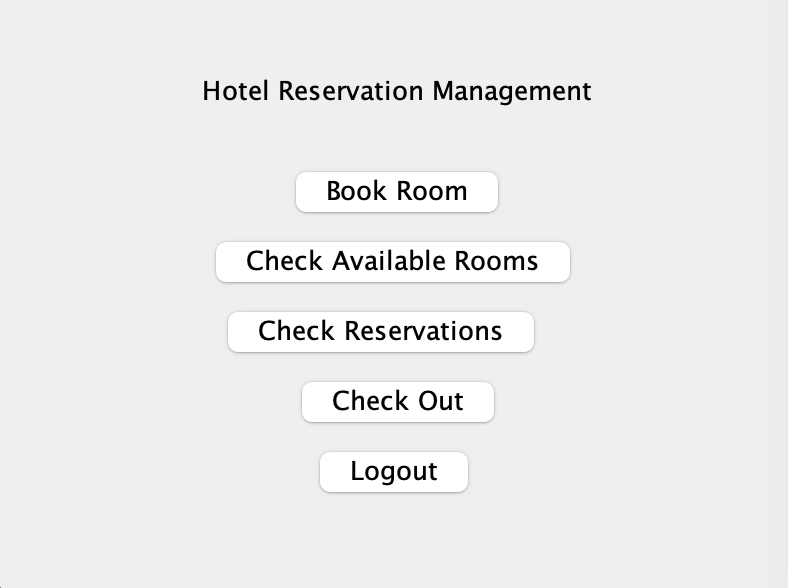
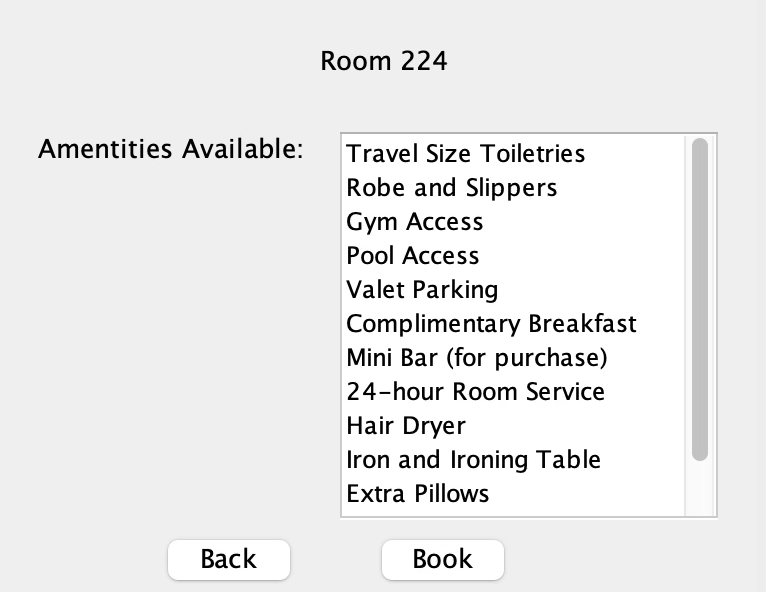
#### **User Interface Design and Implementation**

* 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.