

CSIT321 - Final Year Project

Multi-factor transaction authorisation based on QR code

FYP-23-S1-02

Technical Manual

Group: FYP-23-S1-02		
Name	Student Number	Email Address
Angela Chia Ying Lin	7364209	aylchia001@mymail.sim.edu.sg
Belinda Peh Siow Yi	7023042	bsybxpeh001@mymail.sim.edu.sg
Benishia Angelin	7372917	benishia001@mymail.sim.edu.sg
Chae Youngsuh	6958692	chae002@mymail.sim.edu.sg
Chua Jia Hong	7435320	jhchua017@mymail.sim.edu.sg

Supervisor: Mr. Chua Zong Fu

Assessor: Dr. Ee Kiam Keong

Table Of Contents

1. Introduction	4
1.1. Purpose	4
1.2. Scope	4
1.3. Project Team	4
2. Overview	5
2.1. Project Description	5
2.2. How it works	5
3. Functional Requirements	7
3.1. Functional Requirement for Users	7
3.2. Functional Requirement for Hotel Operators	7
3.3. Functional Requirement for our platform.....	7
4. Test Plan	8
4.1. Purpose	8
4.2. Objectives	8
4.3. Testing Approach	8
4.4. Test Case Design.....	8
4.5. Test Environment	9
5. User Stories	10
6. Use Case Diagrams and Descriptions.....	12
7. Non-Functional Requirements	124
7.1. Usability	124
7.2. Scalability	124
7.3. Compatibility	124
7.4. Availability.....	124
7.5. Security	124
8. Other Requirements	125
8.1. System Features.....	125
8.2. Security Requirements	125
8.3. Regulations	125
8.3.1. Personal Data Protection Act (PDPA)	125
8.3.2. Cybersecurity Act.....	126
9. Marketing Research	126
9.1. What is QR Code?	126
9.2. Security Issues Regarding QR Code	126
9.3. How are we going to tackle this issue?	127
9.4. Design Research.....	127
9.5. Similar Multi-factor transaction authorisation done in other platforms	128

9.5.1 OAuth2	128
9.6. Research on places where application-specific authorisation logic should be built in.....	129
10. Work Breakdown Structure.....	130
11. Gantt Chart.....	133
12. Appendix	136

1. Introduction

This document serves as an overview of our project, containing a comprehensive description of the requirements for the project, including functional and non-functional requirements. The document is structured into several sections, including an overview of the project explaining the purpose and scope of the project, a detailed description of the project requirements, as well as the marketing research done for this project.

1.1. Purpose

The purpose of this project is to create a hotel booking platform which uses an authentication system that requires more than one authentication factor for successful authentication based on the QR codes that are being generated by the System Admin. The authentication system will be executed between a QR code and a mobile application along with the QR code scanner to complete the transaction.

1.2. Scope

The scope of the project includes the overall description of the authentication system, as well as the integration of the system and the creation of a user-friendly interface. The project is subjected to strict security requirements and must meet the regulations set by the Personal Data Protection Act (PDPA) and the Cybersecurity Act.

1.3. Project Team

The table below shows the formation of the team along with the roles, that they have taken in this project:

Name	UOW ID	Email	Roles
Angela Chia Ying Lin	7364209	aylchia001@mymail.sim.edu.sg	Programmer, Developer, Tester & Documenter
Belinda Peh Siow Yi	7023042	bsybxpeh001@mymail.sim.edu.sg	Programmer, Developer, Tester & Documenter
Benishia Angelin	7372917	benishia001@mymail.sim.edu.sg	Programmer, Developer, Tester & Documenter
Chae Youngsuh	6958692	chae002@mymail.sim.edu.sg	Programmer, Developer, Tester & Documenter
Chua Jia Hong	7435320	jchua017@mymail.sim.edu.sg	Programmer, Developer,

			Tester & Documenter
--	--	--	---------------------

2. Overview

This section of the document describes the project idea along with the project description and how our platform will work.

2.1. Project Description

The team aims to create a platform that makes the process of booking a hotel room faster and more efficient along with a keyless check-in, check-out system. The customers using the platform will be entitled to promotions on their hotel bookings while the hotel operator who onboard our platform will be given access to the customer's personal details and preferences. Our platform will allow the hotel operator to understand their customers better while customers will be able to enjoy quality discounts unavailable to them outside the platform.

As our project is centred around the idea of QR Code, the team decided on the idea of providing convenience to hotel customers by replacing a room key with QR codes to unlock the room and check out of the hotel, along with providing customers with discounts for using our platform. Along with Multi-Factor Authentication such as the use of biometrics to secure the hotel customers information.

2.2. How it works

The team plans to create a website and a mobile application. The website serves as a catalogue where customers can view all the available hotel rooms for rent. Once the customer finds a suitable room to book, the customer will be able to reserve a room and make their payment through the website itself.

A QR code will be pasted outside every room for scanning. After checking in to the hotel, the customer will be able to use the QR scanner on our mobile application to unlock the hotel room. After validating from the hotel side that the current customer is authorised to use the room at the appointed date and time, the room will be unlocked for the customer to use.

When the customer wants to check out of the room, there will be a main check-out QR code that is pasted at the hotel lobby for customers to check out of the hotel without having to go through the hassle of spending extra time and waiting in a queue at the hotel counter to check-out.

The diagram below depicts the flow of how our platform works from customer and hotel operator.

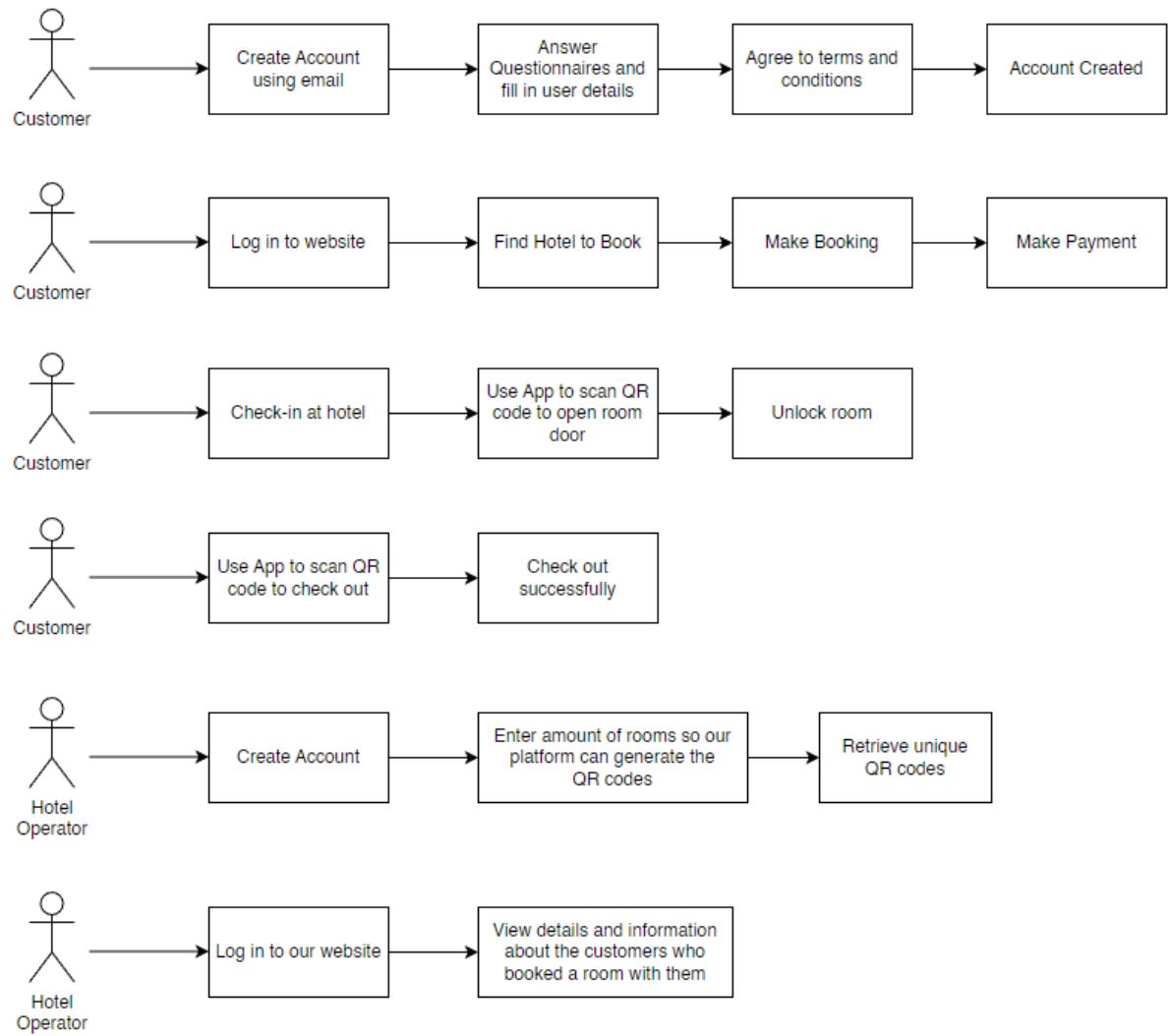


Figure 1: Simple diagram to show the process of our platform

3. Functional Requirements

This section contains the functional requirement of our platform.

3.1. Functional Requirement for Users

Customers will have to create a user account. Our platform - both mobile application and website will need the basic functions such as the login button, logout button and edit customer's details button. Other functions that our website requires is the view and search for hotel listings.

The main focus of our project is the usage of the QR Code. The QR code for each hotel room will be generated by our platform so that when the customer scans the QR code using the QR code reader from the mobile application, it will authenticate and authorise the customer.

Our mobile application will have the functionality of scanning the QR Code using the QR code scanner so the customer can scan the respective QR codes to unlock the hotel room door and check out of the hotel.

3.2. Functional Requirement for Hotel Operators

Similarly, the hotel operators will have a hotel operator account assigned to them. Using this account, they will be able to view the bookings made by the customers, along with their personal information and their preferences.

The hotel operators will retrieve the QR codes from our platform based on the number of rooms the hotel has. The hotel operators have to put the QR codes outside of the hotel room doors.

3.3. Functional Requirement for our platform

Our platform will need to generate QR codes for the hotel and the rooms along with a verification method to authenticate and authorise the hotel customer when they scan a QR code.

4. Test Plan

4.1. Purpose

The purpose of testing is to ensure that the website and the mobile application that is developed is of good and high quality, and to make sure it gives users a good satisfaction while using it. This includes testing of all components of the website and mobile application to assess how it behaves in various situations. Through this way, we can identify problems and bugs that would not have been encountered during the development phase. Testing also helps to prevent bugs, controls development costs and improve performance of website and mobile application.

4.2. Objectives

The main goal of the testing plan is to verify if the CheckInn website and mobile application serves and provides justification as what is described in the specifications.

4.3. Testing Approach

There will be two types of testing approaches taken, that is Functional Testing and Unit Testing. Through these approaches, we can gauge the application's readiness to ensure that it is working like it should be. The below are some helpers that will be used during the testing phase:

- Noting down of steps for each test case in great detail
- Capturing of error messages that should and should not arise
- Having the correct outcomes of each test written down before the testing
- Making sure all output that has been noted is same as the expected outcome

4.4. Test Case Design

The functional requirements will be converted into use cases. The development, validation, and change of test cases and test scripts are all part of this process. The result of the process will be the Test Cases.

Each use case will have its own test cases and results. They would be displayed as a screenshot below each of the use cases. The test case documents that contain detailed description of the process for each user will also be provided separately.

4.5. Test Environment

The hardware and operating system details are required to assess the CheckInn features of both website and mobile application are specified in a test environment. The operating system software for each of these devices used is listed below:

Devices For Website	LGPC - DESKTOP-JIJ7P0A	VivoBook_ASUSLaptop - LAPTOP-NVMU2TT8	ASUS Gaming Laptop (FX503V)
Software Requirements			
Operating System	Windows 11	Windows 11 Home	Windows 10 Home
Hardware Requirements			
Processor	Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz	Intel(R) Core(TM) i7-950H CPU @ 2.60GHz 2.59GHz	Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz 2.80 GHz
GPU	Intel(R) UHD Graphics 620	Intel(R) Graphics	NVIDIA Geforce GTX
Resolution	1920 x 1080	1920 x 1080	1920 x 1080
Network	Intel(R) Wireless-AC 9560 160MHz	Intel(R) Wireless-AC 9560 160MHz	Intel(R) Dual Band Wireless-AC 8265
CPU	Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz	Intel(R) Core(TM) i7-950H CPU @ 2.60GHz	Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz

Devices For Mobile	XiaoMi 10TPro
Software Requirements	
Operating System	Android 12 SKQ1.21
Hardware Requirements	
Processor	Qualcomm Snapdragon 865
GPU	Adreno 650
Resolution	1080 x 2400 pixels
Network	Wi-Fi AC, 4 GHz, 5 GHz

CPU	Octa-core (1x2.84 GHz Cortex-A77 & 3x2.42 GHz Cortex-A77 & 4x1.80 GHz Cortex-A55)
-----	---

5. User Stories

System Admin

- 1) As the **System Admin**, I want to be able to **login** so that I can manage the customers and the hotel chain client.
- 2) As the **System Admin**, I want to be able to **logout** so that I can exit the system.
- 3) As the **System Admin**, I want to be able to **create a user account** so that I can add a new user into the system.
- 4) As the **System Admin**, I want to be able to **view all user accounts** so that I can retrieve the user account details list.
- 5) As the **System Admin**, I want to be able to **update a user account** so that I can edit the account information of a user.
- 6) As the **System Admin**, I want to be able to **search for a specific user account** so that I can retrieve the user account.
- 7) As the **System Admin**, I want to be able to **delete a user account** so that I can delete the user account.
- 8) As the **System Admin**, I want to be able to **suspend a user account** so that I can deactivate the user account due to inactivity for a very long period of time.
- 9) As the **System Admin**, I want to be able to **re-activate a user account** so that I can re-activate the user account.
- 10) As the **System Admin**, I want to be able to **view all user profiles** so that I can retrieve the data based on the user profile.
- 11) As the **System Admin**, I want to be able to **view a user profile** so that I can retrieve the data based on the user profile.
- 12) As the **System Admin**, I want to be able to **search for a specific user profile** so that I can retrieve the user profile.

Hotel Operator

- 13) As the **Hotel Operator**, I want to be able to **login** into the **Website** so that I can access the system.
- 14) As the **Hotel Operator**, I want to be able to **logout** from the **Website** so that I can exit the system.
- 15) As the **Hotel Operator**, I want to be able to **view all Customers' details** from the **Website** so I know who my customers are.
- 16) As the **Hotel Operator**, I want to be able to **view all Customers transactions** from the **Website** so that I know my customer's preferences.

- 17) As the **Hotel Operator**, I want to be able to **add a room** from the **Website** so that the customer can view more available rooms.
- 18) As the **Hotel Operator**, I want to be able to **generate QR codes** from the **Website** so that the customer can scan the QR code to enter the hotel room.
- 19) As the **Hotel Operator**, I want to be able to **view generated QR codes** from the **Website** so that the hotel operator can print the QR code for customer to scan it and enter the hotel room.

Customer

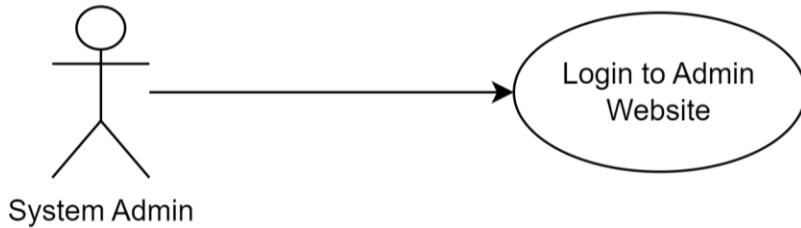
- 20) As the **Customer**, I want to be able to **login** into the **Website** so that I can access the system.
- 21) As the **Customer**, I want to be able to **logout** from the **Website** so that I can exit the system.
- 22) As the **Customer**, I want to be able to **sign up** to the **Website** so that I can access the system.
- 23) As the **Customer**, I want to be able to **update my password** in the **Website** so that I can change the password.
- 24) As the **Customer**, I want to be able to **create a hotel booking** from the **Website** so that I can book a room to stay in.
- 25) As the **Customer**, I want to be able to **view my hotel bookings** from the **Website** so that I can retrieve the hotel booking details.
- 26) As the **Customer**, I want to be able to **view my specific hotel booking** from the **Website** so that I can retrieve the hotel booking details.
- 27) As the **Customer**, I want to be able to **update my hotel booking** from the **Website** so that I can edit the hotel booking details.
- 28) As the **Customer**, I want to be able to **delete my hotel booking** from the **Website** so that I can cancel my hotel booking.
- 29) As the **Customer**, I want to be able to **create payment** for hotel booking on the **Website** so that I can make payment for booking.
- 30) As the **Customer**, I want to be able to **search for rooms or hotel** from the **Website** so that I can retrieve the hotel details to book.
- 31) As the **Customer**, I want to be able to **login** into the **Mobile App** so that I can access the system.
- 32) As the **Customer**, I want to be able to **logout** from the **Mobile App** so that I can exit the system.
- 33) As the **Customer**, I want to be able to **use the QR code scanner** from the **Mobile App** so I can scan the QR code to verify my booking inorder to access the hotel room that I have booked / check-out (sub flow).
- 34) As the **Customer**, I want to be able to **use fingerprint scanner** from the **Mobile App** so that I can access the hotel room using Two-Factor Authentication.
- 35) As the **Customer**, I want to be able to **view my profile** from the **Mobile App** so that I can view my profile.

6. Use Case Diagrams and Descriptions

Use Case # 1

As the **System Admin**,

I want to be able to **login** so that I can manage the customers and the hotel chain client in the **System**.

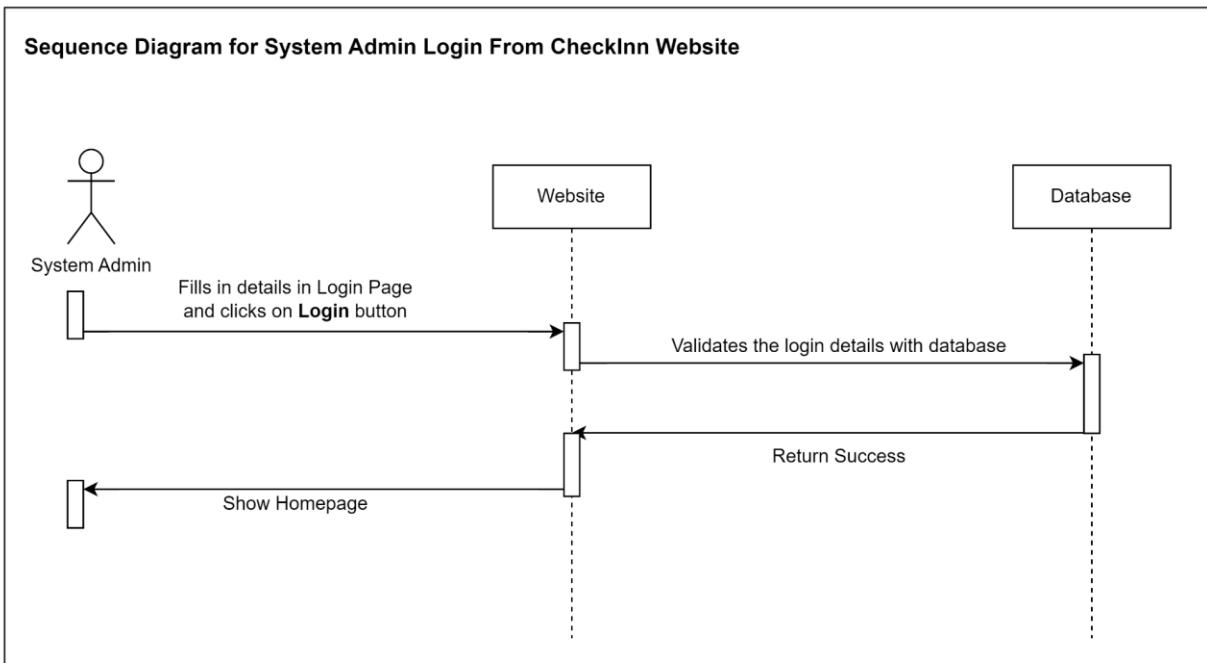


Name: Login	ID: 1
Stakeholders and goals: System Admin - want to login to the system	
Description: The System Admin wants to login to the system to manage customers and hotel operators.	
Actors: System Admin	
Trigger: The System Admin enters the website	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow: <ol style="list-style-type: none"> 1. System will display a login page. 2. System Admin will enter the username and password. 3. System Admin clicks on the “Login” button. 4. System will validate the username and password combination. 5. System will navigate to the homepage. 6. End. 	
Sub-flows:	
Alternative/Exceptional flows: <p><i>Incorrect email and password combination:</i></p> <p>4.1a – The system will pop-up a message dialog box stating “Incorrect Email or Password”. 4.1b – The user will click the “Ok” button to acknowledge and re-enter email and password.</p> <p><i>If blank email:</i></p> <p>4.2a –The system will pop-up a message dialog box stating “Please enter Email”. 4.2b –The user will click the “Ok” button to acknowledge and enter email.</p> <p><i>If blank password:</i></p> <p>4.3a –The system will pop-up a message dialog box stating “Please enter Password”. 4.3b –The user will click the “Ok” button to acknowledge and enter password.</p>	

If email is valid but does not belong to System Admin:

- 4.4a –The system will pop-up a message dialog box stating “User Account Not Found”.
- 4.4b –The user will click the “Ok” button to acknowledge and re-enter email and password.

- Sequence Diagram



- Low-Fidelity Prototype

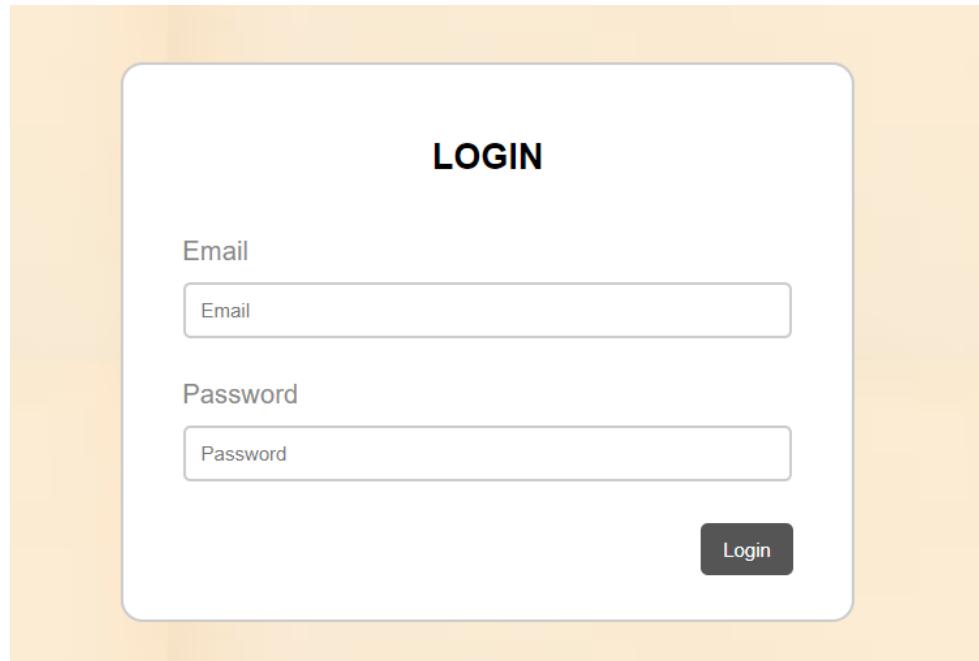
LOGIN

Email

Password

Login

- High-Fidelity Prototype

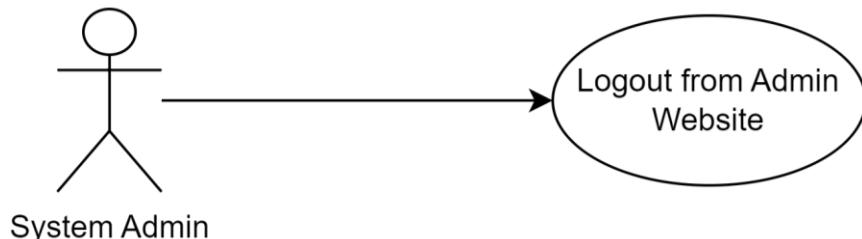


- Test Case

TEST CASE ID:	TC-1						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser						
Purpose	Let user system admin be able to login into the website						
Pass	6						
Fail	0						
Pending	0						
Number of test cases:	6						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin Login Validation:							
TC-1-1	Launch Website	1. User launch website in browser	User sees login page appears on screen	System display login page appears on screen	30/04/2023	Pass	
TC-1-2	Input valid email and password	1. The system will prompt the user for the login credentials: email and password. 2. The user will enter the email and password. 3. The user clicks on the "Login" button. 4. The system will validate the email and password combination. 5. The system will navigate to the landing page.	User will see the website's landing page	System will direct to website's landing page	30/04/2023	Pass	email: Password:
Check System Admin Login Validation:							
TC-1-3	Enters invalid email and password	1. User enter invalid email and password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Incorrect Email or Password"	System pop-up a message dialog box warns "Incorrect Email or Password"	30/04/2023	Pass	email: Password:
TC-1-4	Enters blank email	1. User enter blank email and non-blank password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Please enter Email"	System pop-up a message dialog box warns "Please enter Email"	30/04/2023	Pass	email: Password:
TC-1-5	Enters blank password	1. User enter non-blank email and blank password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Please enter Password"	System pop-up a message dialog box warns "Please enter Password"	30/04/2023	Pass	email: Password:
TC-1-6	Enters valid email but is not System Admin	1. User enter email and password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "User Account Not Found"	System pop-up a message dialog box warns "User Account Not Found"	30/04/2023	Pass	email: Password:

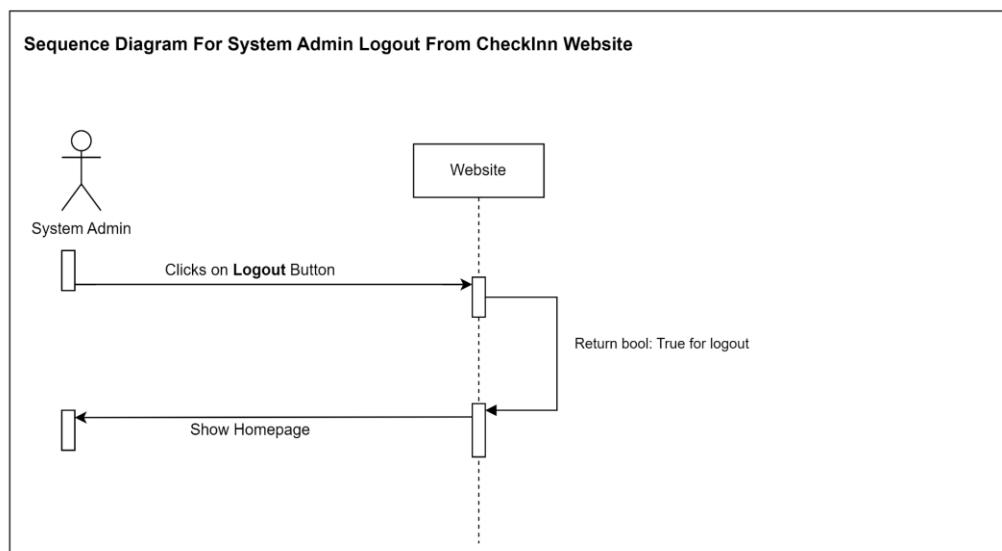
Use Case # 2

As the **System Admin**,
I want to be able to **logout** so that I can exit the **System**.

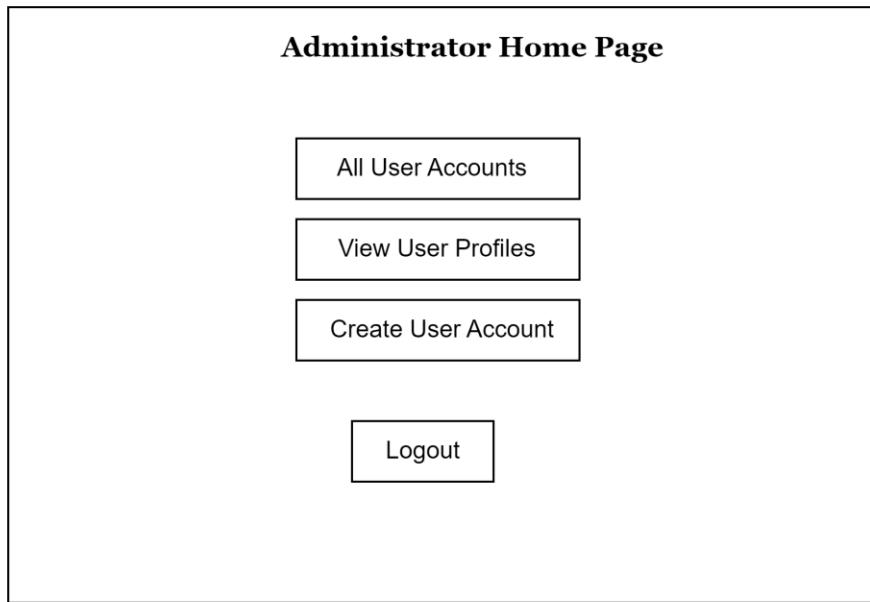


Name: Logout	ID: 2
Stakeholders and goals: System Admin - wants to log out from the system.	
Description: The System Admin wants to log out from the system to stop managing customers and hotel operators.	
Actors: System Admin	
Pre-condition: The system admin must already login into the system	
Trigger: The System Admin clicks on “Log Out” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
1. System Admin clicks on the “Log Out” button. 2. System will direct them to the Login page. 3. End.	
Sub-flows:	
Alternative/Exceptional flows:	

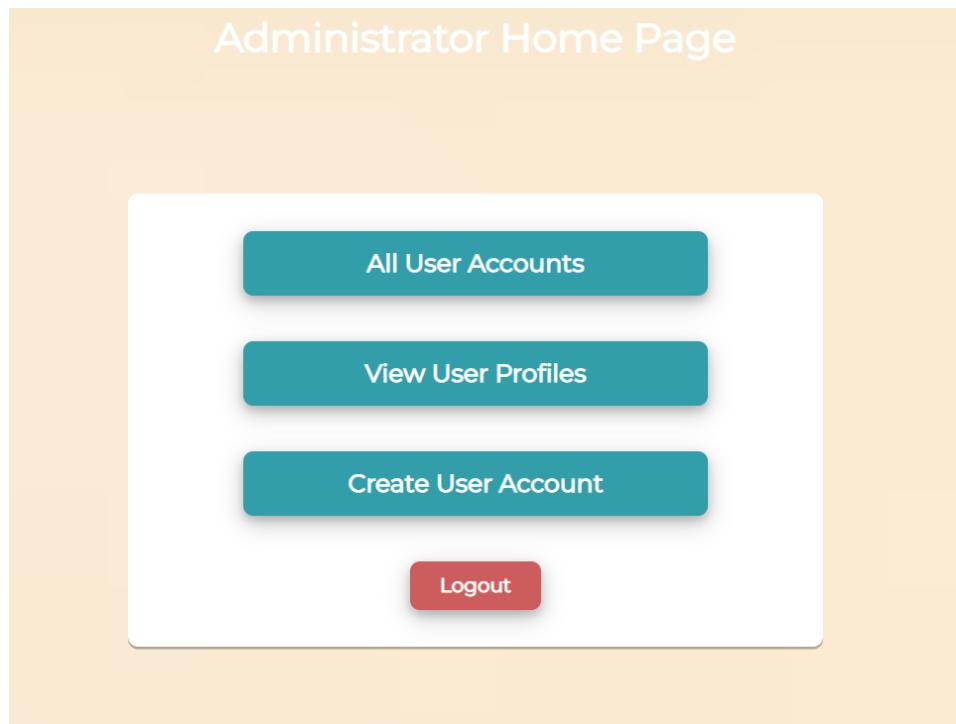
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



- Test Case

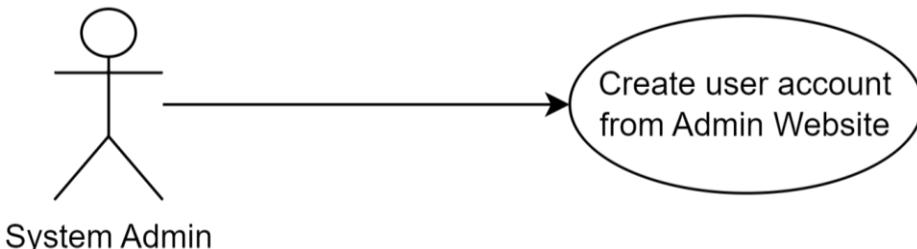


TEST CASE ID:	TC-2						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser						
Purpose	Let user system admin be able to log out of the website						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
TC-2-1	Log out of System Admin account	1. The user clicks on the "Log Out" button. 2. The system will pop-up a message dialog box stating "Log Out successful" 3. The user clicks on the "OK" button. 4. The system will direct user to the "Log in" page.	User is directed user to website's login page	System will direct user to website's login page	30/04/2023	Pass	User returns back to login page

Use Case # 3

As the **System Admin**,

I want to be able to **create a user account** so that I can add a new user into the **System**.



Name: Create a user account	ID: 3
Stakeholders and goals: System Admin – wants to add a new user into the system	
Description: The System Administrator wants to create a user account to add a new user into the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the “Create User Account” button.	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System will display the homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “Create User Account” button. 4. System directs the user to another page for the user to fill up the details. 5. System admin can enter the name, email, contact and select the account type (hotel operator, customer, system admin) in the drop down button. 6. System admin clicks on the “Submit” button. 7. System validates the information given. 8. System creates the account and returns to the homepage. 9. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<p><i>If blank email:</i></p> <p>7.1a –The system will pop-up a message dialog box stating “Email cannot be empty”.</p> <p>7.1b –The user will click the “Ok” button to acknowledge and re-enter email and password.</p>	
<p><i>If blank password:</i></p> <p>7.2a –The system will pop-up a message dialog box stating “Password cannot be empty”.</p> <p>7.2b –The user will click the “Ok” button to acknowledge and re-enter password.</p>	
<p><i>If short password:</i></p> <p>7.3a –The system will pop-up a message dialog box stating “Password needs to be 8 characters long”.</p>	

7.3b –The user will click the “Ok” button to acknowledge and re-enter password.

If blank name:

7.4a –The system will pop-up a message dialog box stating “Name cannot be empty”.

7.4b –The user will click the “Ok” button to acknowledge and re-enter email and password.

If blank date of birth:

7.5a –The system will pop-up a message dialog box stating “Date of Birth cannot be empty”.

7.5b –The user will click the “Ok” button to acknowledge and re-enter password.

If invalid email:

7.6a –The system will pop-up a message dialog box stating “Invalid email”.

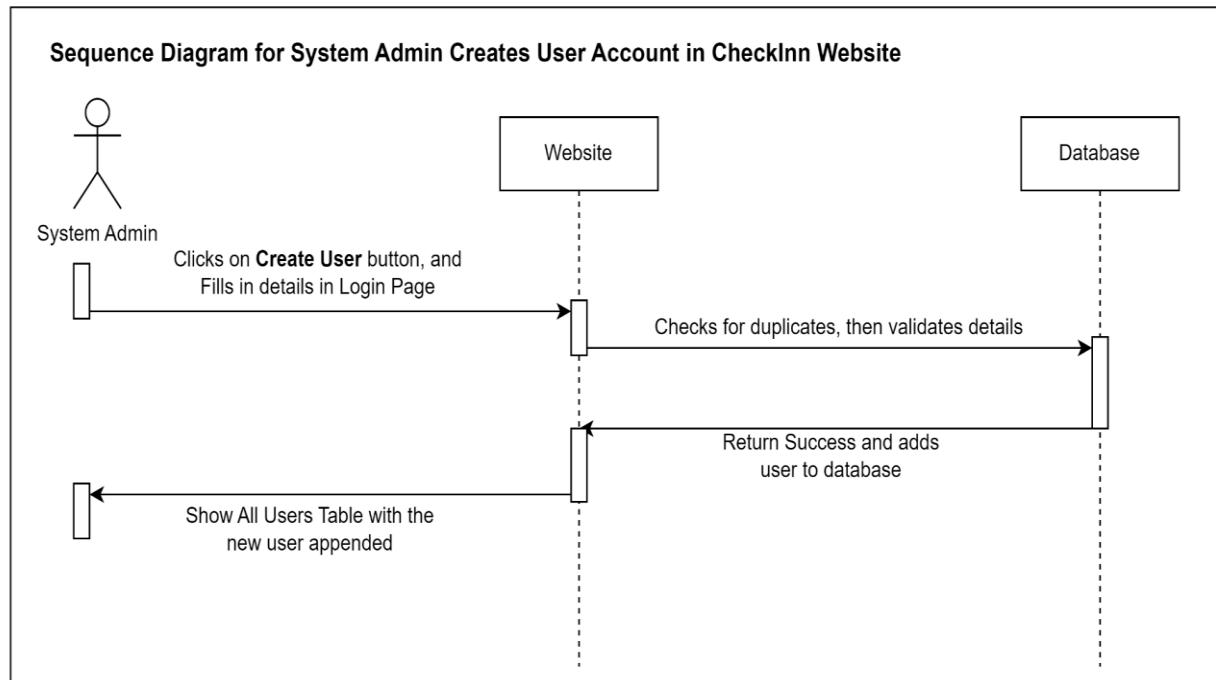
7.6b –The user will click the “Ok” button to acknowledge and re-enter email.

If email has already been used:

7.7a –The system will pop-up a message dialog box stating “Email is already being used. Please enter a different email”.

7.7b –The user will click the “Ok” button to acknowledge and re-enter email.

- Sequence Diagram



- Low-Fidelity Prototype

Create User Account

Email:

Password:

Name:

Date of Birth: dd/mm/yyyy

Profile:

Status: Active

- High-Fidelity Prototype

Create User Account

Email:

Password:

Name:

Date of Birth: dd/mm/yyyy

Profile: Administrator

Status: Active

- Test Case

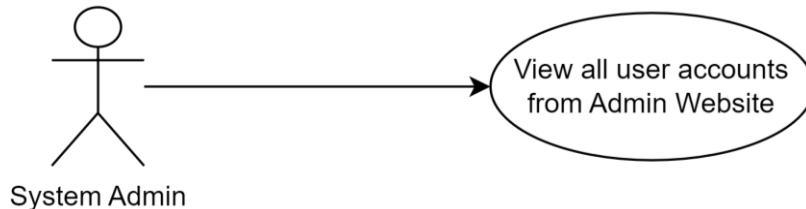


TEST CASE ID:	TC-3						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser						
Purpose	Let user system admin be able to create user account						
Pass	10						
Fail	0						
Pending	0						
Number of test cases:	10						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin create user account:							
TC-3-1	Access Create user account page	1. User clicks on "Create User Account" button	User direct user to "Create User Account" page	System direct user to "Create User Account" page	30/01/2023	Pass	
TC-3-2	Create new user account	1. The user clicks the "Create User Account" button. 2. The system directs the user to "Create User Account" page. 3. The user enters the details needed to create an account and click the "Create user" button. 4. The system directs the user back to the "View All User Accounts" page of the website.	User will see the newly created user account in All Users table in the landing page	System adds the new user account to database AND direct user to landing page displaying the newly created user account in the "All Users" table	30/01/2023	Pass	
Check System Admin return back/exit Create user account page:							
TC-3-3	Returns back to website's landing page from Create user account page	1. User clicks on "Back" button	User is redirected to website's landing page	System direct user back to website's landing page	30/01/2023	Pass	
Check System Admin create user account:							
TC-3-4	Enter blank email	1. User clicks on "Create User" button 2. User input blank email 3. User input password, name and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Email cannot be empty"	System pop-up a message dialog box displaying "Email cannot be empty."	30/01/2023	Pass	
TC-3-5	Enter blank password	1. User clicks on "Create User" button 2. User input blank password 3. User input email, name and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Password cannot be empty"	System pop-up a message dialog box displaying "Password cannot be empty."	30/01/2023	Pass	
TC-3-6	Enter short password	1. User clicks on "Create User" button 2. User input short password 3. User input email, name and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Password needs to be 8 characters long"	System pop-up a message dialog box displaying "Password needs to be 8 characters long"	30/01/2023	Pass	
TC-3-7	Enter blank name	1. User clicks on "Create User" button 2. User input blank name 3. User input email, password and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Name cannot be empty"	System pop-up a message dialog box displaying "Name cannot be empty."	30/01/2023	Pass	
TC-3-8	Enter blank date of birth	1. User clicks on "Create User" button 2. User input blank name 3. User input email, password and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Date of Birth cannot be empty."	System pop-up a message dialog box displaying "Date of Birth cannot be empty."	30/01/2023	Pass	
TC-3-9	Enter invalid email	1. User clicks on "Create User" button 2. User input email 3. User input password, name and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up message dialog box displaying "Invalid Email!"	System pop-up a message dialog box displaying "Invalid Email!"	30/01/2023	Pass	
TC-3-10	Enter used email	1. User clicks on "Create User" button 2. User input used email 3. User input password, name and date of birth 4. User selects user profile in drop down list 5. User clicks on "Submit" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Email is already being used. Please enter a different email"	System pop-up a message dialog box displaying "User with same Email already exists."	30/01/2023	Pass	

Use Case # 4

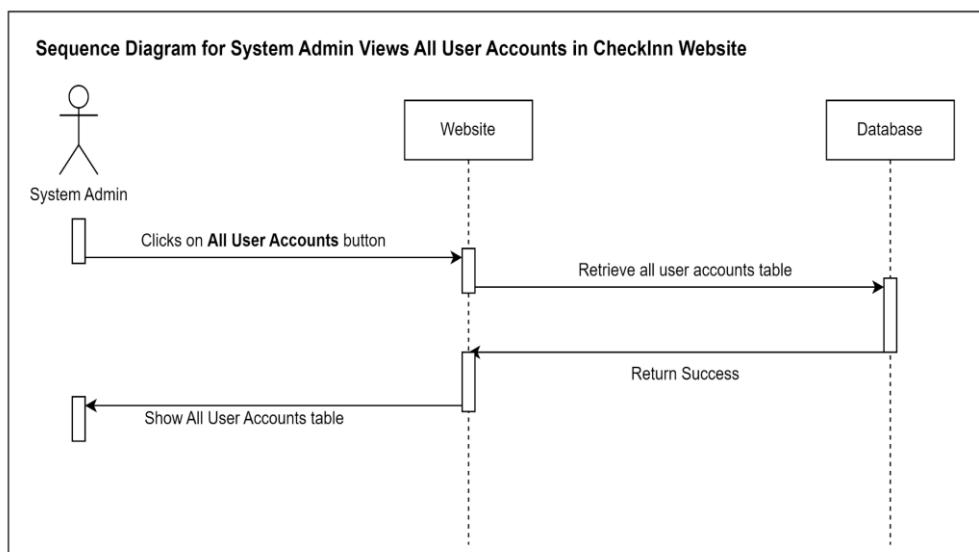
As the **System Admin**,

I want to be able to **view all user accounts** so that I can retrieve the user account details list from the **System**.



Name: View all user accounts	ID: 4
Stakeholders and goals: System Admin - wants to retrieve the user account details list	
Description: The System Administrator wants to view all user accounts to retrieve the user account details list in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the “All User Accounts” button.	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System will display the homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User Accounts” button. 4. System directs the user to another page where all the users are displayed in rows. 5. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

View All User Accounts								
Search Name or Email	Search							
Email	Password	Name	DOB	Profile	Status	Edit	Suspend	Delete
e@gmail.com	*****	eugene	2003-05-16	Hotel Operator	Active	Edit	Suspend	Delete
f@gmail.com	*****	fourth	2010-06-07	Administrator	Inactive	Edit	Suspend	Delete
View User Accounts							Back	

- High-Fidelity Prototype

View All User Accounts										
Search Name or Email										
Email	Password	Name	DOB	Profile	Status	Edit	Suspend	Delete		
customer@test.com	*****	customer	2023-05-16	Customer	Active	Edit	Suspend	Delete		
hbs@test.com	*****	James	2000-10-03	Hotel Operator	Active	Edit	Suspend	Delete		
anica@test.com	*****	anica	2023-05-09	Customer	Active	Edit	Suspend	Delete		
hba@test.com	*****	John	1993-01-25	Hotel Operator	Active	Edit	Suspend	Delete		
hkg@test.com	*****	Kai	1985-03-08	Hotel Operator	Active	Edit	Suspend	Delete		
hma@test.com	*****	Maria	1994-01-06	Hotel Operator	Active	Edit	Suspend	Delete		
admin@test.com	*****	admin	1991-02-06	Administrator	Active	Edit	Suspend	Delete		
hi@test.com	*****	hi	2023-05-03	Customer	Active	Edit	Suspend	Delete		
r@gmail.com	*****	Administrator	2023-05-09	Administrator	Active	Edit	Suspend	Delete		
helly@gmail.com	*****	helly	2023-05-09	Administrator	Active	Edit	Suspend	Delete		
View User Accounts							Back			

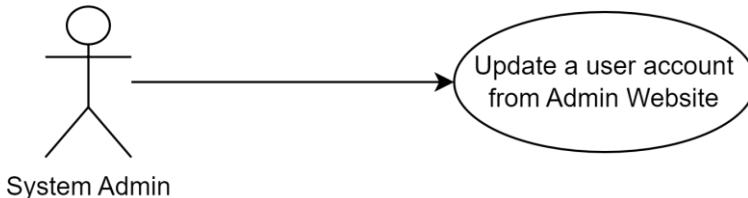
- Test Case

TEST CASE ID:	TC-4									
System Name:	CheckInn Admin Website									
Module Code:	CSIT331									
Test requirement:	1. Launch CheckInn Admin Website in the browser									
Purpose:	Let user system admin view all user accounts									
Pass:	1									
Fail:	0									
Pending:	0									
Number of test cases:	1									
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note			
Check System Admin View all user accounts:										
TC-1-1	Login System Admin account.	1. The user clicks on "All User Accounts". 2. The system directs the user to the landing page that displays all user's account details in a table. 3. The user will scroll through the table list to view all user's account details.	User will direct user to website's landing page and see a table of all user accounts in the landing page	System will direct user to website's landing page and display all user's account details in a table.	30/04/2023	Pass	Landing page displays all user's account details in a scrollable table list.			

Use Case # 5

As the **System Admin**,

I want to be able to **update a user account** so that I can edit the account information of a user in the **System**.



Name: Update user accounts	ID: 5
Stakeholders and goals: System Admin - wants to update a user account in the system	
Description: The System Administrator wants to update a user account to edit the account information of a user in the system.	
Actors: System Admin	
Pre-Condition: User account must be created	
Trigger: The system administrator clicks on the “Edit” button.	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System displays the homepage that shows3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User Accounts” button. 4. System directs the user to another page where all the users are displayed in rows. 5. System admin needs to select the user from the list. 6. System will display the details of the selected user. 7. System admin clicks on the “Edit” button and changes the necessary details. 8. System admin clicks on the “Submit” button. 9. System will save the updated account details. 10. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<p><i>If blank password:</i></p> <p>7.1a –The system will pop-up a message dialog box stating “Password cannot be empty”.</p> <p>7.1b –The user will click the “Ok” button to acknowledge and re-enter password.</p>	
<p><i>If short password:</i></p> <p>7.2a –The system will pop-up a message dialog box stating “Password needs to be 8 characters long”.</p> <p>7.2b –The user will click the “Ok” button to acknowledge and re-enter password.</p>	
<p><i>If blank name:</i></p> <p>7.3a –The system will pop-up a message dialog box stating “Name cannot be empty”.</p>	

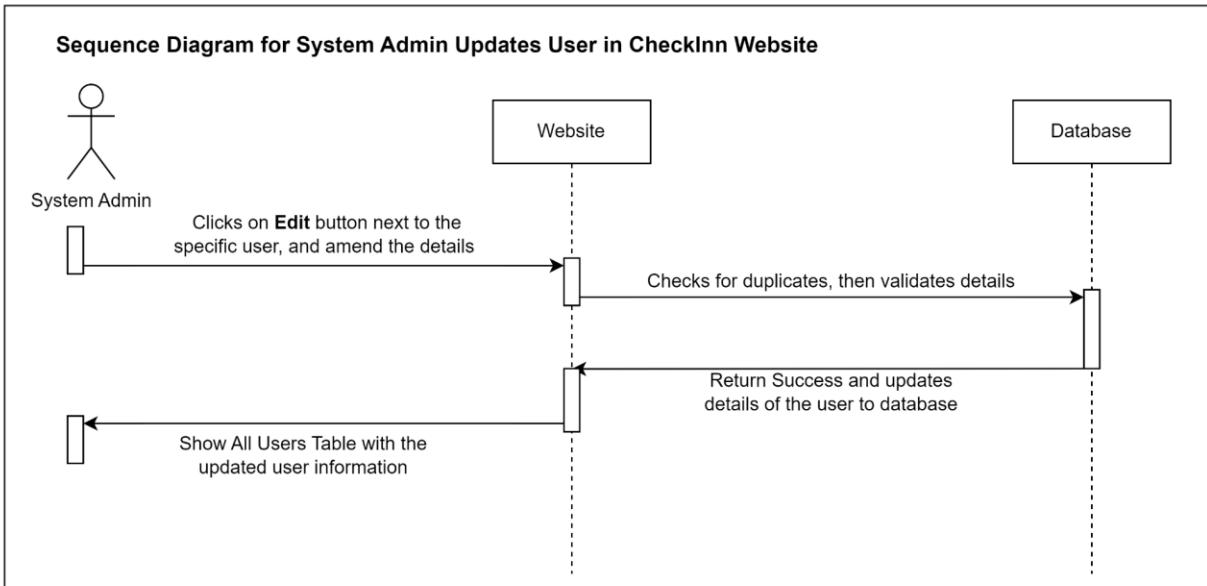
7.3b –The user will click the “Ok” button to acknowledge and re-enter email and password.

If blank date of birth:

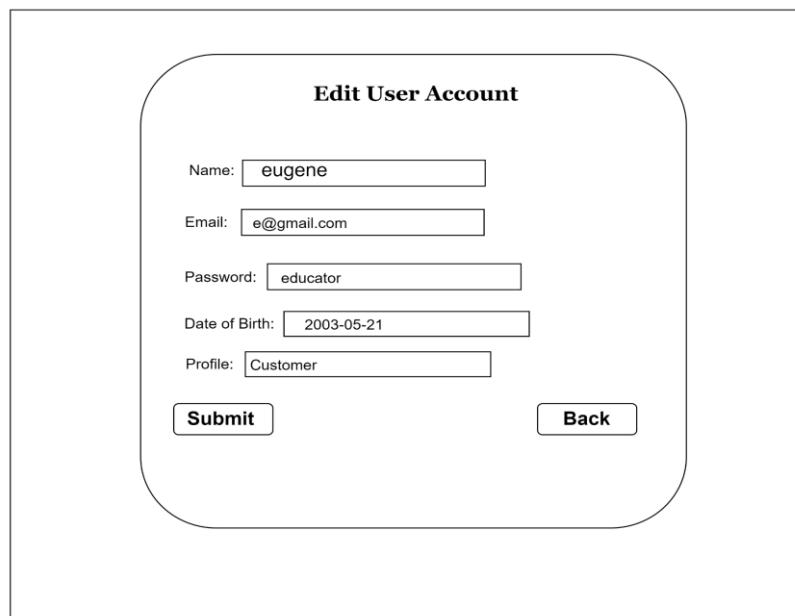
7.4a –The system will pop-up a message dialog box stating “Date of Birth cannot be empty”.

7.4b –The user will click the “Ok” button to acknowledge and re-enter password.

- Sequence Diagram



- Low-Fidelity Prototype



The low-fidelity prototype shows a rounded rectangular form titled "Edit User Account". Inside the form, there are five input fields with placeholder text: "Name: eugene", "Email: e@gmail.com", "Password: educator", "Date of Birth: 2003-05-21", and "Profile: Customer". Below these fields are two buttons: "Submit" on the left and "Back" on the right.

- High-Fidelity Prototype

Edit User Account

Name: Customer

Email: customer@test.com

Password: customer

Date of Birth: 03/05/2023

Profile: Administrator ▾

Submit **Back**

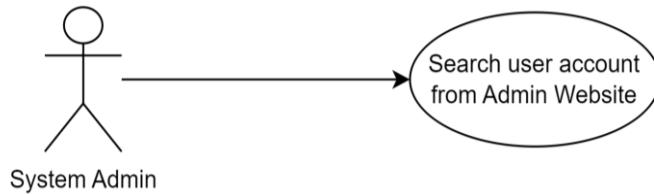
- Test Case

TEST CASE ID:	TC-5						
System Name:	CheckIn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckIn Admin Website in the browser 2. Has Successfully logged in 3. Clicks on "All User Accounts" button						
Purpose:	Let user system admin updates user's account						
Pass	0						
Fail	0						
Pending	0						
Number of test cases:	6						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin edit user's account:							
TC-5-1	Edit User Account	1. The user selects a user's account in the table. 2. The user clicks on the "Edit" button. 3. The system directs the user to the "Edit User Account" page. 4. The user can make any amendments to the user's account. 5. The user clicks on the "Save" button. 6. The system directs the user back to the landing page of the website, with the updated information shown in table	User see a the edited user's account in the table in the landing page	System updates the user details in the database and direct user to landing page displaying the edited user's account in the table	30/04/2023	Pass	-
Check System Admin return back/exit Edit User page:							
TC-5-2	Returns back to website's landing page from Edit user account form.	1. User clicks on "Back" button	User is directed to user back to website's landing page	System direct user back to website's landing page	30/04/2023	Pass	-
Check System Admin edit user's account:							
TC-5-3	Enter blank password	1. User left password blank. 2. User clicks on "Submit" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Password cannot be empty."	System pop-up a message dialog box displaying "Password cannot be empty."	30/04/2023	Pass	-
TC-5-4	Enter short password	1. User input short password 2. User clicks on "Submit" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Password needs to be 8 characters long"	System pop-up a message dialog box displaying "Password needs to be 8 characters long"	30/04/2023	Pass	-
TC-5-5	Enter blank name	1. User left name blank 2. User clicks on "Submit" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Name cannot be empty."	System pop-up a message dialog box displaying "Name cannot be empty."	30/04/2023	Pass	-
TC-5-6	Enter blank date of birth	1. User left date of birth blank. 2. User clicks on "Submit" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Date of Birth cannot be empty."	System pop-up a message dialog box displaying "Date of Birth cannot be empty."	30/04/2023	Pass	-

Use Case # 6

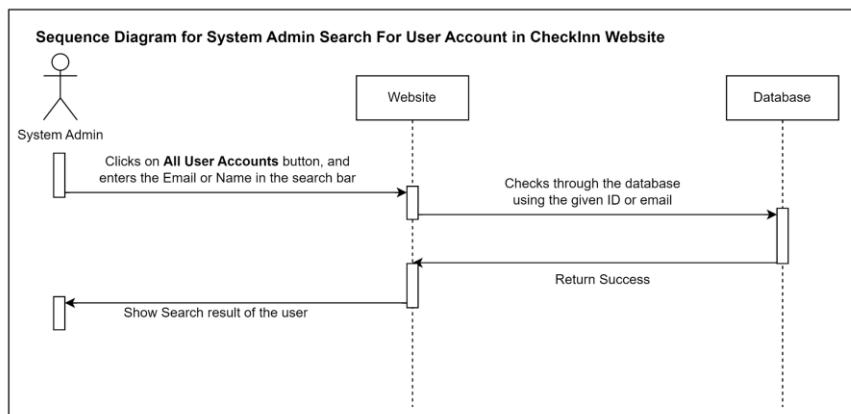
As the **System Admin**,

I want to be able to **search for a specific user account** so that I can retrieve the user account in the **System**.



Name: Search for specific user accounts	ID: 6
Stakeholders and goals: System Admin - wants to search for a specific user account.	
Description: The System Administrator wants to search for a specific user account to retrieve the user account in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the Search Bar	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User accounts” button. 4. System directs the user to another page where all the users are displayed in rows and a search bar. 5. System admin clicks on the search bar and enters the user account that they wish to search for and click the search button. 6. System search for the user account and displays the user accounts that match the searched name. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

Search in User Accounts using Name:

View All User Accounts

<input type="text" value="eugene"/> <input type="button" value="Search"/>								
Email	Password	Name	Date of Birth	Position	Status	Edit	Suspend	Delete
e@gmail.com	*****	eugene	2003-05-21	Hotel Operator	Inactive	Edit	Suspend	Delete

[View User Accounts](#) [Back](#)

- High-Fidelity Prototype

Search in User Accounts using Name:

All User Accounts

<input type="text" value="Search Name or Email"/> <input type="button" value="Search"/>								
Email	Password	Name	Date of Birth	Position	Status	Edit	Suspend	Delete
customer@test.com	password	customer	0000-00-00	Customer	Active	Edit	Suspend	Delete

[View User Accounts](#) [Back](#)

- Low-Fidelity Prototype

Search in User Accounts using Email:

View All User Accounts

f@gmail.com	Search							
Email	Password	Name	Date of Birth	Position	Status	Edit	Suspend	Delete
f@gmail.com	*****	Fourth	2010-06-07	Administrator	Inactive	Edit	Suspend	Delete

[View User Accounts](#) [Back](#)

- High-Fidelity Prototype

Search in User Accounts using Email:

All User Accounts

<input type="text" value="Search Name or Email"/> <input type="button" value="Search"/>								
Email	Password	Name	Date of Birth	Position	Status	Edit	Suspend	Delete
ho@test.com	hoteloperator	hotel	2000-10-03	Hotel Operator	Active	Edit	Suspend	Delete
View User Accounts Back								

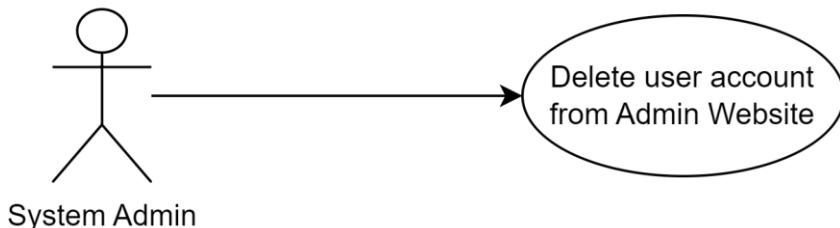
- Test Case

Test Case Details							
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin search user account:							
TC-6-1	Search User account using Email	1. The user clicks on the "Search for user account" button. 2. The system directs the user to the "Search" page. 3. The user can enter the Email inside the search text box. 4. The user clicks on the search button. 5. The system would display the accounts in the "Accounts" table that matches the search input.	User directs user to the "Search Page" and see search results in scrollable table with closest similarities	System directs user to the "Search Page" and displays search result in scrollable table with closest similarities result	30/04/2023	Pass	
TC-6-2	Search User account using Name	1. The user clicks on the "Search for user account" button. 2. The system directs the user to the "Search" page. 3. The user can enter the Name inside the search text box. 4. The user clicks on the search button. 5. The system would display the accounts in the "Accounts" table that matches the search input.	User directs user to the "Search Page" and see search results in scrollable table with closest similarities	System directs user to the "Search Page" and displays search result in scrollable table with closest similarities result	30/04/2023	Pass	

Use Case # 7

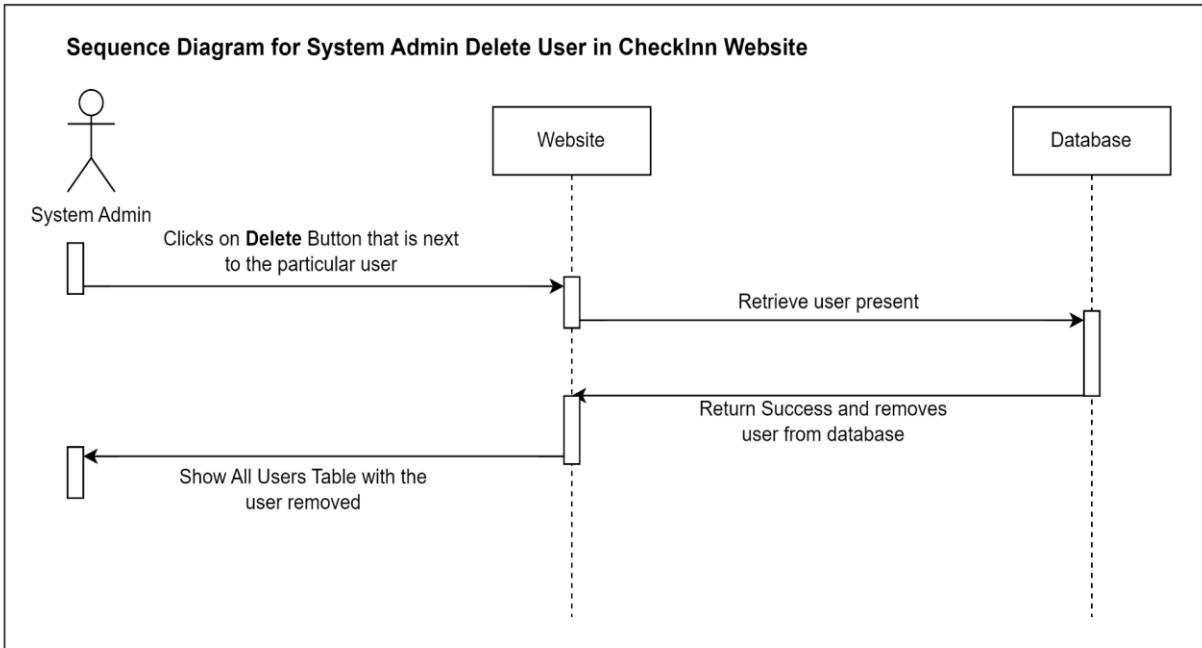
As the **System Admin**,

I want to be able to **delete a user account** so that I can delete the user account in the **System**.



Name: Delete a user account	ID: 7
Stakeholders and goals: System Admin - wants to delete a user account.	
Description: The System Administrator wants to delete a user account to delete the user account in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the “Delete” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User Accounts” button. 4. System directs the user to another page where all the users are displayed in rows. 5. System admin clicks on the “Delete” button. 6. System confirms the account deletion with the system admin. 7. System admin clicks on the “Confirm” button. 8. System deletes the account. 9. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

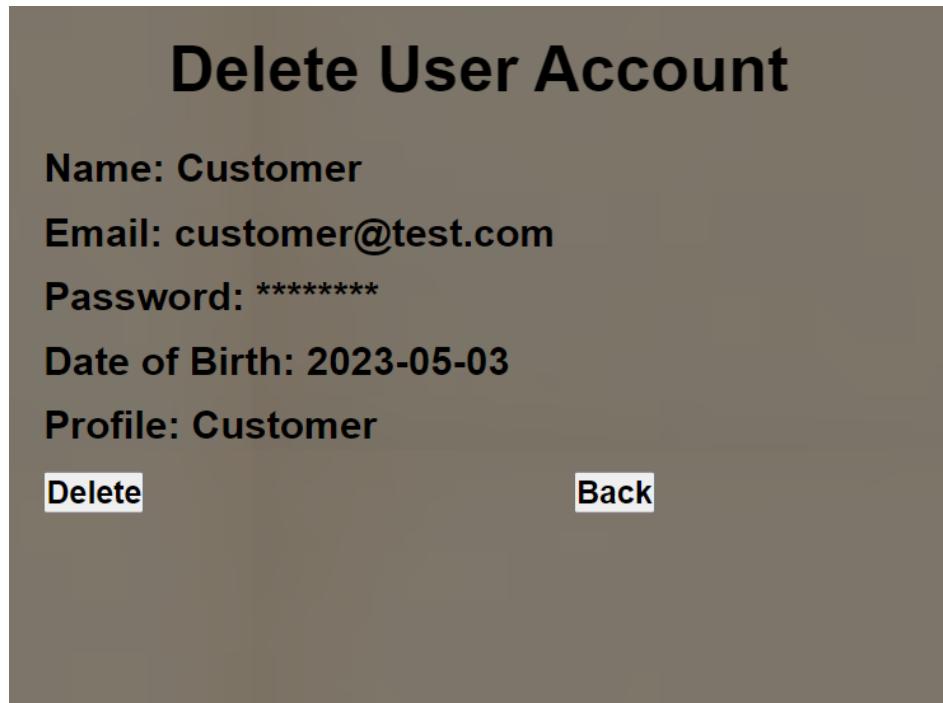
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



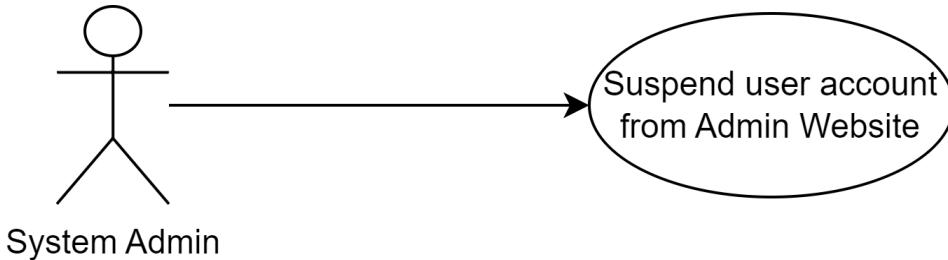
- Test Case

Test Case ID:	TC-7						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser 2. Has Successfully logged in 3. Clicks on 'All User Accounts' button						
Purpose	Let user system admin delete user account						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin view all user account page:							
TC-7-1	Delete a user account	1. The user clicks on "Delete" button next to the specific user account	User sees the updated table of all user account in 'All User Accounts' page	System removes the specific user account and displays the updated table of all user profiles	30/04/2023	Pass	

Use Case # 8

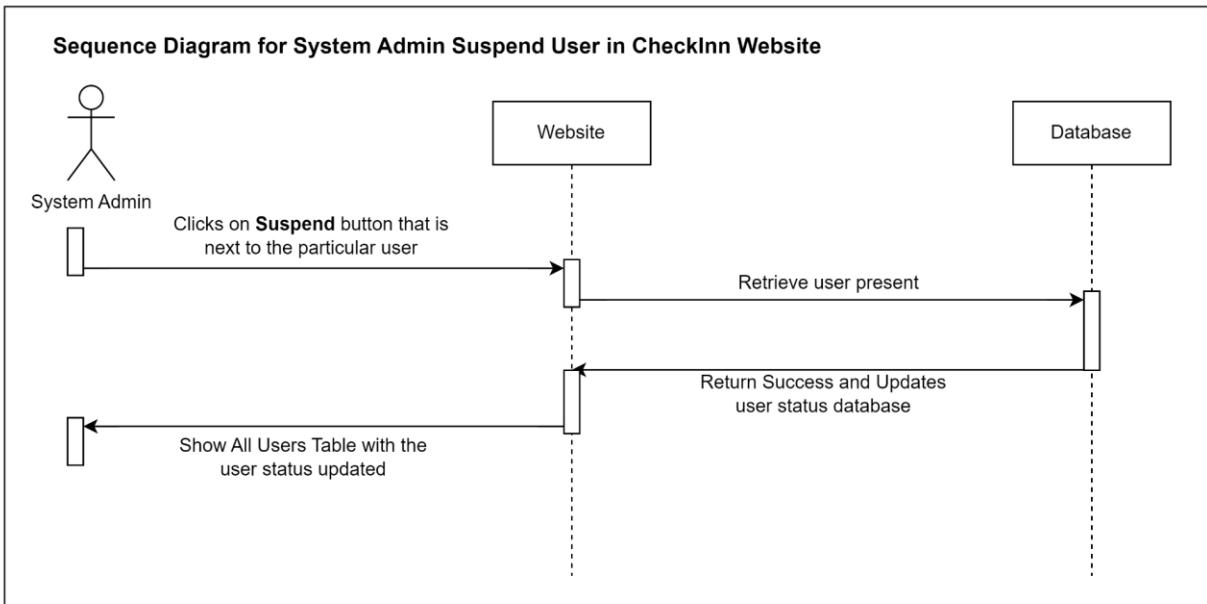
As the **System Admin**,

I want to be able to **suspend a user account** so that I can deactivate the user account due to inactivity for a very long period of time in the **System**.



Name: Suspend a user account	ID: 8
Stakeholders and goals: System Admin - wants to suspend a user account.	
Description: The System Administrator wants to suspend a user account to deactivate the user account due to inactivity for a very long period of time in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the “Suspend” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User Accounts” button. 4. System directs the user to another page where all the users are displayed in rows. 5. System admin clicks on the “Suspend” button. 6. System updates the account status as “Inactive” and shows the updated page. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

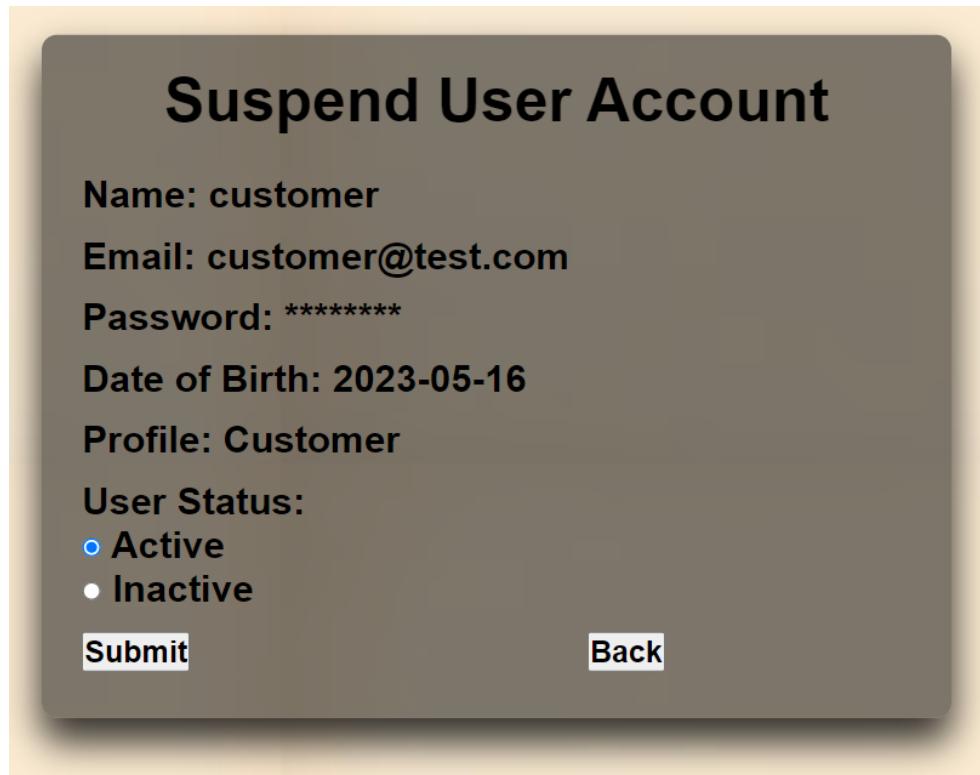
Suspend User Account

Name: Eugene
Email: e@gmail.com
Password: *****
Date of Birth: 2003-05-21
Profile: Hotel Operator
User Status:

- Active
- Inactive

Submit
Back

- High-Fidelity Prototype



Suspend User Account

Name: customer

Email: customer@test.com

Password: *****

Date of Birth: 2023-05-16

Profile: Customer

User Status:

Active

Inactive

Submit **Back**

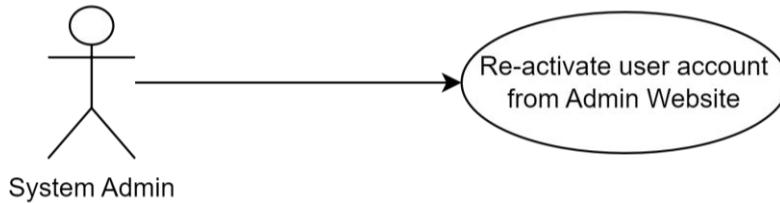
- Test Case

Test Case ID:	TC-B						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser 2. Has Successfully logged in 3. Clicks on "All User Accounts" button						
Purpose	Let user system admin suspend user account						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin suspend a user account:							
TC-B-1	Suspend a user account	1. The user clicks on "Suspend" button next to the specific user account	User sees the updated table of all user account in "All User Accounts" page	System removes the specific user account and displays the updated table of all user profiles	30/04/2023	Pass	Logged in as System Admin

Use Case # 9

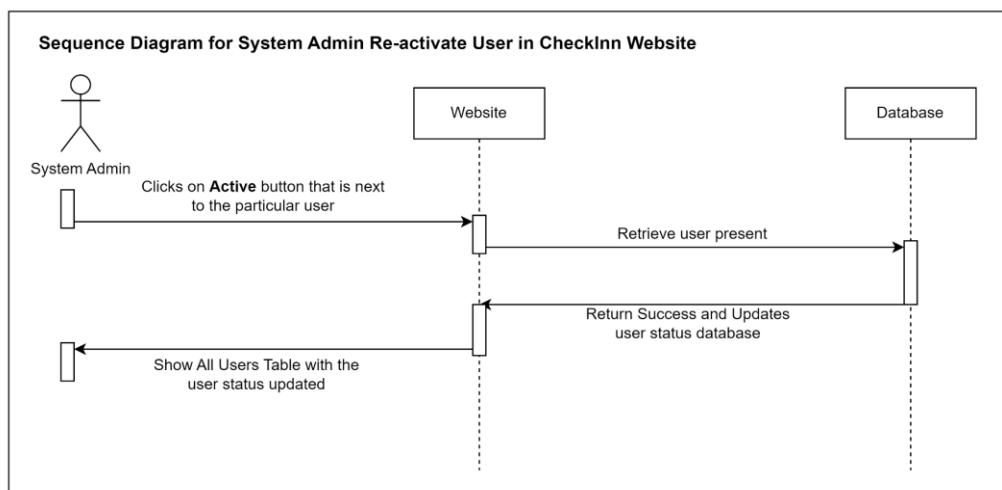
As the **System Admin**,

I want to be able to **re-activate a user account** so that I can re-activate the user account in the **System**.



Name: Reactivate a user account	ID: 9
Stakeholders and goals: System Admin - wants to reactivate a user account.	
Description: The System Administrator wants to reactivate a user account to allow the user to continue using the website or application in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the “Active” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “All User Accounts” button. 4. System directs the user to another page where all the users are displayed in rows. 5. System admin clicks on the “Suspend” button. 6. System updates the account status as “Active” and shows the updated page. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype



Suspend User Account

Name: Eugene

Email: e@gmail.com

Password: *****

Date of Birth: 2003-05-21

Profile: Hotel Operator

User Status:

- Active
- Inactive

Submit **Back**

- High-Fidelity Prototype

Suspend User Account

Name: customer

Email: customer@test.com

Password: *****

Date of Birth: 2023-05-16

Profile: Customer

User Status:

- Active
- Inactive

Submit **Back**

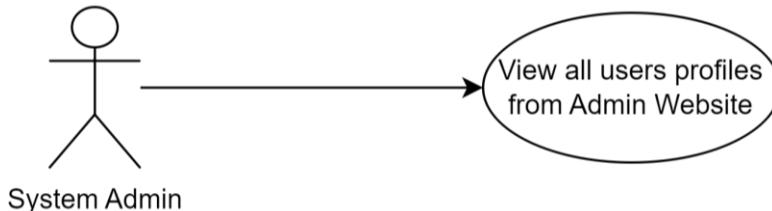
- Test Case

Test Case ID:	TC-9						
System Name:	CheckInn Admin Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Admin Website in the browser 2. Has Successfully logged in 3. Clicks on "All User Accounts" button						
Purpose	Let user system admin re-activate user account						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin re-activate user account:							
TC-9-1	Re-activate a user account	1. The user clicks on "Active" button next to the specific user account	User sees the updated table of all user account in "All User Accounts" page.	System removes the specific user account and displays the updated table of all user profiles	30/04/2023	Pass	*

Use Case # 10

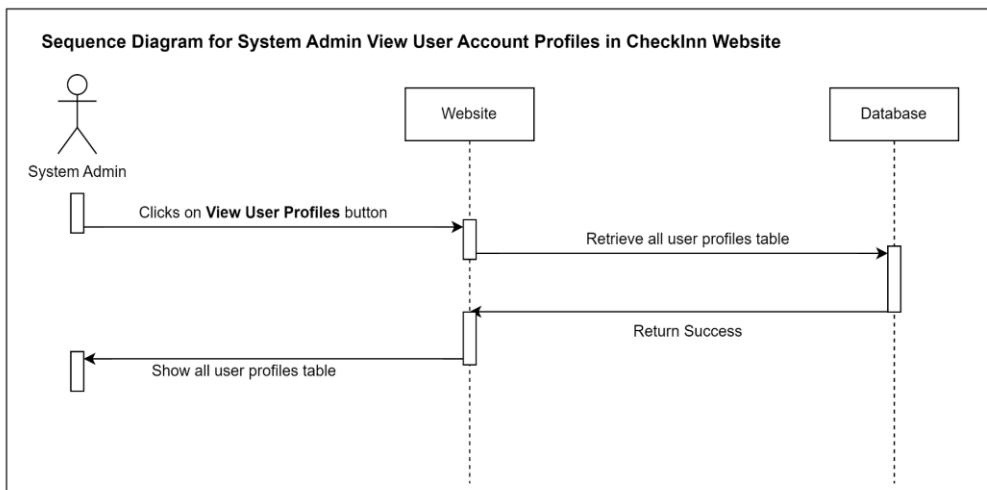
As the **System Admin**,

I want to be able to **view all user profiles** so that I can retrieve the data based on the user profile in the **System**.



Name: View all user profiles	ID: 10
Stakeholders and goals: System Admin - wants to view all user profiles	
Description: The System Administrator wants to view user profiles to retrieve the data based on the user profile in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on “User Profiles” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows3 buttons: “All User Accounts”, “View User Profiles” and “Create User Account”. 3. System admin clicks on the “View User Profiles” button. 4. System directs the user to another page where all the users' profiles are displayed in rows. 5. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

All User Profiles

Email	Name	Profile	View
e@gmail.com	eugene	Hotel Operator	View
f@gmail.com	fourth	Administrator	View

- High-Fidelity Prototype

All User Profiles

Email	Name	Profile	View
customer@test.com	customer	Customer	View
hbs@test.com	James	Hotel Operator	View
anica@test.com	anica	Customer	View
hba@test.com	John	Hotel Operator	View
hkg@test.com	Kai	Hotel Operator	View
hma@test.com	Maria	Hotel Operator	View
admin@test.com	admin	Administrator	View
hi@test.com	hi	Customer	View
r@gmail.com	Administrator	Administrator	View
helly@gmail.com	helly	Administrator	View

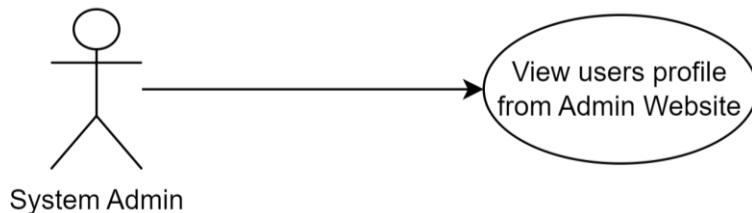
- Test Case

Test Case ID:	TC-10						
System Name:	CheckInn Admin Website						
Module Code:	(SCI 321)						
Test requirements:	1. Launch CheckInn Admin Website in the browser 2. Successfully login to the System Admin website						
Purpose:	Let user system admin view all user profile						
Pass:	1						
Fail:	0						
Pending:	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin view all user profile page:							
TC-10-1	Access View all User Profile Page	1. The user clicks on the "User Profiles" button. 2. The system directs the user to the "All User Profile" page. 3. The system displays a list of all user's profiles. 4. The user will scroll through the table list to view all user's account details.	User sees a table of all user profiles in "All User Profiles" page	System directs user to "View all User Profiles" page and displays a table of all user profiles	30/01/2023	Pass	

Use Case # 11

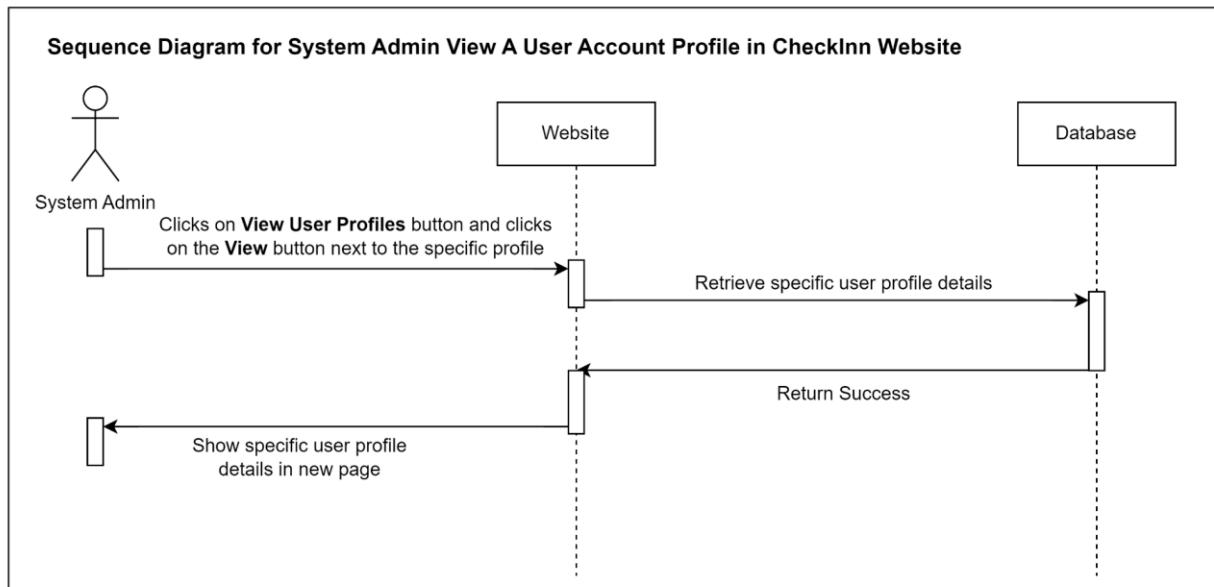
As the **System Admin**,

I want to be able to **view a user profile** so that I can retrieve the data based on the user profile in the System.

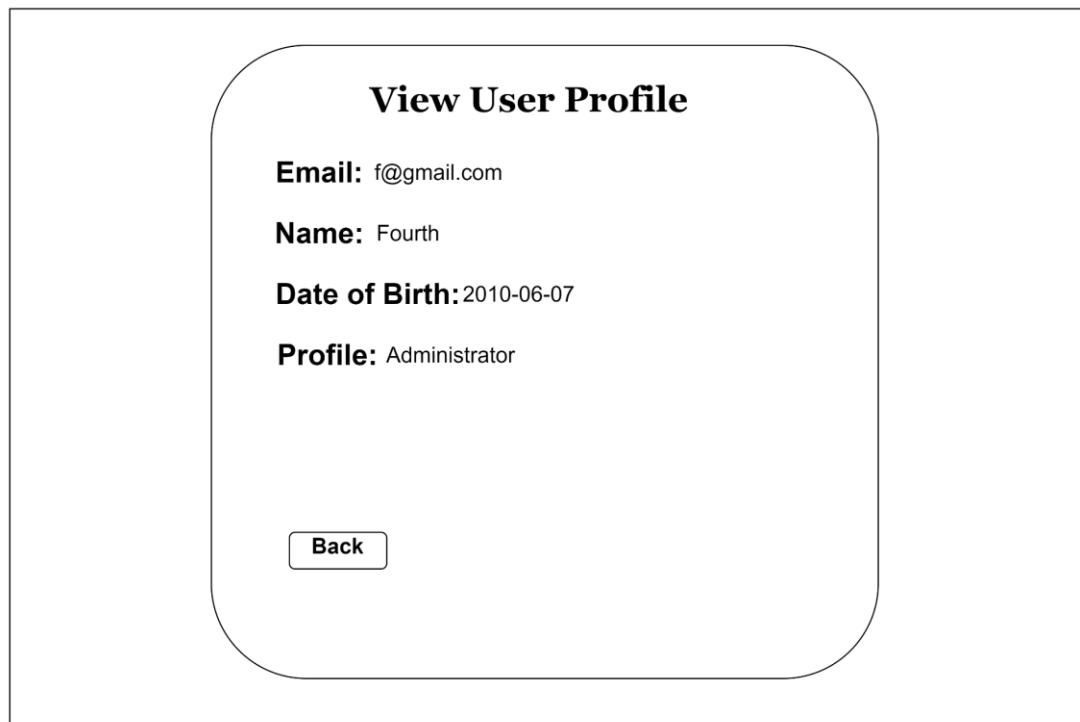


Name: View a user profile	ID: 11
Stakeholders and goals: System Admin - wants to view a user profile	
Description: The System Administrator wants to view a user profile to retrieve the data based on the user profile in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on “View” button	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 2 buttons: “All User Accounts” and “User Profiles” 3. System admin clicks on the “User Profiles” button. 4. System directs the user to another page where all the users' profiles are displayed in rows. 5. System admin clicks on the “View” button. 6. System directs the user to another page where the specific user profile is shown in detail. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

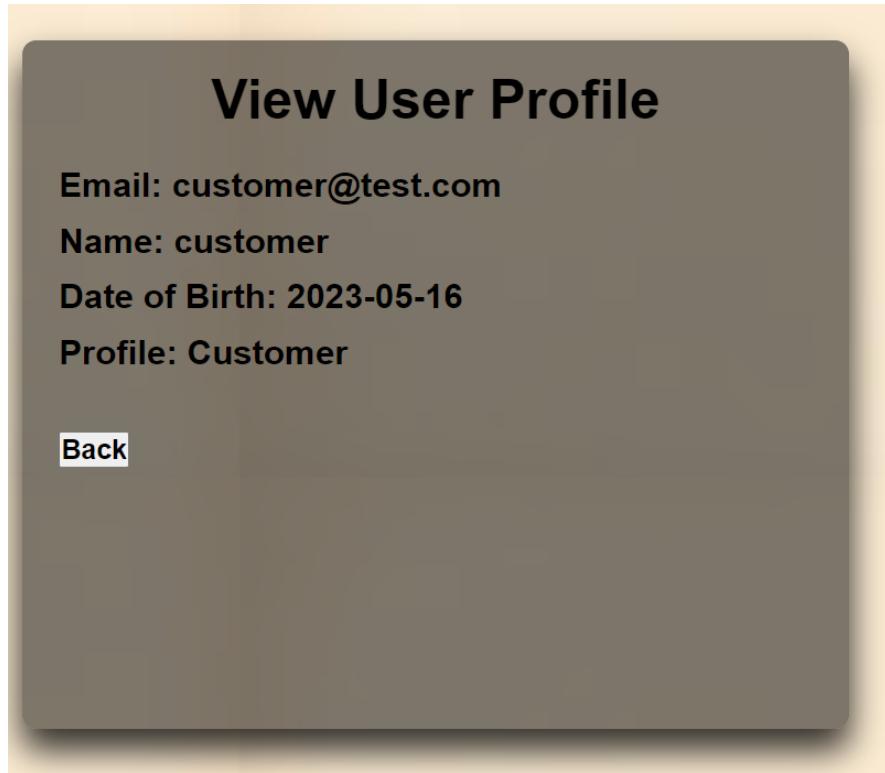
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



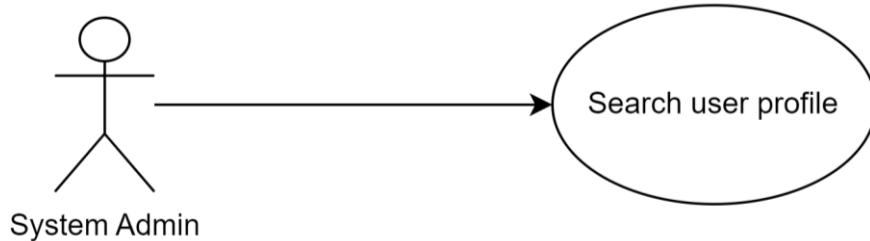
- Test Case

Test Case ID:	TC-11	System Name:	CheckInn Admin Website	Module Code:	CSCI 321	Test requirements:	1. Launch CheckInn Admin Website in the browser 2. Successfully logins to the System Admin website	Purpose:	Let user system admin view a user profile	Pass:	1	Fail:	0	Pending:	0	Number of test cases:	1
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note										
Check System Admin view a user profile page:																	
TC-11-1	Access View all User Profile Page	1. The user clicks on "User Profiles" button 2. The system directs the user to the "All User Profile" page. 3. The system displays a list of all user's profiles. 4. The user clicks on "View" button.	User sees details of a specific user profile details in "User Profile" page	System directs user to "View User Profiles" page and displays a details of specific user profile	30/04/2023	Pass	Logged in as System Admin										

Use Case #12

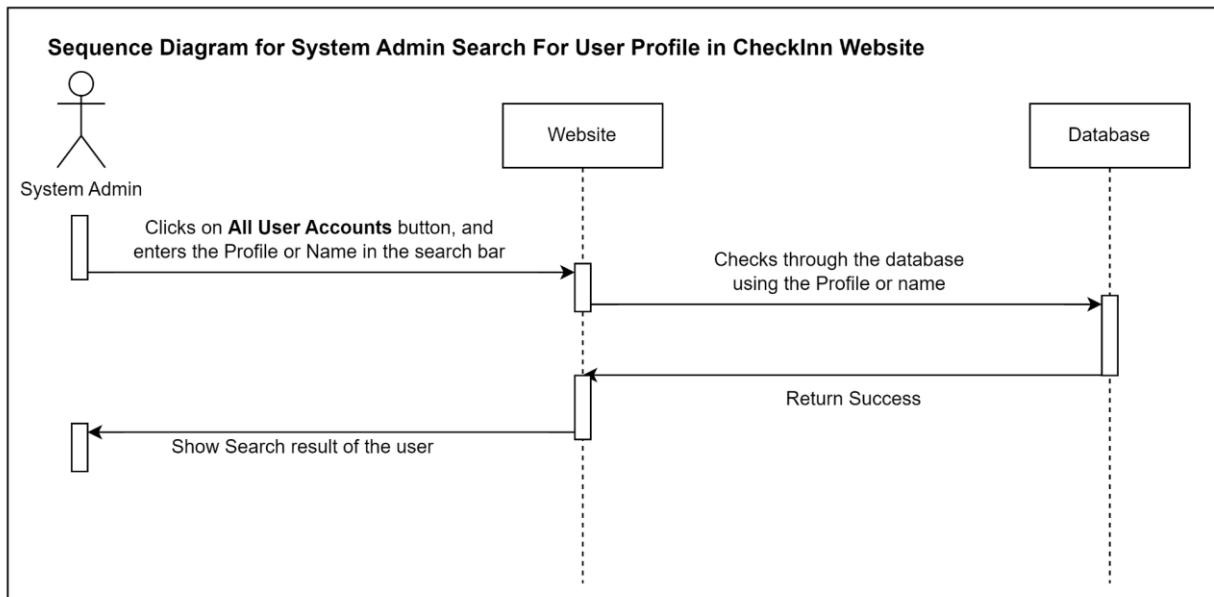
As the **System Admin**,

I want to be able to **search for a specific user profile** so that I can retrieve the specific user profile in the **System**.



Name: Search for specific user profile	ID: 12
Stakeholders and goals: System Admin - wants to search for a specific user profile.	
Description: The System Administrator wants to search for a specific user profile to retrieve the user profile in the system.	
Actors: System Admin	
Trigger: The system administrator clicks on the Search Bar	
Prerequisite: Successfully logged in as System Admin in Administrative Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System admin logs into the System. 2. System display homepage that shows 2 buttons: “All User Accounts” and “User Profiles”. 3. System admin clicks on the “User Profiles” button. 4. System directs the user to another page where all the user profiles are displayed in rows and a search bar. 5. System admin clicks on the search bar and enters the profile or email that they wish to search for and click the search button. 6. System search for the user profile and displays the user profiles that match the searched profile or name respectively. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram

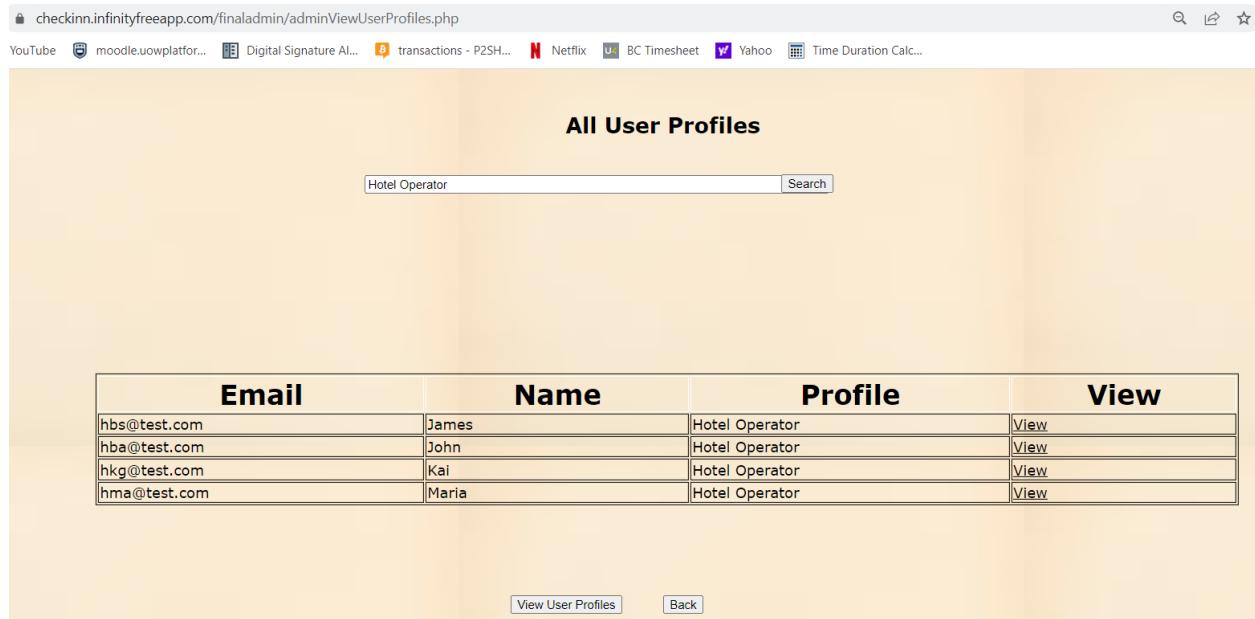


- Low-Fidelity Prototype

All User Profiles

beni	Search												
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Email</th> <th>Name</th> <th>Proffile</th> <th>View</th> </tr> </thead> <tbody> <tr> <td>beni@gmail.com</td> <td>beni</td> <td>Hotel Operator</td> <td style="text-align: right;">View</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;"></td> </tr> </tbody> </table>		Email	Name	Proffile	View	beni@gmail.com	beni	Hotel Operator	View				
Email	Name	Proffile	View										
beni@gmail.com	beni	Hotel Operator	View										
View User Profiles Back													

- High-Fidelity Prototype



The screenshot shows a web browser window with the URL checkinn.infinityfreeapp.com/finaladmin/adminViewUserProfiles.php. The page title is "All User Profiles". At the top, there is a search bar with the placeholder "Hotel Operator" and a "Search" button. Below the search bar is a table with four rows, each representing a user profile. The columns are labeled "Email", "Name", "Profile", and "View". The data in the table is as follows:

Email	Name	Profile	View
hbs@test.com	James	Hotel Operator	View
hba@test.com	John	Hotel Operator	View
hkg@test.com	Kai	Hotel Operator	View
hma@test.com	Maria	Hotel Operator	View

At the bottom of the page, there are two buttons: "View User Profiles" and "Back".

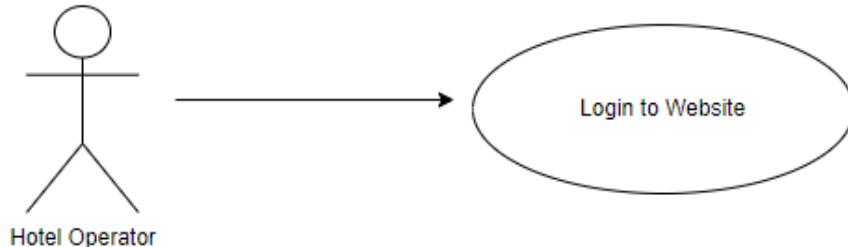
- Test Case

TEST CASE ID:	TC-12					
System Name:	CheckInn Admin Website					
Module Code:	CSC 321					
Test requirement:	1. Launch CheckInn Admin Website in the browser 2. Successfully logs to the System Admin website					
Purpose	Let user system admin search user profile.					
Pass	2					
Fail	0					
Pending	0					
Number of test cases:	2					
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result
Check System Admin search user profile:						
TC-12-1	Search User Profile using Profile.	1. The system directs to Search User Profile page 2. The user can enter a Profile inside the search text box. 3. The user clicks on the "Search" button. 4. The system would display the profiles in the "Profile" table that matches the search input.	User sees users with the user profile searched results on table list in "Search User Profile" page	System displays user with the user profile that matches on table list in "Search User Profile" page	30/04/2023	Pass
TC-12-2	Search User Profile using Name.	1. The system directs to Search User Profile page 2. The user can enter a Name inside the search text box. 3. The user clicks on the "Search" button. 4. The system would display the profiles in the "Profile" table that matches the search input.	User sees users with the user profile searched results on table list in "Search User Profile" page	System displays user with the user profile that matches on table list in "Search User Profile" page	30/04/2023	Pass

Use Case # 13

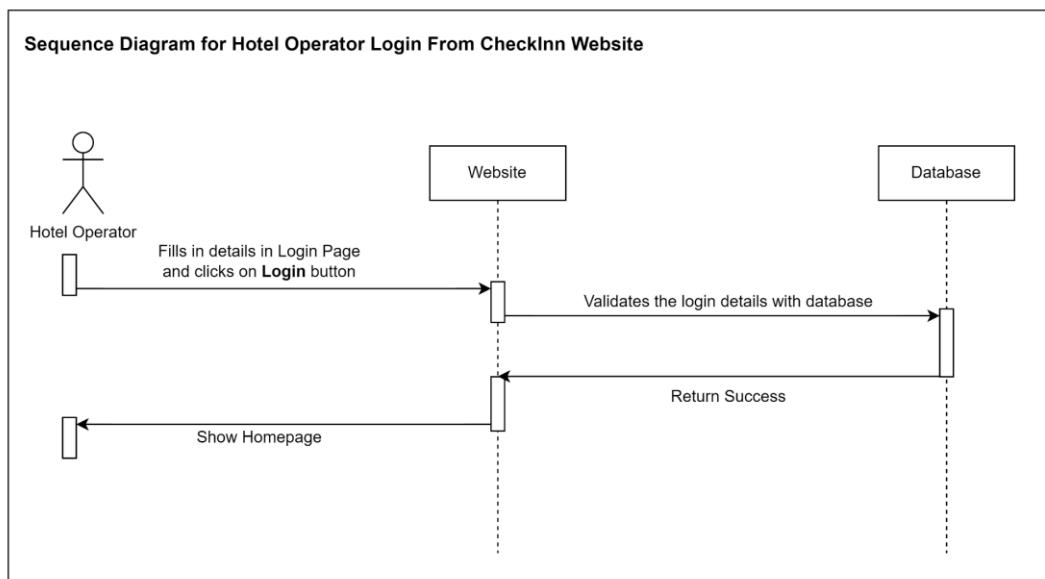
As the **Hotel Operator**,

I want to be able to **login** into the **Website** so that I can access the system.

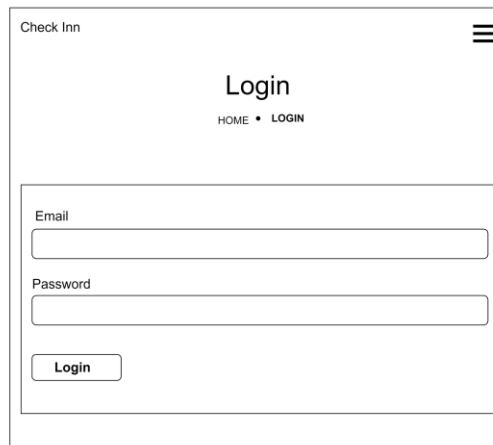


Name: Login	ID: 13
Stakeholders and goals: Hotel Operator - wants to login to the website.	
Description: The Hotel Operator wants to log in to the website to access the system.	
Actors: Hotel Operator	
Trigger: The hotel operator enters the website URL.	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow: <ol style="list-style-type: none"> 1. System will display a login page. 2. Hotel Operator will enter the username and password on the website. 3. Hotel Operator clicks on the “Login” button. 4. System will validate the username and password combination. 5. System will navigate to the homepage. 6. End. 	
Sub-flows:	
Alternative/Exceptional flows: <p><i>Incorrect email and password combination:</i></p> <p>4.1a – The system will pop-up a message dialog box stating “Incorrect Email or Password”. 4.1b – The user will click the “Ok” button to acknowledge and re-enter email and password.</p> <p><i>If blank email:</i></p> <p>4.2a –The system will pop-up a message dialog box stating “Please enter Email”. 4.2b –The user will click the “Ok” button to acknowledge and enter email.</p> <p><i>If blank password:</i></p> <p>4.3a –The system will pop-up a message dialog box stating “Please enter Password”. 4.3b –The user will click the “Ok” button to acknowledge and enter password.</p> <p><i>If email is valid but does not belong to Hotel Operator:</i></p> <p>4.4a –The system will pop-up a message dialog box stating “User Account Not Found”. 4.4b –The user will click the “OK” button to acknowledge and re-enter email and password.</p>	

- Sequence Diagram

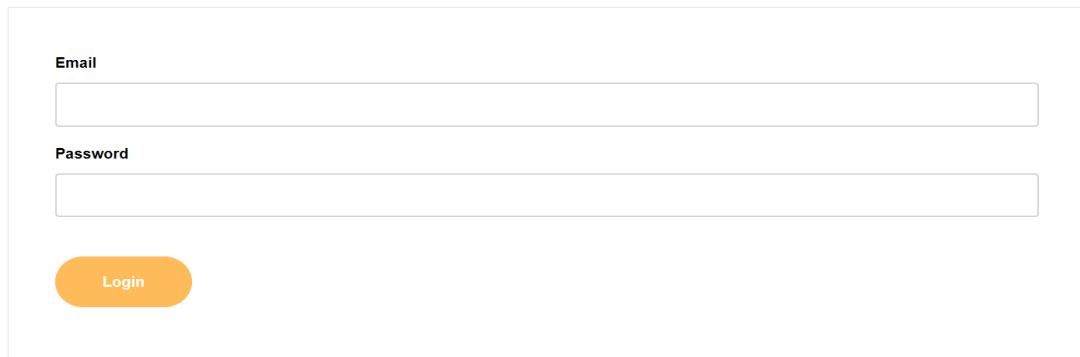
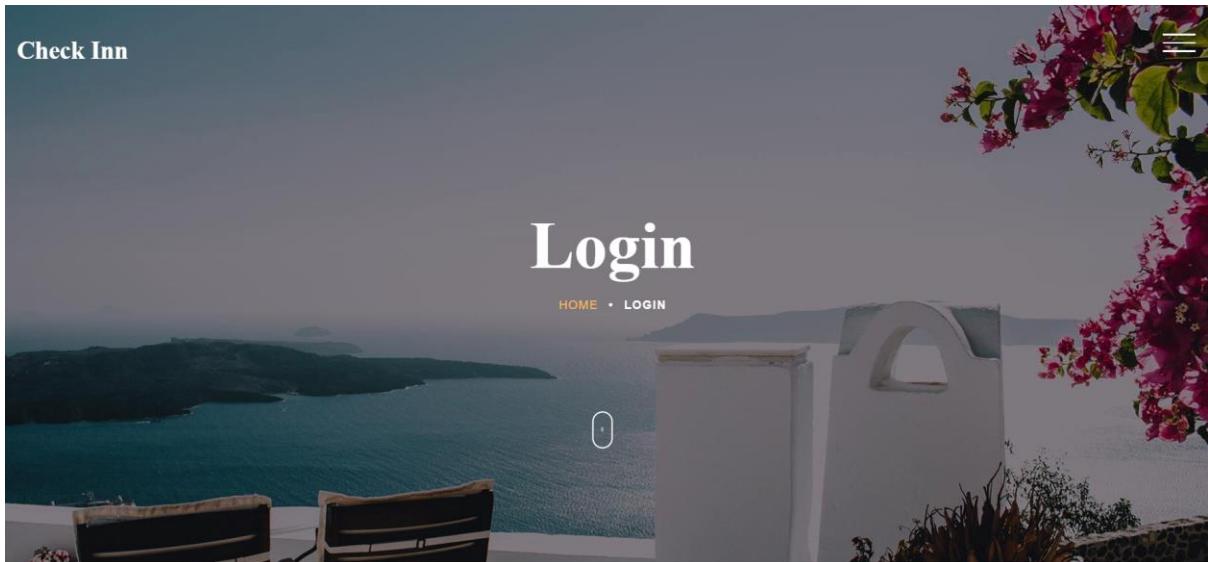


- Low-Fidelity Prototype



The low-fidelity prototype of the CheckInn Login page features a header with the text "Check Inn" and a menu icon. Below the header, the word "Login" is centered above a navigation bar with "HOME • LOGIN". The main content area contains two input fields labeled "Email" and "Password", followed by a "Login" button.

- High-Fidelity Prototype



The image provides a detailed view of the login form. It consists of three main fields: "Email" with a placeholder "Email", "Password" with a placeholder "Password", and a "Login" button. The "Email" and "Password" fields are enclosed in a light gray box. The "Login" button is a yellow oval shape with the word "Login" in white. The entire form is set against a white background.

Email	<input type="text" value="Email"/>
Password	<input type="password" value="Password"/>
<input type="button" value="Login"/>	

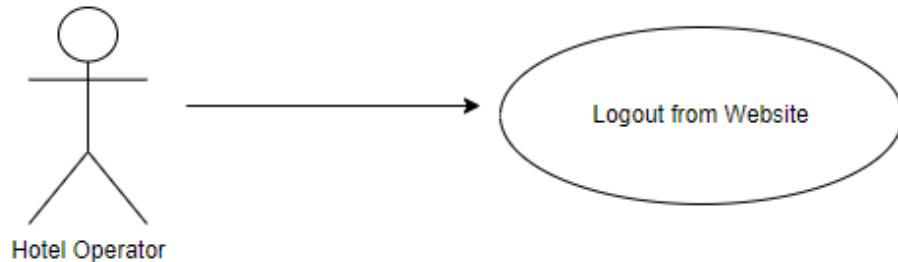
- Test Case

TEST CASE ID:	TC-13						
System Name:	CheckInn Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Website in the browser						
Purpose	Let user Hotel Operator be able to login into the website						
Pass	5						
Fail	0						
Pending	0						
Number of test cases:	5						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator Login Validation:							
TC-13-1	Launch Website	1. User launch website in browser	User sees login page appears on screen	System display login page appears on screen	08/05/2023	Pass	
TC-13-2	Input valid email and password	1. The system will prompt the user for the login credentials: email and password. 2. The user will enter the email and password 3. The user clicks on the "Login" button. 4. The system will validate the email and password combination. 5. The system will navigate to the landing page.	User will see the website's landing page	System will direct to website's landing page	08/05/2023	Pass	
Check Hotel Operator Login Validation:							
TC-13-3	Enters invalid email and password	1. User enter invalid email and password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Incorrect Email or Password"	System pop-up a message dialog box warns "Incorrect Email or Password"	08/05/2023	Pass	
TC-13-4	Enters blank email	1. User enter blank email and non-blank password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Please enter Email"	System pop-up a message dialog box warns "Please enter Email"	08/05/2023	Pass	
TC-13-5	Enters blank password	1. User enter non-blank email and blank password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Please enter Password"	System pop-up a message dialog box warns "Please enter Password"	08/05/2023	Pass	

Use Case # 14

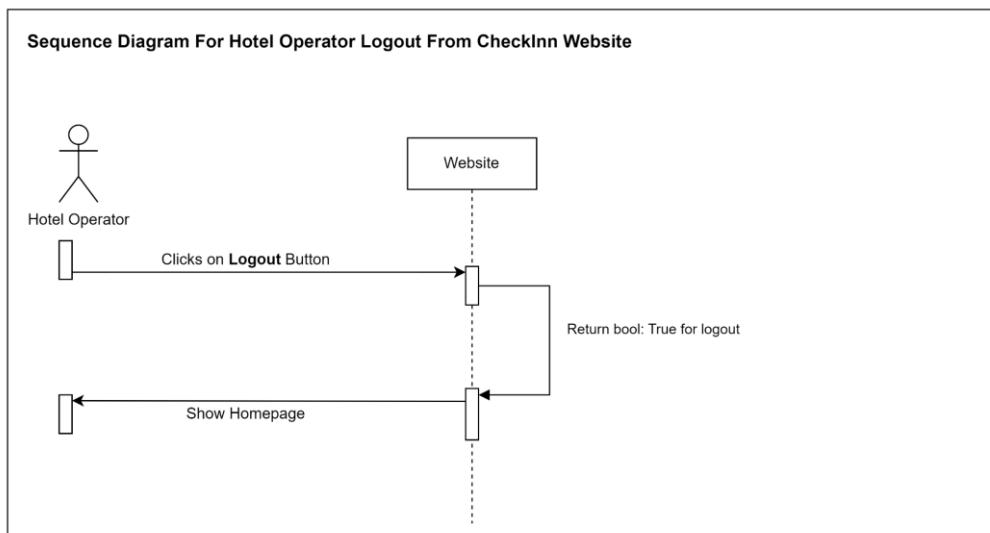
As the **Hotel Operator**,

I want to be able to **logout** from the **Website** so that I can exit the system.



Name: Logout	ID: 14
Stakeholders and goals: Hotel Operator - logout from the website	
Description: The Hotel Operator wants to logout from the website to exit the system.	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on “Log Out” button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow:	
1. Hotel Operator clicks on the “Log Out” button	
2. System will direct them to the Login page	
3. End.	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

Check Inn

X

Home
MyPage
Booking History
Customer Accounts
QR Codes
Logout

- High-Fidelity Prototype

Check Inn

X

Home
MyPage
Booking History
Customer Accounts
QR codes
Logout

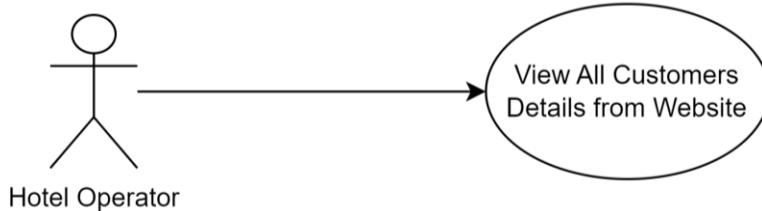
- Test Case

TEST CASE ID:	TC-14						
System Name:	CheckInn Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Website in the browser						
Purpose	Let user Hotel Operator be able to log out of the website						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator log out:							
TC-14-1	Log out of Hotel Operator account	1. The user clicks on the "Log Out" button. 2. The system will pop-up a message dialog box stating "Log Out successful". 3. The user clicks on the "OK" button. 4. The system will direct user to the "Log in" page.	User is directed user to website's login page	System will direct user to website's login page	08/05/2023	Pass	User returns back to login page

Use Case # 15

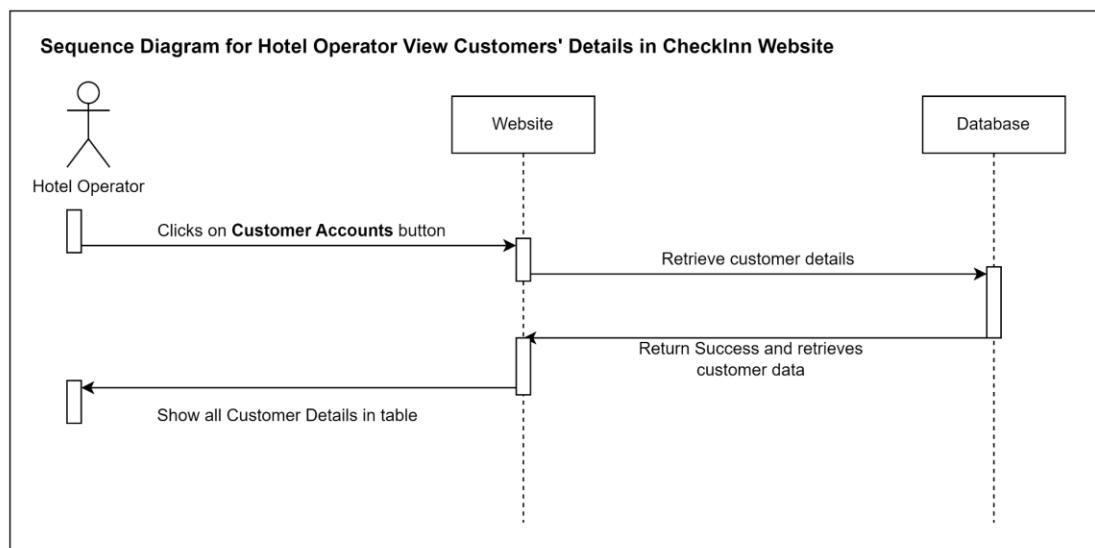
As the **Hotel Operator**,

I want to be able to **view** all Customers details from the **Website** so I know who my customers are.

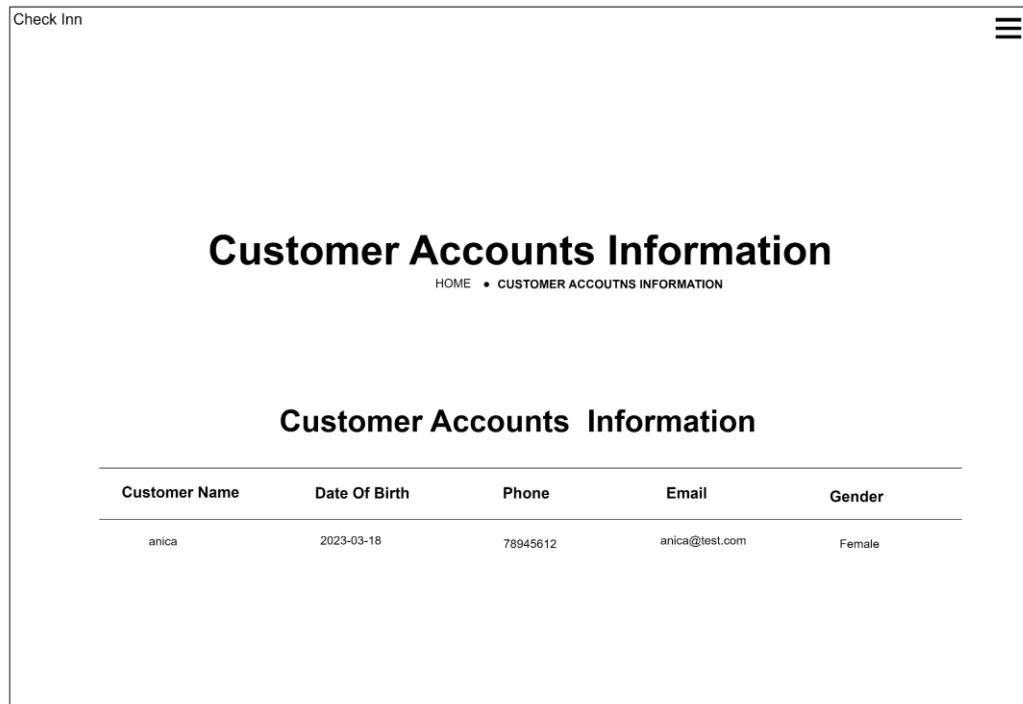


Name: View All Customers Details	ID: 15
Stakeholders and goals: Hotel Operator - view all customers details from the website	
Description: The Hotel Operator wants to view all customers details from the website to know who their customers are.	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on “Customer Accounts” button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. Hotel Operator logs into the System. 2. System will display the homepage. 3. Hotel Operator clicks on the “Customer Accounts” button. 4. System directs the user to another page where all Customer Accounts are displayed in a scrollable table page 5. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



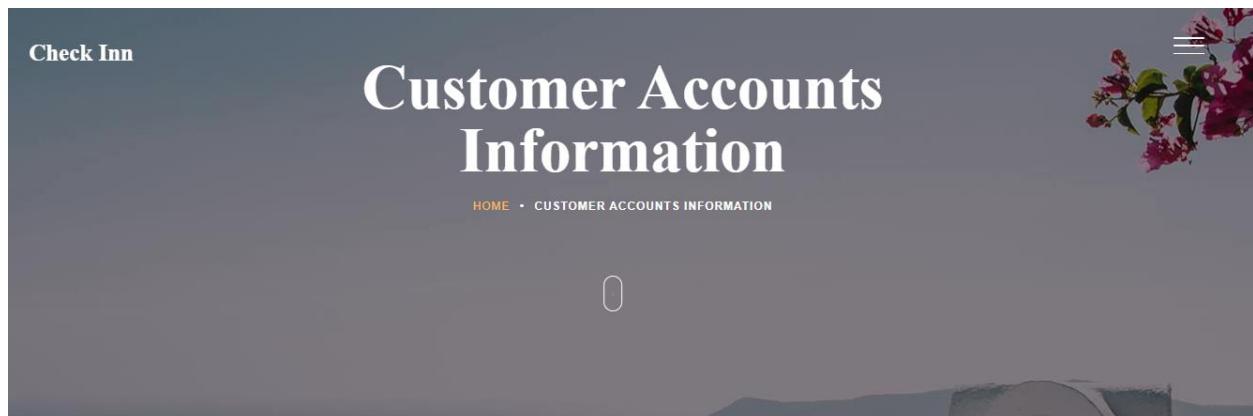
- Low-Fidelity Prototype



The wireframe shows a header with 'Check Inn' and a menu icon. Below it is a title 'Customer Accounts Information' with a subtitle 'HOME • CUSTOMER ACCOUNTS INFORMATION'. A main section titled 'Customer Accounts Information' contains a table with one row of data:

Customer Name	Date Of Birth	Phone	Email	Gender
anica	2023-03-18	78945612	anica@test.com	Female

- High-Fidelity Prototype



The high-fidelity prototype has a dark background with a nature-themed header featuring a tree branch with pink flowers. It includes a 'Check Inn' button and a menu icon. The main content area features a large title 'Customer Accounts Information' and a subtitle 'HOME • CUSTOMER ACCOUNTS INFORMATION'. Below the title is a placeholder text '0'.

Customer Accounts Information

Customer Name	Date Of Birth	Phone	Email	Gender	Notes
customer	2002-02-06	85104592	customer@test.com	Female	

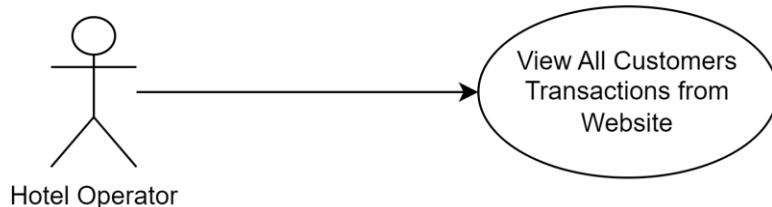
- Test Case

TEST CASE ID:	TC-15						
System Name:	CheckInn Website						
Module Code:	CSIT344						
Test requirement:	1. Launch CheckInn Website in the browser 2. Logged in successfully as Hotel Operator						
Purpose:	Let user Hotel Operator view all user accounts						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator View all Customers Details:							
TC-15-1	View All Customers Details	1. The user clicks on "Customer Accounts" button 2. The system displays all Customer accounts in a scrollable table page. 3. The user will scroll through the table list to view all user's account details.	User will direct user to website's landing page and see a table of all user accounts in the landing page	System will direct user to website's landing page and display all user's account details in a table	08/05/2023	Pass	Landing page displays all user's account details in a scrollable table list

Use Case # 16

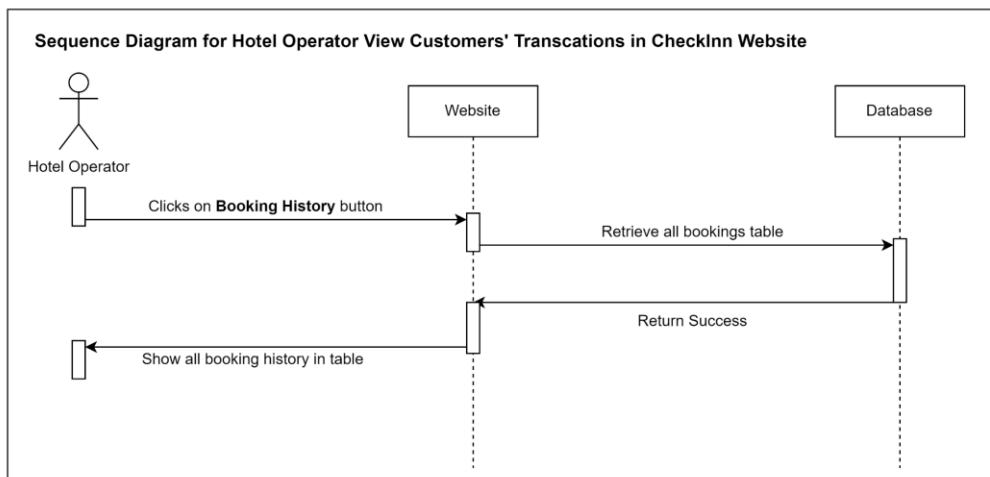
As the **Hotel Operator**,

I want to be able to **view** all Customers transactions from the **Website** so that I know my customer's preferences.

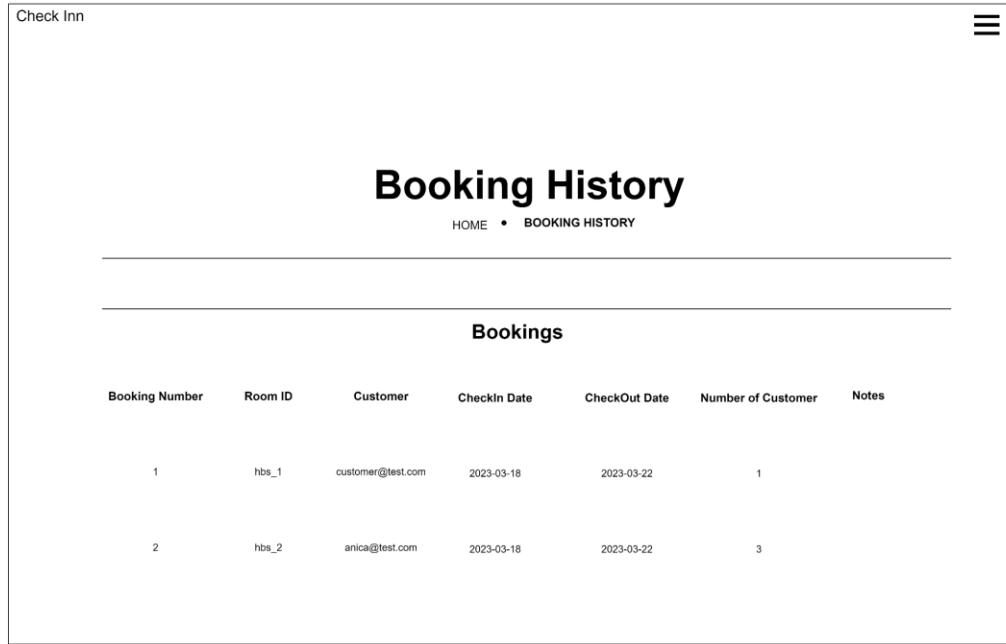


Name: View All Customers transactions	ID: 16
Stakeholders and goals: Hotel Operator - view all customers transactions from the Website	
Description: The Hotel Operator wants to view all customers transactions from the Website to know the customer's preferences.	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on "Booking History" button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. Hotel Operator logs into the System. 2. System will display the homepage. 3. Hotel Operator clicks on the "Booking History" button. 4. System directs the user to another page where all the customers are displayed in rows. 5. Hotel Operator needs to select the customer's name from the list. 6. System displays customer's detailed information and hotel transactions. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype



The screenshot shows a low-fidelity prototype of a web application. At the top left is a "Check Inn" button. At the top right is a menu icon consisting of three horizontal lines. The main title "Booking History" is centered above a navigation bar with "HOME" and "BOOKING HISTORY". Below this is a section titled "Bookings" with a table. The table has columns: Booking Number, Room ID, Customer, Checkin Date, CheckOut Date, Number of Customer, and Notes. Two rows of booking data are listed:

Booking Number	Room ID	Customer	Checkin Date	CheckOut Date	Number of Customer	Notes
1	hbs_1	customer@test.com	2023-03-18	2023-03-22	1	
2	hbs_2	anica@test.com	2023-03-18	2023-03-22	3	

- High-Fidelity Prototype



Bookings

Booking Number	Room ID	Customer	Checkin Date	CheckOut Date	Number of Customer	Notes
1	hbs_1	customer@test.com	2023-04-25	2023-04-27	1	
3	hbs_5	anica@test.com	2023-04-25	2023-04-28	1	

- Test Case

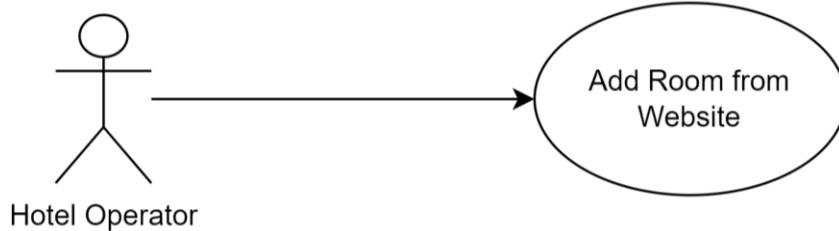
TEST CASE ID:	TC-4						
System Name:	CheckInn Website						
Module Code:	CSIT344						
Test requirement:	1. Launch CheckInn Website in the browser						
Purpose	Let user Hotel Operator view all user accounts						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						

ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator View all user accounts:							
TC-4-1	Login Hotel Operator account	1. The user clicks on "Booking History" 2. The system directs the user to the landing page that displays all user's account bookings and details.	User will direct user to website's landing page and see a table of all bookings in the landing page	System will direct user to website's landing page and display all bookings in a table	08/05/2023	Pass	

Use Case # 17

As the **Hotel Operator**,

I want to be able to **add room** from the **Website** so that the customer can view more available rooms.



Name: Add Room	ID: 17
Stakeholders and goals: Hotel Operator - add room.	
Description: The Hotel Operator wants to add room from the website for the customer to view more options of rooms	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on “Add Room” button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow: <ol style="list-style-type: none"> 1. Hotel Operator logs into the website. 2. System will display the homepage. 3. Hotel Operator clicks on “Add Room” button. 4. System directs the user to the “Add Room” page. 5. Hotel Operator enters room type, price and number of guests can stay. 6. Hotel Operator clicks on the “Add” button. 7. System validates the input. 8. System displays an updated page with room added. 9. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<i>If blank Room Type:</i> 7.1a - The system will pop-up a message dialog box stating “Please enter Room Type”. 7.1b –The user will click the “Ok” button to acknowledge and enter Room Type.	
<i>If blank Price:</i> 7.2a - The system will pop-up a message dialog box stating “Please enter Price”. 7.2b –The user will click the “Ok” button to acknowledge and enter Price.	
<i>If blank Number Of Guests Can Stay:</i> 7.3a - The system will pop-up a message dialog box stating “Please enter Number of Guests Can Stay”. 7.3b –The user will click the “Ok” button to acknowledge and enter Number of Guests Can Stay.	
<i>If all blank:</i> 7.4a - The system will pop-up a message dialog box stating “Please enter all the details”.	

7.4b –The user will click the “Ok” button to acknowledge and enter room type, price and number of guests can stay

If invalid Price:

7.5a - The system will pop-up a message dialog box stating “Please enter a number”.

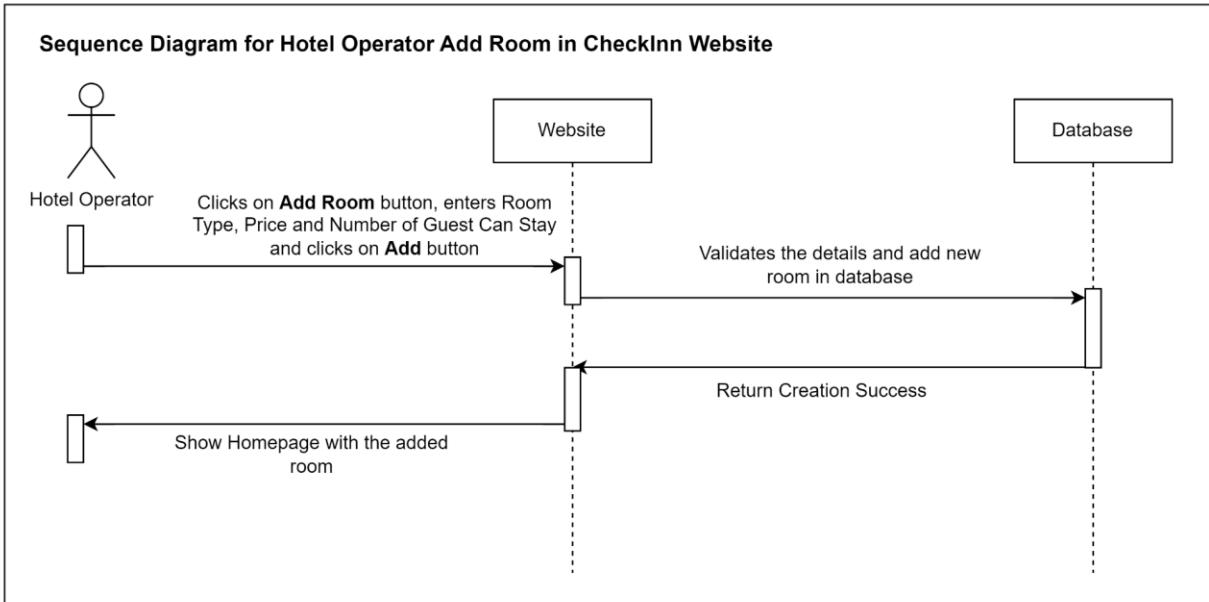
7.5b –The user will click the “Ok” button to acknowledge and re-enter Price.

If invalid Number Of Guests Can Stay:

7.6a - The system will pop-up a message dialog box stating “Please enter a number”.

7.6b –The user will click the “Ok” button to acknowledge and re-enter Number Of Guests Can Stay.

- Sequence Diagram



- Low-Fidelity Prototype

Check Inn

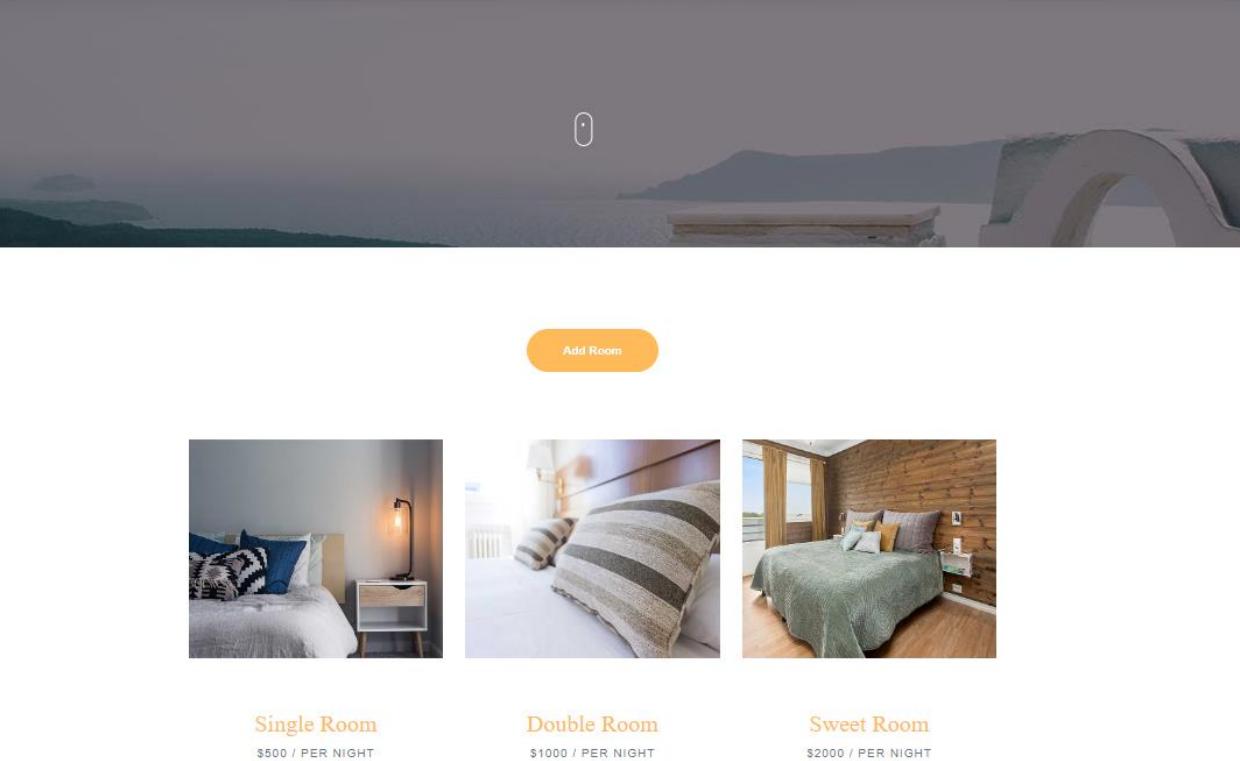
Add Room

HOME • HOTEL ROOMS

Room Type	Price
Deluxe	\$3000
Number Of Guests Can Stay	
2	
Upload a Picture: <input type="button" value="Choose File"/> No file chosen	
<input type="button" value="Back"/>	<input type="button" value="Add"/>

- High-Fidelity Prototype

Check Inn



Add Room



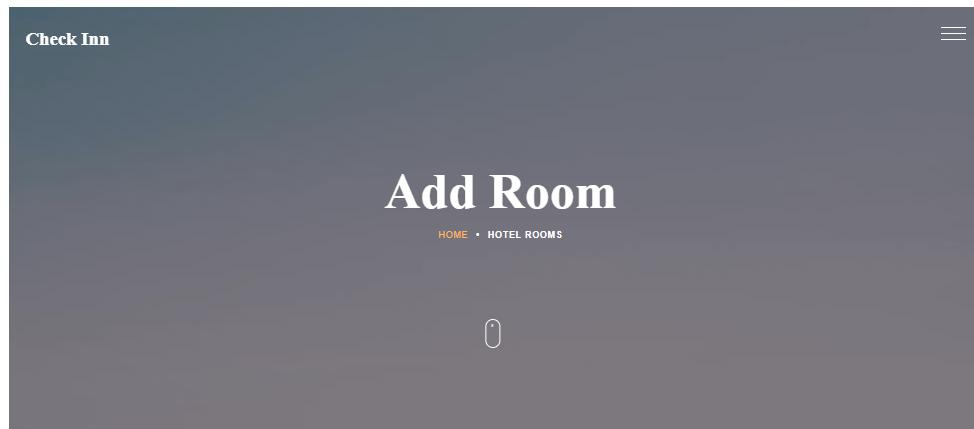
Single Room
\$500 / PER NIGHT



Double Room
\$1000 / PER NIGHT



Sweet Room
\$2000 / PER NIGHT



Room Type	Price
<input type="text"/>	<input type="text"/>
Number Of Guests Can Stay	
<input type="text" value="1"/>	
Upload a File: <input type="button" value="Choose File"/> No file chosen	
<input type="button" value="Back"/>	<input type="button" value="Add"/>

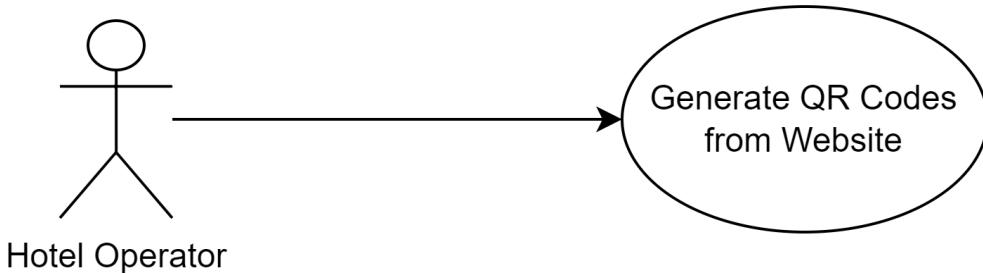
● Test Case

TEST CASE ID:	TC-17						
System Name:	CheckInn Website						
Module Code:	CSCI 321						
Test requirements:	1. Launch CheckInn Website in the browser 2. Successfully logs in to the Hotel Operator website						
Purpose:	Let user Hotel Operator be able to add new room.						
Pass	8						
Fail	0						
Pending	0						
Number of test cases:	8						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator generate QR Code:							
TC-17-1	Add Room	1. The user clicks the "Add Room" button. 2. The system directs the user to "Add Room" page. 3. The user enters room type, price and number of guests can stay and clicks the "Add" button. 4. The system validates the details.	User will see the newly generated QR Codes in View QR Codes page	System adds the new QR Codes to database AND direct user to landing page displaying the newly generated QR Codes in the "View QR Codes" table	08/05/2023	Pass	
Check Hotel Operator return back/exit View QR Code page:							
TC-17-2	Returns back to website's landing page from View QR Codes page	1. User clicks on "Back" button	User is redirected to website's landing page	System direct user back to website's landing page	08/05/2023	Pass	
Check Hotel Operator generate QR Code:							
TC-17-3	Enter blank Room Type	1. User input blank Room Type 2. User input other necessary details. 3. User clicks on "Add" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Room Type"	System pop-up a message dialog box displaying "Please enter Room Type"	08/05/2023	Pass	
TC-17-4	Enter blank Price	1. User input blank Price 2. User input other necessary details. 3. User clicks on "Add" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Price"	System pop-up a message dialog box displaying "Please enter Price"	08/05/2023	Pass	
TC-17-5	Enter blank number of guests can stay	1. User input blank Number Of Guests Can Stay 2. User input other necessary details. 3. User clicks on "Add" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Number Of Guests Can Stay"	System pop-up a message dialog box displaying "Please enter Number Of Guests Can Stay"	08/05/2023	Pass	
TC-17-6	Leaves everything blank	1. User leaves everything blank 2. User clicks on "Add" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter all the details"	System pop-up a message dialog box displaying "Please enter all the details"	08/05/2023	Pass	
TC-17-7	Enter invalid Price	1. User input alphabets or special characters 2. User input other necessary details. 3. User clicks on "Add" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter a number"	System pop-up a message dialog box displaying "Please enter a number"	08/05/2023	Pass	
TC-17-8	Enter invalid number of guests can stay	1. User input symbols or special characters 2. User input other necessary details. 3. User clicks on "Add" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter a number"	System pop-up a message dialog box displaying "Please enter a number"	08/05/2023	Pass	

Use Case # 18

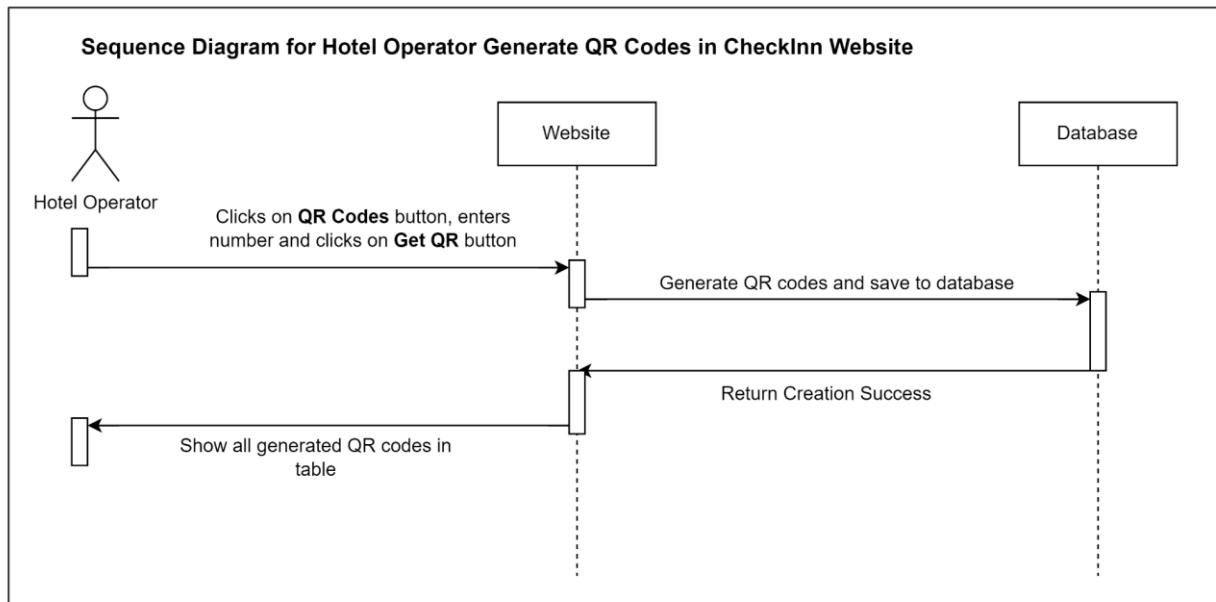
As the **Hotel Operator**,

I want to be able to **generate QR codes** from the **Website** so that the customer can scan the QR code to enter the hotel room.

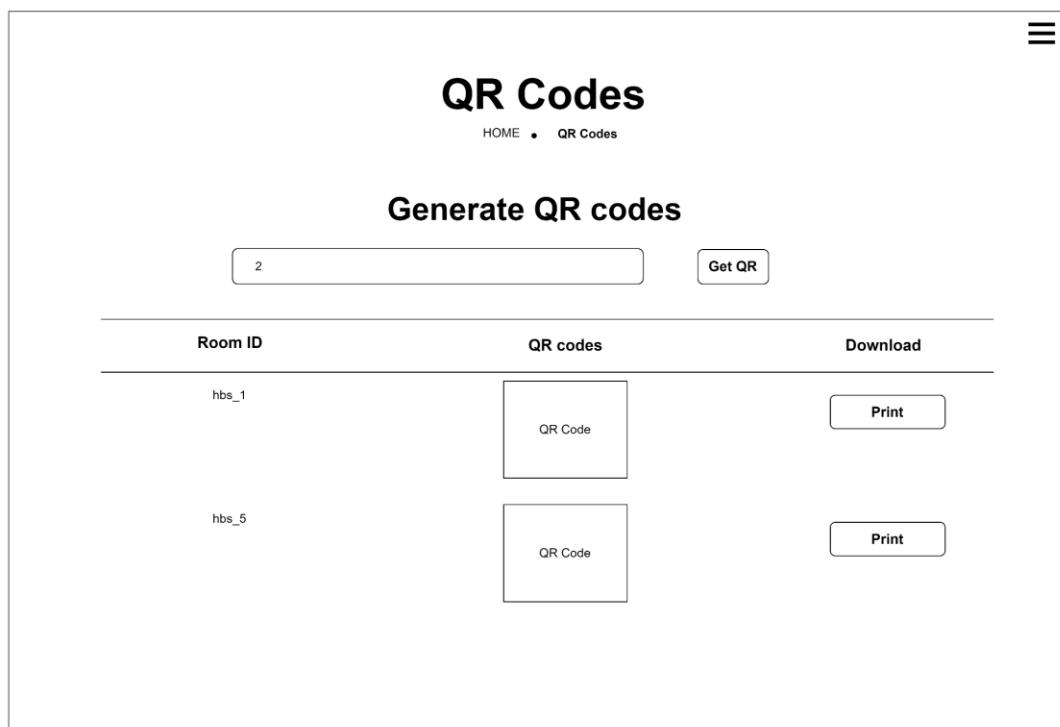


Name: Generate QR codes	ID: 18
Stakeholders and goals: Hotel Operator - generate QR codes.	
Description: The Hotel Operator wants to generate QR codes from the website for the customer to scan the QR code and enter the hotel room.	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on “Get QR” button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. Hotel Operator logs into the website. 2. System will display the homepage. 3. Hotel Operator clicks on “QR Codes” button. 4. System directs the user to the “QR Codes” page. 5. Hotel Operator enters the number of rooms inside the text bar. 6. Hotel Operator clicks on the “Get QR” button 7. System validates the input. 8. System displays an updated page with generated QR codes. 9. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<i>If invalid number:</i> <p>5.1a - The system will pop-up a message dialog box stating “Please enter a number”.</p> <p>5.1b –The user will click the “Ok” button to acknowledge and re-enter a number.</p>	

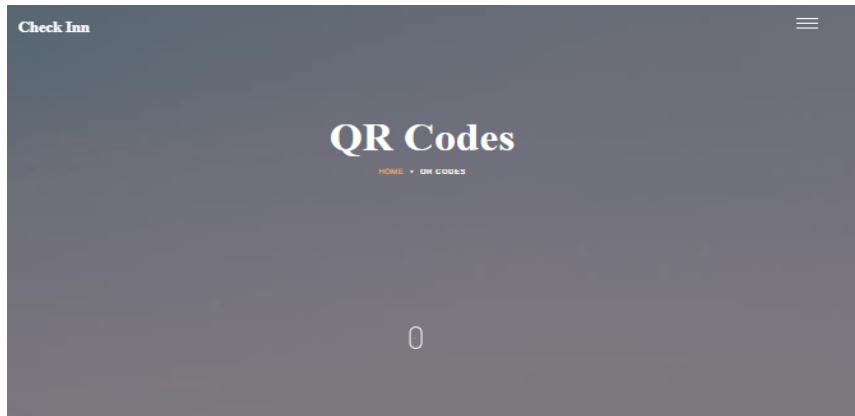
- Sequence Diagram



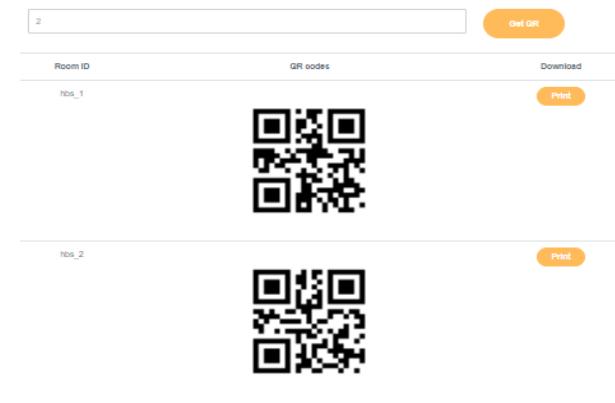
- Low-Fidelity Prototype



- High-Fidelity Prototype



Generate QR codes



This screenshot shows a "Generate QR codes" form. At the top, there's a search bar with the value "2" and an orange "Get QR" button. Below the search bar is a table with two rows. Each row has a "Room ID" column (hs1_1 and hs1_2), a "QR code" column (two square QR codes), and a "Print" button. The table also has columns for "Room ID", "QR code", and "Download".

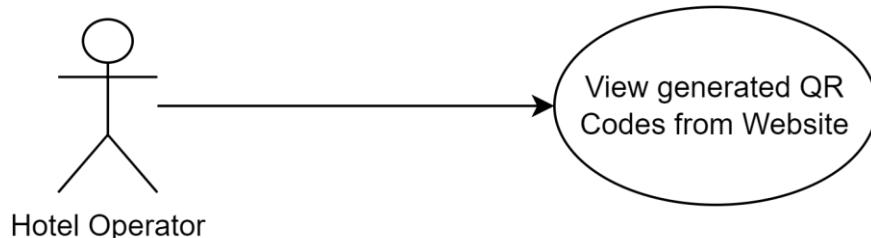
- Test Case

TEST CASE ID:	TC-18						
System Name:	CheckInn Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Website in the browser 2. Successfully logs in to the Hotel Operator website						
Purpose	Let user Hotel Operator be able to generate QR Code						
Pass	4						
Fail	0						
Pending	0						
Number of test cases:	4						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator generate QR Code:							
TC-18-1	Access View QR Codes page	1. User clicks on "QR Codes" button	User direct user to "View QR Codes" page	System direct user to "View QR Codes" page	08/05/2023	Pass	
TC-18-2	Generate QR Codes	1. The user clicks the "QR Codes" button. 2. The system directs the user to "QR Codes page. 3. The user enters number of rooms and clicks the "Get QR" button. 4. The system validates the details	User will see the newly generated QR Codes in View QR Codes page	System adds the new QR Codes to database AND direct user to landing page displaying the newly added QR Codes in the View QR Codes table	08/05/2023	Pass	
Check Hotel Operator return back/exit View QR Code page:							
TC-18-3	Returns back to website's landing page from View QR Codes page	1. User clicks on "Back" button	User is redirected to website's landing page	System direct user back to website's landing page	08/05/2023	Pass	
Check Hotel Operator generate QR Code:							
TC-18-4	Enter invalid number	1. User input alphabets or special characters 2. User clicks on "Get QR Code" button 3. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter a number"	System pop-up a message dialog box displaying "Please enter a number"	08/05/2023	Pass	

Use Case # 19

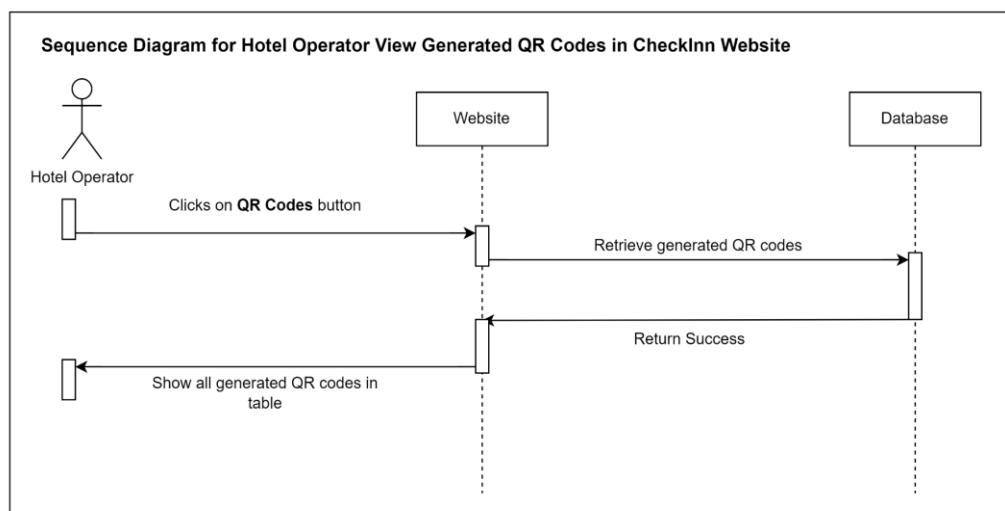
As the **Hotel Operator**,

I want to be able to **view the generated QR codes** from the **Website** so that the hotel operator can print the QR code for customer to scan it and enter the hotel room.



Name: Retrieve generated QR codes	ID: 19
Stakeholders and goals: Hotel Operator - view the generated QR codes.	
Description: The Hotel Operator wants to view the generated QR codes from the website for the hotel operator can print the QR code for customer to scan it and enter the hotel room.	
Actors: Hotel Operator	
Trigger: The hotel operator clicks on “View QR Codes” button	
Prerequisite: Successfully logged in as Hotel Operator in Website	
Normal flow:	
1. Hotel Operator logs into the website. 2. System will display the homepage. 3. Hotel Operator clicks on the “QR Codes” button. 4. System directs the user to the “QR Codes” page. 5. End.	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype



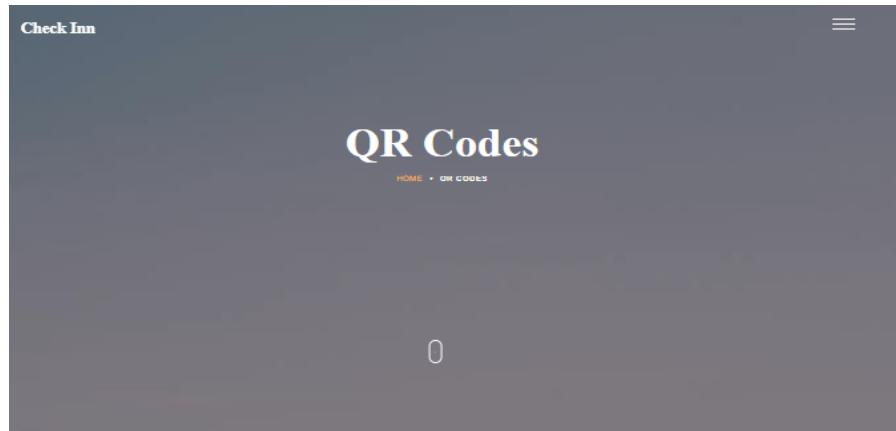
QR Codes

HOME • QR Codes

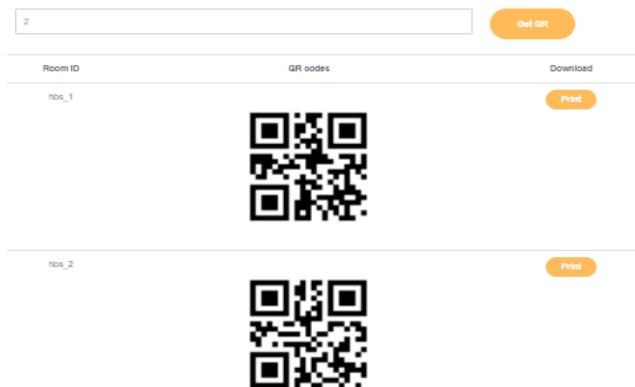
Generate QR codes

Room ID	QR codes	Download
hbs_1		<input type="button" value="Print"/>
hbs_5		<input type="button" value="Print"/>

- High-Fidelity Prototype



Generate QR codes



This screenshot shows a "Generate QR codes" section. It includes a search bar with the value "2", a "Get QR" button, and a table with two rows. Each row contains a "Room ID" (hos_1 and hos_2), a "QR codes" column showing a QR code image, and a "Download" and "Print" button. A horizontal line separates this from the previous screenshot.

- Test Case

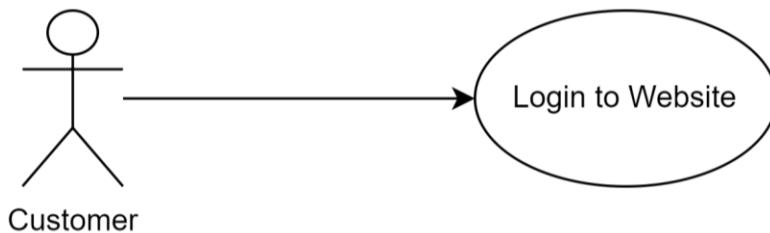
Test Case ID:	TC-19						
System Name:	CheckInn Website						
Module Code:	CSCI 321						
Test requirement:	1. Launch CheckInn Website in the browser 2. Successfully logs in to the Hotel Operator website						
Purpose	Let user Hotel Operator view all QR codes						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Hotel Operator view all QR codes page:							
TC-19-1	View All generated QR Codes	1. The user clicks the "QR Codes" button.	User sees a table of all generated QR Codes in "QR Codes" page	System directs user to "View QR Codes" page and displays a table of all generated QR Codes	08/05/2023	Pass	Logged in as Hotel Operator

Website

Use Case # 20

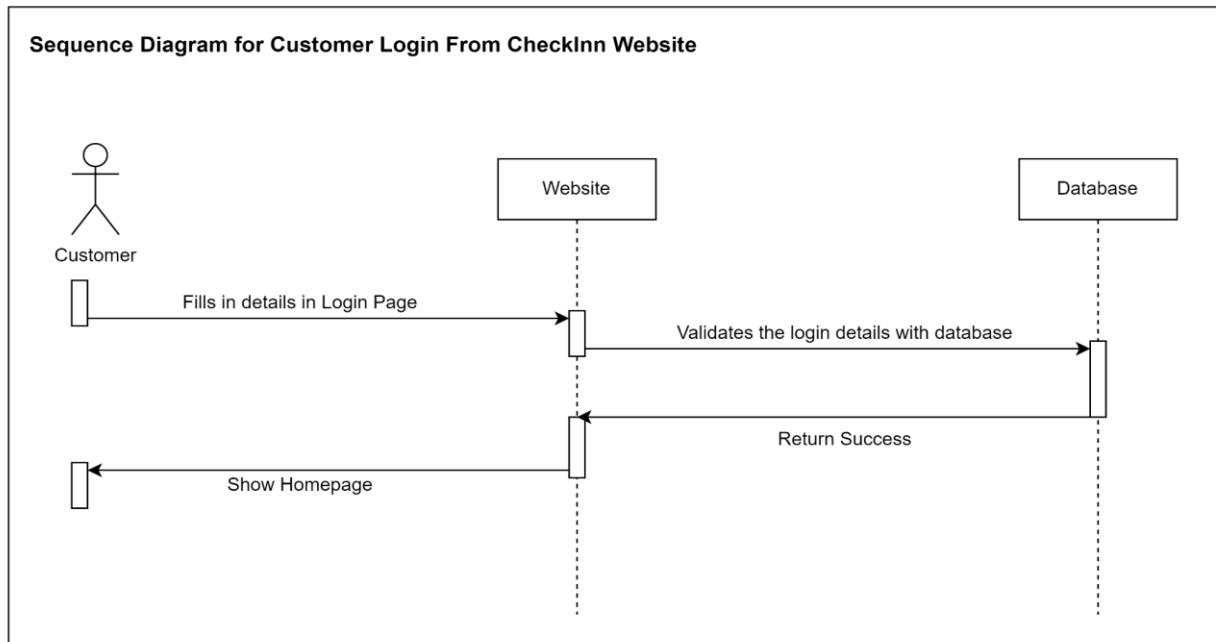
As the **Customer**,

I want to be able to **login** into the **Website** so that I can access the system.

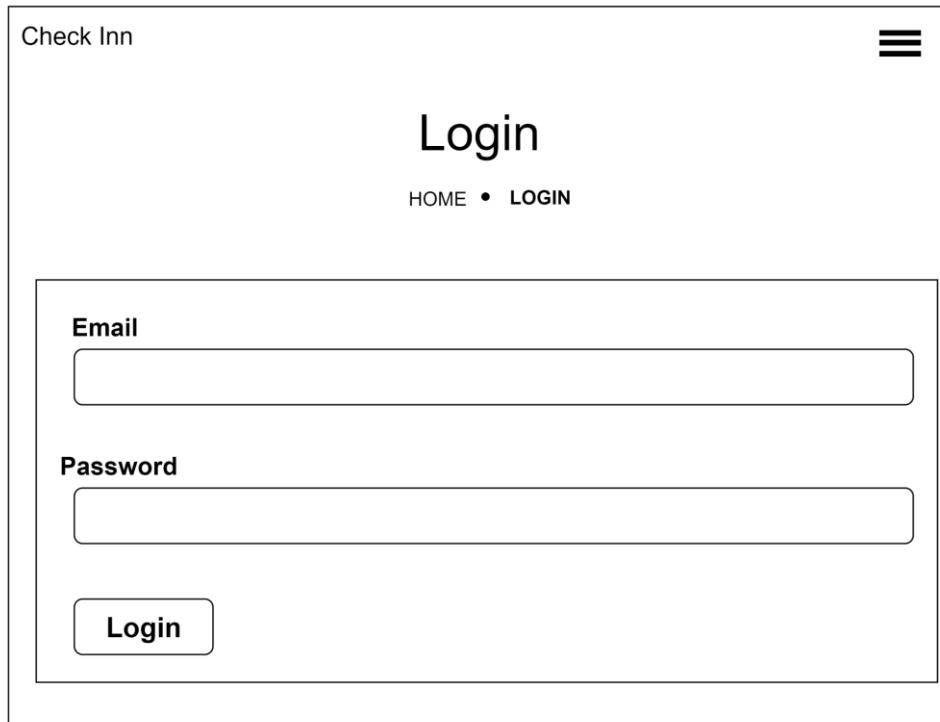


Name: Login	ID: 20
Stakeholders and goals: Customer - wants to login to the website.	
Description: The Customer wants to log into the website to access the system.	
Actors: Customer	
Trigger: The Customer enters the website URL.	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. Customer will enter the website URL. 2. System will display a login page. 3. Customer will enter the username and password. 4. Customer clicks on the “Login” button. 5. System will validate the username and password combination. 6. System will navigate to the homepage. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows: <p><i>Incorrect email and password combination:</i></p> <p>5.1a – The system will pop-up a message dialog box stating “Incorrect Email or Password”. 5.1b – The user will click the “Ok” button to acknowledge and re-enter email and password.</p> <p><i>If blank email:</i></p> <p>5.2a –The system will pop-up a message dialog box stating “Please enter Email”. 5.2b –The user will click the “Ok” button to acknowledge and enter email.</p> <p><i>If blank password:</i></p> <p>5.3a –The system will pop-up a message dialog box stating “Please enter Password”. 5.3b –The user will click the “Ok” button to acknowledge and enter password.</p> <p><i>If email is valid but does not belong to Customer:</i></p> <p>5.4a –The system will pop-up a message dialog box stating “User Account Not Found”. 5.4b –The user will click the “Ok” button to acknowledge and re-enter email and password.</p>	

- Sequence Diagram



- Low-Fidelity Prototype



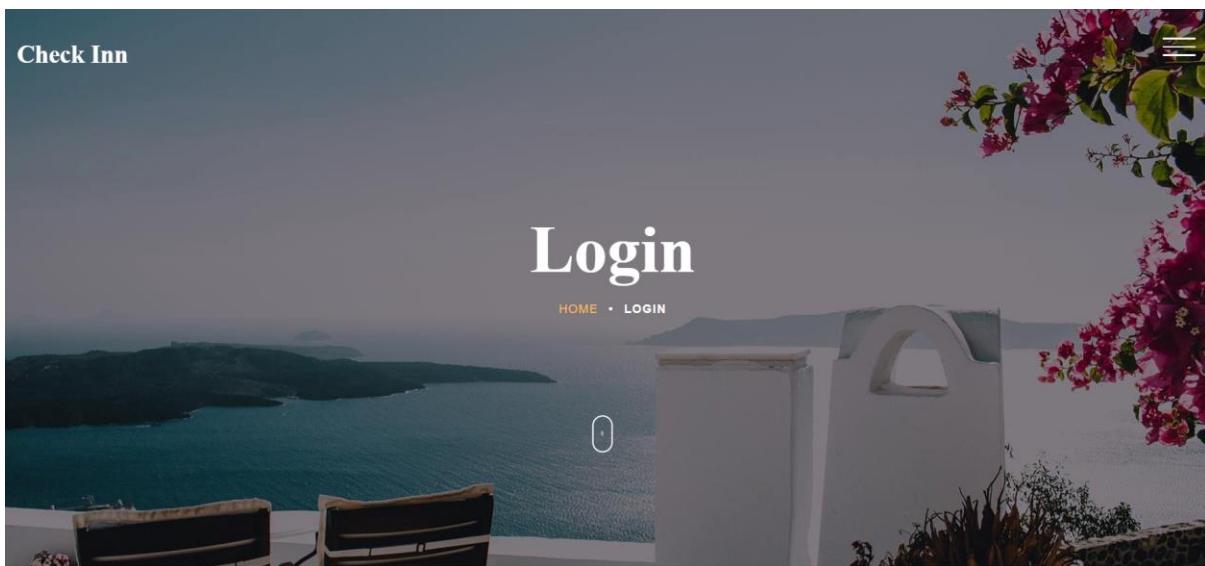
The low-fidelity prototype of the CheckInn login page features a header with the brand name "Check Inn" and a navigation menu icon. The main title "Login" is centered above a "HOME • LOGIN" link. Below the title is a large input field labeled "Email" with a placeholder for entering an email address. Underneath it is another input field labeled "Password" with a placeholder for entering a password. At the bottom of the form is a prominent "Login" button.

- High-Fidelity Prototype



UNIVERSITY
OF WOLLONGONG
AUSTRALIA

SIM GLOBAL
EDUCATION



Email

Password

[Login](#)

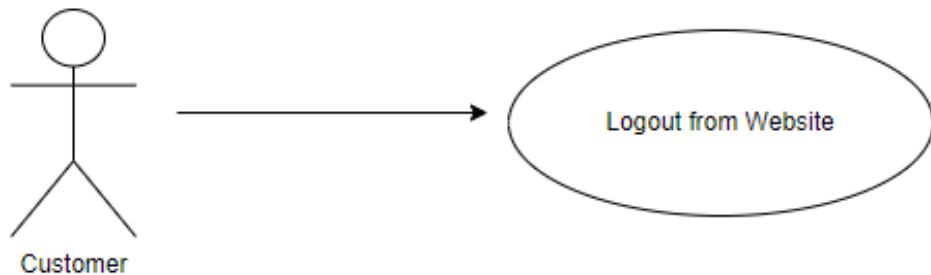
- Test Case

TEST CASE ID:	TC-20						
System Name:	CheckInn Website.						
Module Code:	CSIT321						
Test requirement:	1. Launch CheckInn Website in the browser						
Purpose	Let user Customer be able to login into the website						
Pass	5						
Fail	0						
Pending	0						
Number of test cases:	5						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer Login Validation:							
TC-20-1	Launch Website	1. User launch website in browser	User sees login page appears on screen	System display login page appears on screen	18/05/2023	Pass	
TC-20-2	Input valid email and password	1. The user will click on "Login" button 2. The system will prompt the user for the login credentials: email and password. 3. The user will enter the email and password 4. The user clicks on the "Login" button. 5. The system will validate the email and password combination. 6. The system will navigate to the landing page.	User will see the website's landing page	System will direct to website's landing page	18/05/2023	Pass	email: Password:
Check Customer Login Validation:							
TC-20-3	Enters invalid email and password	1. User enter invalid email and password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Incorrect Email or Password"	System pop-up a message dialog box warns "Incorrect Email or Password"	18/05/2023	Pass	email: Password:
TC-20-4	Enters blank email	1. User enter blank email and non-blank password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Please enter Email"	System pop-up a message dialog box warns "Please enter Email"	18/05/2023	Pass	email: Password:
TC-20-5	Enters blank password	1. User enter non-blank email and blank password credentials 2. User click "Login" button	User sees a pop-up a message dialog box warns "Please enter Password"	System pop-up a message dialog box warns "Please enter Password"	18/05/2023	Pass	email: Password:

Use Case # 21

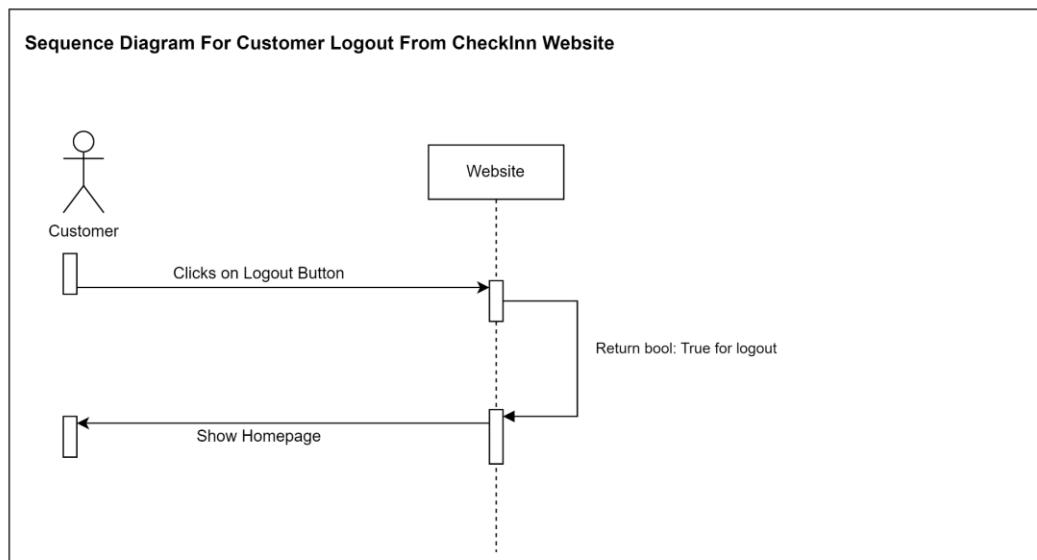
As the **Customer**,

I want to be able to **logout** from the **Website** so that I can exit the system.



Name: Logout	ID: 21
Stakeholders and goals: Customer - wants to log out from the website.	
Description: The Customer wants to log out from the website.	
Actors: Customer	
Trigger: The Customer clicks on “Log Out” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow:	
1. Customer clicks on the “Log Out” button 2. System will direct them to the Login page 3. End.	
Sub-flows:	
Alternative/Exceptional flows:	

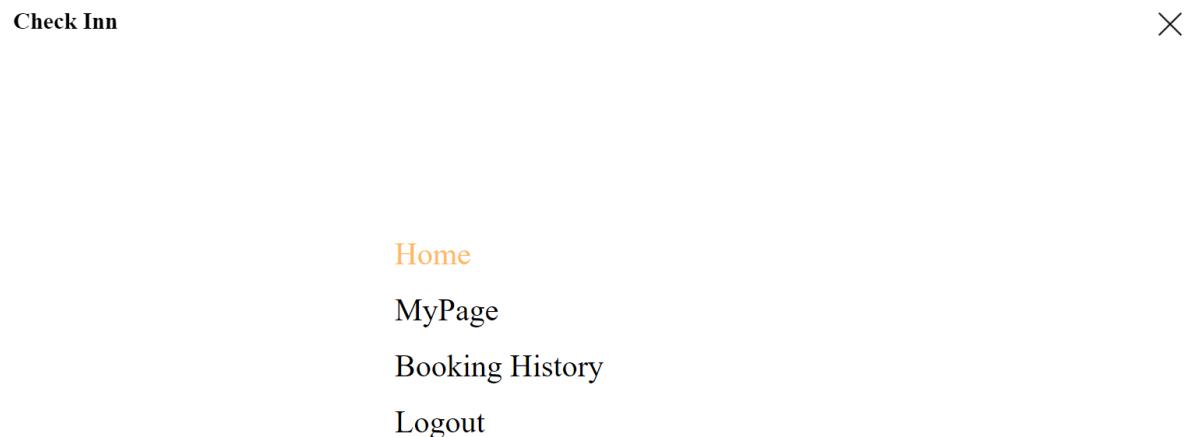
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



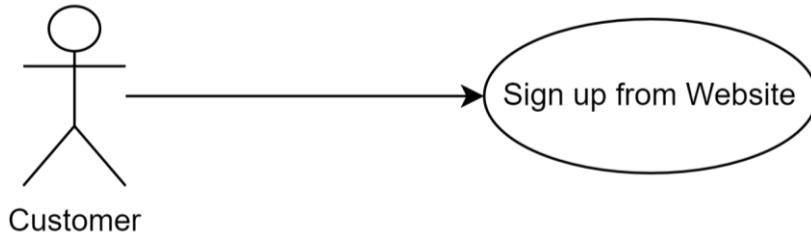
- Test Case

TEST CASE ID:		TC-21					
System Name:		CheckInn Website					
Module Code:		CST321					
Test requirement:		1. Launch CheckInn Website in the browser 2. Successfully logged in as Customer					
Purpose		Let user Customer be able to log out of the website					
Pass		1					
Fail		0					
Pending		0					
Number of test cases:		1					
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer log out:							
TC-21-1	Log out of Customer account	1. The user clicks on the "Log Out" button. 2. The system will pop-up a message dialog box stating "Log Out successful" 3. The user clicks on the "OK" button. 4. The system will direct user to the Homepage.	User is directed user to website's login page	System will direct user to website's login page	18/05/2023	Pass	User returns back to login page

Use Case # 22

As the **Customer**,

I want to be able to **sign up** to the **Website** so that I can access the system.



Name: Sign up	ID: 22
Stakeholders and goals: Customer - wants to sign up from the website.	
Description: The Customer wants to sign up to CheckInn website from the website.	
Actors: Customer	
Trigger: The Customer clicks on “Sign up” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. Customer clicks on the “Sign up” button 2. System will direct them to the Sign up page 3. Customer enters necessary details 4. Customer clicks on the “Sign up” button 5. System validates the 6. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<i>If blank email:</i> <p>5.1a –The system will pop-up a message dialog box stating “Please enter Email”.</p> <p>5.1b –The user will click the “Ok” button to acknowledge and enter email.</p>	
<i>If blank password:</i> <p>5.2a –The system will pop-up a message dialog box stating “Please enter Password”.</p> <p>5.2b –The user will click the “Ok” button to acknowledge and enter password.</p>	
<i>If blank name:</i> <p>5.3a –The system will pop-up a message dialog box stating “Please enter Name”.</p> <p>5.3b –The user will click the “Ok” button to acknowledge and enter name.</p>	
<i>If blank date of birth:</i> <p>5.4a –The system will pop-up a message dialog box stating “Please enter Date of Birth”.</p> <p>5.4b –The user will click the “Ok” button to acknowledge and enter date of birth.</p>	
<i>If blank phone number:</i> <p>5.5a –The system will pop-up a message dialog box stating “Please enter Phone Number”.</p> <p>5.5b –The user will click the “Ok” button to acknowledge and enter phone number.</p>	

If email is already being used:

5.5a –The system will pop-up a message dialog box stating “Email already being used. Please use another email”.

5.5b –The user will click the “Ok” button to acknowledge and re-enter email.

If invalid email:

5.6a –The system will pop-up a message dialog box stating “Please enter valid Email”.

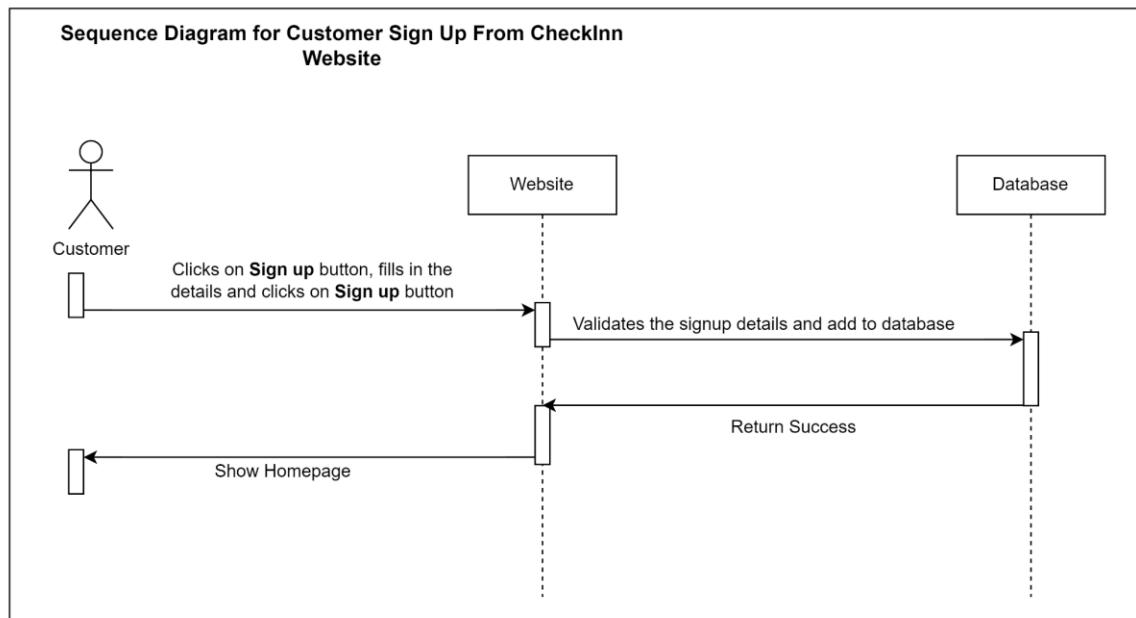
5.6b –The user will click the “Ok” button to acknowledge and re-enter email.

If short password:

5.7a –The system will pop-up a message dialog box stating “Password must be more than 8 characters in length”.

5.7b –The user will click the “Ok” button to acknowledge and enter password.

- Sequence Diagram



- Low-Fidelity Prototype

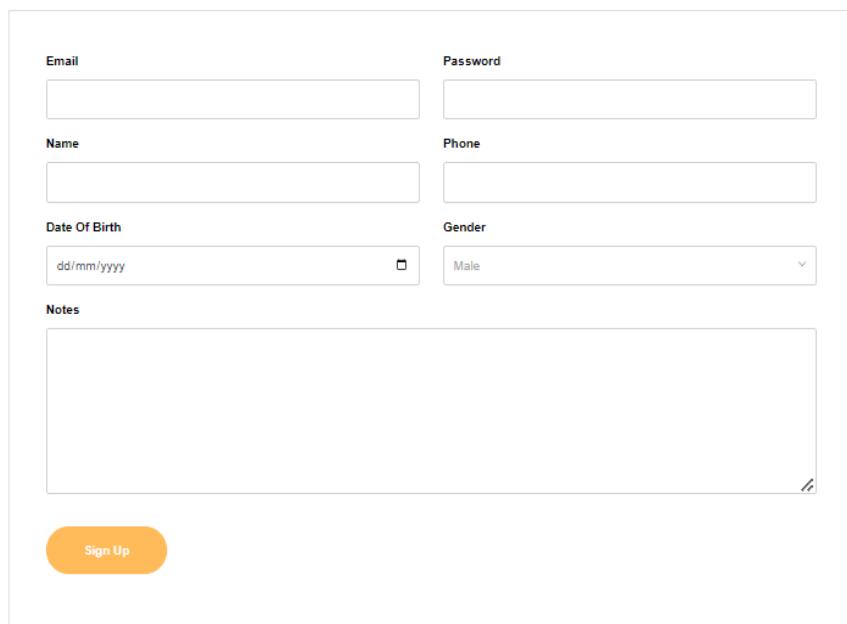
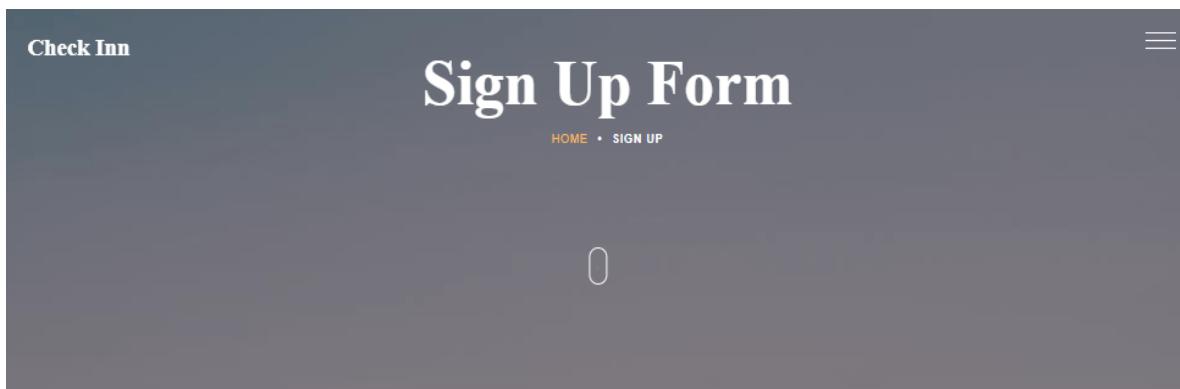
Check Inn ≡

Sign up Form

HOME • SIGN UP

ID	Password
<input type="text"/>	<input type="text"/>
Name	Phone
<input type="text"/>	<input type="text"/>
Age	Gender
<input type="text"/> v	<input type="text"/>
Email	
<input type="text"/>	
Notes	
<input type="text"/>	
<input type="button" value="Sign up"/>	

- High-Fidelity Prototype



A wireframe diagram of a sign-up form. It includes fields for Email (text input), Password (text input), Name (text input), Phone (text input), Date Of Birth (date input with a calendar icon), Gender (dropdown menu with "Male" selected), Notes (text area), and a "Sign Up" button at the bottom.

Email	Password
<input type="text"/>	<input type="text"/>
Name	Phone
<input type="text"/>	<input type="text"/>
Date Of Birth	Gender
<input type="text"/> dd/mm/yyyy	<input type="button" value="▼"/>
Notes	
<input type="text"/>	
<input type="button" value="Sign Up"/>	

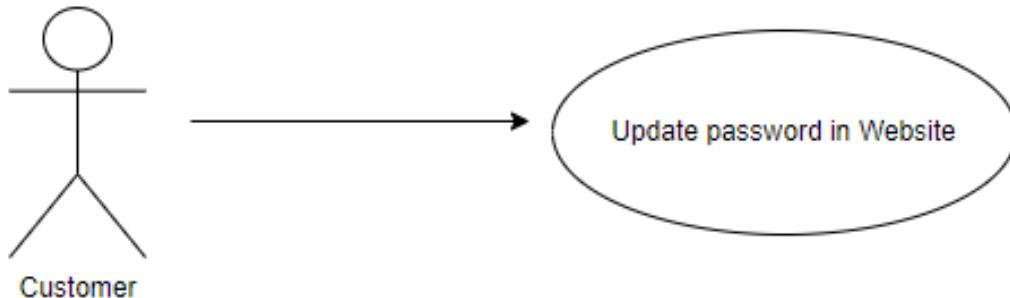
- Test Case

TEST CASE ID:	TC-22						
System Name:	CheckInn Website						
Module Code:	CSIT3322						
Test requirement:	1. Launch CheckInn Website in the browser						
Purpose	Let user Customer be able to login into the website						
Pass	10						
Fail	0						
Pending	0						
Number of test cases:	10						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer Sign up Validations:							
TC-22-1	Launch Website	1. User launch website in browser	User sees main page appears on screen	System display main page appears on screen	18/05/2022	Pass	
TC-22-2	Input valid email, password, name, phone number, date of birth and gender	1. The user will click on "Sign up" button. 2. The system will prompt the user for the sign up credentials: email, password, name, phone number, date of birth and gender 3. The user will enter email, password, name, phone number, date of birth and gender 4. The user clicks on the "Sign up" button. 5. The system will validate the details and add user to database. 6. The system will navigate to the landing page.	User will see the website's landing page	System will direct to website's landing page	18/05/2022	Pass	
Check Customer Sign up Validation:							
TC-22-3	Enter blank email	1. User clicks on "Sign up" button 2. User input blank email 3. User input password, name, phone number and date of birth 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Email"	System pop-up a message dialog box displaying "Please enter Email."	18/05/2022	Pass	
TC-22-4	Enter blank password	1. User clicks on "Sign up" button 2. User input blank password 3. User input email, name, phone number and date of birth 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Password"	System pop-up a message dialog box displaying "Please enter Password"	18/05/2022	Pass	
TC-22-5	Enter blank name	1. User clicks on "Sign up" button 2. User input blank name 3. User input password, phone number and date of birth 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Name"	System pop-up a message dialog box displaying "Please enter Name"	18/05/2022	Pass	
TC-22-6	Enter blank date of birth	1. User clicks on "Sign up" button 2. User input blank name 3. User input password, name and phone number 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Date of Birth"	System pop-up a message dialog box displaying "Please enter Date of Birth"	18/05/2022	Pass	
TC-22-7	Enter blank phone number	1. User clicks on "Sign up" button 2. User input blank phone number 3. User input email, password, name and date of birth 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Phone Number"	System pop-up a message dialog box displaying "Please enter Phone Number."	18/05/2022	Pass	
TC-22-8	Enter used email	1. User input used email 2. User input password, re-enter password, name, and email 3. User selects gender in drop down list 4. User clicks on "Sign up" button 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Email is already being used. Please enter a different email"	System pop-up a message dialog box displaying "User with some Email already exists."	18/05/2022	Pass	
TC-22-9	Enter invalid email	1. User input invalid email 2. User input password, name, phone number and date of birth 3. User selects gender in drop down list 4. User clicks on "Sign up" button 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up message dialog box displaying "Please enter valid Email"	System pop-up a message dialog box displaying "Please enter valid Email".	18/05/2022	Pass	
TC-22-10	Enter short password	1. User clicks on "Sign up" button 2. User input short password 3. User input email, password, name, phone number and date of birth 4. User selects gender in drop down list 5. User clicks on "Sign up" button 6. The system validates the details	User sees a pop-up a message dialog box displaying "Password needs to be 8 characters long"	System pop-up a message dialog box displaying "Password needs to be 8 characters long"	18/05/2022	Pass	

Use Case # 23

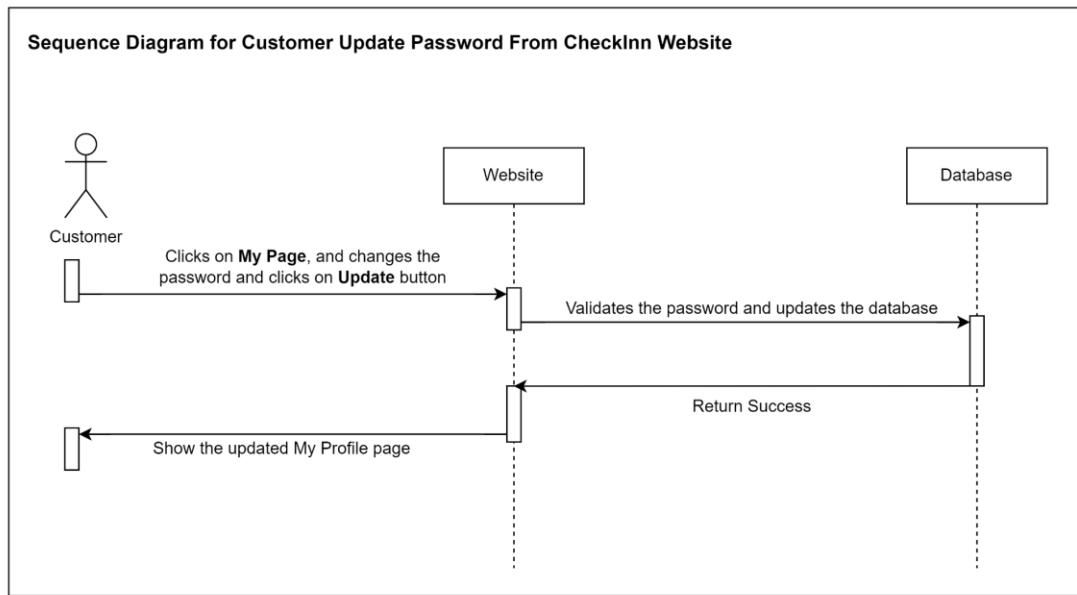
As the **Customer**,

I want to be able to **update my password** in the **Website** so that I can change the password.



Name: Update my password	ID: 23
Stakeholders and goals: Customer - wants to update password from the Website.	
Description: The Customer wants to update password from the Website to edit the password.	
Actors: Customer	
Trigger: The Customer clicks on “Change Password” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. Customer logs into the website. 2. System navigates the customer to the hotel website homepage. 3. Customer clicks on the “MyPage” button. 4. System directs to customer profile page. 5. Customer clicks on “Update”. 6. Customer enters a new password. 7. System validates the password. 8. System updates the password in the database. 9. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<i>If blank password:</i> 5.1a –The system will pop-up a message dialog box stating “Please enter Password”. 5.1b –The user will click the “Ok” button to acknowledge and enter password.	
<i>If short password:</i> 5.2a –The system will pop-up a message dialog box stating “Password needs to be 8 characters long”. 5.2b –The user will click the “Ok” button to acknowledge and enter password.	

