



APPENDIX A

KV6003: Individual Computing Project

Project Terms of Reference

Scott Mains (w18003567)

Computer Science with Web Development

Table Reservation System for a Restaurant

General Computing Project



CONTENTS

Background

Proposed Work

Aims and Objectives

Skills

Bibliography

Resources

Structure and Contents of Project Report



BACKGROUND

Since the COVID-19 pandemic, it has become a necessity to reserve tables at restaurants to avoid queues and prevent congestion from customer walk-ins due to the new government health guidelines. It is also useful for restaurant owners to minimize queues from walk-ins so that the customers don't have to wait long to be seated and in turn be disappointed. A table reservation system can help tackle these problems by allowing the user to reserve a table for a specific timeslot. The table will then become available again for another user once the allotted time has occurred. With the data collected from the table bookings, the restaurant owner can also see when his restaurant is most popular on a seasonal, weekly, and daily basis. On top of this, the user will have the ability to create an account and opt-in for special offers from the restaurant. This data can be very important in a customer relationship context, as they can promote their brand on popular trends with special offers etc. It is important for the restaurant to try and build a customer relationship, as it is a sure-fire way to boost sales.

A client has reached out to me to design a system that allows the user to book a table at their restaurant based on such perimeters as date, time, and table size. They have requested that the reservation system is to be integrated into a website that promotes their restaurant, with a booking page that takes the user directly to the table booking form. Being able to view the table layout visually would also be a bonus, so that they can choose exactly where they want to sit inside the restaurant. The client has requested an interface for him as the admin, so they can view all the bookings that are submitted on a daily and monthly basis. The admin page can also display analytics for the data collected from the reservation system; so that they can see extent of when most bookings occur and on what days. The user can also create an account when reserving a table at the restaurant. With the account, the user can build up loyalty points with every successful booking they have at the restaurant. This is great for both the restaurant owner and the user, as the user gets a discount, and the owner gets invaluable data and a returning customer.

There are two different literature focuses for my project. The first aspect is the actual functionality of the booking system - how you can integrate user-centered design and HCI principles to make it as fluid and easy to use as possible. I also want to look at literature regarding existing booking systems and the best way to go about creating one. This will involve looking at existing reservation system reports and seeing what I can use in my own design. The other is the CRM aspect, in which a business should integrate it to identify and understand the need of customers. Both components complement each other as the user data gained from the booking system can be used as the basis of the CRM system.

The idea of the project was first created from the need of my client. They are a new restaurant owner with the need of a booking system so that they can fill up the tables at their restaurant without the need of people contacting him directly. The CRM aspect was something I proposed to them after they contacted me with their request. I feel like the idea of a CRM system is interesting and a great strategy to boost his overall sales; the world is becoming entirely digital now since the Millennial era and CRM is a great way to target users in the digital world to gain and retain customers. It has been shown that when a customer has a positive attitude towards a brand, they will continue to purchase from them. Although obvious, this is something I want to try and implement into my own design. I will do this with a loyalty system and by making the overall product as fluid and user friendly as possible.



PROPOSED WORK

There are three different parts to the project that need to be carried out. The first part is the front-end of the web application. This is going to use HTML, CSS and JavaScript to display the client-side booking system that the user will see and use to input their data. The front-end will also include an admin dashboard area. The second part is the back-end database which takes the information from the booking form and inputs it into a database. The last part is going to be the CRM system, and how it can use the user data generated from the database for things such as special offers and a loyalty system.

The front-end web application will initially display a home page, and then there will be a button for the user to click on to navigate to the booking form. The booking form will then display a calendar plugin created with JavaScript, asking the user to input a date and time, and then displaying all the available tables on that specific slot with a visual representation of the table layout. Once they have chosen a slot, they will be asked if they want to create an account or use guest credentials. The front-end aspect will also involve creating a dashboard for the restaurant owner (admin) to see all the confirmed booking reservations for the month. From this dashboard, they will also be able to control different settings and booking restrictions such as cancelling specific days for bookings due to holidays or other reasons. They can also cancel existing booking reservations and set specific tables as unavailable for certain days.

