

HOTEL MANAGEMENT SYSTEM

System Requirements Documentation

Kristen Thomas

INFO-C451

Contents

Cover Page	Pg. 1
Table of Contents	Pg. 2
Problem Statement & System Requirements	Pg. 3
Functional Requirements Specification	Pg. 4
System Sequence Diagram	Pg. 5
Activity Diagram	Pg. 7
User Interface Specification	Pg. 10
Project Plan	Pg. 14

Customer Problem Statement

Hotel management systems are utilized by most hotels around the world to handle the large volume of guests, rooms, and other information they have to handle securely. Since hotels are the most utilized accommodations when traveling, it is important that they can accurately store all of the guests personal and booking information. And as a lot of guests will check in immediately at their designated check in time, it is important to have a way to streamline this process. Creating a kiosk or app where guests can access their information and check in or check out will allow for an expedited process when staying or leaving a hotel.

System Requirements

- **Objectives:** The objectives of the system are to allow for express check in and check out, secure storage of guest and booking information, and payment processing. Managers will also be able to utilize this system to help guests with their information, but this system will ultimately free up their time to attend to other matters. For front desk workers, it will decrease the wait times of guests checking in and can provide help to those who need it.
- **System Requirements:** The system needs to be able to access guest information quickly, be secure when storing information, and be scalable to larger hotel chains.
- **Typical Customers:** Any travelers and hotel guests can use this system. Avid travelers will definitely make good use of this system to expedite the check-in process. People who don't like to wait in lines or don't want to interact with the front desk staff will also benefit from this system.
- **Project planning:**
 - Software: Java or JavaScript, MySQL
 - Hardware: PC
 - Network Requirement: high-speed internet connection with cellular backup, cloud storage
- **Development approach:** I will most likely be using Java or JavaScript, and some SQL server to store the information.

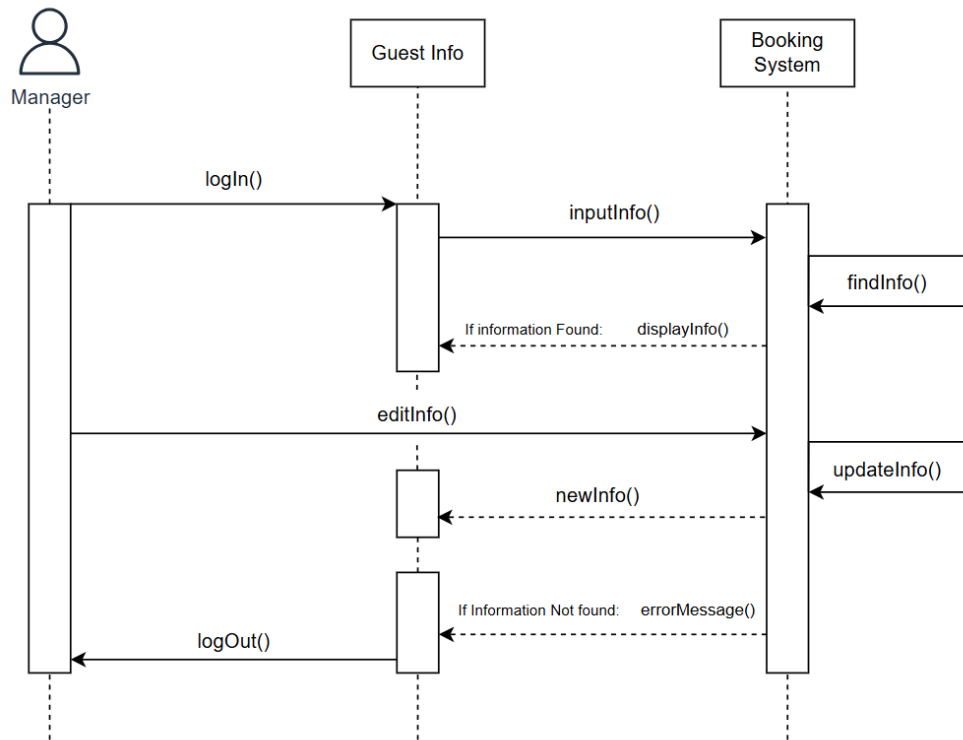
Functional Requirements Specification

No.	Priority Weight	Description
REQ-1	High	Should be able to book guests in all 2,000 rooms.
REQ-2	High	Different room types: 1 king bed, 2 queen beds, accessible rooms, deluxe suites
REQ-3	High	Guests must be at least 21 to check in to the hotel
REQ-4	High	Guests can't check in if hotel is full or don't have proper identification or payment is insufficient
REQ-5	High	Hotel is cash/paper free, must use debit/credit cards to pay incidental
REQ-6	Medium	Customers can use a room key or the hotel app to access room
REQ-7	Medium	Online receipts must be visible to guests to view or email
REQ-8	Medium	Guests can go back and search up their own room information
REQ-9	Low	Display hotel information
REQ-10	High	Collect incidental charge
REQ-11	High	Backend information for hotel staff including how many cards the kiosk contains and system statuses
REQ-12	Medium	Room must be calculated at nightly rate, if not already paid

