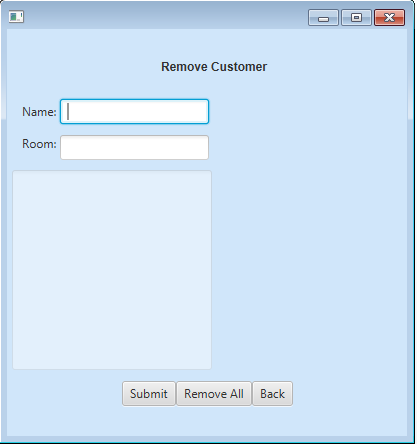
After inputting all the information, the message would be displayed to the manager.



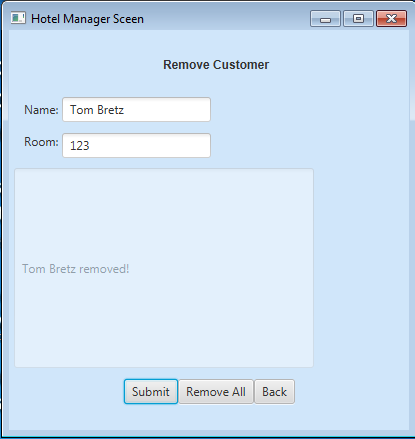
If the user clicks on Display All button, he or she can see the updated information of the customers.



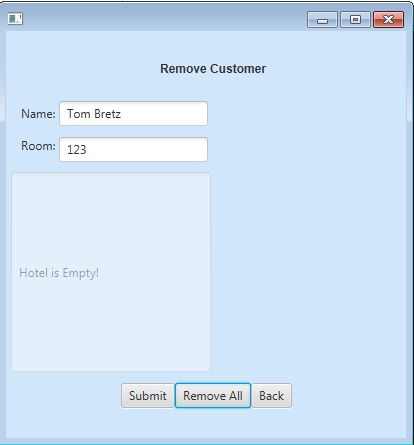
If the manager clicks on Remove Customer button, he or she can see this window, where he or she have to insert the name and the room number of the customer. They can also use features such as Submit, Remove All, and Back button.



After the manager clicks on Submit, the message will be displayed informing that the customer has been removed.



After Clicking on Remove All button, the manager is notified that the rooms in hotel are empty for customers.



Thapana Berpan and Francesco Ward: Design

Class Diagrams

Person

---------------------------------------------------------------------------------------------------------------------

- customerName: String

- cell: String

- howLongTheyStay: int

- roomNum: int

---------------------------------------------------------------------------------------------------------------------

+ Person(customerName: String, cell: String, howLongTheyStay: int, roomNum: int)

+ getCustomerName(): String

+ getCell(): String

+ getHowLongTheyStay (): int

+ setCell(cell: String): void

+ setHowLongTheyStay(howLongTheyStay: int): void

+ setRoomNum(roomNum: int): void

+ setCustomerName(customerName: String): void

+ hashCode(): int

+ equals(obj: Object): int

+ compareTo(person: Person): int

AppModel

---------------------------------------------------------------------------------------------------------------------

+TOTAL\_ROOMS: int

+FIXED\_CHARGE: double

- CustomerList: ArrayList<Person>

- roomsAvailable: int

---------------------------------------------------------------------------------------------------------------------

+ AppModel()

+ getCustomerList(): ArrayList<Person>

+ setCustomerList(CustomerList: ArrayList<Person>): void

+ getNumberOfCustomers(): int

+ getRoomsAvailable(): int

+ checkCustomer(name: String, room: int): Boolean

+ addCustomer(name: String, room: int, nights: int, cell: String)

+ updateCustomer(name: String, room: int, nights: int, cell: String)

+ removeCustomer(name: String, room: int) String

+ removeAll(): void

+ displayEquals(name: String): String

+ retrieveCustomersFromRoom(room: int)

+ displayCustomersInArray(cust: String[]);

+ displayAll(): String

+ writeCollection(): boolean

+ readCollection(): boolean

AppView

---------------------------------------------------------------------------------------------------------------------

-submitAdd: Button

- submitCheck: Button

- displayAll: Button

---------------------------------------------------------------------------------------------------------------------

+ start(stage: Stage)