The back end of the system is going to use a MySQL database integrated with PHP. When the user inputs their data on the client-side, the information will be inputted into the database. The information from the database will then be used and displayed in the admin dashboard. The backend is essentially all the data that will be displayed on the front-end. It will hold all the data regarding the booking slots and user details. For example, once the user confirms their table reservation, every bit of information will be stored in the database for that specific person, and the email address they inputted can be used to send them a booking confirmation to their slot.

The final part is the CRM system, and this will use the data in the database as means to increase customer relationship with the restaurant. One part of this is using the email data that the user can opt-in to when they book a slot and sending them special offers on specific dates that they usually attend the restaurant. Another part of this is integrating a loyalty system for the user. Each time they successfully book a slot at the restaurant, they can gain loyalty points which eventually builds up to a discount.

To carry out all the above, I will need to investigate how existing table reservation systems are made using a literature review. My literature review will need to cover all the different aspects of the web application: the booking system, the calendar plugin, how a CRM system is implemented and privacy and data protection. Learning about all these things in a literature review will help me develop my own version by bringing all the things I've learnt together.



AIMS

- To investigate how a table reservation system is created and how the data can be used in Customer Relationship Management.
- To build a web application for a pizza restaurant that integrates a table reservation system and a CRM for the restaurant owner.

OBJECTIVES

1. Create a literature and technology review:
 - Look at existing booking/reservation systems and identify their good and bad features. Take ideas that could be implemented in my own design.
 - Examine existing plugins used for calendars which can be integrated in the reservation system.
 - Find existing CRM systems used in restaurants and find ways I can use user data created from the reservation system.
 - Find academic articles on CRM system and table reservation systems.
 - UCD design for restaurant website and booking systems.
 - Look at privacy and data protection issues regarding storing user data.
2. Establish and prioritize the requirements of the product:
 - Find out client's requirements
 - Create storyboard and initial vision of the application
 - Find target audience(s)
 - Persona(s) and Scenarios
 - User requirements document
3. Create designs for the web application:
 - Initial design storyboard and wireframes.
 - Low & High-fidelity prototypes
 - Use case diagrams
 - Class diagrams
 - Entity Relationship diagrams
4. Create the product based on the design specification.
5. Test the product:
 - Functional testing
 - Usability testing (Users and admin)
6. Make changes to the application based on the testing process.
7. Evaluate the product.
8. Evaluate the processes and my own performance



SKILLS

I am currently undergoing my degree with a focus on web development, so I have decided to focus my project on a system integrated into a web application. The web modules I have undertook during my time at Northumbria University are going to play a major role in how I create my project. The following modules: Web Programming and Web Technologies, both have extensive content that I can take from. Another module that will be extremely useful is Human Computer Interaction. This module will be useful in the Research & Planning and the design aspect of the system.

For the front-end part of my project, I will need to enhance my knowledge on React so that I can create the client-side booking system and calendar. I am already familiar with CSS and HTML so creating the overall website shouldn't be too hard, but I will need to expand my knowledge on Bootstrap to make the website more cohesive and responsive.

For the back end, I am still new to implementing databases and using PHP to integrate web applications. I will need to study more in-depth both things to successfully complete my project. The Web Application Integration module I'm learning during this semester will come in handy with this.

Finally, I am completely new to the concept of a CRM system. I will enhance my knowledge through the literature review and learn how to use the data gained from the booking system to develop a CRM system for the restaurant owner.

SOURCES OF INFORMATION

SFENRIANTO, LORIO PURNOMO. "View of Customer Relationship Management (CRM) Analysis and Design to Provide Customer Service in The Culinary Field (Case Study Restaurant XYZ)." *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, <https://turcomat.org/index.php/turkbilmamat/article/view/5787/4827>.

Okine, Aaron. "A WEB- BASED RESERVATION SYSTEM." *Theseus*, https://www.theseus.fi/bitstream/handle/10024/36707/Okine_Aaron.pdf?sequence.

