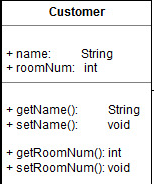
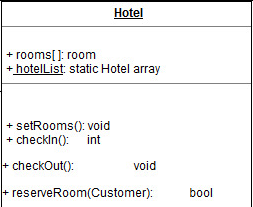
CSC 260 Group Project

Group Members: Riley Fitzpatrick, Noah Gross, Christopher Glanzer, and Kedar Neopaney

**Introduction:**

 For our project, we decided to do problem number three; a room reservation system for a non-profit retreat and conference center. We are using draw.io to create all of our UML diagrams. (All UML diagrams will be located at the end of this document, with the exception of the Main use case specification, which will be located in the dropbox).

**Classes:**

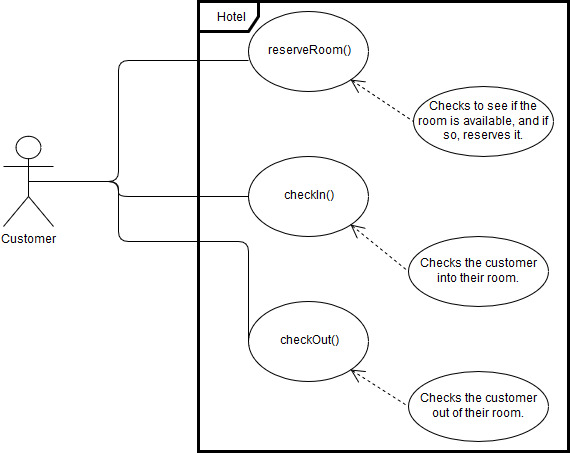
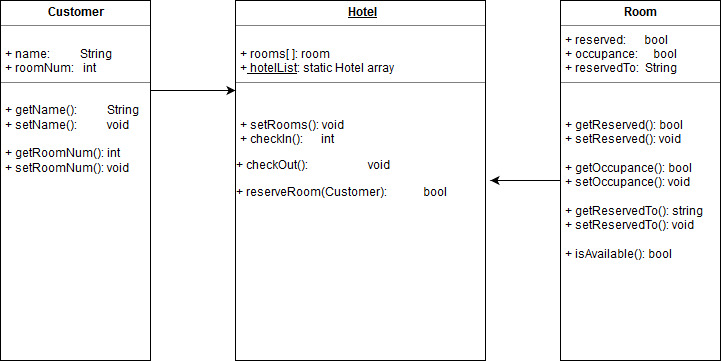
* The Customer class – This class will hold values such as their name and room number. There are also getters and setters for name and room number to get/set the appropriate values.
* The Hotel class – The hotel class will hold an array for all of the rooms. Functions located in this class will be the setRooms, checkIn, checkOut, and reserveRoom (which will all be covered in greater detail later in the document).
* The Room class – The room class will hold mostly boolean values. Values such as reserved and occupance are to report back telling the user if the room is either reserved/occupied; and the string reservedTo is a value for the customer’s name the is currently using it. The room class is also home to getters/setters for reserved, occupance, and reserved to. There is an additional function in the room class called isAvailable to determine whether or not a room is available.

**Functions:**

* setRooms() – setRooms is to initialize the rooms in the array, setting all of the reserved, occupance, and reservedTo values to 0.
* checkIn() – The checkIn function allows the user to check into their room. The function will also change the values of reserved, occupance to 1, and the value of reservedTo to the value of the customer’s name. If the customer tries to check into a room that doesn’t exist, the function will return an error message relaying that information back to the user.
* checkOut() – The checkOut function will allow the user to check out of their room. This function will also reset the values of reserved and occupance to 0, and change the value of reservedTo from the previous customer’s name to null. If the customer tries to check out of a room that currently has nobody in it, the function will return with an error message and relay this information to the user.
* reserveRoom(Customer) – The reserveRoom function is mainly for people who want to reserve a room ahead of time without actually being at the hotel. This function will set the value of reserved to 1. If the customer tries to reserve a room that is already occupied, the function will return with an error message and relay that information back to the user.
* isAvailable() – The isAvailable function is a simple test to determine if a room is occupied or reserved. If the room is occupied or reserved, it will return with a 0, if it is available, it will return with a 1.

**UML Diagrams/Pictures**

Below are pictures of our UML diagrams and any other required documentation (there will be one image to a page).

* UML Use Case Diagram:
* Domain Level Class Diagram:
* Domain Level Sequence Diagram:

