Kara Walters

System Requirements Documentation

TABLE OF CONTENTS

CUSTOMER PROBLEM:

Problem Statement: When booking a hotel room, guests are expecting to be checked into a room that looks like the photos they have seen on the website. There should allow for guests to book the exact room they will be staying in with accurate photos and amenities. Guests should also have the ability to alter their reservations.

Glossary of Terms*:*

Suites - A sitting room attached to the sleeping area with a king-sized bed, kitchenette, and bathroom. Perfect for those wanting a luxurious stay.
Standard - A (sleeping) area featuring a queen-sized bed and bathroom. Perfect for 1 or 2 people.
Single - A (sleeping) area featuring a full-sized bed and bathroom. Perfect for 1 or 2 (1 adult and 1 child) people.
Double - A (sleeping) area featuring a king-sized bed and bathroom. Perfect for 2 people.
Type - The category of room (Ex: Suite, Standard, Single, Double)

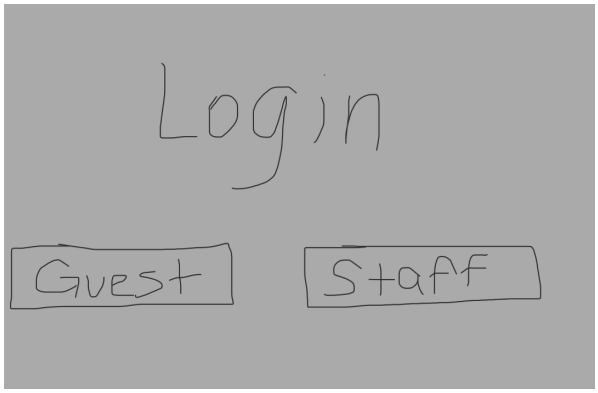
Functional Requirements:

No.	Priority Weight	Description
REQ-1	High	A list of all room types should be available (Suite, Standard, Single, and Double)
REQ-2	High	A list of all amenities available in all rooms (Toiletries, hair dryer, microwave, coffee maker, mini fridge, beverages/snacks for purchase (Suites only), temperature control, extra pillows/blankets, alarm clock, pool access, gym access, robes/slippers (Suites only)
REQ-3	High	Multiple photos of each room
REQ-4	High	A list of currently available rooms by type
REQ-5	High	A guest should be able to

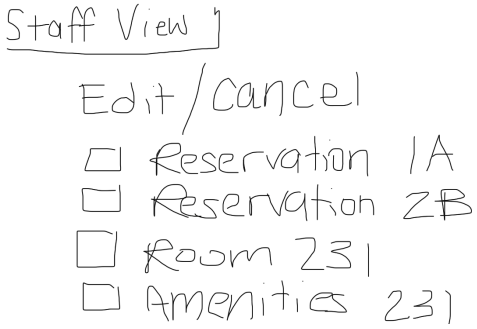
		view prices of rooms available
REQ-6	High	A guest should be able to cancel their reservation
REQ-7	High	A guest should be able to edit their reservation
REQ-8	Medium	A guest should be able to add people to the reservation
REQ-9	Medium	A guest should be able to check in at any convenient time
REQ-10	Medium	A guest should be able to check out late
REQ-11	Low	A guest should be able to leave a rating for the hotel via the system
REQ-12	Low	A guest should be able to leave a tip (digital) via the app
F	4	The system should be able to handle up to 100 logins at a time
U	1	The system should satisfy the guests need for accurate hotel bookings
R	3	The system should be accurate in what it shows the guests
P	2	The system should have fast response times in order to keep the guest engaged when in use
S	5	The system should be regularly updated to ensure accuracy

User Interface Requirements:

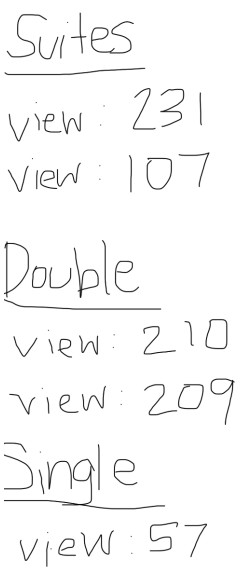
Login Screen: The login screen should allow for guests and hotel personnel to login within the system.



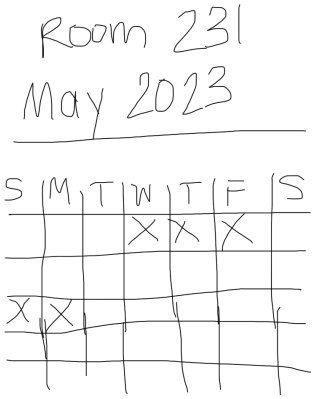
Hotel Management View: This view is only available for hotel personnel and allows views and edits of reservations made by guests, rooms, and amenities.



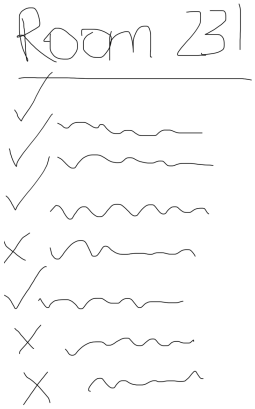
Room Selector: The room selector menu should show all of the rooms available and show their price and type.



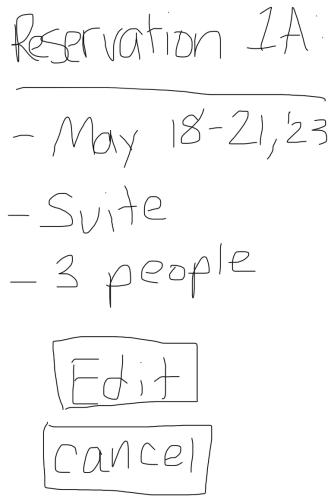
Reservation Options: Reservation options will be available and show dates the room type is or is not available.



Amenities List: A list of amenities for each room should be available.



Reservation Management: This screen is only available for guests who have made reservations. This will only show current reservations.



Plan of Work: The development process is currently happening for the Hotel Reservation Management System. I am currently in the process of trying to put together different logins for the different types of users within the system (guests, hotel managers, hotel staff, cleaning personnel).

FUNCTIONAL REQUIREMENTS:

Stakeholders:
-Hotel Chains
-Hotel Managers
-Hotel Guests

Primary Actors:
Hotel Managers – Hotel managers will be able to log in to a specific account that allows for full control of the system. They can add/modify/delete rooms, amenities, and reservations.
Hotel Staff – Hotel staff will be able to log into a specific account that allows for them to add, modify, and delete reservations for guests in need.
Guests – Guests will be able to log in to an account that will allow them to view rooms and amenities. They will be able to create, modify, and delete their own reservations.

Secondary Actors:
Visitors – Visitors will only have access to view room types as well as amenities.

Use Cases:

Hotel Managers (18 total)

- Add room
- Add amenity
- Add reservation
- Modify room
- Modify reservation
- Modify amenity
- Delete room
- Delete reservation
- Delete amenity

Hotel Staff (6 total)

- Add reservation
- Modify reservation
- Delete reservation

Guests (10 total)

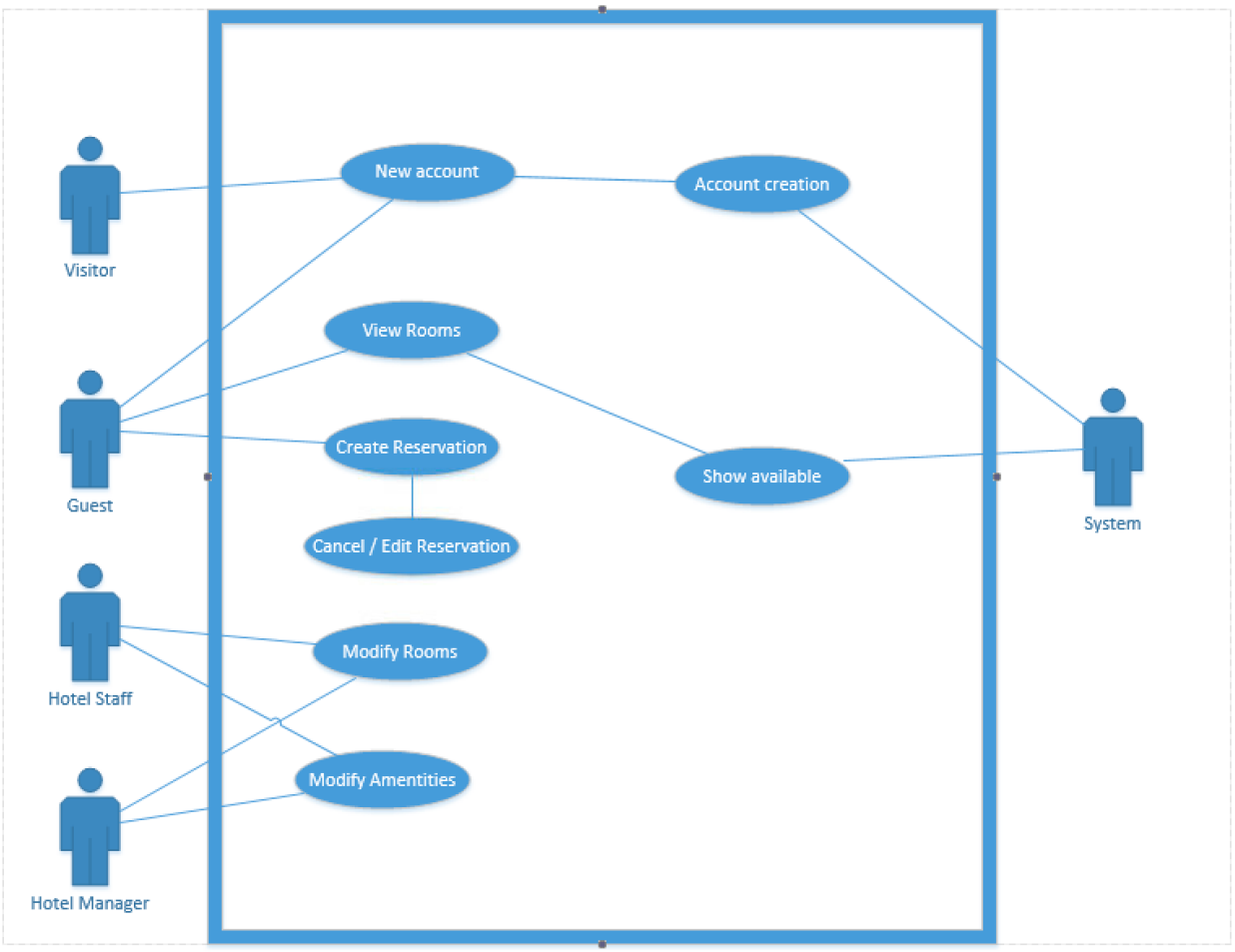
- Add reservation
- Modify reservation
- Delete reservation
- View rooms
- View amenities

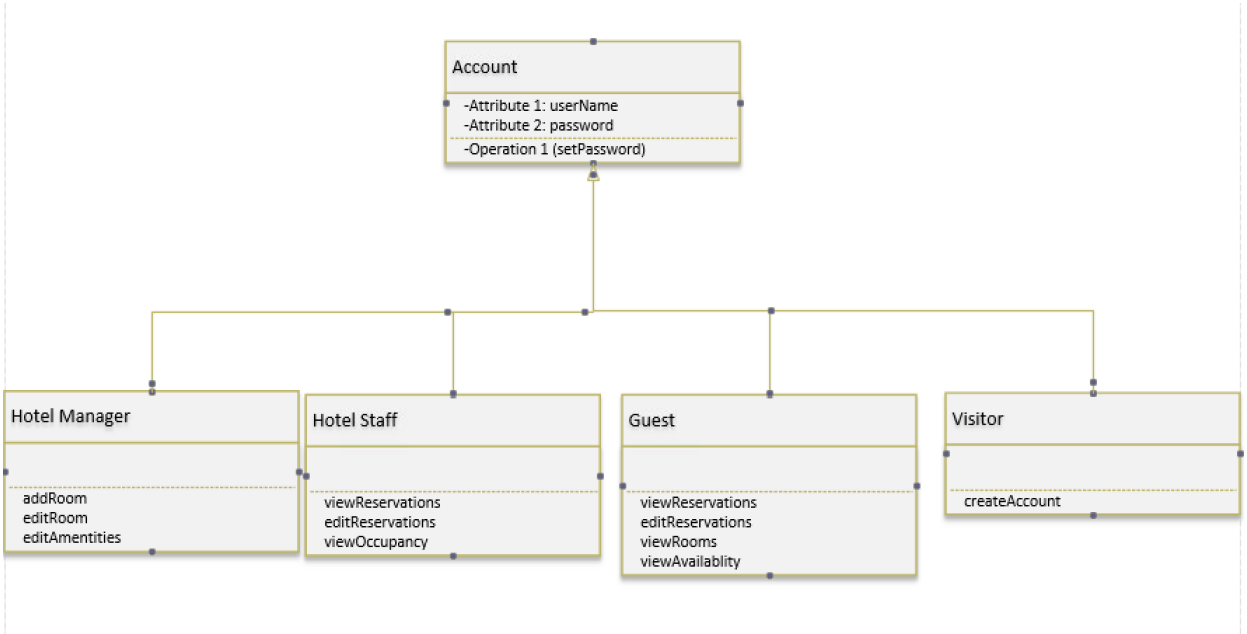
Visitors (4 total)