"EU General Data Protection Regulation: Changes and Implications for Personal Data Collecting Companies - ScienceDirect." *ScienceDirect.Com | Science, Health and Medical Journals, Full Text Articles and Books.*, <https://www.sciencedirect.com/science/article/abs/pii/S0267364917301966>.

Mahdin, Hairulnizam. *SWEET8BAKERY BOOKING SYSTEM*. <https://core.ac.uk/download/pdf/231742562.pdf>. Accessed 9 Nov. 2021.



RESOURCES

I will require the following resources to complete my project:

- Computer – I have this at home and will be able to use the ones available at the university as well.
- Web Server – I can use Newnumyspace for my server needs. I will also be using XAMPP at home as a local server and then upload to newnumyspace at the final stage.
- UML Diagram Tool – I will need a diagram creation tool to design all the different UML diagrams required in the design process.
- Adobe XD – This will be used to create all the wireframes and prototypes used in the design stage

STRUCTURE AND CONTENTS OF PROJECT REPORT

My report will be structured into the following sections:

- Abstract – Overview of the report and the project. It will tell the reader how the report is going to be structured and telling them if the project was an overall success.
- Introduction – Introduces the purpose and scope of the project. This section will tell the reader what I want to achieve with the system and how I intend to do this. I will also explore the background of the project and what the client wants.
- Research and Planning:
 - Literature Review – this section will look at all the literature I've gathered that is relevant to my project. This will help look at their successes and failures and how I can implement their knowledge in my own design.
 - Requirements Capture – This will look include the requirements proposed by the client and look at potential target audience. I will then tailor my design regarding the user requirement plan.
 - Tools and Techniques – Look at all the methods and tools needed to develop my project. This will include all the programming languages and the database system I will use and how I will go about using them to achieve my goal.
- Design, Implementation and Testing:
 - Design – I will talk about how I designed the product in this section. The design will be based upon the findings of the Research and planning stage and will include all the design documents I have created. The final design of the system will then be identified, and I will talk about how I came to this final design.
 - Implementation – This section will focus on how I developed the system and all the practices and processes I went through to achieve the outcome.
 - Testing – I will look at all the testing processes I went through once my system has been created. I will then look at how successful the testing phase was and what changes I made.



- Evaluation – I will have a thorough evaluation of the final product and the processes I went through. I will first focus on the creation of the product and if it matches the requirements set out in the research and planning stage. I will then evaluate how I managed the project and if the overall aims and objectives were met.
- Conclusions and Recommendations – This final section will talk about if I met the overall outcomes of the project. It will also touch over all the successes and failures I endured. I will propose future improvements that could be made to make the project even more successful.

APPENDIX B

Target Group:	Potential consumers of pizza restaurant (Couples, Families)			
Competitors/Comparitors				Most Important findings:
Strengths	OpenTable	Simplybook.me	BookingNinja	<p>Most Important findings:</p> <p>Social media connectivity</p> <p>Intuitive for the customer - make it party size, date, then times available.</p> <p>Appointment reminders sent to the user via email or SMS</p> <p>Timer set on the booking to encourage the user to book quickly or else they will miss out.</p> <p>Restaurant information.</p>
	Very intuitive design for finding the available tables at a restaurant. The user is able to choose party size, date, and time for when they want to reserve a table. If there isn't a table available at that time, the system proposes availability within 2.5 hours of the time initially selected.	Only need an email to reserve booking and can be connected with either facebook or google for express checkout.	Provides the business with its own branded application that the user can download and use to book their services.	
	Once the user has found the slot they want, the system reserves the spot for the user for up to 5 minutes. This allows the user to take their time when entering their details and lets them decide if they want that specific time before someone else has the opportunity to swipe their slot.	Specific discounts can be applied depending on the time they want to book the table. This could be great for restaurants that have a lull of customers at certain times of the day and can encourage them to book a table during these times.	Connectivity with social platforms making booking a lot easier.	
	The user has the opportunity to either book the table as a guest or to sign up an account to make the booking. Creating an account incorporates a loyalty system which allows the user to collect points for future discounts.	Another clean layout only uses 3 colors similar to OpenTable: green, white, and black. Other colors are used sparingly to make certain bits of information stand out - such as how the safety precautions are in blue.	The website is completely personalised for the business rather than being a plugin. This means that the website can be a seamless booking experience for the user.	
	Very clean layout which is easily readable. The color scheme is a very basic red, white and black so all text stands out easily. The web application is very responsive to all different screen-sizes.		Reminders are scheduled for the user to alert them when their booking is. Can be sent via email or SMS. This will decrease the chance of people missing their booking slot.	
Weaknesses	Lots of additional information in regards to the booking. It provides the user with safety precaution information, a detailed description of the venue, reviews, location via maps and provides them everything they need to know in regards to the place they will be eating at.			<p>Keep irrelevant information away from the user.</p> <p>Do not present the user with so many options at one time, try minimize this.</p> <p>Don't lock full sections under a paywall.</p> <p>Don't add features that may annoy the user which in turn would put them off using the app.</p>
		Doesn't allow the user to choose how many people the booking is for	Booking not as intuitive as OpenTable. The user first chooses date, then time then the party size and it doesn't seem like the user will know if the table is available until the end of the process which could cause annoyance.	