- Sequence Diagram



- Low-Fidelity Prototype



The low-fidelity prototype of the 'My Page' form is displayed on a mobile device screen. The header reads 'Check Inn' and 'My Page'. Below the header, there are two rows of input fields: 'ID' (customer) and 'Password' (*****). The next row contains 'Name' (Customer) and 'Phone' (8888 8888). The third row contains 'Email' (customer@test.com). The fourth row contains 'Date Of Birth' (10/9/2001) and 'Gender' (Female). A 'Notes' section at the bottom contains the text: 'Hello! I will like to put an extra bed in my room'. At the bottom are 'Back' and 'Update' buttons.

- High-Fidelity Prototype

Check Inn

ID	Password
<input type="text" value="customer@test.com"/>	<input type="password" value="password"/>
Name	Phone
<input type="text" value="customer"/>	<input type="text" value="85104592"/>
Email	
<input type="text" value="customer@test.com"/>	
Date Of Birth	Gender
<input type="text" value="06/02/2002"/> 	<input type="text" value="Female"/> 
Notes	
<div style="border: 1px solid #ccc; height: 100px; width: 100%;"></div>	

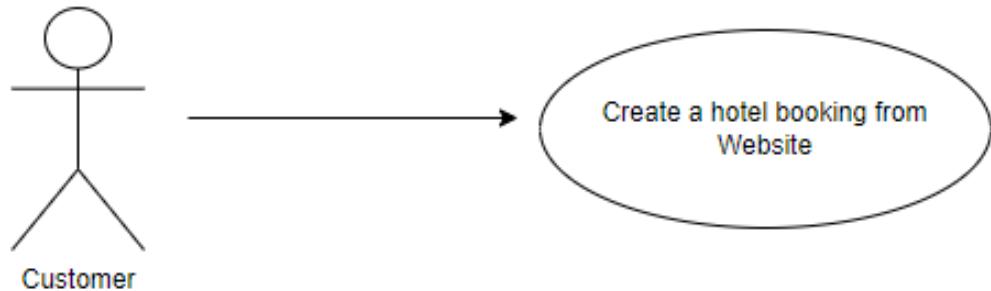
- Test Case

Test Case ID:		TC-23						
System Name:		CheckInn Website						
Module Code:		CSIT421						
Test requirement:		1. Launch CheckInn Website in the browser 2. Has Successfully logged in 3. Clicks on "All User Accounts" button						
Purpose		Let user Customer updates user's account						
Pass		4						
Fail		0						
Pending		0						
Number of test cases:		4						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note	
Check Customer Edit password:								
TC-23-1	Edit Password	1. User clicks on "My Profile" button 2. User input password 3. User clicks on "Update" button 4. The system validates the details	User sees a the edited profile page in the landing page	System updates the user password in the database and direct user to profile page displaying the edited profile page	18/05/2023	Pass		
Check Customer return back/exit Edit Password:								
TC-23-2	Returns back to website's landing page from Edit Password form.	1. User clicks on "Back" button	User is directed to user back to website's landing page	System direct user back to website's landing page	18/05/2023	Pass		User account will not be edited in this manner
Check Customer Edit Password:								
TC-23-3	Enter blank password	1. User clicks on "My Profile" button 2. User input blank password 3. User clicks on "Update" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Please enter Password"	System pop-up a message dialog box displaying "Please enter Password"	18/05/2023	Pass		
TC-23-4	Enter short password	1. User clicks on "My Profile" button 2. User input short password 3. User clicks on "Update" button 4. The system validates the details	User sees a pop-up a message dialog box displaying "Password needs to be 8 characters long"	System pop-up a message dialog box displaying "Password needs to be 8 characters long"	18/05/2023	Pass		

Use Case # 24

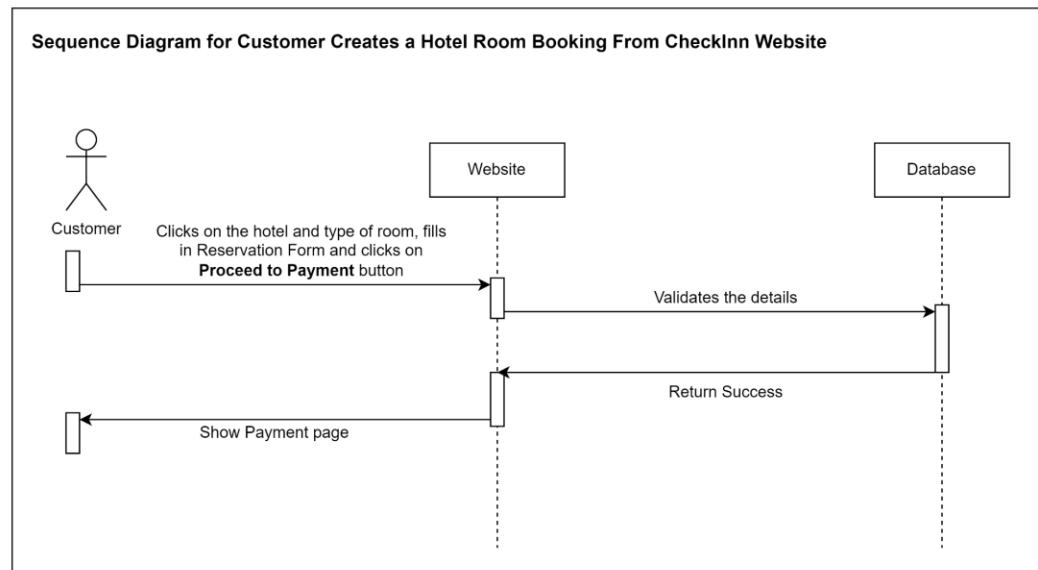
As the **Customer**,

I want to be able to **create a hotel booking** from the **Website** so that I can book a room to stay in.



Name: Create a hotel booking	ID: 24
Stakeholders and goals: Customer - wants to create a hotel booking from the Website.	
Description: The Customer wants to create a hotel booking from the hotel website to book a room to stay in.	
Actors: Customer	
Trigger: The Customer clicks on “Create hotel booking” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. Customer logs into the website. 2. System navigates the customer to the hotel website homepage. 3. Customer click on the hotel and the room that they want. 4. Customer keys in the check-in & check-out details. 5. Customer clicks on the “Proceed to Payment” button. 6. System validates the details. 7. System adds the details of booking in the database. 8. End. 	
Sub-flows: Create Payment	
Alternative/Exceptional flows:	
<p><i>If blank Date Check In:</i></p> <p>6.1a –The system will pop-up a message dialog box stating “Please add Date Check In”. 6.1b –The user will click the “Ok” button to acknowledge and enter date check in.</p>	
<p><i>If blank Date Check Out:</i></p> <p>6.2a –The system will pop-up a message dialog box stating “Please add Date Check Out”. 6.2b –The user will click the “Ok” button to acknowledge and enter date check out</p>	
<p><i>If blank Date Check In & Date Check Out:</i></p> <p>6.3a –The system will pop-up a message dialog box stating “Please add the dates”. 6.3b –The user will click the “Ok” button to acknowledge and enter dates for check in and check out.</p>	

- Sequence Diagram



- Low-Fidelity Prototype

Check Inn

☰

Reservation Form

HOME • RESERVATION

Name	Phone
<input type="text"/>	<input type="text"/>
Email	
<input type="text"/>	
Date Check In	Date Check Out
<input type="text"/> 	<input type="text"/> 
Adults	Room Type
<input type="text"/> v	<input type="text"/>
Notes	
<input type="text"/>	
Back	Proceed to Payment

- High-Fidelity Prototype

Check Inn

Name	Phone
customer	85104592
Email	
customer@test.com	
Date Check In	Date Check Out
dd/mm/yyyy	dd/mm/yyyy
Adults	Room Type
1	Single
Notes	
<input type="button" value="Back"/> <input type="button" value="Reserve Now"/>	

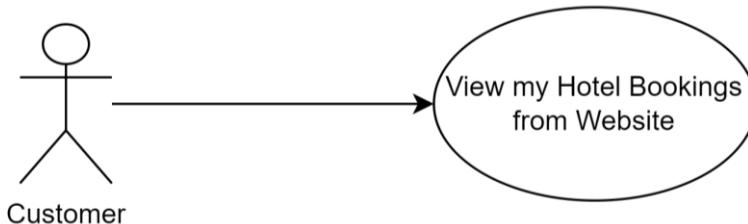
- Test Case

TEST CASE ID:	TC-24	System Name:	CheckInn Website	Module Code:	CSIT321	Test requirements:	1. Launch CheckInn Website in the browser 2. Successfully logged in as Customer	Purpose	Let user Customer be able to create booking	Pass	5	Fail	0	Pending	0	Number of test cases:	5
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note										
Check Customer create booking:																	
TC-24-1	Launch Website	1. User launch website in browser	User sees main page appears on screen	System display main page appears on screen	18/05/2023	Pass											
TC-24-2	Create new booking	1. The user clicks on hotel and room they want. 2. The system directs the user to "Reservation Form" page. 3. The user enters the necessary details. 4. The system validates the details.	User will see the Payment page	System direct user to Payment page	18/05/2023	Pass											
Check Customer return back/exit Create booking page:																	
TC-24-3	Return back to website's landing page from Reservation Form page	1. User clicks on "Back" button	User is redirected to updated Home page	System direct user back to website's landing page	18/05/2023	Pass	Booking will not be created in this manner										
Check Customer Create booking:																	
TC-24-3	Enter blank Date Check In	1. The user clicks on hotel and room they want. 2. The system directs the user to "Reservation Form" page. 3. The user input blank Date Check In. 4. The user enters the other necessary details. 5. The system validates the details	User sees a pop-up a message dialog box displaying "Please add Date Check In"	System pop-up a message dialog box displaying "Please add Date Check In"	18/05/2023	Pass											
TC-24-4	Enter blank Date Check Out	1. The user clicks on hotel and room they want. 2. The system directs the user to "Reservation Form" page. 3. The user input blank Date Check Out. 4. The user enters the other necessary details. 5. The system validates the details	User sees a pop-up a message dialog box displaying "Please add Date Check Out"	System pop-up a message dialog box displaying "Please add Date Check Out"	18/05/2023	Pass											
TC-24-5	Enter blank Date Check In and Date Check Out	1. The user clicks on hotel and room they want. 2. The system directs the user to "Reservation Form" page. 3. The user input blank Date Check In and Date Check Out. 4. The user enters the other necessary details. 5. The system validates the details	User sees a pop-up a message dialog box displaying "Please add the Dates"	System pop-up a message dialog box displaying "Please add the Dates"	18/05/2023	Pass											

Use Case # 25

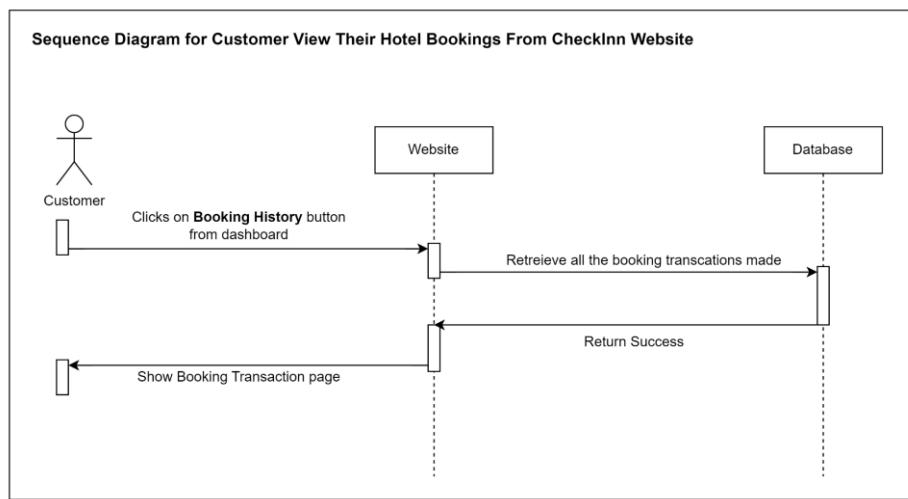
As the **Customer**,

I want to be able to **view my hotel bookings** from the **Website** so that I can retrieve the hotel booking details.

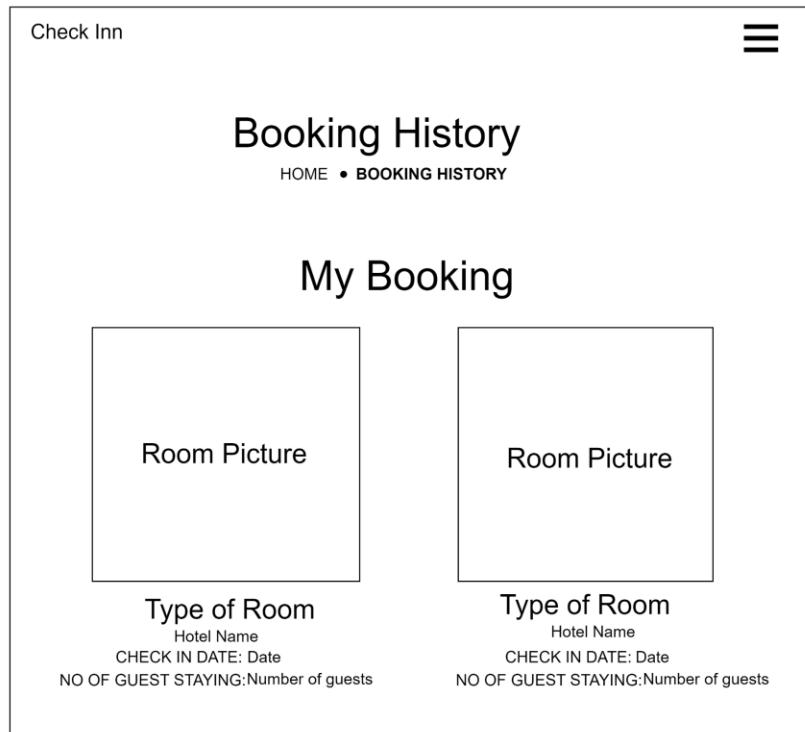


Name: View my hotel bookings	ID: 25
Stakeholders and goals: Customer - wants to view my hotel bookings	
Description: The Customer wants to view their hotel booking(s) from the Website to retrieve the hotel booking details.	
Actors: Customer	
Trigger: The Customer clicks on “My Bookings” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the homepage. 4. Customer clicks on the “My Bookings” button. 5. System directs the customer to the “My Bookings” page. 6. Customer will be able to view their hotel booking(s). 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

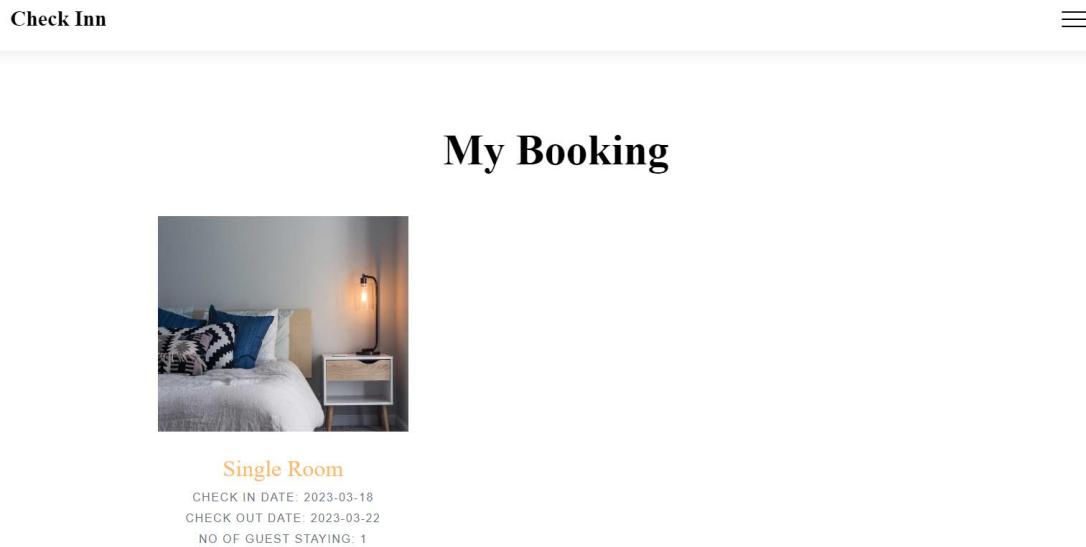
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



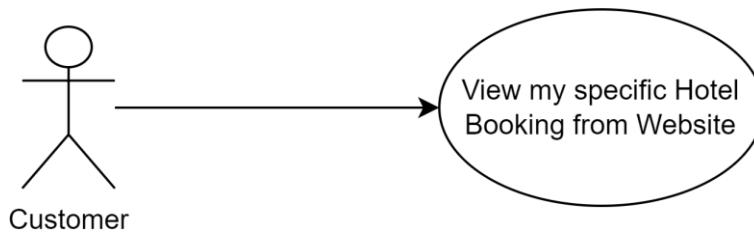
- Test Case

TEST CASE ID:	TC-25						
System Name:	CheckInn Website						
Module Code:	CSIT344						
Test requirement:	1. Launch CheckInn Website in the browser 2. Successfully logged in as Customer						
Purpose	Let user Customer view all hotel bookings						
Pass	1						
Fail	0						
Pending	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer View all hotel bookings:							
TC-25-1	View All Hotel Bookings	1. The user clicks on "My Bookings" button 2. The system directs the user to the Booking History page with details in a table. 3. The user will scroll through the table list to view all hotel bookings details.	User will direct user to Booking History page and see a table of all bookings with details in the landing page.	System will direct user to Booking History page and display all bookings with details in a table.	18/05/2023	Pass	Landing page displays all bookings in a scrollable table list.

Use Case # 26

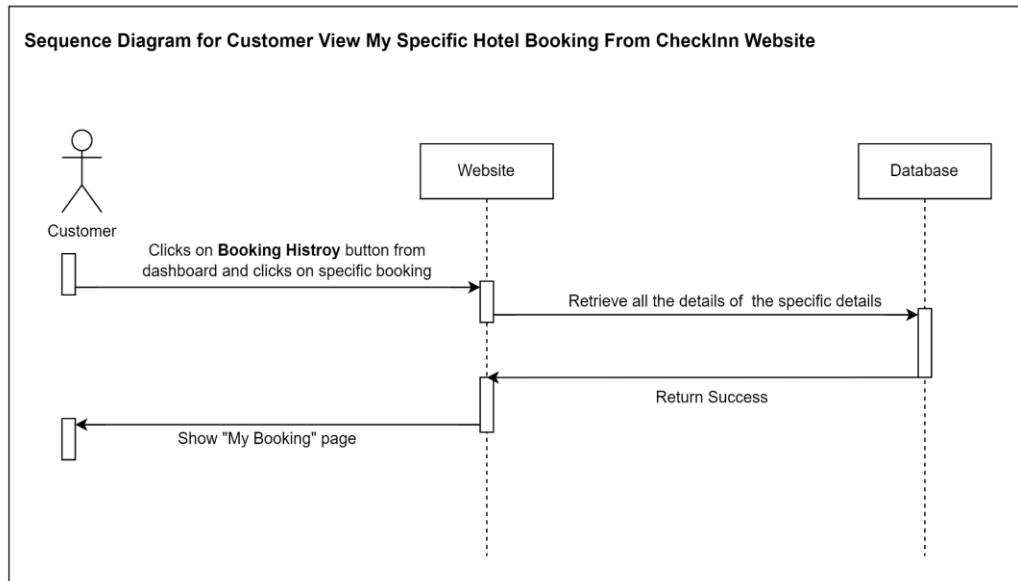
As the **Customer**,

I want to be able to **view my specific hotel booking** from the **Website** so that I can retrieve the hotel booking details.



Name: View my specific hotel booking	ID: 26
Stakeholders and goals: Customer - wants to view my specific hotel booking	
Description: The Customer wants to view their hotel booking from the Website to retrieve the hotel booking details.	
Actors: Customer	
Trigger: The Customer clicks on “My Bookings” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the homepage. 4. Customer clicks on the “My Bookings” button. 5. System directs the customer to the “My Bookings” page. 6. Customer clicks on specific booking. 7. System directs the customer to the specific hotel booking page with details. 8. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

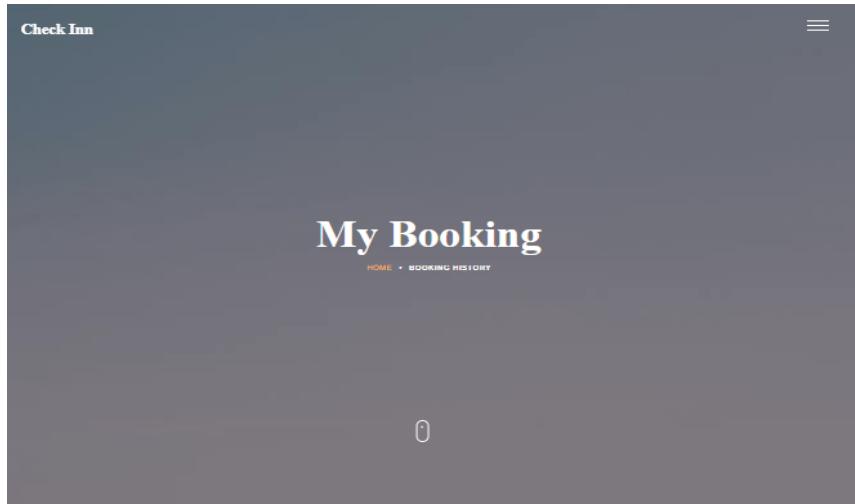
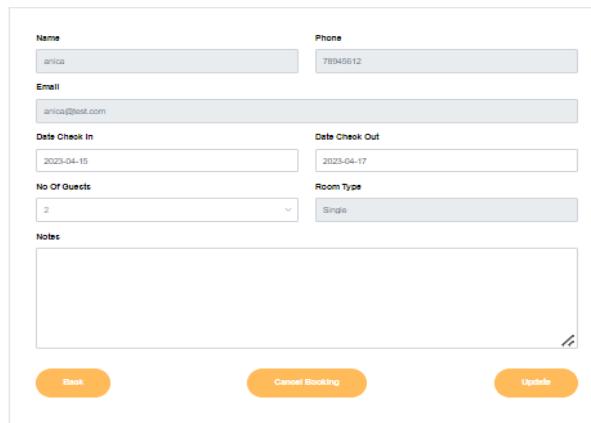
Check Inn

My Booking

HOME • BOOKING HISTORY

Name	Phone	
<input type="text" value="Name"/>	<input type="text" value="Phone Number"/>	
Email		
<input type="text" value="Email Address"/>		
Date Check In	Date Check Out	
<input type="text" value="Date Check In"/> 	<input type="text" value="Date Check Out"/> 	
No Of Guests	Room Type	
<input type="text" value="Number of Adults"/> v	<input type="text" value="Room Type"/>	
Notes		
<input type="text" value=""/>		
<input type="button" value="Back"/>	<input type="button" value="Cancel Booking"/>	<input type="button" value="Update"/>

- High-Fidelity Prototype

This is a detailed view of the "My Booking" form. It includes fields for Name (Anica), Phone (78945612), Email (anica@test.com), Date Check In (2023-04-15), Date Check Out (2023-04-17), No Of Guests (2), Room Type (Single), and Notes (empty). Below the form are three buttons: "Book" (orange), "Cancel Booking" (orange), and "Update" (orange).

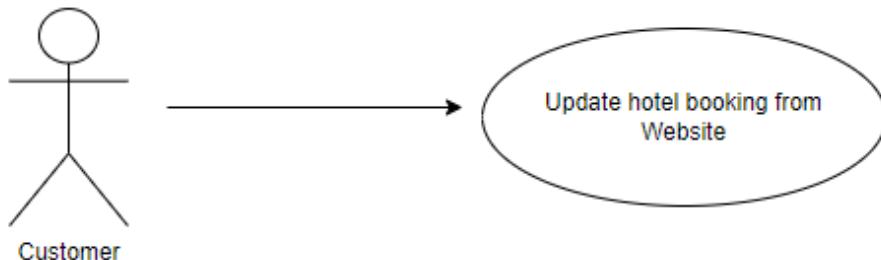
- Test Case

TEST CASE ID:	TC-26						
System Name:	CheckInn Website						
Module Code:	CSIT3441						
Test requirement:	1. Launch CheckInn Website in the browser 2. Successfully logged in as Customer						
Purpose:	Let user Customer view a specific hotel booking						
Pass:	1						
Fail:	0						
Pending:	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer View a specific hotel booking:							
TC-26-1	View a Specific Hotel Booking	1. The user clicks on "My Bookings" button 2. The system directs the user to the Booking History page with details in a table. 3. The user will scroll through the table list to view all hotel bookings details. 4. The user clicks on a specific booking	User will direct user to My Booking page and see a booking with details in the landing page	System will direct user to My Booking page and display all details of booking	18/05/2023	Pass	

Use Case # 27

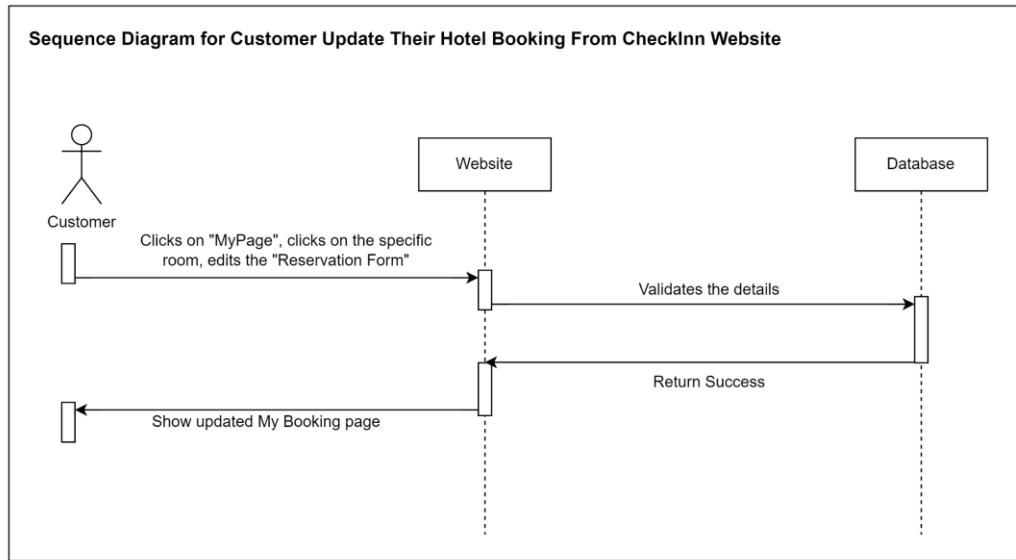
As the **Customer**,

I want to be able to **update my hotel booking** from the **Website** so that I can edit the hotel booking details.

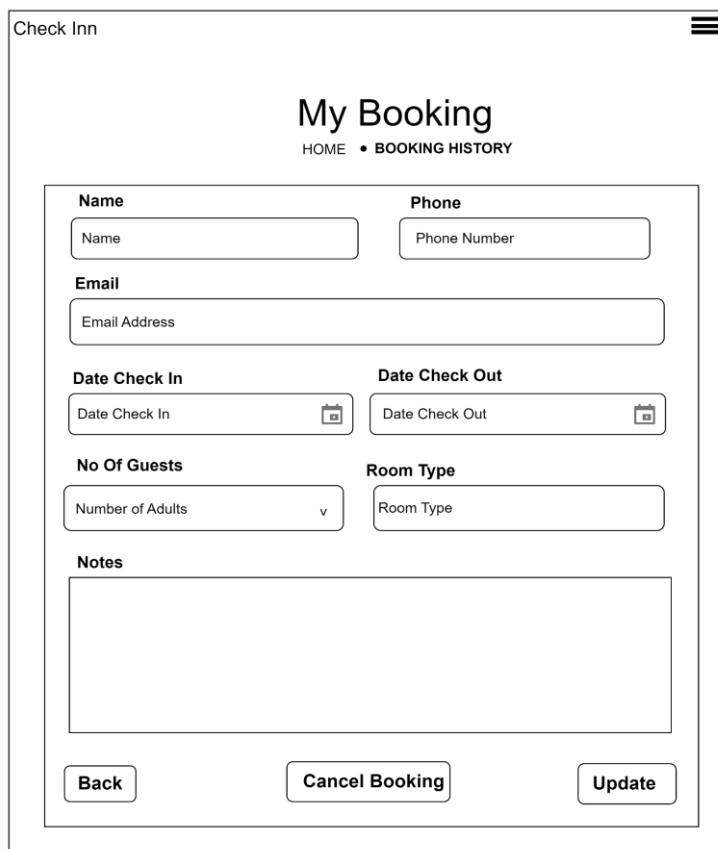


Name: Update my hotel booking	ID: 27
Stakeholders and goals: Customer - wants to update hotel booking from the Website.	
Description: The Customer wants to update hotel booking from the Website so that I can edit the hotel booking details.	
Actors: Customer	
Trigger: The Customer clicks on the hotel booking	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the homepage. 4. Customer clicks on the “MyPage” button. 5. System directs the customer to the “Booking History” page. 6. Customer clicks on the specific hotel booking they want to edit. 7. System directs the user to “Edit My Booking” page 8. Customer makes the edits. 9. Customer clicks on the “Update” button. 10. System validates the edits. 11. System displays updated Booking History Page. 12. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype



The low-fidelity prototype of the "My Booking" page is titled "Check Inn" at the top left. At the top center, it says "My Booking" with "HOME • BOOKING HISTORY" below it. The page contains several input fields and buttons:

- Name**: Two input fields for "Name" and "Phone Number".
- Email**: An input field for "Email Address".
- Date Check In** and **Date Check Out**: Each has a date input field followed by a calendar icon.
- No Of Guests**: A dropdown menu for "Number of Adults" with a downward arrow icon.
- Room Type**: An input field for "Room Type".
- Notes**: A large text area for notes.
- Buttons**: At the bottom are three buttons: "Back", "Cancel Booking", and "Update".

- High-Fidelity Prototype

Check Inn



Name

Phone

Email

Date Check In

Date Check Out

No Of Guests

Room Type

Notes

Back
Cancel Booking
Update

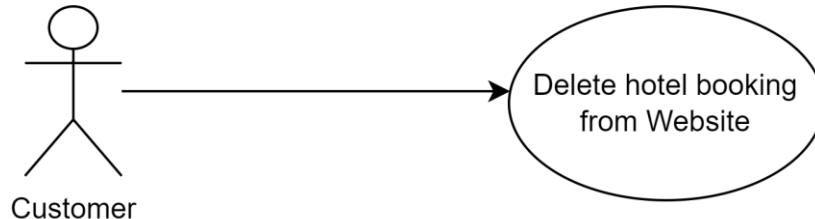
● Test Case

TEST CASE ID:	TC-27						
System Name:	CheckInn Website						
Module Code:	CST1222						
Test requirement:	1. Launch CheckInn Website in the browser 2. Has successfully logged in 3. Clicks on "MyPage" button						
Purpose	Let user Customer updates hotel booking						
Pass	2						
Fail	0						
Pending	0						
Number of test cases:	2						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer Update Hotel Booking:							
TC-27-1	Update hotel booking	1. The user selects a hotel booking 2. System directs user to "Edit My Booking" page 3. Customer makes the edits. 4. Customer clicks on the "Update" button. 5. System validates the edits	User see a. the updated hotel booking table in the landing page	System updates the hotel bookings in the database and direct user to landing page displaying the updated hotel booking table	18/05/2023	Pass	-
Check Customer return back/exit Update Hotel Booking page:							
TC-27-2	Returns back to website's landing page from Reservation form	1. User clicks on "Back" button	User is directed to user back to website's landing page	System direct user back to website's landing page	18/05/2023	Pass	-

Use Case # 28

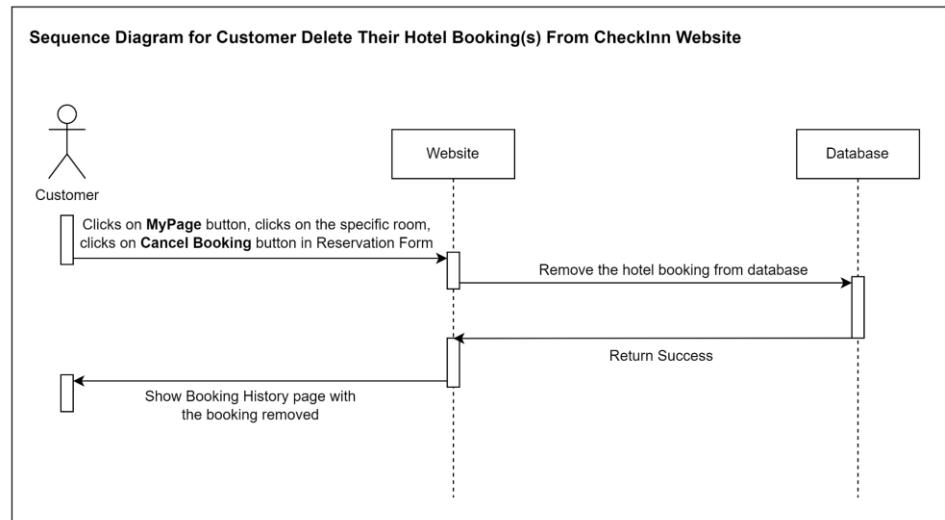
As the **Customer**,

I want to be able to **delete my hotel booking** from the **Website** so that I can cancel my hotel booking.



Name: Delete my hotel booking	ID: 28
Stakeholders and goals: Customer - wants to Delete hotel booking from the Website.	
Description: The Customer wants to Delete hotel booking from the Website so that I can edit the hotel booking details.	
Actors: Customer	
Trigger: The Customer clicks on “Cancel Booking” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the homepage. 4. Customer clicks on the “MyPage” button. 5. System directs the customer to the “Booking History” page. 6. Customer clicks on the specific hotel booking they want to edit. 7. System directs the user to “Edit My Booking” page 8. Customer clicks on the “Cancel Booking” button. 9. System displays updated Booking History Page. 10. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



- Low-Fidelity Prototype

Check Inn

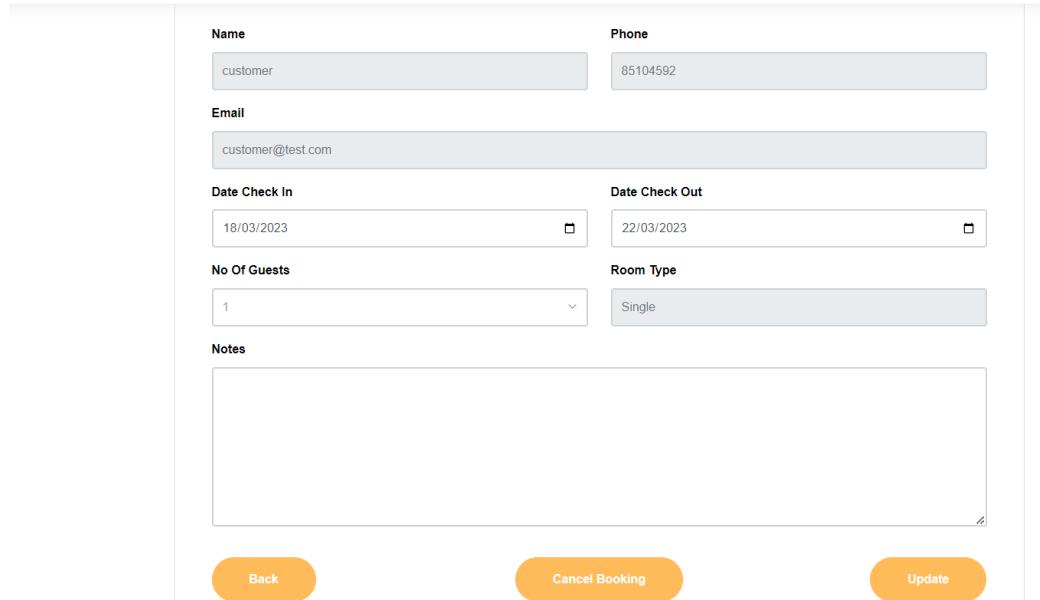
My Booking

HOME • BOOKING HISTORY

Name	Phone	
<input type="text"/>	<input type="text"/>	
Email		
<input type="text"/>		
Date Check In	Date Check Out	
<input type="text"/>	<input type="text"/>	
No Of Guests	Room Type	
<input type="text"/>	<input type="text"/>	
Notes		
<input type="text"/>		
<input type="button" value="Back"/>	<input type="button" value="Cancel Booking"/>	<input type="button" value="Update"/>

- High-Fidelity Prototype

Check Inn

The form consists of several input fields and buttons:

- Name:** customer
- Phone:** 85104592
- Email:** customer@test.com
- Date Check In:** 18/03/2023
- Date Check Out:** 22/03/2023
- No Of Guests:** 1
- Room Type:** Single
- Notes:** (Empty text area)

Buttons at the bottom:

- Back
- Cancel Booking
- Update

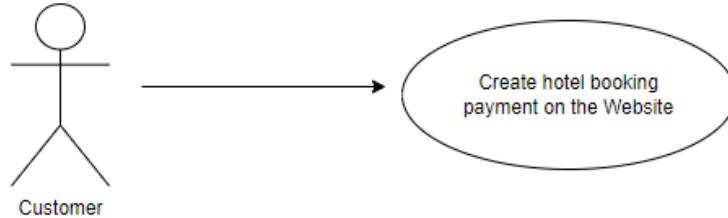
- Test Case

Test Case ID:	TC-28						
System Name:	CheckInn Website						
Module Code:	CSIT321						
Test requirement:	1. Launch CheckInn Website in the browser 2. Has Successfully logged in						
Purpose	Let user Customer delete a hotel booking						
Pass	1						
Fail	0						
Pending	0						
Number of test cases	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer view all user account pages							
TC-28-1	Delete a hotel booking	1. The user selects a hotel booking. 2. System directs user to "Edit My Booking" page. 3. Customer makes the edits. 4. Customer clicks on the "Cancel Booking" button.	User sees a updated hotel bookings table in the Booking History page.	System updates the hotel bookings in the database and direct user to Booking History page displaying the updated hotel bookings table .	18/03/2023	Pass	Logged in as Customer

Use Case # 29

As the **Customer**,

I want to be able to **create payment** for hotel booking on the **Website** so that I can make payment for booking.