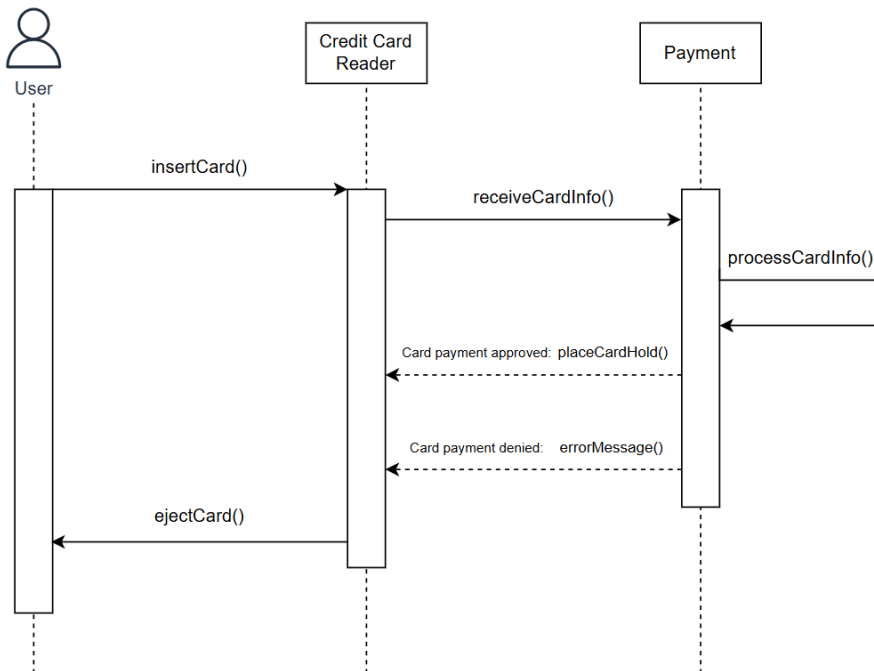
System Sequence Diagram

- Secure Payment Authorization
 - Actor: Customer
 - Objects: Credit Card, Payment
 1. Customer inserts credit card
 2. Payment system receives credit card information
 3. Card is ejected and payment is processed
 - If approved, payment is placed on card and the customer is notified
 - If denied, error message is sent
- Management retrieving guest and booking information
 - Actor: Management
 - Object: Booking system/database, guest information
 1. Management logs in with manager ID
 2. Manager will enter minimum required information
 3. If the information is in the database, manager can view and edit info
 4. If the information is not in the database, error message is returned