+ loadMainScreen(): VBox

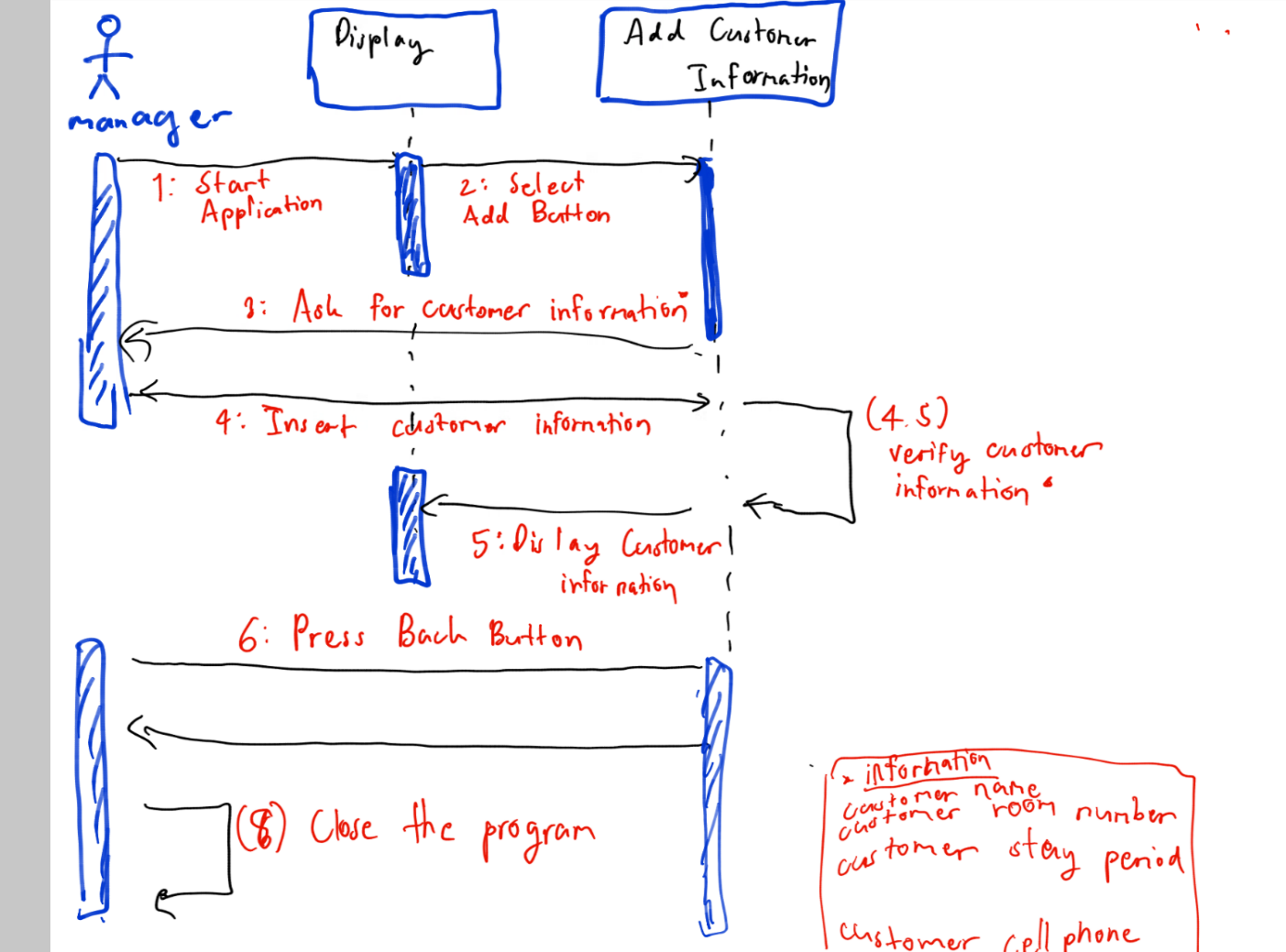
+ goToCheckCustomerStatus(): VBox

+ goToAddCustomer(): VBox

+ goToRemoveCustomer(): VBox

+ goToUpdateCustomer(): VBox

Sequential Diagram



Juhee Kim and To Yin Yu :Coding

**Brief Description of Coding Process**

* Coding Language: Java
* GUI platform: JavaFX
* Software Architecture: Netbeans IDE, MVC Architecture. OOP with classes
* Programming paradigm: Pair programming which is part of Agile programming. Splitting into pairs to complete specific sections of the Project.