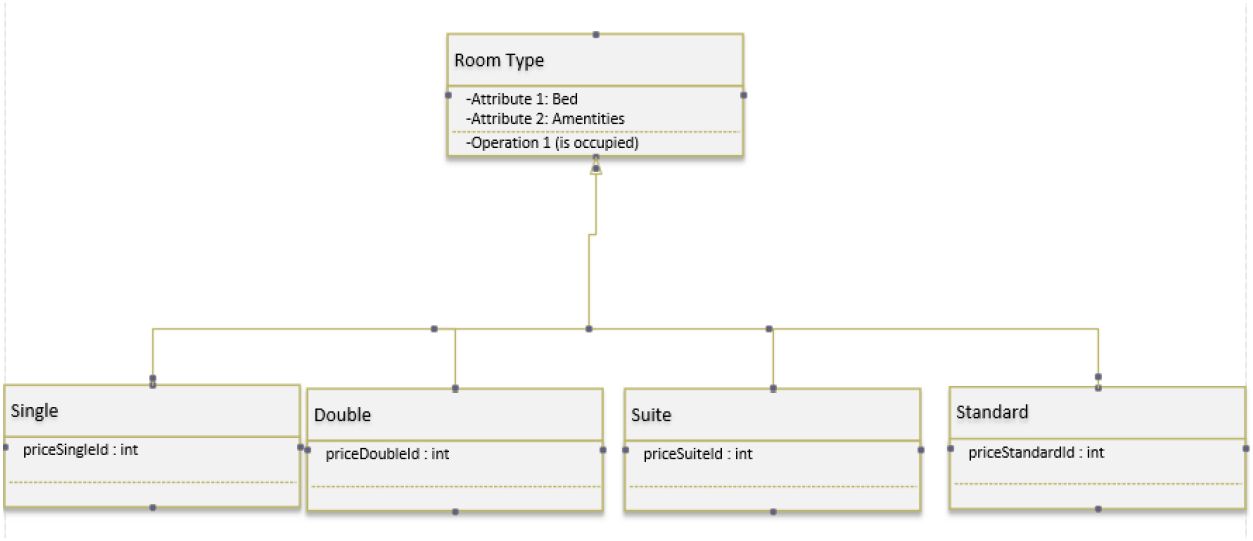
- View rooms
- View amenities

System (4 total)

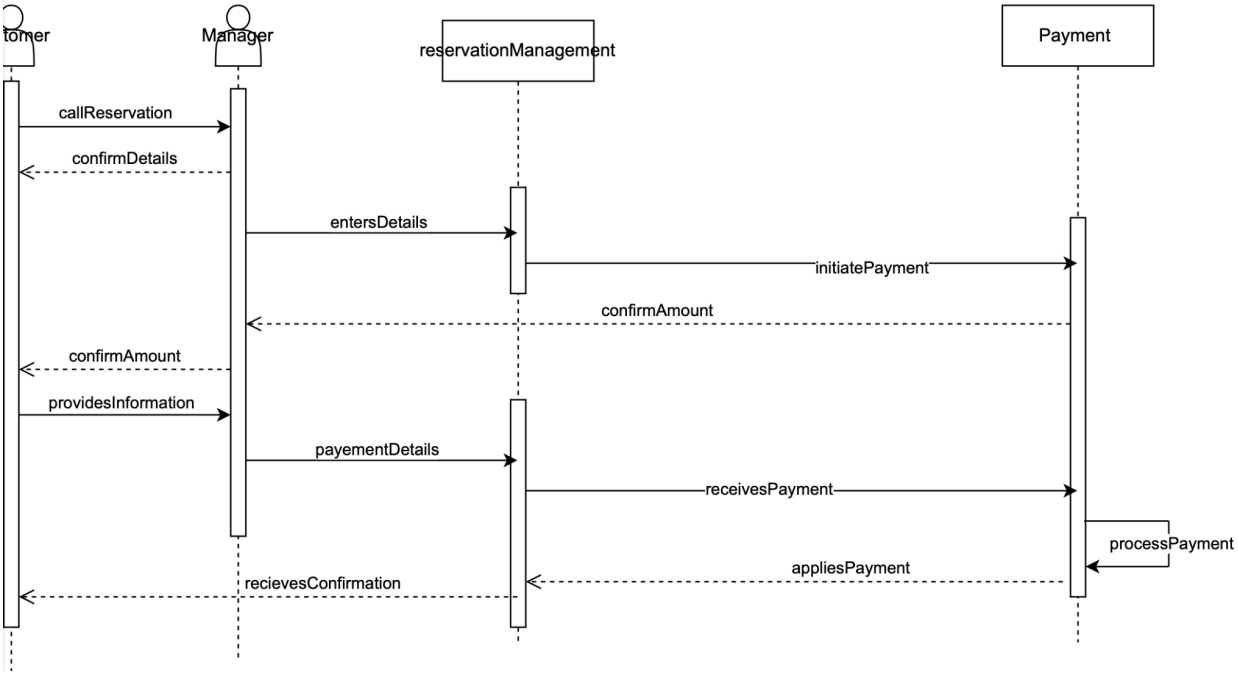
- Save changes made within system
- Show availability


Visitor
Guest
Hotel Staff
Hotel Manager
New account
Account creation
View Rooms
Create Reservation
Cancel / Edit Reservation
Show available
Modify Rooms
Modify Amentities
System

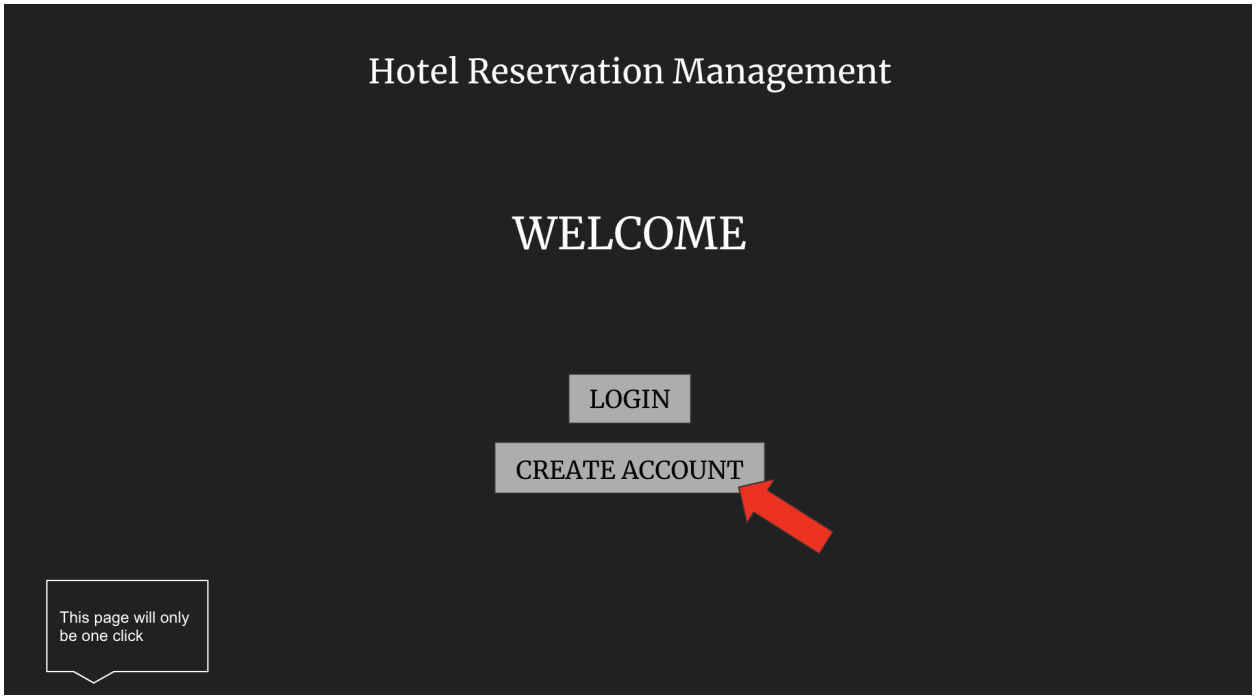

Account
-Attribute 1: userName
-Attribute 2: password
-Operation 1 (setPassword)
Hotel Manager
addRoom
editRoom
editAmentities
Hotel Staff
viewReservations
editReservations
viewOccupancy
Guest
viewReservations
editReservations
viewRooms
viewAvailablity
Visitor
createAccount

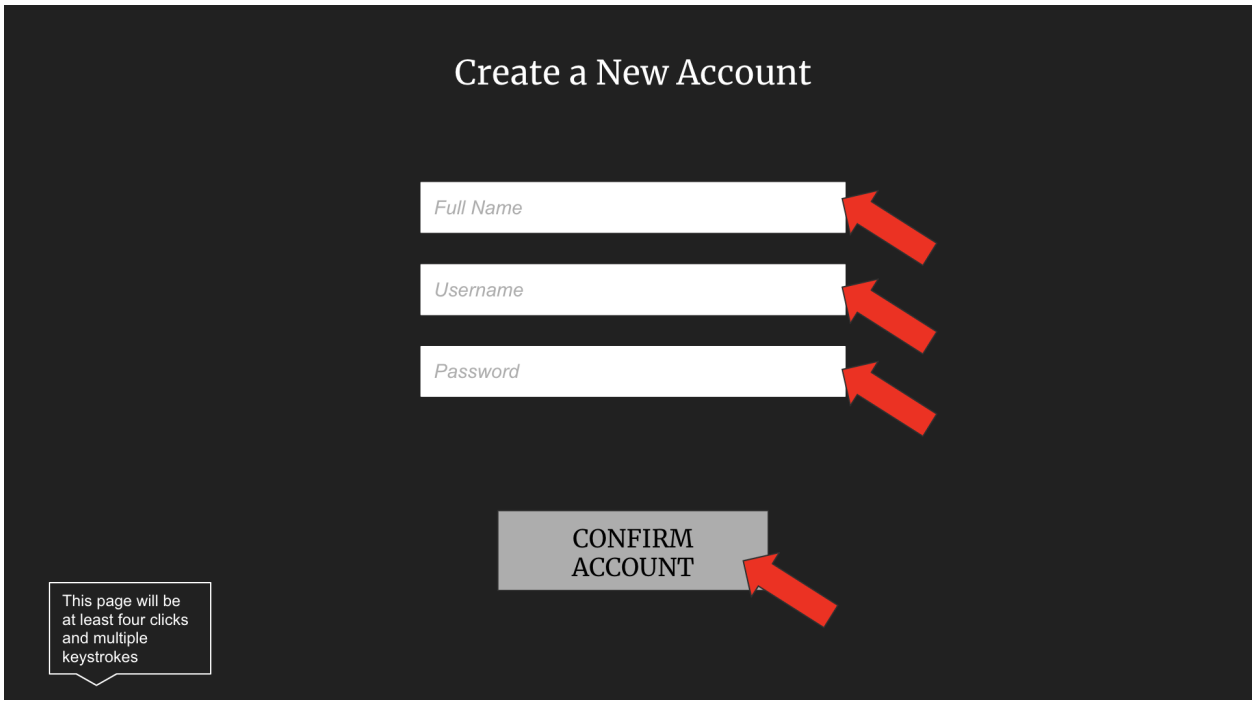


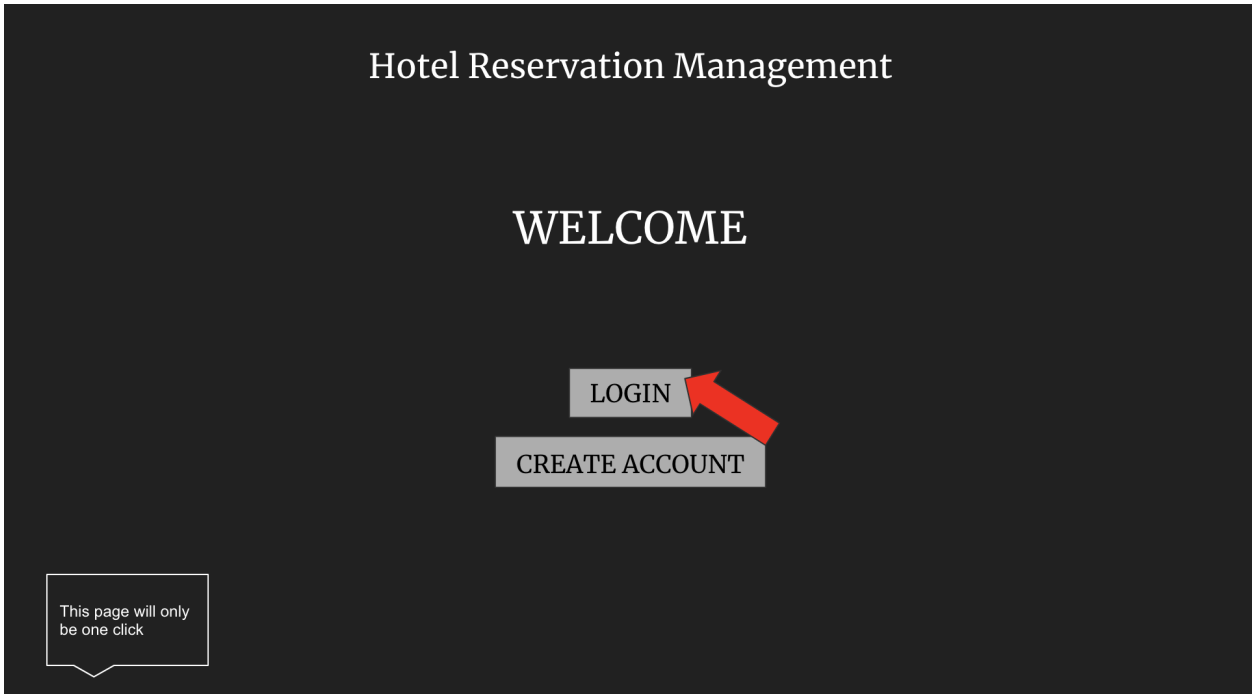
SYSTEM SEQUENCE DIAGRAM:

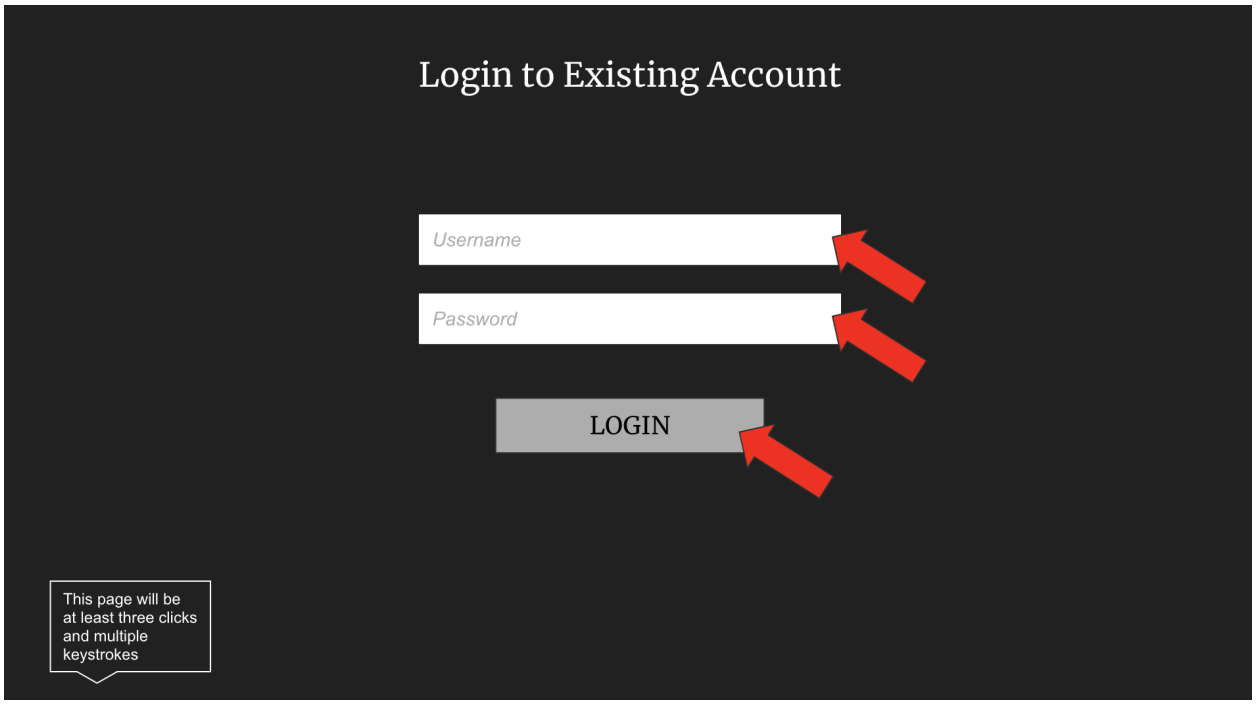


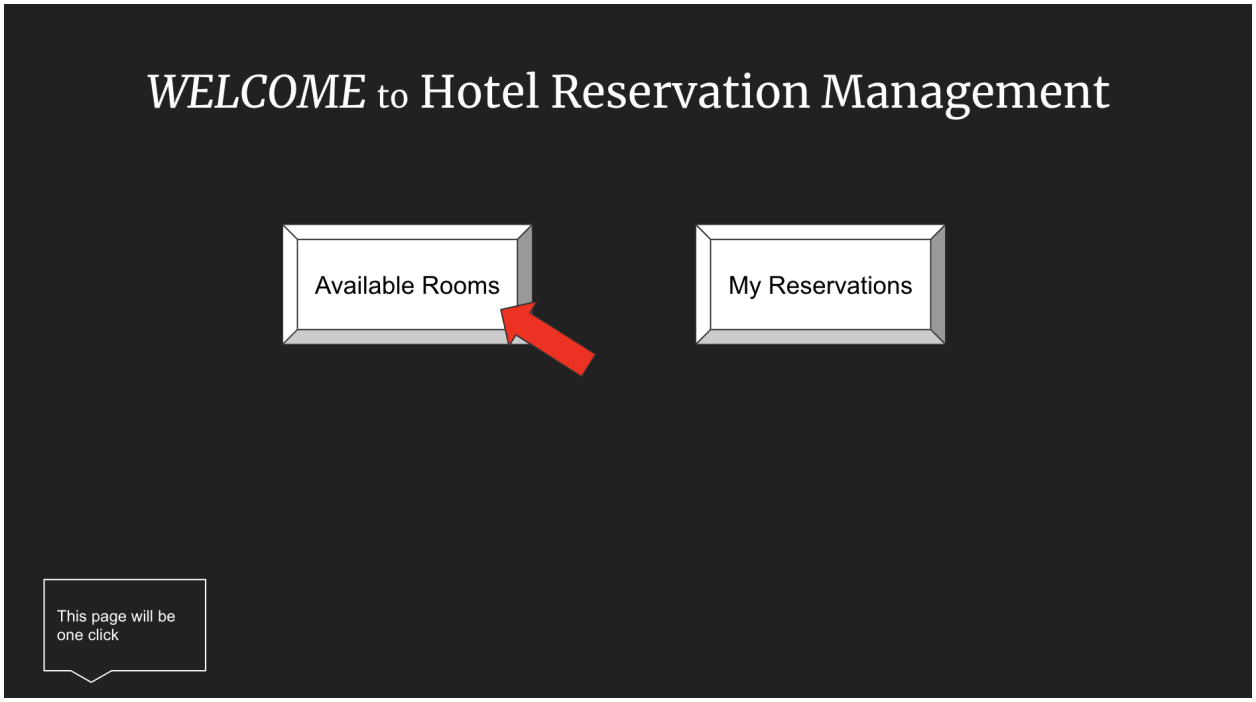
USER INTERFACE SPECIFICATION:

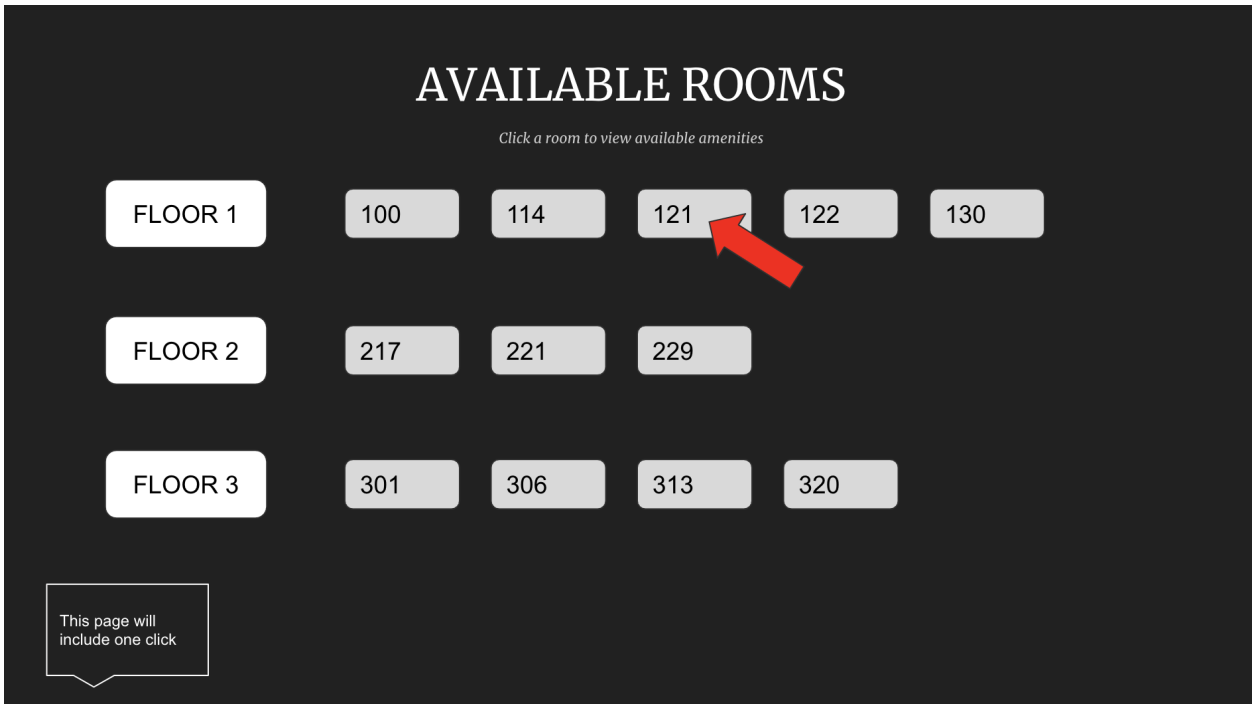


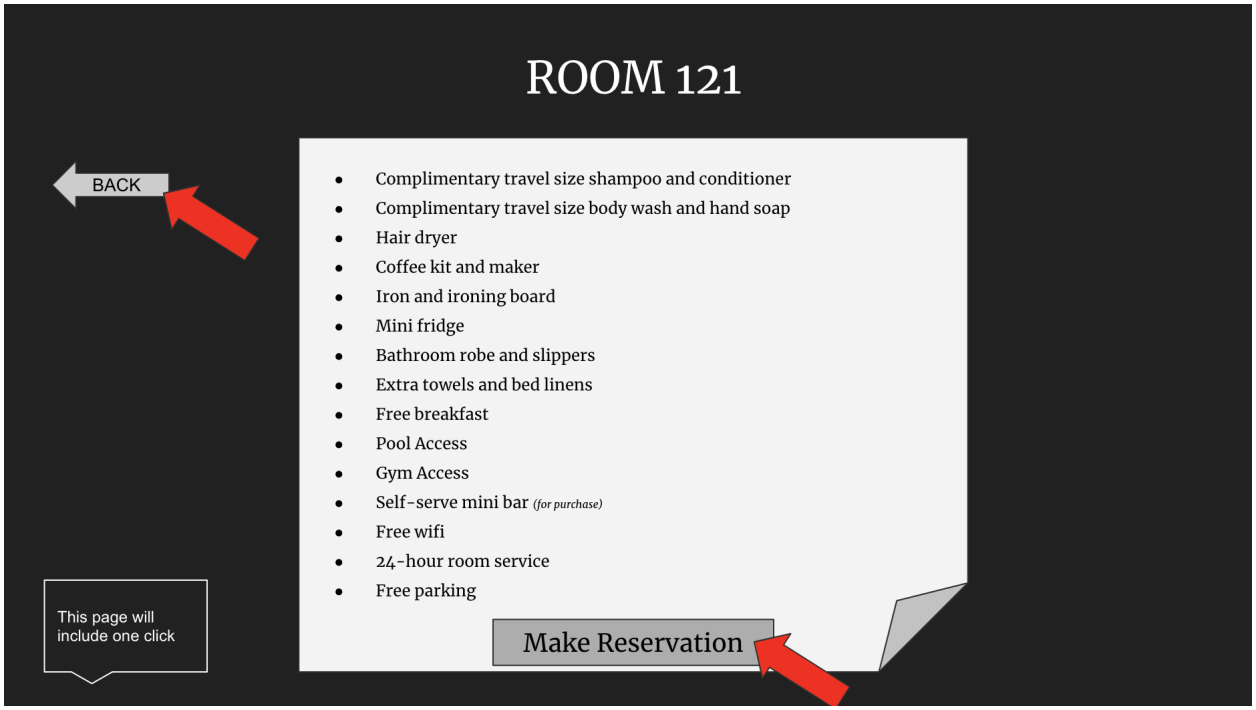


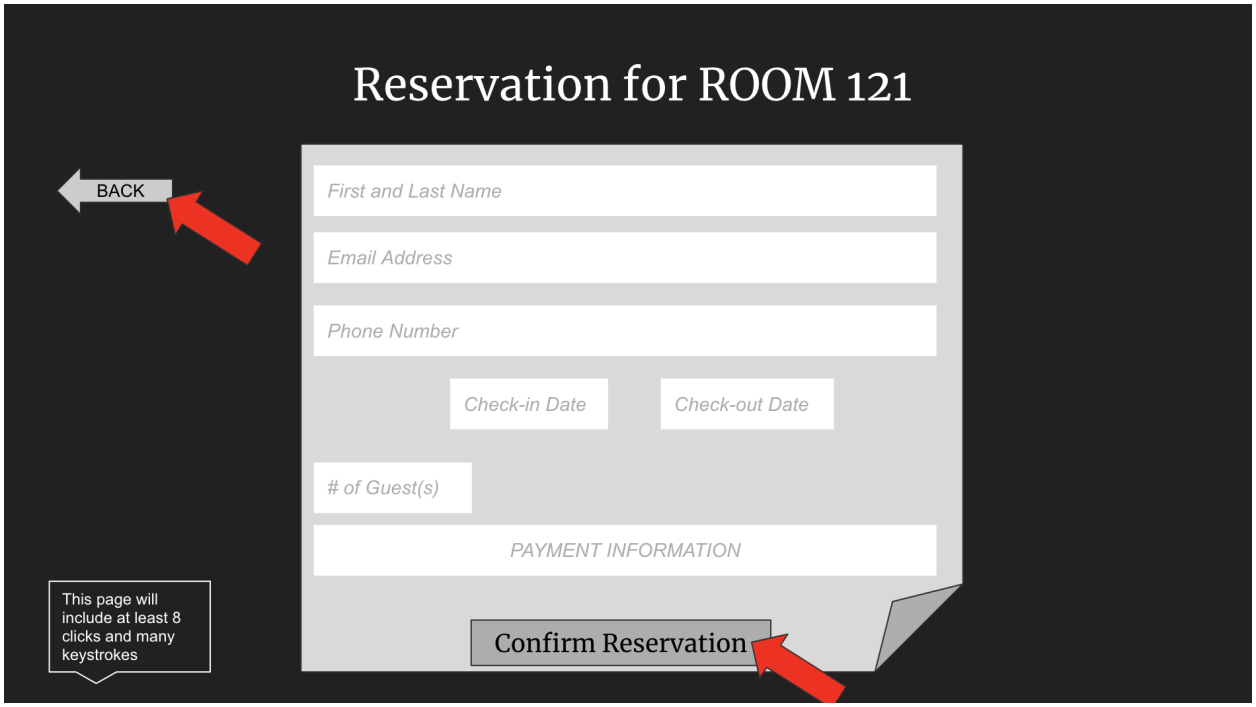

Hotel Reservation Management
WELCOME
LOGIN
CREATE ACCOUNT
This page will only be one click

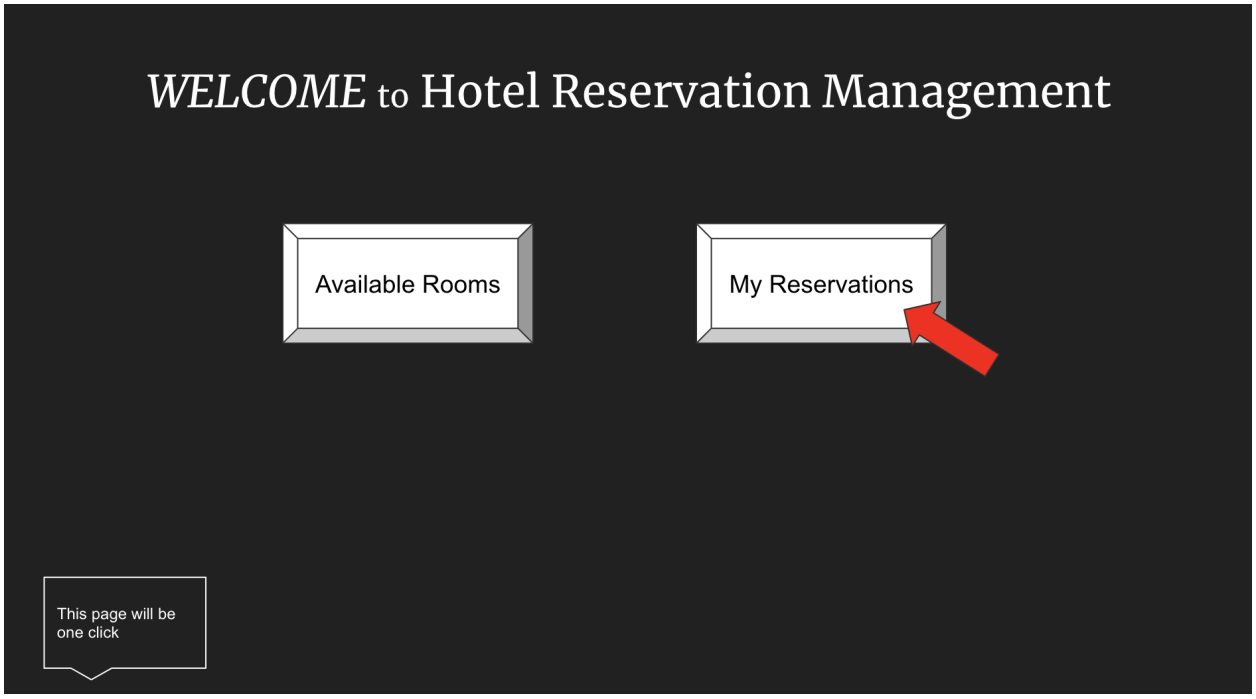

Login to Existing Account
Username
Password
LOGIN
This page will be at least three clicks and multiple keystrokes


WELCOME to Hotel Reservation Management
Available Rooms
My Reservations
This page will be one click

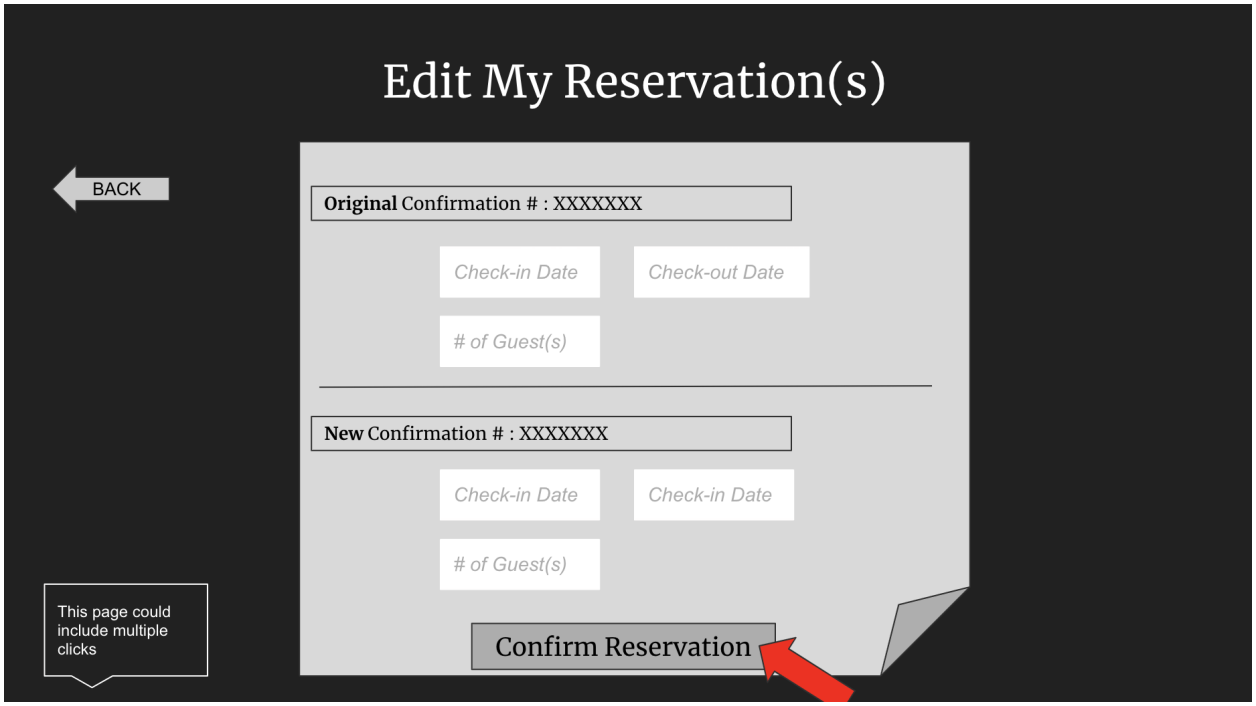

AVAILABLE ROOMS
Click a room to view available amenities
FLOOR 1
100
114
121
122
130
FLOOR 2
217
221
229
FLOOR 3
301
306
313
320
This page will include one click


ROOM 121
BACK
Complimentary travel size shampoo and conditioner
Complimentary travel size body wash and hand soap
Hair dryer
Coffee kit and maker
Iron and ironing board
Mini fridge
Bathroom robe and slippers
Extra towels and bed linens
Free breakfast
Pool Access
Gym Access
Self-serve mini bar (for purchase)
Free wifi
24-hour room service
Free parking
Make Reservation
This page will include one click


Reservation for ROOM 121
BACK
First and Last Name
Email Address
Phone Number
Check-in Date
Check-out Date
# of Guest(s)
PAYMENT INFORMATION
Confirm Reservation
This page will include at least 8 clicks and many keystrokes


WELCOME to Hotel Reservation Management
Available Rooms
My Reservations
This page will be one click


My Reservation(s)
BACK
Confirmation # : XXXXXXX
Check-in Date
Check-out Date
# of Guest(s)
Edit Reservation
This page could include multiple clicks



Traceability Matrix

No.	Priority Weight (1 - 5: 1: lowest, 5: highest)	Description
REQ1	5	Hotel managers can log in and out of system
REQ2	5	Hotel staff can log in and out of system
REQ3	5	Guests can log in and out of system
REQ4	4	Hotel managers can add room
REC5	4	Hotel managers can add amenity
REC6	4	Hotel managers can delete amenity
REC7	4	Hotel staff can add reservation
REC8	4	Hotel staff can modify reservation
REC9	4	Hotel staff can delete reservation
REC10	5	Guests can add reservation
REC11	4	Guests can modify reservation
REC12	5	Guests can delete reservation
REC13	4	Guests can view room
REC14	4	Guests can view amenities
REC15	3	Visitors can view room
REC16	3	Visitors can view amenities

Req't	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14
REQ1	5				X										
REQ2	5				X										
REQ3	5				X										
REQ4	4	X													
REC5	4		X												
REC6	4			X											
REC7	4					X									
REC8	4						X								
REC9	4							X							
REC10	5								X						
REC11	4									X					
REC12	5										X				
REC13	4											X			
REC14	4												X		
REC15	3													X	
REC16	3														X
Max PW		4	4	4	5	4	4	4	5	4	5	4	4	3	3
Total PW		4	4	4	15	4	4	4	5	4	5	4	4	3	3

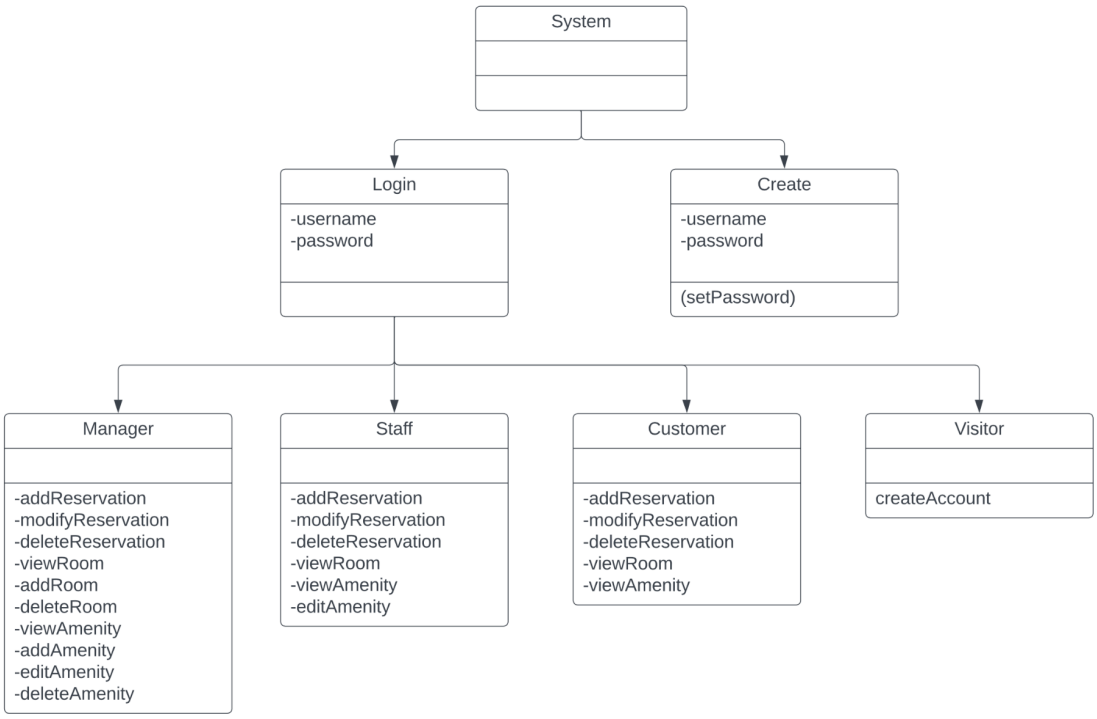
No.	Description	
UC1	Add room	*Hotel Managers*
UC2	Add amenity	
UC3	Delete amenity	
UC4	Login	
UC5	Add reservation	*Hotel Staff*
UC6	Modify reservation	
UC7	Delete reservation	
UC8	Add reservation	*Guests*
UC9	Modify reservation	
UC10	Delete reservation	
UC11	View rooms	
UC12	View amenities	
UC13	View rooms	*Visitors*
UC14	View amenities	

System Architecture and System Design

Architectural Styles

My design features client-server and service-oriented architectural styles. The design provides a service to the customers that use it. The program is also client-server because it would utilize the internet as the client.

Identifying Subsystems



Persistent Data Storage

The system will require data outliving a single execution of the system in order for customers to make/modify/delete their reservations. These persistent objects (customer information) will stay within a relational database (SQL).

Global Control Flow

Execution orders:

My system is both procedure-driven and event-driven, it just depends at what point the customer is at within the system. For example, a user will have to login every single time in order to use the system. Once logged in, it becomes event-driven, meaning that

the system waits in a loop for events and every user can generate actions in different orders.

Time dependency:

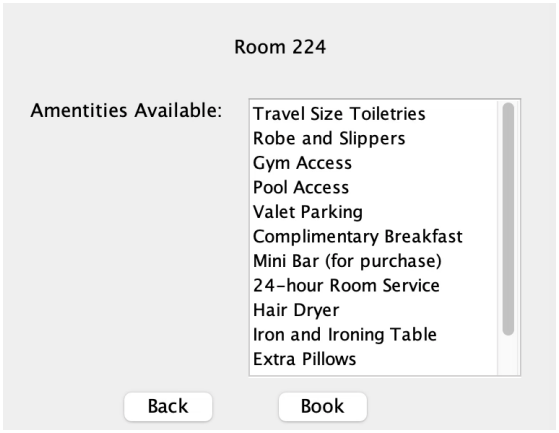
The only time there is a time dependency in the system is when someone is making, modifying, or deleting their reservation. There would be a time dependency at this stage in order to eliminate customers from holding a room “hostage”, and so mistakes in the modify or delete stage do not occur.

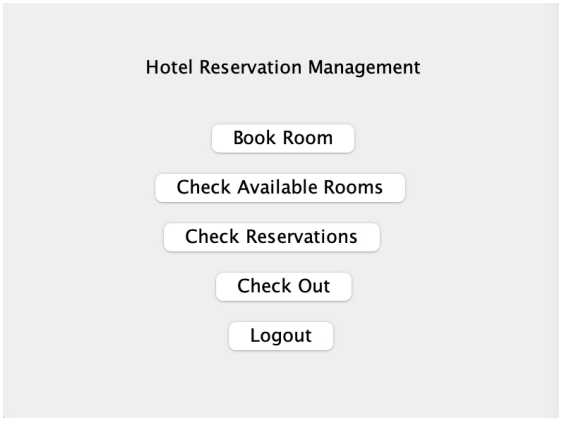
Hardware Requirements

While the system is just a hotel reservation system, it does not need a minimum screen display. It is recommended that the resolution for the system would be 1920 x 1080. This is recommended because of the photos of the rooms and the amenities. In order to get a good look of everything, it is advised that the user would have a full HD view. A network bandwidth of 56 Kbps is recommended.

User Interface Design and Implementation

- After going back through the initial screen mock-ups, I went through and visited other hotel reservation websites to see if anything obvious was missing. I wanted to make sure that while the design is simple, it is still as functional as other reservation sites.





Design of Tests

Test Cases:

- Create account
  - View rooms
  - View amenities
  - Book a room
- Log into account
  - View reservation
  - Edit reservation
- Log into account (hotel staff)
  - Add and edit amenities
  - Add and edit reservations

- Test coverage should be around 80-90% for the model. I'm hoping that the test cases focus on the main usage of the program.

- Integration testing: I want to test the main, larger test cases first (creating an account and logging in). I then will deep dive into the test cases and work towards the smaller ones. I feel like this is a good strategy to work with because it makes sure that my program can handle the larger tasks and then focuses on the smaller ones. If it can't handle the larger tasks, it won't be able to handle the smaller ones.
- Aftering making sure that the functional requirements are met, I will move my focus toward the user interface requirements. This part is important because just because a program works does not mean users will be inclined to use it. If I focus on the user interface, I can make the program look and act more user friendly. I think I would yield better results from the users also.