Management retrieving guest and booking information



Payment Authorization



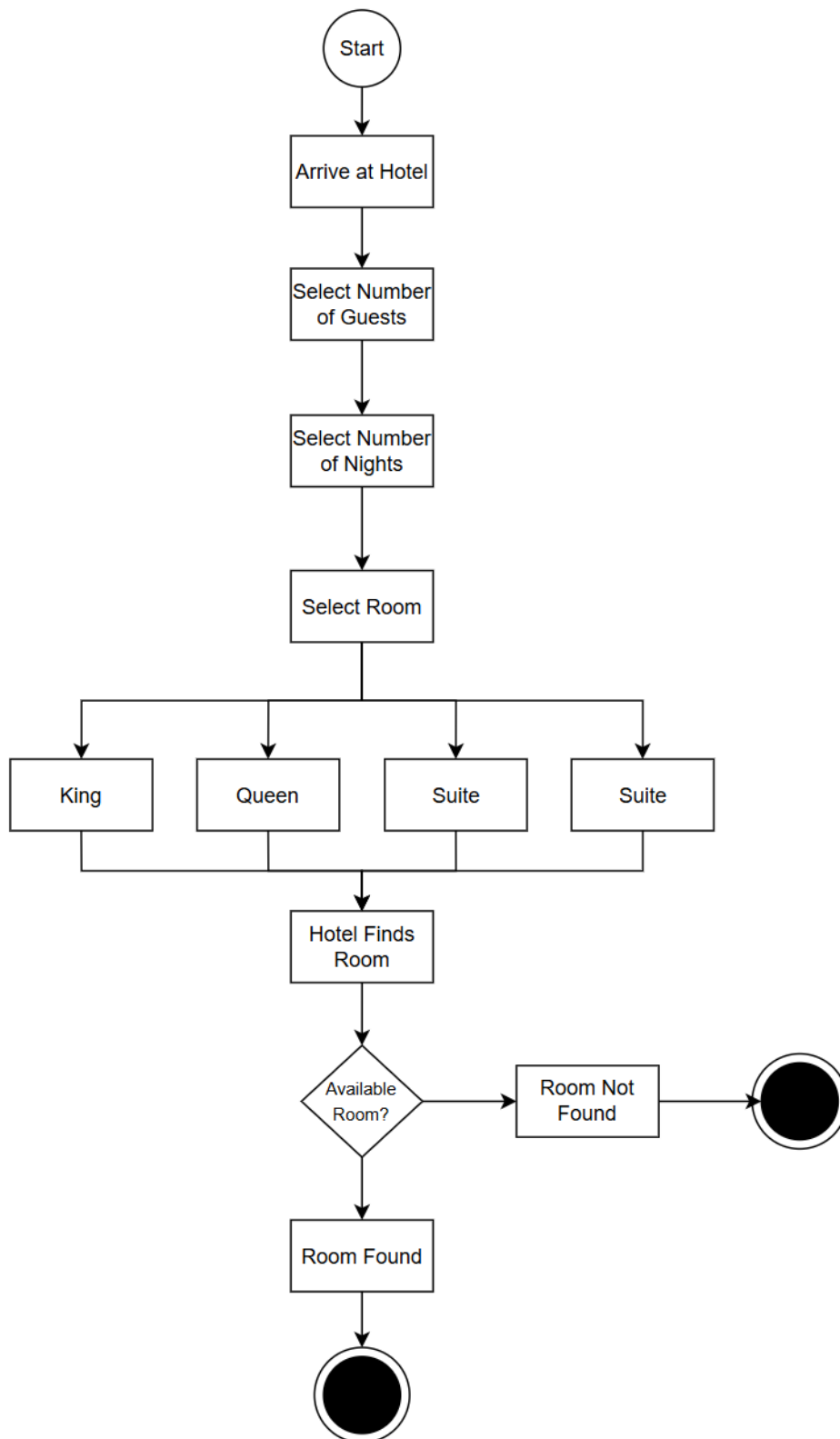
Activity Diagram

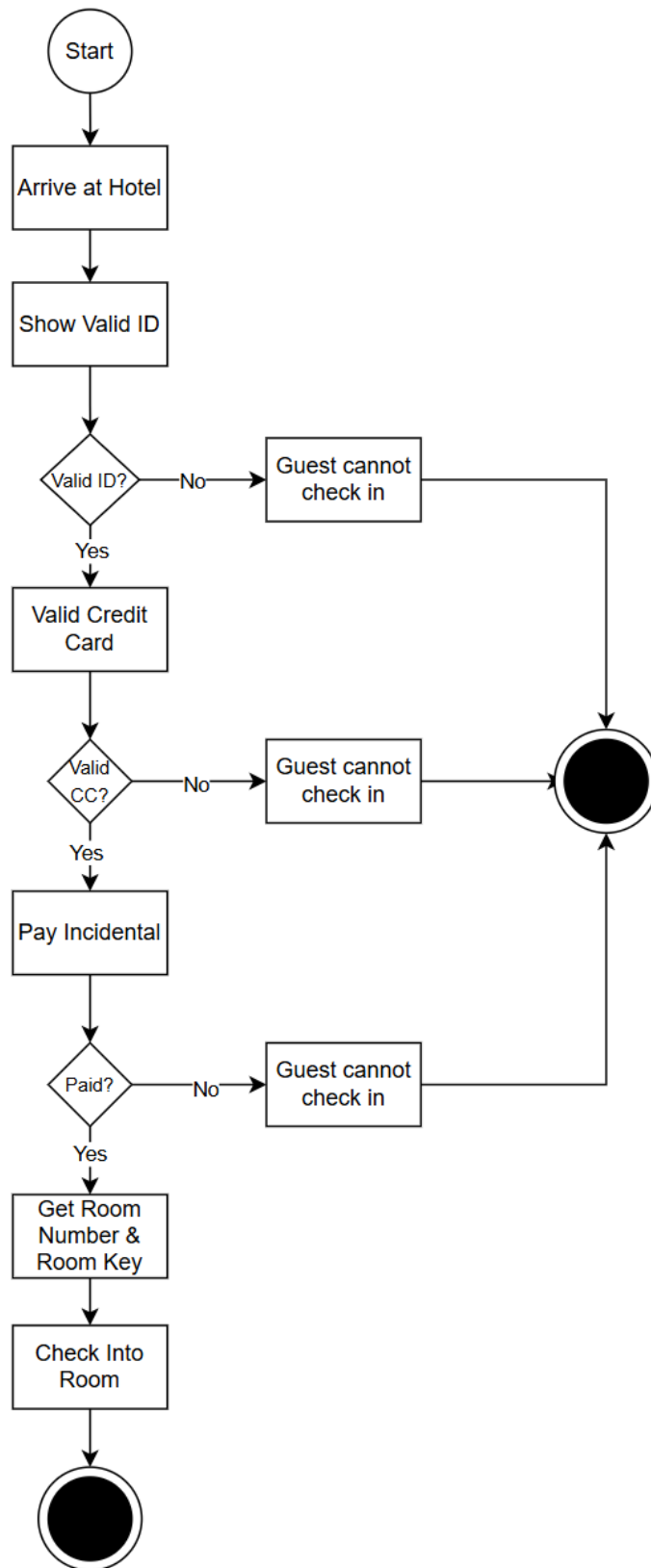
Guest books room

- States
 - Initial State: Customer arrives at the hotel
 - Final State:
 1. Customer successfully books room
 2. There are no more rooms available to book
- Actions
 - The customer arrives at the hotel. They select room type, party size, and total night stay. They are assigned a room based off of their input.