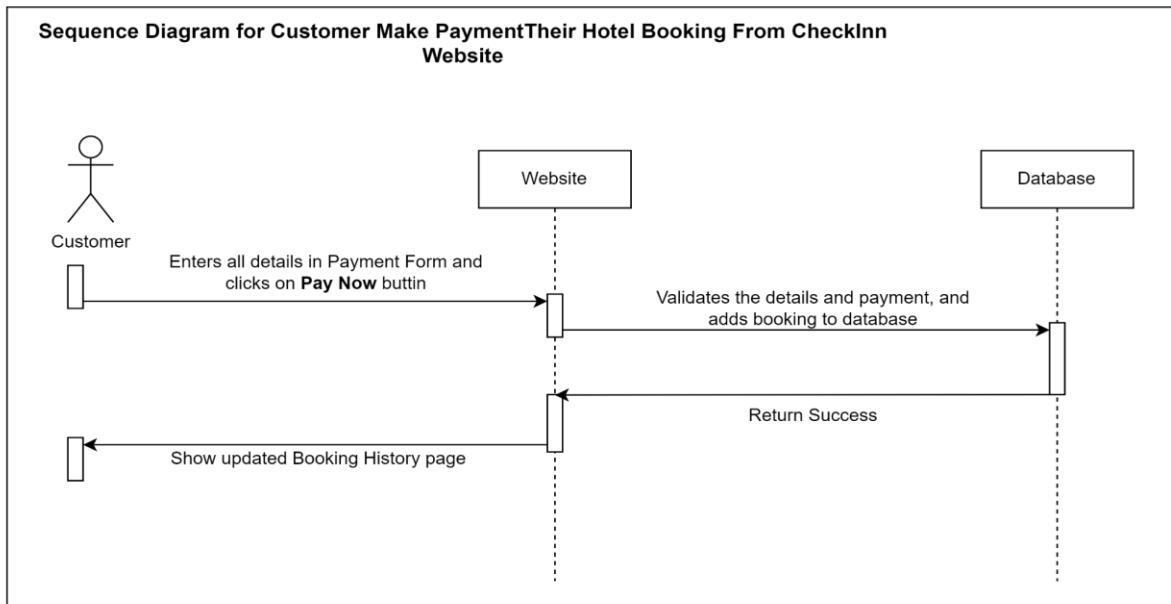
Name: Create hotel booking payment	ID: 29
Stakeholders and goals: Customer - wants to create payment for hotel booking.	
Description: The Customer wants to create payment for hotel booking to secure booking.	
Actors: Customer	
Trigger: The Customer clicks on “Proceed to Payment” button	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow: <ol style="list-style-type: none"> 1. System will display payment form page. 2. Customer will enter the necessary details 3. Customer clicks on Pay Now button 4. System validates the details 5. System directs the customer to the “Booking History” page. 6. End. 	
Sub-flows:	
Alternative/Exceptional flows:	
<i>If blank Card Number:</i> <p>4.1a – The system will pop-up a message dialog box stating “Please enter Card Number”.</p> <p>4.1b – The user will click the “Ok” button to acknowledge and enter Card Number.</p>	
<i>If blank Expiry Date:</i> <p>4.2a –The system will pop-up a message dialog box stating “Please enter Expiry Date”.</p> <p>4.2b –The user will click the “Ok” button to acknowledge and enter Expiry Date.</p>	
<i>If blank CVC:</i> <p>4.3a –The system will pop-up a message dialog box stating “Please enter CVC”.</p> <p>4.3b –The user will click the “Ok” button to acknowledge and enter CVC.</p>	
<i>Invalid Card Number:</i> <p>4.4a – The system will pop-up a message dialog box stating “Invalid Card Number”.</p> <p>4.4b – The user will click the “Ok” button to acknowledge and re-enter Card Number.</p>	
<i>Invalid Expiry Date:</i>	

- 4.5a –The system will pop-up a message dialog box stating “Invalid Expiry Date”.
 4.5b –The user will click the “Ok” button to acknowledge and re-enter Expiry Date.

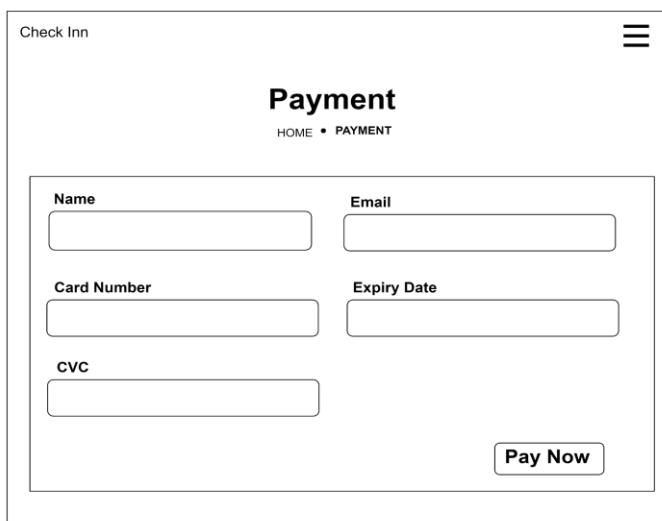
Invalid CVC:

- 4.6a –The system will pop-up a message dialog box stating “Invalid CVC”.
 4.6b –The user will click the “Ok” button to acknowledge and re-enter CVC.

- Sequence Diagram

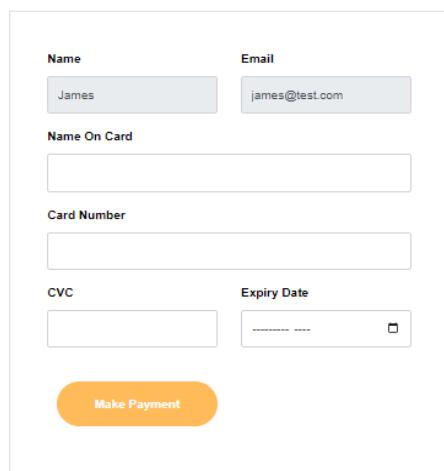
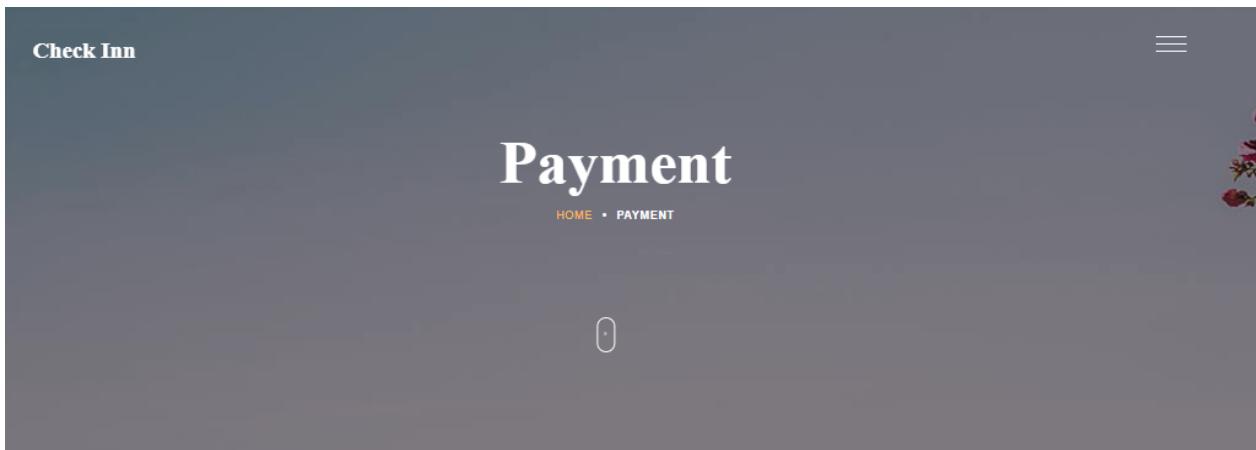


- Low-Fidelity Prototype



The low-fidelity prototype shows a 'Payment' page under 'Check Inn'. The page has a header with 'Check Inn' and a menu icon. Below the header, the word 'Payment' is centered, with 'HOME • PAYMENT' underneath. The main content area contains four input fields: 'Name' (with a placeholder box), 'Email' (with a placeholder box), 'Card Number' (with a placeholder box), and 'Expiry Date' (with a placeholder box). Below these is a 'CVC' field (with a placeholder box) and a large 'Pay Now' button at the bottom right.

- High-Fidelity Prototype



A form for entering payment information. It includes fields for Name (James), Email (james@test.com), Name On Card (empty), Card Number (empty), CVC (empty), Expiry Date (empty), and a "Make Payment" button.

Name	Email
James	james@test.com
Name On Card	
<input type="text"/>	
Card Number	
<input type="text"/>	
CVC	Expiry Date
<input type="text"/>	<input type="text"/> <input type="button" value=""/>
<input type="button" value="Make Payment"/>	

HOTEL ID:

hba

ROOM ID:

hba_1

NUMBER OF NIGHT STAYING:

1

TOTAL PRICE:

\$2100

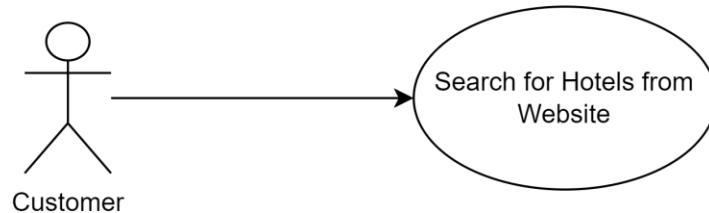
- Test Case

TEST CASE ID:	TC-29						
System Name:	CheckInn Website						
Module Code:	CSTT321						
Test requirement:	1. Launch CheckInn Website in the browser 2. Successfully logged in as Customer						
Purpose	Let user Customer be able to create payment for hotel booking						
Pass	7						
Fail	0						
Pending	0						
Number of test cases:	7						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer create payment for hotel booking:							
TC-29-1	Create payment	1. The user views "Payment Form" page. 2. The user enters the necessary details. 3. The system validates the details	User will see the Payment confirmed message	System direct user to Payment confirmed message	18/05/2023	Pass	✓
Check Customer create payment for hotel booking:							
TC-29-2	Enter blank Card Number	1. User views "Payment Form" page. 2. User leave Card Number blank 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Please enter Card Number"	System pop-up a message dialog box displaying "Please enter Card Number"	18/05/2023	Pass	✓
TC-29-3	Enter blank Expiry Date	1. User views "Payment Form" page. 2. User leave Expiry Date blank 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Please enter Expiry Date"	System pop-up a message dialog box displaying "Please enter Expiry Date"	18/05/2023	Pass	✓
TC-29-4	Enter blank CVC	1. User views "Payment Form" page. 2. User leave CVC blank 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Please enter CVC"	System pop-up a message dialog box displaying "Please enter CVC"	18/05/2023	Pass	✓
TC-29-5	Enter invalid Card Number	1. User views "Payment Form" page. 2. User enters invalid Card Number 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Invalid Card Number"	System pop-up a message dialog box displaying "Invalid enter Card Number"	18/05/2023	Pass	✓
TC-29-6	Enter invalid Expiry Date	1. User views "Payment Form" page. 2. User enter invalid Expiry Date 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Invalid Expiry Date"	System pop-up a message dialog box displaying "Invalid enter Expiry Date"	18/05/2023	Pass	✓
TC-29-7	Enter invalid CVC	1. User views "Payment Form" page. 2. User enter invalid CVC 3. User enters the necessary details. 4. System validates the details	User sees a pop-up a message dialog box displaying "Invalid enter CVC"	System pop-up a message dialog box displaying "Invalid enter CVC"	18/05/2023	Pass	✓

Use Case # 30

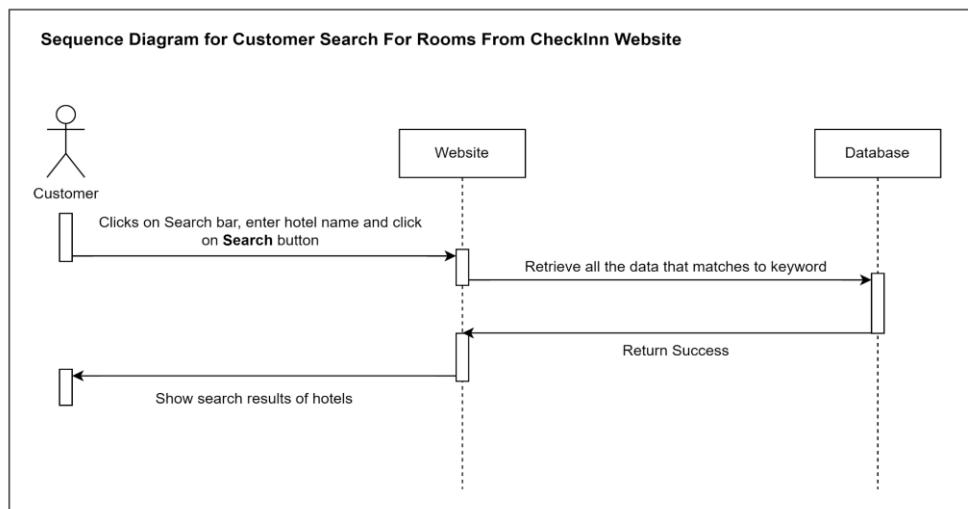
As the **Customer**,

I want to be able to **search for hotels** on the **Website** so that I can retrieve the hotel details to book.



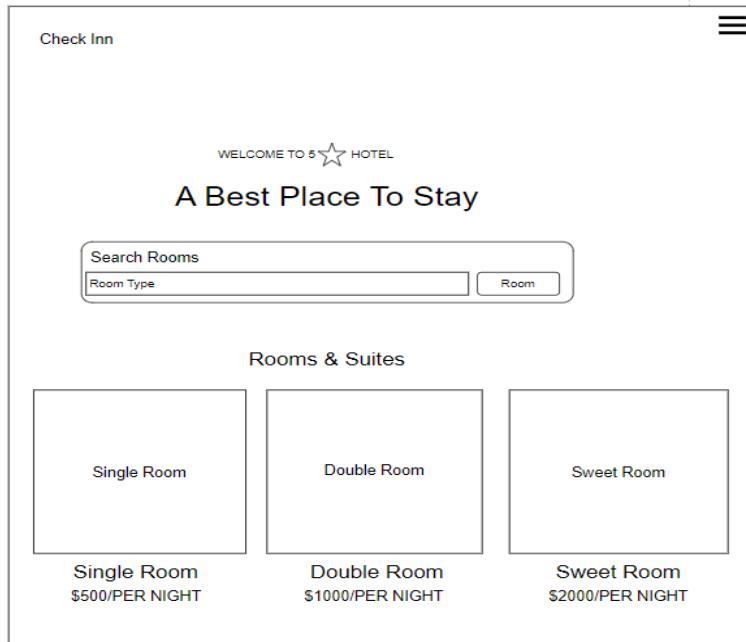
Name: Search for rooms	ID: 30
Stakeholders and goals: Customer - wants to search hotels on the Website.	
Description: The Customer wants to search hotels from the Website to retrieve the hotel details to book.	
Actors: Customer	
Trigger: The Customer clicks on the search bar	
Prerequisite: Successfully logged in as Customer in Website	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer log into their account. 3. System directs the customer to the homepage. 4. Customer search for hotel name in the search bar. 5. Customer clicks on “Search” button 6. System directs the user to landing page with the match search results 7. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram



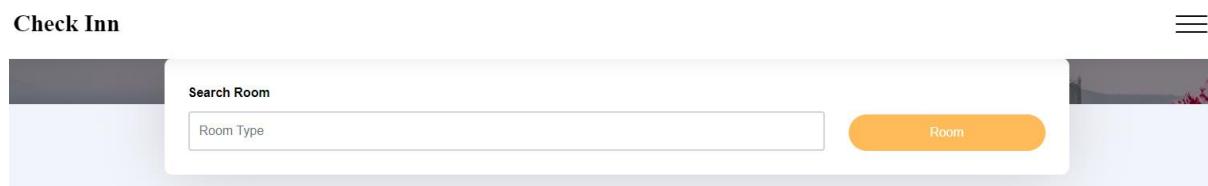
- Low-Fidelity Prototype

In the homepage, there will be a search bar, key in the room type:



- High-Fidelity Prototype

In the homepage, there will be a search bar, key in the room type:



Rooms & Suites



- Test Case

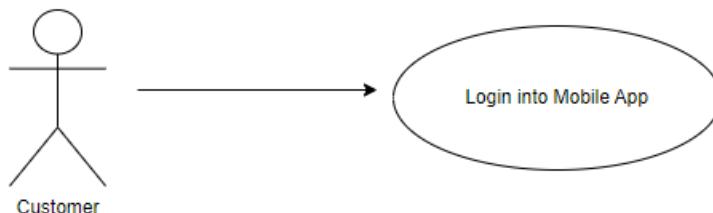
Test Case ID:	TC-30						
System Name:	CheckInn Website						
Module Code:	CSIT3211						
Test requirement:	1. Launch CheckInn Website in the browser 2. Has Successfully logged in						
Purpose:	Let user Customer search for hotels or rooms						
Pass:	1						
Fail:	0						
Pending:	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check System Admin search hotels or rooms:							
TC-30-1	Search Hotel Using Hotel Name	1. The user clicks on the "Search" button. 2. The system directs the user to the "Search" page. 3. The user can enter the Hotel Name inside the search text box. 4. The user clicks on the "Search" button. 5. The system would display the hotels that matches the search input.	User directs user to the "Search Page" and see search results in scrollable table with closest similarities	System directs user to the "Search Page" and displays search result in scrollable table with closest similarities result	18/05/2023	Pass	-

Mobile App

Use Case # 31

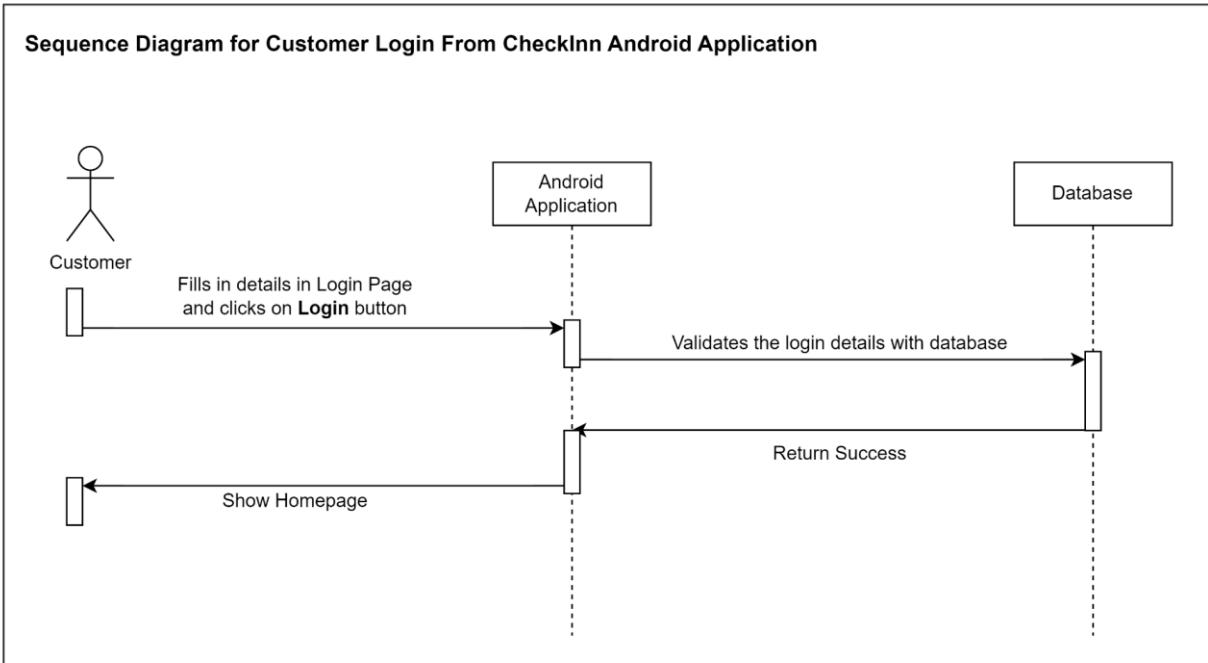
As the **Customer**,

I want to be able to **login** into the **Mobile App** so that I can access the system.

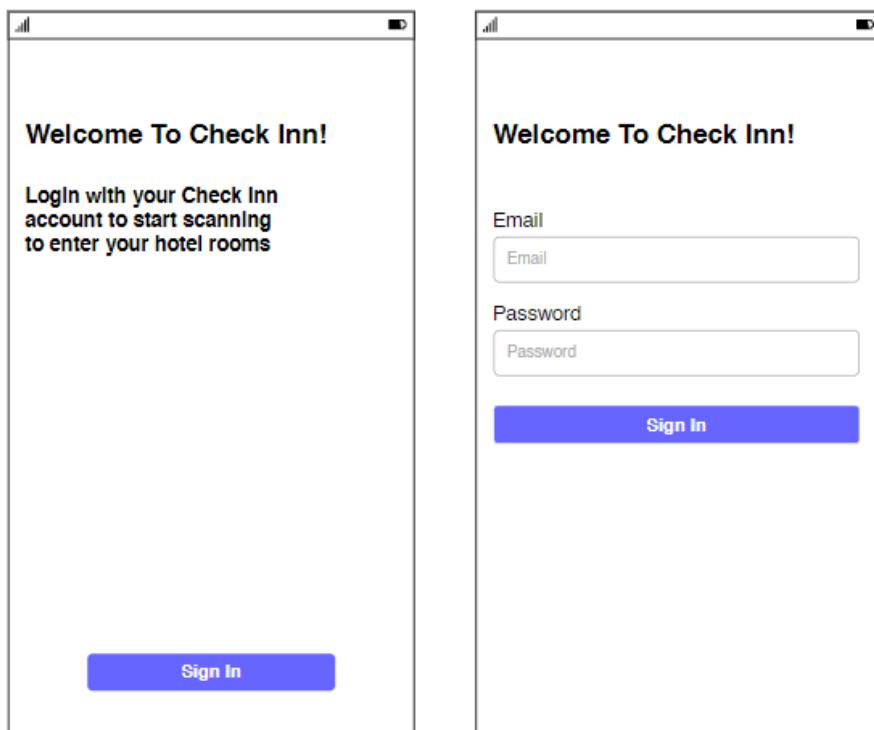


Name: Login to the Mobile App	ID: 31
Stakeholders and goals: Customer - wants to login into the Mobile App.	
Description: The Customer wants to login into the Mobile App to access the system.	
Actors: Customer	
Trigger: The Customer opens the Mobile App.	
Prerequisite:	
Normal flow: <ol style="list-style-type: none"> 1. Customer will open the Mobile App. 2. System will display a login page. 3. Customer will enter the username and password. 4. Customer clicks on the “Login” button. 5. System will validate the username and password combination. 6. System will navigate to the mobile app homepage. 7. End. 	
Sub-flows:	
Alternative/Exceptional flows: <p><i>Incorrect email and password combination:</i></p> <p>4.1a – The system will pop-up a message dialog box stating “Incorrect Email or Password”. 4.1b – The user will click the “Ok” button to acknowledge and re-enter email and password.</p> <p><i>If blank email:</i></p> <p>4.2a –The system will pop-up a message dialog box stating “Please enter Email”. 4.2b –The user will click the “Ok” button to acknowledge and enter email.</p> <p><i>If blank password:</i></p> <p>4.3a –The system will pop-up a message dialog box stating “Please enter Password”. 4.3b –The user will click the “Ok” button to acknowledge and enter password.</p>	

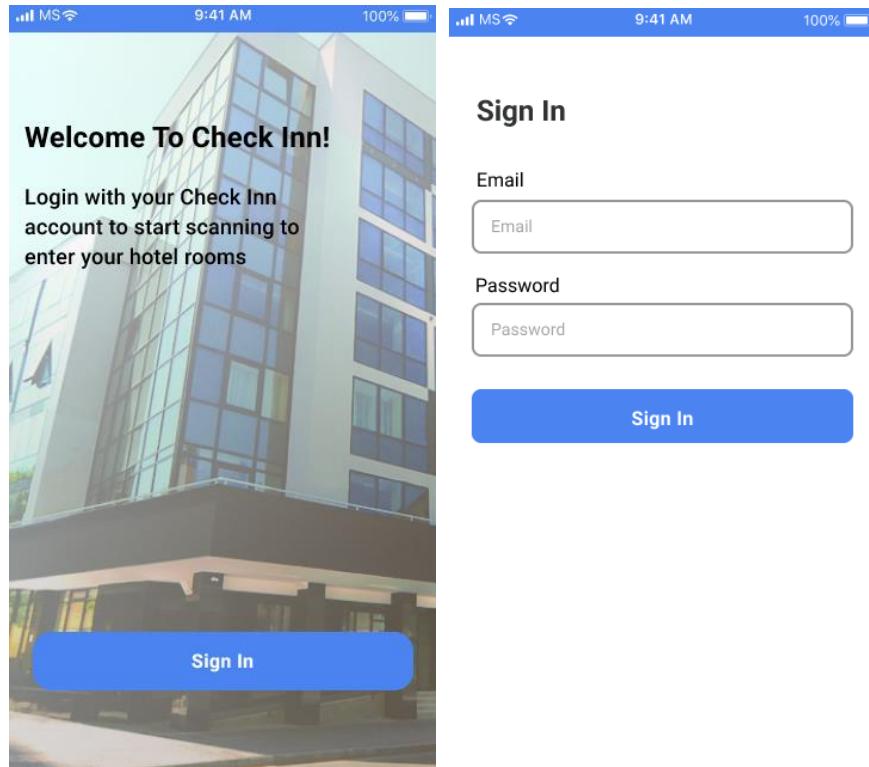
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



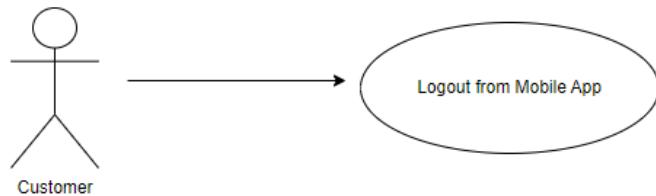
- Test Case

TEST CASE ID:	TC-31						
System Name:	CheckInn application						
Module Code:	CSIT321						
Test requirements:	1. Launch CheckInn Application in Android phone						
Purpose	Let user Customer be able to login into the application						
Pass	5						
Fail	0						
Pending	0						
Number of test cases:	5						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer Login Validation:							
TC-31-1	Launch application	1. User launch application in browser	User sees login page appears on screen	System display login page appears on screen	18/05/2023	Pass	
TC-31-2	Input valid email and password	1. The system will prompt the user for the login credentials: Email and password. 2. The user will enter the email and password. 3. The user clicks on the "Login" button. 4. The system will validate the email and password credentials. 5. The system will navigate to the landing page.	User will see the application's landing page	System will direct to application's landing page	18/05/2023	Pass	
Check Customer Login Validation:							
TC-31-3	Enters invalid email and password	1. User enter invalid email and password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Incorrect Email or Password"	System pop-up a message dialog box warns "Incorrect Email or Password"	18/05/2023	Pass	
TC-31-4	Enters blank email	1. User enter blank email and non-blank password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Please enter Email"	System pop-up a message dialog box warns "Please enter Email"	18/05/2023	Pass	
TC-31-5	Enters blank password	1. User enter non-blank email and blank password credentials 2. User click "Login" button 3. message appears 4. User click "OK" button to acknowledge message	User sees a pop-up a message dialog box warns "Please enter Password"	System pop-up a message dialog box warns "Please enter Password"	18/05/2023	Pass	

Use Case # 32

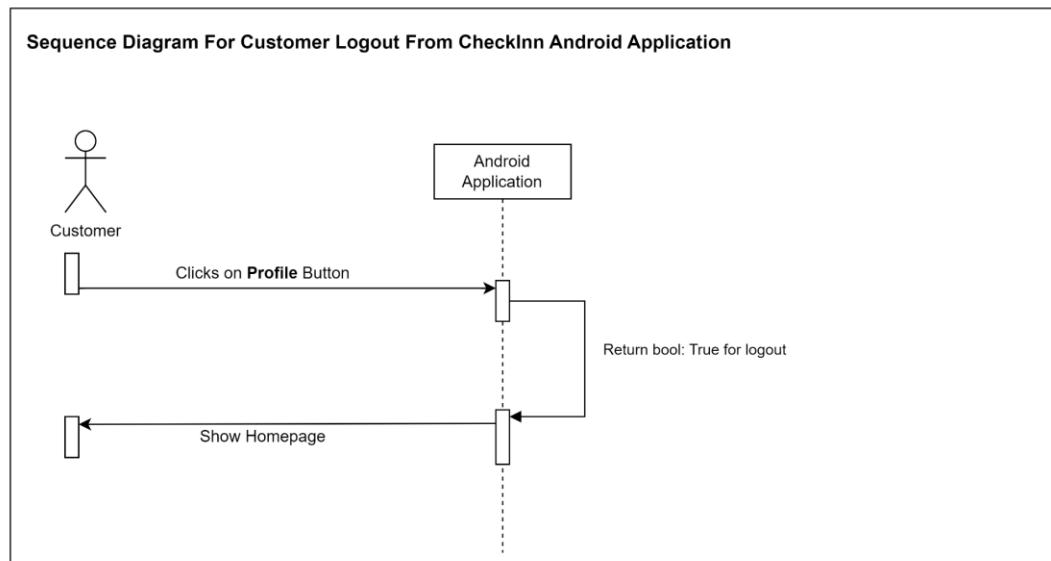
As the **Customer**,

I want to be able to **logout** from the **Mobile App** so that I can exit the system.

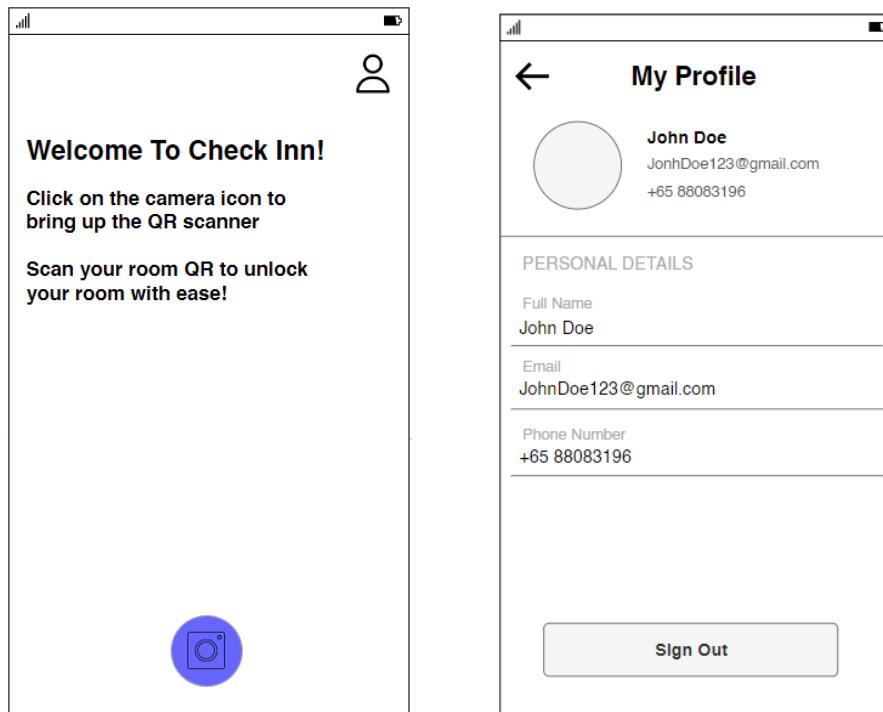


Name: Logout from the Mobile App	ID: 32
Stakeholders and goals: Customer - wants to log out from the Mobile App.	
Description: The Customer wants to logout from the Mobile App.	
Actors: Customer	
Trigger: The Customer clicks on “Logout” button	
Prerequisite: Successfully logged in as Customer in Mobile App	
Normal flow:	
<ol style="list-style-type: none"> 1. Customer clicks on “Profile” button 2. System will direct them to Profile Page 3. Customer clicks on “Log Out” button 4. System will direct them to the Login page 5. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

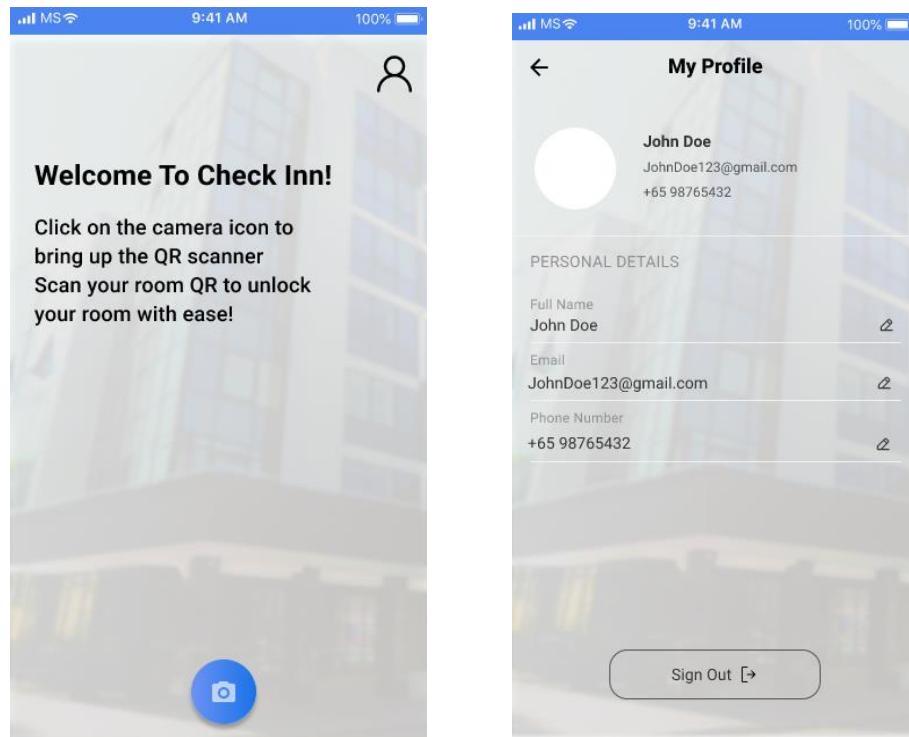
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



- Test Case

Test Case ID: TC-32							
System Name: CheckInn application							
Module Code: CSIT321							
Test requirement: 1. Launch CheckInn Application in Android phone 2. Successfully logged in as Customer in Mobile App							
Purpose: Let user Customer be able to log out of the application							
Pass: 1							
Fail: 0							
Pending: 0							
Number of test cases: 1							
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer log out:							
TC-32-1	Log out of Customer account	1. The user clicks on the "Profile" button. 2. System will direct them to Profile Page. 3. Customer clicks on "Log Out" button 4. System will direct them to the Login page.	User is directed user to application's login page	System will direct user to application's login page	18/05/2023	Pass	

Use Case # 33

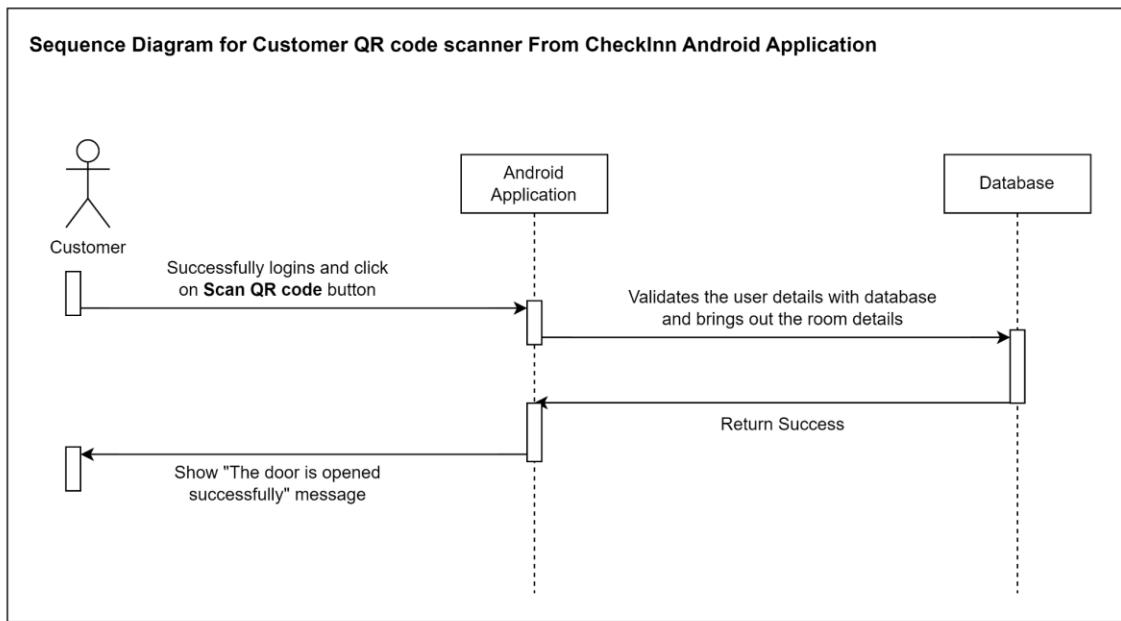
As the **Customer**,

I want to be able to **use the QR code scanner** from the **Mobile App** so I can scan the QR code to access the hotel room that I have booked / check-out (sub flow).