APPENDIX C

Target Group:		Staff at pizza restaurant			Most Important findings:
		Competitors/Comparitors			
		OpenTable	Simplybook.me	Booking Ninja	
Strengths	Provides lots of operation solutions. Has the ability to analyze turnover times and can add users to waitlists and text them when their table is ready.	Rather than be a hub for lots of different restaurants, simplybook.me is much more personalized for the specific brand and makes it so the business can integrate it directly into their website or have them create a website for them.	Customer no-show prevention. Automatic confirmation emails and guest reliability scores.		Set Max occupancy of restaurnt Add CRM aspects such as add tags to top spenders etc. No-show prevention
	Great CRM aspects that maximize their relationship with their customers. Make better plans when guests pre-order party size-specific or seasonal menus. Send customized emails to regulars with special offers. Personalize the experience with top-spenders with tags and guest notes	Can split the restaurant into different sections so that it can be more personalised for the customer. For example, you can label some tables as "By window", and the user can select tables specific to being next to a window.	Can set the amount of occupancy you want within the restaurant. This is perfect for when you want to allow walk-ins as well. For example, if you have 40 available seats, you can allow up to 30 up-front bookings for allowance of walk-ins. You also might want to give certain tables more than the average time. Because of this, you dont want the occupancy to be total availability		
Weaknesses		Pricing is very expensive in comparison to it's competitors	Not specific to restaurant scheduling so doesn't offer as many CRM capabilities in comparison.		
		Some consumers say the interface dashboard is quite confusing initially.			



"I want to start using online, but its really daunting!"

Age: 42

Occupation: Plumber

Relationship Status: Married

Location: Peterlee

Character: The Overthinker

INTELLIGENT

HESISTANT

NOVICE

STIRRED

ABOUT MARK

Mark is a middle-aged family man that isn't took familiar with online booking systems. He wants to book a slot at the restaurant but hasn't used anything other than phone calls to reserve a table. He notices that the restaurant have just launched a new online booking system, and he's willing to try it for the first time.

GOALS

- Book a slot at the restaurant with no issues.
- Have confirmation that he was successful with the booking
- Be able to book directly from his mobile.

FRUSTRATIONS

- Prefers speaking to an actual person rather than online as he is more confident that everything will work out.
- Doesn't have a desktop computer so can't use a multitude of web applications that aren't responsive.

EXPECTATIONS

- An app that can tell him exactly how to book into the restaurant.
- Feedback that he successfully booked into the restaurant.



"I want to keep fit and healthy but I don't have the time to manage it!"

Age: 24

Occupation: Business Analyst

Relationship Status: Engaged

Location: Newcastle-upon-Tyne

Character: The Workaholic

PERFECTIONIST

FOCUSSED

AMBITIOUS

MOTIVATED

ABOUT CLAIRE

Claire is a young professional that is working in tech. She is familiar with using web applications and has booked a table at a restaurant before. She works long hours and wants to enjoy a meal with her boyfriend after a hard day at work. Because of this, she wants a seamless experience in booking a table at a restaurant which allows her to do it all online without having to interact with the restaurant at all.