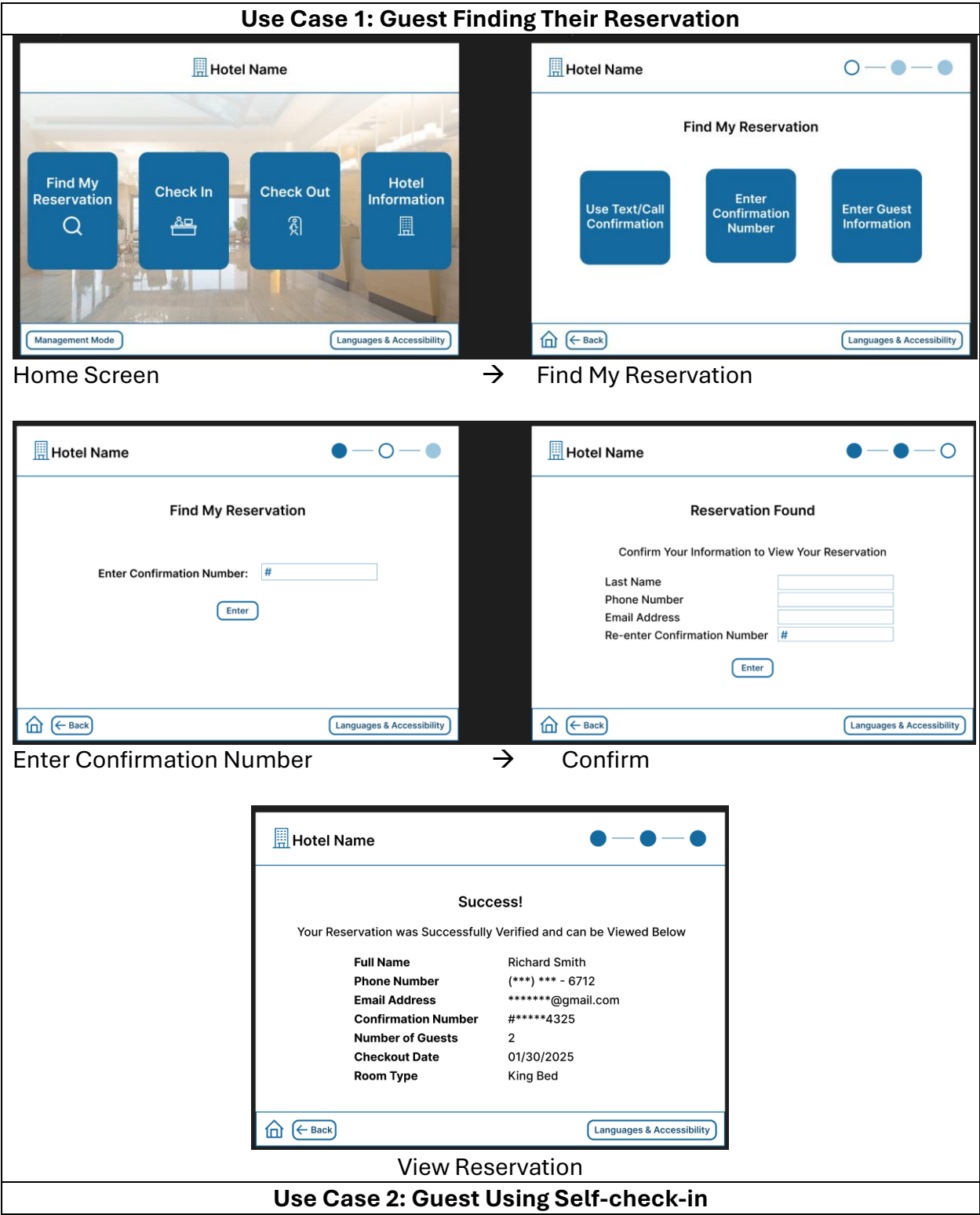
Guest uses self-check in

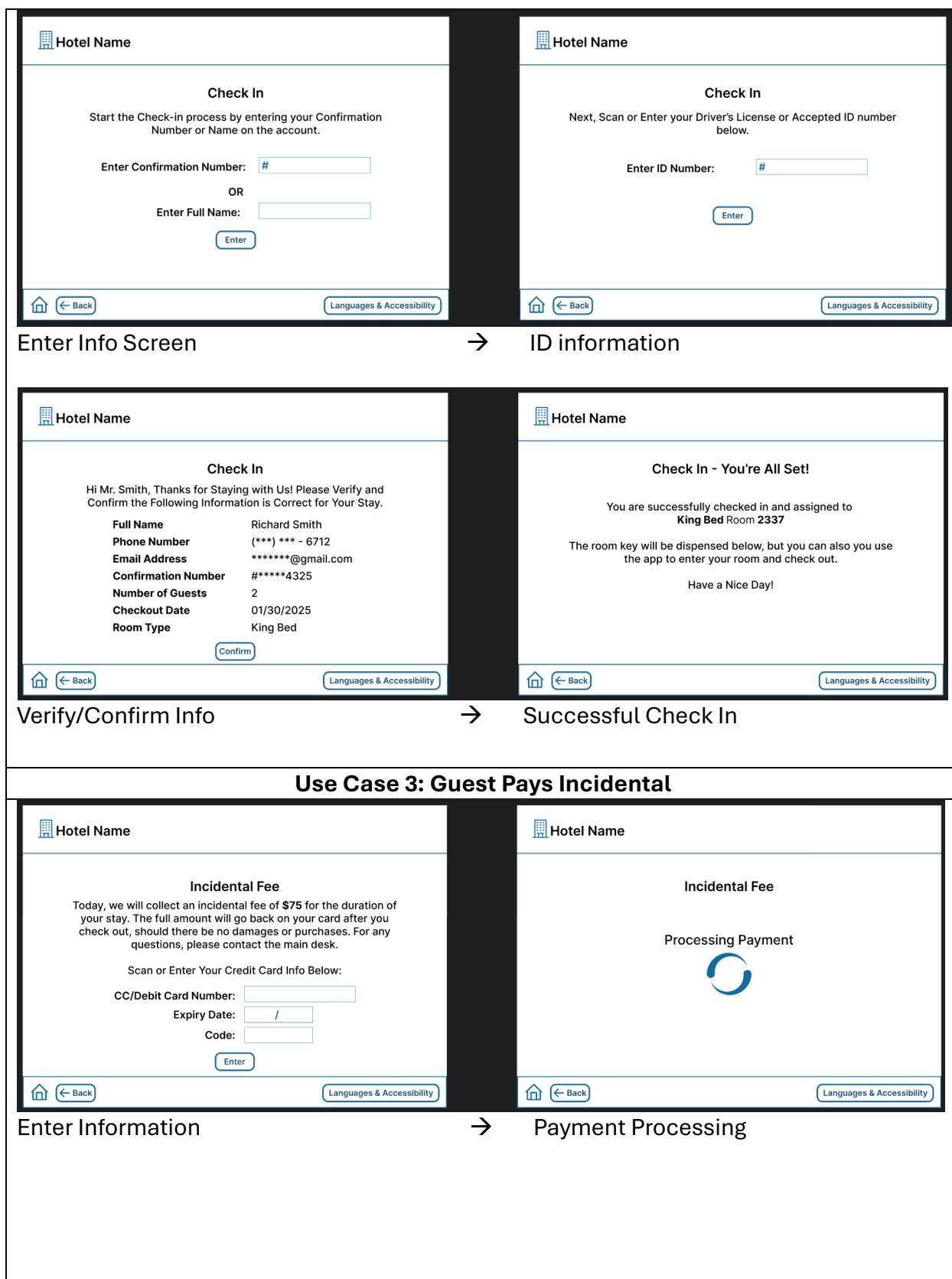
- States
 - Initial State: Guest arrives at the hotel
 - Final State:
 1. Guest successfully checked in
 2. Guest does not check in
- Actions
 - Guest arrives at the hotel. Guest shows their valid ID and enters their credit card information to pay the incidental for the room. Guest receives room number and room key through the app. Guest receives e-receipt