AppModel Class: Thapana Berpan, Francesco Ward

AppView Class: Joey Jeong, To Yin Yu

Person Class: Juhee Kim, Tamanna Sharma

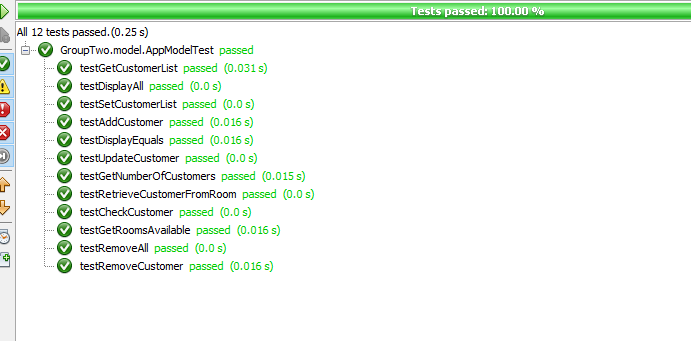
To view methods and code please refer to the Netbeans Project. To view documentation please refer to the zipped Javadocs uploaded with the project.

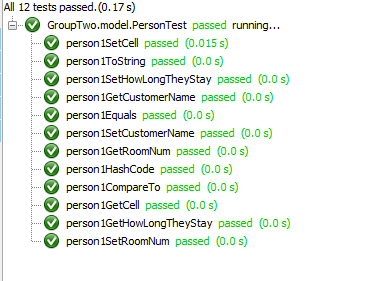
**Testing**

# Junit Testing

Junit testing coverage all method concerning AppModel and Person class. But the test with the most cases is a non netbean generating method that we not modified and its concentrated in AppModel class.

The testing has coverage of cases concerning if the input is null for variable. If the object is equal or not. If the customer have the same underline value or not. If the customer have been remove or not. If the customer is in the same list all not.





GUI Testing

|  |  |  |
| --- | --- | --- |
| Action | Expected Result | Success |
| When the user enters letters instead of numbers in the room and nights.  (Go to Add Customer) | It gives warning to the user telling them to enter valid numbers for room and night. | Yes. |
| When the user enters letters instead of numbers in the cell or when the user enters nothing in the cell or when the user enters negative numbers in the cell.  (Go to Add Customer) | It gives warning to the user telling them to enter valid numbers for cell. | No. |
| When the user does not enter anything in the name.  (Go to Add Customer) | It gives warning to the user to enter name. | Yes. |
| When the user enters nothing in the room and nights.  (Go to Add Customer.) | It gives warning to the user to enter valid numbers for room and nights. | Yes. |
| When the user enters negative numbers in room and nights.  (Go to Add Customer.) | It gives warning to the user that they have entered invalid numbers. | Yes. |
| When the user enters letters instead of update Room and Update nights.  (Go to Update Customer) | It gives warning to the user telling them to enter valid numbers for room and nights. | Yes. |
| When the user enters letters instead of numbers in update cell.  (Go to Update Customer) | It gives warning to the user telling them to enter valid numbers for cell. | No. |
| When the user enters nothing in the Input Name or enters the name which is not checked in.  (Go to Update Customer) | It gives warning to the user that customer was not found. | Yes. |
| When the user enters negative numbers in Update Room, Update nights and update cell.  (Go to Update Customer) | It gives warning to the user that the user has entered invalid numbers. | No. |
| When the user enters letters instead of numbers in room.  (Go to Check Customer Status) | It gives warning to the user telling them to enter valid room number. | Yes. |
| When the user enters the customer name who has not checked in.  (Go to Check Customer Status) | It gives warning to the user telling that the customer entered is NOT in the db. | Yes. |
| When the user enters nothing in the customer Name.  (Go to Check Customer Status) | It gives warning to the user to enter name of the customer. | No. |
| When the user presses the button ‘Display All’.  (Go to Check Customer Status) | It displays all the customers who are staying in the hotel and their information. Also displays total customers and rooms available. | Yes. |
| When the user wants to remove the customer name who has not checked in.  (Go to Remove Customer) | It gives warning to the user telling that Customer is not found. | Yes. |
| When the user enters the letters instead of numbers in room or when the user enters nothing in the room.  (Go to Remove Customer) | It gives warning to the user telling them to enter a valid room number. | Yes. |
| When the user presses the button ‘Remove all’.  (Go to Remove Customer) | It removes the all the customer’s information and tell the user that hotel is empty. | Yes. |

References

Instructor: Linda Zuvich

Introduction to Java Programming Comprehensive Version, 10th Edition

 MyProgrammingLab Website

<https://docs.oracle.com/javase/7/docs/api/>

<https://docs.oracle.com/javase/8/docs/api/>