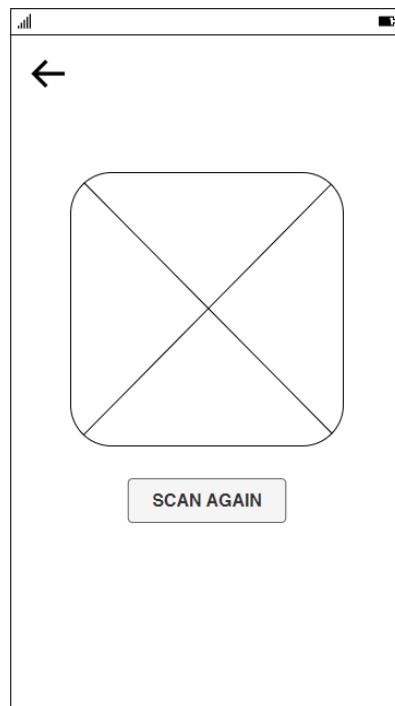


Name: Use QR code scanner from the Mobile App	ID: 33
Stakeholders and goals: Customer - wants to use QR code scanner through the Mobile App.	
Description: The Customer wants to use the QR code scanner through the Mobile App to access the hotel room or check-out.	
Actors: Customer	
Trigger: The Customer clicks on “Scan QR code” button	
Prerequisite: Successfully logged in as Customer in Mobile App	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the mobile app homepage. 4. Customer clicks on the “Scan QR code” button. 5. System directs the customer to the QR code scanner. 6. Customer scans the QR code and clicks on “Confirm” button. 7. System validates the details 8. System prompts the user for fingerprint. 9. End. 	
Sub-flows: Fingerprint authentication	
Alternative/Exceptional flows:	
<p><i>Invalid access:</i></p> <p>7.1a – The system will pop-up a message dialog box stating “User access denied”.</p> <p>7.1b – The user will click the “Ok” button to acknowledge</p>	

- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



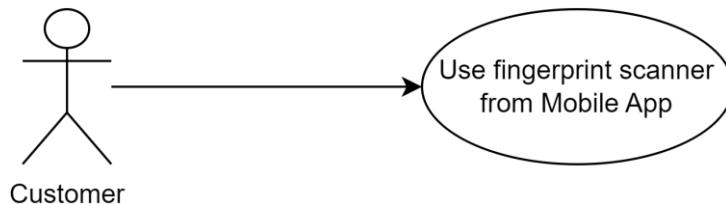
- Test Case

TEST CASE ID:	TC-33						
System Name:	CheckInn Website						
Module Code:	CSIT321						
Test requirement:	1. Launch CheckInn Application in Android phone 2. Successfully logged in as Customer in Mobile App						
Purpose:	Let user Customer use QR code scanner to check-in/check-out from Mobile App						
Pass:	2						
Fail:	0						
Pending:	0						
Number of test cases:	2						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer use QR code scanner to check-in/check-out:							
TC-33-1	Use QR code scanner to check-in/check-out	1. Customer clicks on "Scan QR Code" button 2. System displays a scanner 3. Customer scans the QR code using the scanner and clicks on "Confirm" button 4. System validates the details 5. System prompts user for fingerprint	User sees two buttons: "Open the hotel room" and "Check-out" and clicks either one	System validates the user details and displays two buttons: "Open the hotel room" and "Check-out".	18/05/2023	Pass	
Check QR code provided by Customer for Validation:							
TC-33-2	Invalid user	1. Customer clicks on "Scan QR Code" button 2. System displays a scanner 3. Customer scans the QR code using the scanner and clicks on "Confirm" button 4. System validates the details	User sees a pop-up a message dialog box warns "User access denied"	System validates the user details and displays message dialog box warns "User access denied"	18/05/2023	Pass	

Use Case # 34

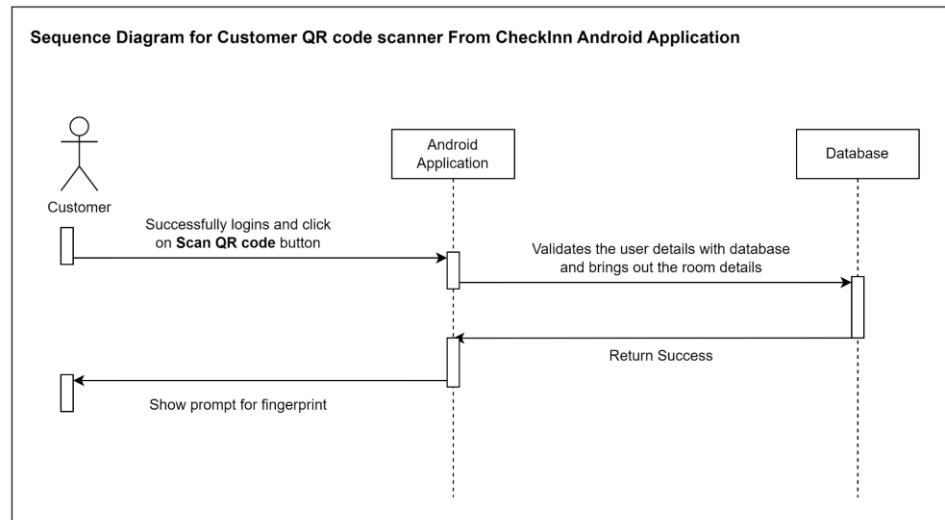
As the **Customer**,

I want to be able to **use fingerprint authentication** from the **Mobile App** so I can access the hotel room using Two-Factor Authentication.

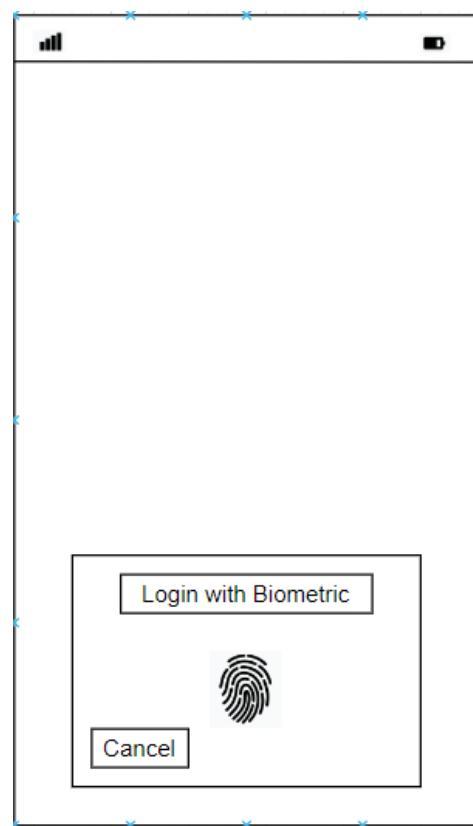


Name: Use fingerprint scanner from Mobile App	ID: 34
Stakeholders and goals: Customer - wants to use fingerprint scanner through the Mobile App.	
Description: The Customer wants to use fingerprint scanner to authentication through the Mobile App to again access the hotel room.	
Actors: Customer	
Trigger: The Customer successfully finished the QR Code Scanner	
Prerequisite: Successfully logged in as Customer in Mobile App	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the mobile app homepage. 4. Customer clicks on the “Scan QR code” button. 5. System directs the customer to the QR code scanner. 6. Customer scans the QR code and clicks on “Confirm” button. 7. System validates the details. 8. System prompts the customer to provide their fingerprints. 9. Customer provides their fingerprints. 10. System validates the details. 11. End. 	
Sub-flows: QR Code Scanner	
Alternative/Exceptional flows:	
<i>Invalid fingerprint:</i> 10.1a – The system will pop-up a message dialog box stating “Please try again”. 10.1b – The user will click the “Ok” button to acknowledge	

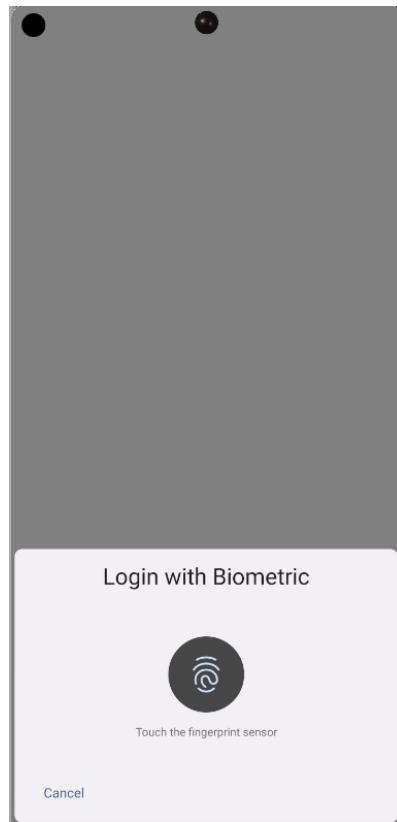
- Sequence Diagram



- Low-Fidelity Prototype



- High-Fidelity Prototype



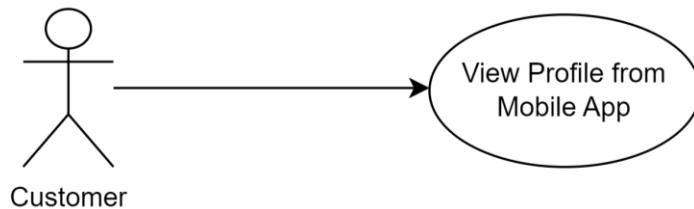
- Test Case

Test Case ID:	TC-34						
System Name:	CheckInn Website						
Module Code:	CSIT321						
Test requirement:	1. Launch CheckInn Application in Android phone. 2. Successfully logged in as Customer in Mobile App						
Purpose:	Let user Customer access their hotel room using Two-Factor Authentication						
Pass	2						
Fail	0						
Pending	0						
Number of test cases:	2						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Note
Check Customer can access room from Mobile App:							
TC-34-1	View Unlocked Page	1. The user provides their fingerprints 2. The system validates their fingerprints 3. The system directs the user to the Unlocked page	User sees that their granted access to the room in "Unlocked" page	System retrieves data from database and directs user to "Unlocked" page and displays the unlocked sign	18/05/2023	Pass	*
Check Customer can access room from Mobile App:							
TC-34-2	Invalid Fingerprints	1. The user provides no input or half fingerprint 2. The system validates their fingerprints	User sees a pop-up a message dialog box warns "Please try again"	System validates the user fingerprints and displays message dialog box warns "Please try again"	18/05/2023	Pass	*

Use Case # 35

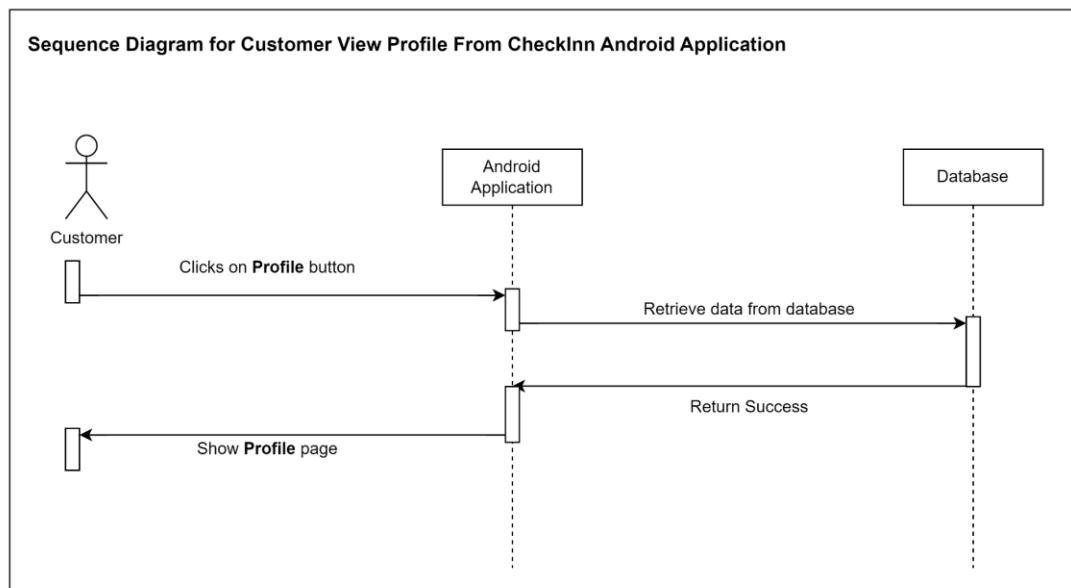
As the **Customer**,

I want to be able to **view my Profile** from the **Mobile App** so I can view my profile..

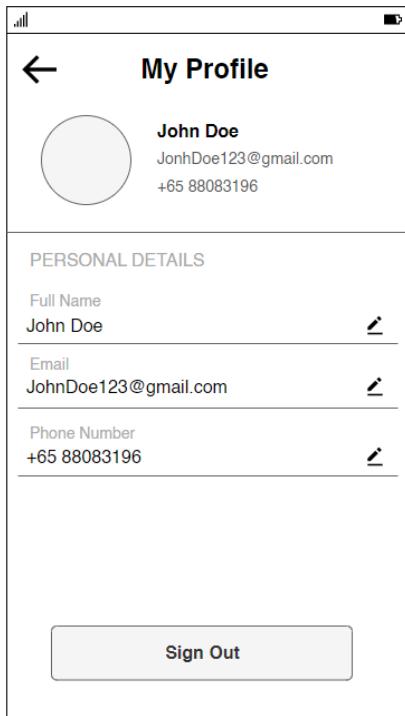


Name: View Profile from Mobile App	ID: 35
Stakeholders and goals: Customer - wants to view Profile through the Mobile App.	
Description: The Customer wants to view Profile through the Mobile App.	
Actors: Customer	
Trigger: The Customer clicks on “Profile” button	
Prerequisite: Successfully logged in as Customer in Mobile App	
Normal flow:	
<ol style="list-style-type: none"> 1. System will display the login page. 2. Customer logs into the account. 3. System directs the customer to the mobile app homepage. 4. Customer clicks on the “Profile” button. 5. System directs the customer to the Profile page. 6. End. 	
Sub-flows:	
Alternative/Exceptional flows:	

- Sequence Diagram

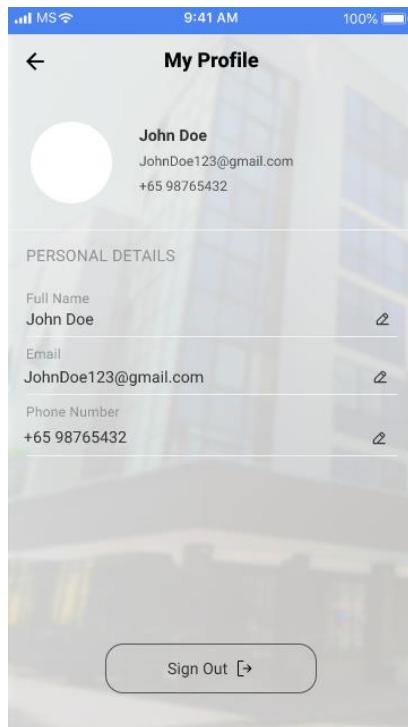


- Low-Fidelity Prototype



The image shows a low-fidelity wireframe prototype of a mobile application's profile screen. At the top, there is a header bar with a back arrow icon and the title "My Profile". Below the header, there is a circular placeholder for a user's profile picture. To the right of the placeholder, the user's name "John Doe" is displayed, along with their email address "JohnDoe123@gmail.com" and phone number "+65 88083196". A horizontal line separates this section from the "PERSONAL DETAILS" section below. The "PERSONAL DETAILS" section contains three input fields: "Full Name" (with "John Doe" entered), "Email" (with "JohnDoe123@gmail.com" entered), and "Phone Number" (with "+65 88083196" entered). Each input field has a small edit icon (pencil) to its right. At the bottom of the screen is a rectangular "Sign Out" button.

- High-Fidelity Prototype



- Test Case

Test Case ID:	TC-35						
System Name:	CheckInn Website						
Module Code:	CSIT321						
Test requirement:	1. Launch CheckInn Application in Android phone 2. Successfully logged in as Customer in Mobile App						
Purpose:	Let user Customer view profile						
Pass:	1						
Fail:	0						
Pending:	0						
Number of test cases:	1						
ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Test date	Result	Notes
Check Customer view a gender page:							
TC-35-1	View Profile	1. The user clicks on the "Profile" button 2. The system directs the user to the Profile page	User sees "Profile" page	System retrieves data from database and directs user to "Profile" page	18/05/2022	Pass	*

7. Non-Functional Requirements

This section contains the non-functional requirement for our platform, such as

7.1. Usability

The platform, both on the website and mobile application, should be user-friendly and easy to use and navigate. This will be achieved using a simple and intuitive user interface with clear and concise navigation, making it accessible and straightforward to use. The use of a consistent design with similar layouts, fonts, and colours is to reduce confusion and improve the user experience.

7.2. Scalability

The platform should be able to handle a huge number of users - both our customers, hotel operators and an increasing number of bookings. The platform's database will be designed with this in mind.

7.3. Compatibility

The platform should be compatible with multiple devices and browsers. Currently, the team is only planning on making the mobile application available on android devices. The system should be optimised for mobile devices, with a responsive design that adapts to different screen sizes and orientations. Our website should be compatible with multiple web browsers, such as Google Chrome, Mozilla Firefox, and Microsoft Edge to ensure a consistent user experience for all customers.

7.4. Availability

The platform should be available 24/7 and have a high uptime to ensure the customer experience is never compromised due to downtime caused by our servers. To ensure this, the team will choose the most reputable and reliable database servers and web hosting platforms.

7.5. Security

Since our platform requires customers to log in with sensitive information such as passwords to access the platform, our system should protect sensitive customer information and secure payments. Sensitive customer information and payments will be encrypted to prevent unauthorised access.

The QR code generation process should also be encrypted using secure algorithms. The sensitive data encoded in the QR code such as the customer's passwords and payment information, will also be encrypted to prevent unauthorised access to sensitive information

8. Other Requirements

This section contains the other requirements for our platform which does not fit in the previous sections of the document

8.1. System Features

The front-end system would consist of the website and the mobile application. This will also work hand-in-hand with our back-end database system. The majority of the services will be done on the website itself however the customer has to use the mobile application to authenticate themselves to unlock the hotel room door and check-out of the hotel. Hence, our mobile application and the website need to work together for the customer to carry out the authorisation of QR codes.

8.2. Security Requirements

As for the security perspective, each customer of the hotel will only be able to make one booking at a time. This prevents the customer from making many dummy bookings and helps the hotel to prepare and adhere to the customers' necessary requirements. By preventing customers from booking more than once at one time, it would also help the hotel to prevent business loss from happening due to the customer's absence and accommodate more customers.

Before confirming the booking, we would request the customer to pay their full accommodation fees to confirm their booking and reserve the room. Through this, we can prevent customers from not showing up for their reservation, as the customer has paid the full amount ahead of time.

8.3. Regulations

8.3.1. Personal Data Protection Act (PDPA)

The Personal Data Protection Act (PDPA) in Singapore helps to ensure that an individual's personal data is collected, used and disclosed in a way that respects their privacy rights. By appointing a Data Protection Officer (DPO) and implementing reasonable security arrangements, organisations can demonstrate their commitment to complying with the PDPA and protecting each individual personal data.

It is important to note that the PDPA applies to personal data stored in both electronic and non-electronic formats, which provides a comprehensive framework for protecting personal data in Singapore.

The exclusion of certain categories of data such as business contact information and information collected by public agencies is designed to balance the need to protect personal data with the need for organisations to collect and use data for legitimate purposes.

By following the PDPA guidelines, organisations can help to build trust with their customers and clients by demonstrating their commitment to responsible data management practices.

8.3.2. Cybersecurity Act

The Singapore Cybersecurity Act plays a critical role in protecting sensitive information and ensuring the overall security of the digital space in Singapore.

By regulating the owners of critical information infrastructure and cybersecurity service providers, the Cybersecurity Act helps to create a secure and trustworthy digital environment for businesses and individuals to operate in.

Moreover, the Cybersecurity Act requirements for certain cybersecurity incidents and data breaches help to keep authorities informed and enable them to respond quickly to potential threats. The practices and performance standards imposed by the act help to ensure that businesses and organisations are following the best practices for cybersecurity.

In the current digital age where exchange of information and data is becoming increasingly common, it is very important to prioritise cybersecurity and ensure the protection of sensitive information.

By implementing measures such as the Cybersecurity Act, we can help to build a safe and secure digital landscape for all.

9. Marketing Research

9.1. What is QR Code?

QR Codes are two-dimensional barcodes that may accommodate more data than traditional barcodes because information is stored in both the horizontal and vertical axes of the code. The additional data is what gives QR Codes their wide range of applications. A smart device's camera, which is readily available, can be used to scan QR Codes. Most frequently, online links or arbitrary data like text or images are stored in QR Codes. The data can still be retrieved even if up to 30% of the QR Code is damaged or difficult to see.

9.2. Security Issues Regarding QR Code

<https://www.jotform.com/blog/qr-code-security/>

The security issues associated with QR Codes come from their destination rather than the codes themselves because QR Codes cannot be hacked. Hackers can replace valid QR Codes with malicious ones or manipulate the system so the customer is directed to a phishing website rather than the real one. Additionally, a lot of individuals are unable to identify between a safe QR Code and one that is malicious, which raises a lot of security issues.

9.3. How are we going to tackle this issue?

To tackle this issue, our team will split into two groups where the red group will be doing weekly penetration testing and spoof testing while the blue group will be doing weekly patches with tools and processes. The red group will inform the blue group of the loopholes that they have found in the system. This is a pseudo activity for the project.

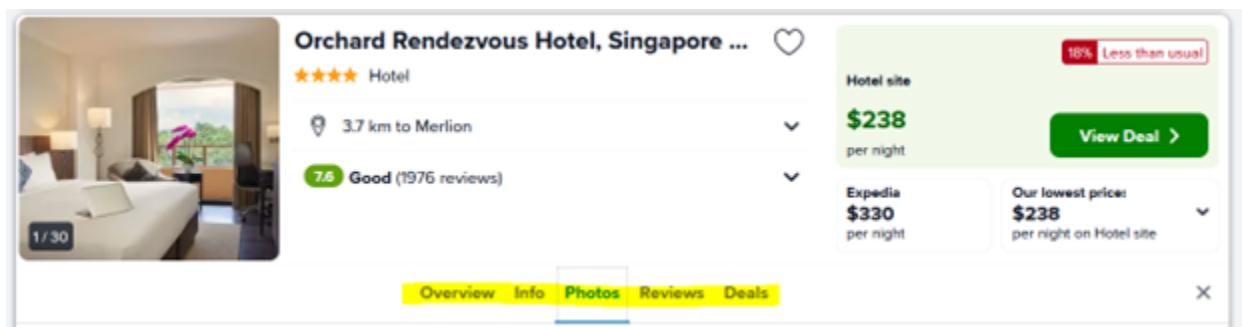
The red group will act as the hacker to do penetration testing into the system, they will write down the problems that they have seen and the blue group will be the defenders/troubleshooters. They will defend the system and see how the system security can be improved.

After testing, the team will gather together to revise and rectify the system according to the weaknesses that the groups spotted to assure the customers that our system is safe to use.

9.4. Design Research



Our team did online research on some common and popular hotel booking websites and found out some of the important attributes when a customer is reserving the hotel. The attribute such as date for check in and check out, type of room, price per night and guest rating can be considered to implement into the website.



Some other features such as overview, information of the hotel room, photo, reviews, and hotel room deals are also important factors when customers select their hotel which are also a good consideration to implement into the website. Overview provides an overall description such as the location, price tending, amenities & facilities. Information provides the detailed location of the hotel. The photo provides images of the hotel room, facilities, and dining room. Reviews provide the rating overview and reviews of the guests that have stayed in that hotel.

The idea of QR code to unlock the hotel room and checking out of the hotel that my team came out with is due to Covid-19. Covid-19 is an ongoing global pandemic that has affected all the country since 2020. This virus has resulted in a lot of death cases in all the country. This virus is transmitted through the air and the

risk of breathing it is high when the people are in proximity. To reduce the risk of getting Covid-19, people must reduce their point of contact in the public and follow strict community health protocols and practices.

By using QR code to unlock the door, customers can avoid touching the hotel room key. Using the QR Code also provides more security. This accelerates the check-in process for the customer, making it a seamless hassle-free experience. Checking out of the hotel with a QR code allows the customer to avoid interaction with the hotel front desk which reduces the risk of getting Covid-19.

9.5. Similar Multi-factor transaction authorisation done in other platforms

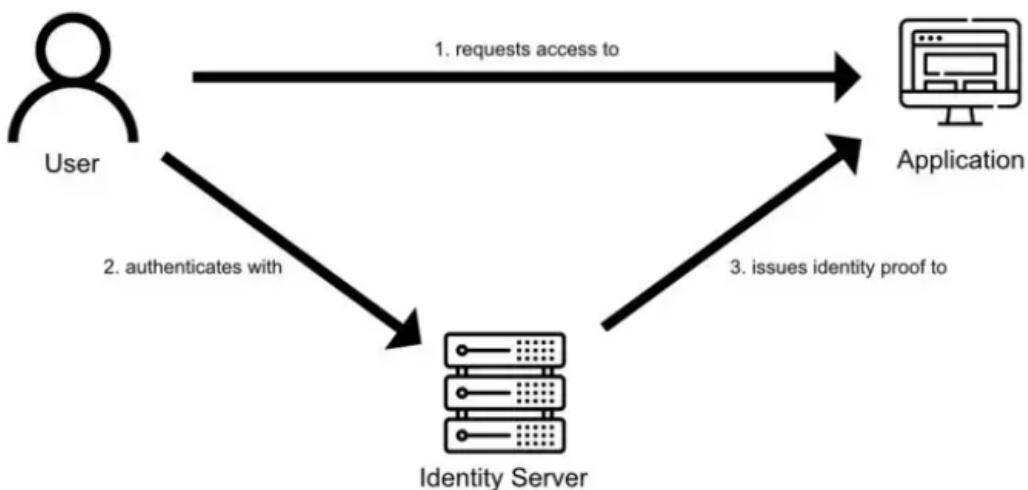
Other than using QR code, many other platforms use passwords, cards, retina scans, voice recognition and fingerprints to authorise the identity of the user for transactions to happen. Not only passwords will be used, but like some cases where the transaction is part of another transaction then authorisation would not be necessary as the computer or user would have been identified earlier.

9.5.1 OAuth2

OAuth2 is an industry-standard authorisation protocol. It offers specific authorisation flows for web applications, desktop applications, mobile phones and living room devices. The flow is considered high-level flow if it is done between user, application and identity server.

An example of high-level flow is shown below:

- 1) User requests access to the application
- 2) User authenticates with the Identity Server
- 3) Identity Server issues identity proof to Application



To briefly explain this flow:

- 1) User is redirected to the Identity Server where both authentication and authorisation are handled

- 2) Client (that is the application that requests the user information) gets authorisation by the user to get the necessary information. That is done by configuring the right scopes. Scopes resemble the type of data that a specific client has access to.
- 3) Scopes are requested by application during the authentication process.
- 4) Once the authentication is completed, the application might be able to give authorisation for the requested data.
- 5) When given authorisation, the data will be added to the payload of the token and passed to the application.
- 6) As there is a possibility that a user can have multiple roles, an identity server is set up for all of the clients or users of the application. Each user would be assigned to different roles depending their rights level
- 7) The identity server could share the assigned roles to a specific user in token

9.6. Research on places where application-specific authorisation logic should be built in

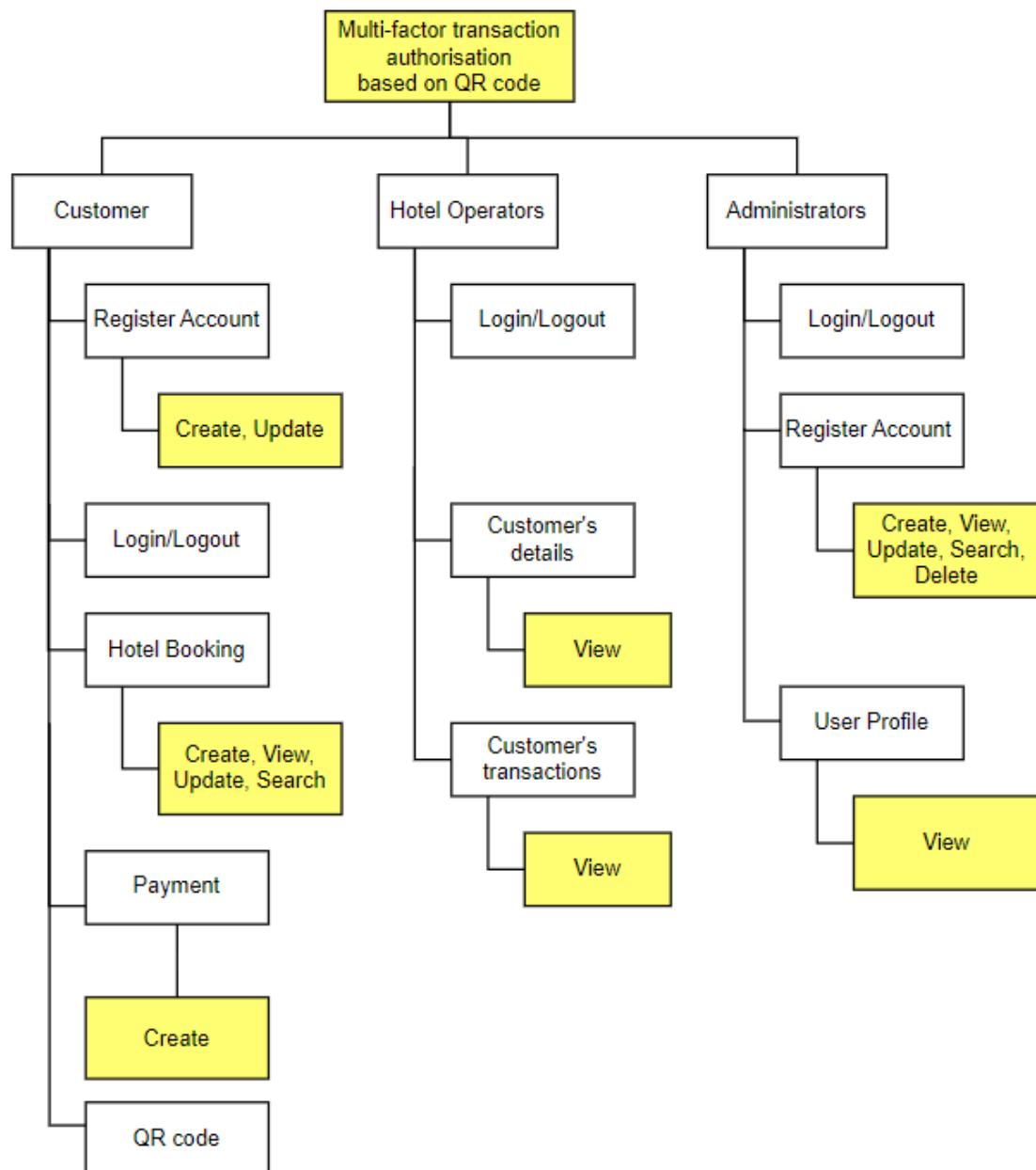
As the authentication is built in a separate centralised responsible service, it becomes harder for application-specific authorisation logic. In a small architecture, the authorisation and authentication services should not be exposed directly to the consuming application, if not managing the connections will not be manageable.

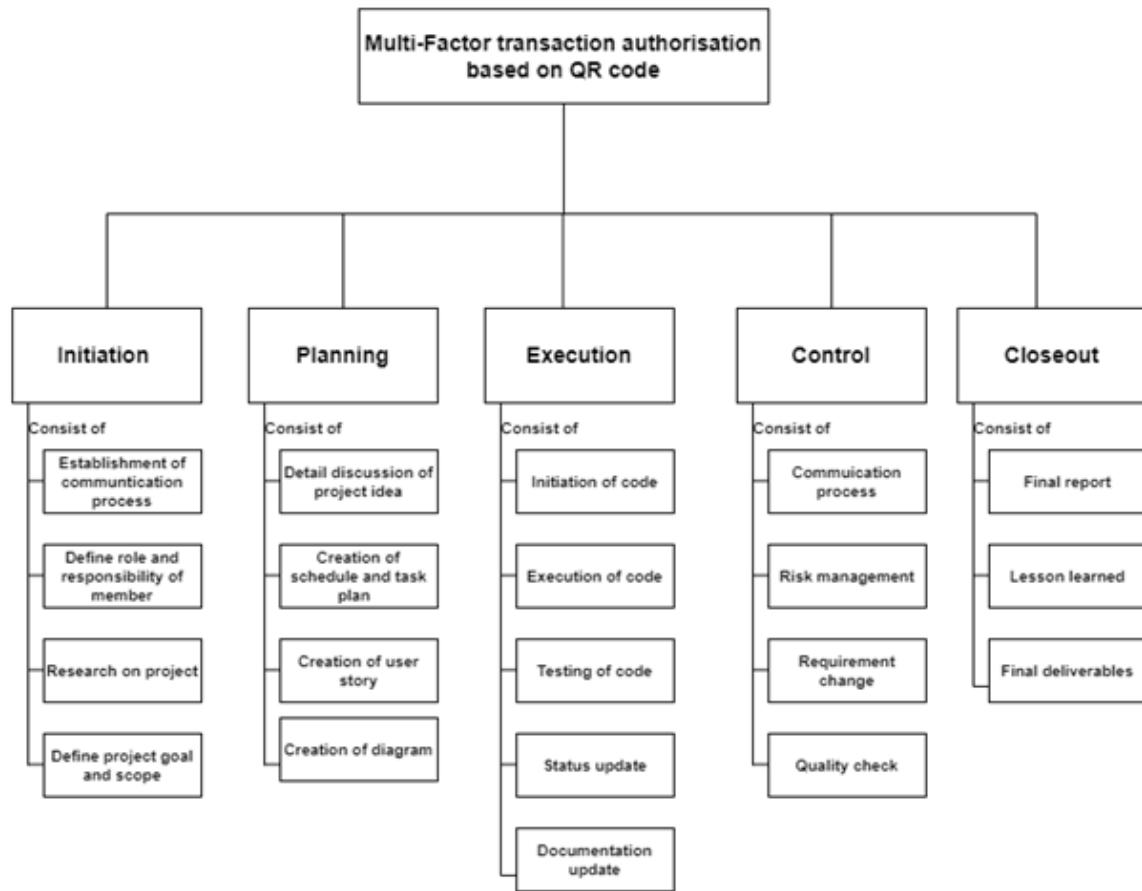
Through implementing an API gateway, it helps to create a single point for consumers to communicate with. The API gateway routes the requests to the upstream microservices. If multiple consumers use the application or software at one time, then creating specific API gateways could be a solution to create separate specific endpoints for each consumer. Hence this variation is called the Backends for Frontends pattern by which endpoints are specifically created and used for each and every customer. The only minus of this variation is that it adds another separate service per customer that needs to be maintained.

Another way to manage the authorisation in microservices is by API gateway should pass the JWT along with the request towards the microservice. This can be done for extensive application-specific authorisation checks. The JWT would contain roles assigned to users or customers. API gateway would be handling the authentication still and the validation of token will be done when the microservice receives the request. By searching through the JWT for the role, the microservice can now check whether the user is authorised for the wanted request. Through this way, the application-specific only request can only be implemented in one place. The downfall of this service is that when roles change frequently, there is more workload.



10. Work Breakdown Structure





In the Initiation Phase, our team will be establishing a chain of communication with our teammate and project supervisor. Each team member will be given a role and responsibility based on the project. Our team will conduct a detailed research on the project and come out with the project requirement that inline with the project scope. Our team will define the project goal and list of requirements for our project.

In the Planning Phase, project ideas will be generated, and detailed team discussion is done to assess the feasibility of the project idea. The task schedule and plan will be going forward once the approval of the project idea is being done by our team. Detailed outline of user stories and diagrams help to provide more aid and visualise to our team when writing a program for our project.

In the Execution Phase, program codes are initialised in this stage. The code will be executed, and testing of code will be done to ensure each requirement is working. To ensure our schedules are on the same track, a weekly status update is done. Weekly update of the documentation on the progress is done to ensure the workload is lesser for the final report.

In the Control Phase, our team may encounter some miscommunication with each other as the project progresses. To reduce the risk of miscommunication, our team will have an informal short update of what they are doing every two days within ourselves. Some of the project requirements may require a longer period. Thus, it may result in other requirements not being able to finish on time.

To reduce the risk of failing the project requirements, our team will work on the program code together rather than doing it individually. After the prototype demonstration, our team will anticipate a change request. By knowing that there will be an upcoming change request, our team will be able to anticipate some of the problems that are bound to happen along the way and create solutions that might help to resolve the issue.

Our team will also conduct a quality check to fulfil our basic requirements and functionalities.

Finally for the Closure Phase, our team will write the final report, lessons learned and give a final presentation on our final product.

11. Gantt Chart

We have split the Gantt Chart into three parts.

1) Research and Discovery

2) Design and Development Phase:

3) Documentation and Testing Phase

12. Appendix

Hotel Room Door

A IOT device is installed in the wall and it is wired to an electric door strike lock. A small motor on the strike is connected to a power supply and the current holds the strike plate in the locked position to keep the door closed and secured.

The IOT device is connected to the online server through the Wi-Fi network. The QR code will be pasted on the IOT device.

Once the customer is authenticated to unlock the hotel room, it will update the database of the unlock door status from closed to open. Once the IOT device detects the open status from the database, it will send a signal to the electric door strike lock to unlock the door.