Guest Books Room

Guest Uses Self-Check-In

User Interface Specification






 Hotel Name


Incidental Fee- Payment Successful!

Your Payment was Successfully posted and you may now complete the check-in process.

 [← Back](#) [Languages & Accessibility](#)

Payment Successful

Use Case 4: Express Checkout

 Hotel Name


Express Checkout


Please Enter Your Room Number and Last Name to start the check out process.

Enter Room Number:

Enter Last Name:

[Enter](#)


 [← Back](#) [Languages & Accessibility](#)

 Hotel Name

Success!


You are successfully checked out. You may keep your keys or drop them in the box below.

Thank you for staying with us!

 [← Back](#) [Languages & Accessibility](#)

Enter information → Successful checkout

Use Case 5: Management Views Guest Information

 Hotel Name


Management


Enter your Manager username and PIN to proceed.

Username:

PIN:

[Enter](#)

 [← Back](#) [Languages & Accessibility](#)

 Hotel Name


Management

Create/Edit a Reservation





Kiosk Management

Search by Guest Information

Search by Confirmation Number

 [← Back](#) [Languages & Accessibility](#)

Management log in → Management interface

 Hotel Name	 Hotel Name																						
<h3>Guest Search</h3> <p>Please enter any available information to search for a Guest.</p> <table><tr><td>Last Name</td><td><input type="text"/></td></tr><tr><td>Phone Number</td><td><input type="text"/></td></tr><tr><td>Email Address</td><td><input type="text"/></td></tr><tr><td>Phone Number</td><td><input type="text"/></td></tr></table> <p><input type="button" value="Enter"/></p>	Last Name	<input type="text"/>	Phone Number	<input type="text"/>	Email Address	<input type="text"/>	Phone Number	<input type="text"/>	<h3>Guest Found</h3> <p>1 Guest was found from the search results.</p> <table><tr><td>Full Name</td><td>Richard Smith</td></tr><tr><td>Phone Number</td><td>(**) *** - 6712</td></tr><tr><td>Email Address</td><td>*****@gmail.com</td></tr><tr><td>Confirmation Number</td><td>#*****4325</td></tr><tr><td>Number of Guests</td><td>2</td></tr><tr><td>Checkout Date</td><td>01/30/2025</td></tr><tr><td>Room Type</td><td>King Bed</td></tr></table>	Full Name	Richard Smith	Phone Number	(**) *** - 6712	Email Address	*****@gmail.com	Confirmation Number	#*****4325	Number of Guests	2	Checkout Date	01/30/2025	Room Type	King Bed
Last Name	<input type="text"/>																						
Phone Number	<input type="text"/>																						
Email Address	<input type="text"/>																						
Phone Number	<input type="text"/>																						
Full Name	Richard Smith																						
Phone Number	(**) *** - 6712																						
Email Address	*****@gmail.com																						
Confirmation Number	#*****4325																						
Number of Guests	2																						
Checkout Date	01/30/2025																						
Room Type	King Bed																						
 ← Back Languages & Accessibility	 ← Back Languages & Accessibility																						

Search by guest info → Guest Search Results

Project Plan

- Development plan:
 - W1-2: Create a structure and basic requirements for the system
 - W3-4: Build the main screen and an outline of the different customer side options
 - W5-7: Work on the manager-side tasks and log-in info
 - W8: test features and complete midterm
 - W9-11: improve features and work on the interface/UI
 - W12-14: Test use cases and finish up the program
 - W15: Final