GOALS

- Claire wants to book a slot with ease and not have to worry about any malfunctions.
- She wants to see how many loyalty points she can exchange for rewards as she has booked multiple times before.

FRUSTRATIONS

- Doesn't like having to ring up and speak to the restaurant directly as she's restless after a day at work and just wants it done quickly.
- Has lots of different food and restaurant booking apps on her phone and she loses track of all the loyalty she has accumulated

EXPECTATIONS

- To be able to book directly from the web application and view all her points shes gained from her user profile

APPENDIX F

Persona Scenarios

Mark Dean (Wanting to book at the restaurant for the first time) –

Context: Mark wants to book a slot at the restaurant for the first time. He learnt from social media that the restaurant is now taking online bookings so he checks the website.

Claire Smith (Wanting to see her loyalty points after booking numerous times) –

Context: Claire has already used the web application numerous times to book a slot at the restaurant. She now wants to see how many loyalty points she has gained and what rewards she can spend them on.

Reservation System for a Restaurant

This survey is to gather customer expectations and requirements for an online booking system for a restaurant.

* Required

* This form will record your name, please fill your name.

User Expectations of a Booking System

I am currently working on developing a web application that will allow the user to reserve a table at a pizza restaurant.

I want to hear your opinions on what makes a good online reservation system so I can further improve its content and functionalities and meet the user's requirements. Please fill out this quick survey and let us know your thoughts (your answers will be anonymous).

Thank you for your time.

Principal Investigator: Scott Mains

Student ID No: w18003567

1. If you would like to take part in this study, please read the statement below and tick 'I agree'

I understand the nature of the study, and what is required from me. I understand that after I participate I will receive a debrief providing me with information about the study and contact details for the researcher. I understand I am free to withdraw from the study at any time, without having to give a reason for withdrawing, and without prejudice. I agree to provide information to the investigator and understand that my contribution will remain confidential. I also consent to the retention of this data under the condition that any subsequent use also be restricted to research projects that have gained ethical approval from Northumbria University. I agree to the University of Northumbria at Newcastle recording and processing this information about me. I understand that this information will be used only for the purpose(s) set out in the information sheet supplied to me, and my consent is conditional upon the University complying with its duties and obligations under the Data Protection Act 2018 which incorporates General Data Protection Regulations (GDPR). You can find out more about how we use your information at Privacy Notices *

- I agree

2. How old are you?

- 18 - 24
- 25 - 31
- 32 - 39
- 40+

3. Have you ever used an online reservation system to book a table at a restaurant?

- Yes
- No

4. Would you rather use an online reservation system or book via other methods?

Online Reservation System

Phone

Social Media

Other

5. Would you be more inclined to use an online booking system if it involved a loyalty point system for future discount offers?

Yes

No

6. Would you book using your mobile phone?

Yes

No

7. Do you have any suggestions for functionality in the booking form?

8. Are there any other features you think would make you want to book again? Other than the loyalty system.

This content is neither created nor endorsed by Microsoft. The data you submit will be sent to the form owner.



APPENDIX H

Client Interview Questions

1. For background information, what is your role in the restaurant?
2. How long have you been a restaurant owner?
3. Have you ever used a reservation system for your restaurant before?
4. If yes, what parts of the system did you find useful?
5. Are there any parts that you didn't find useful or necessary?
6. If a reservation system were to be made from scratch, what features would you consider necessary?
7. Would you want the bookings to be on a table-by-table basis? Or set up considering the max capacity of the venue, allowing you to readjust seating to suit the party size?
8. Are there any features that you think could help build customer relationships that I could incorporate into the application?
9. Are there any other features you would like to see on the “admin dashboard”?

APPENDIX I

As a user, I would like to be able to create an account on the web application as seamlessly as possible.



As a user, I would like to be able to log into the system

As a user, I would like to be able to edit my user details, so I can update my information for new bookings

As a user, I would like to be able to create a new booking

As a user, I would like to be able to see what time slots are full at a restaurant to avoid overbooking.

As a user, I would like my new booking to be confirmed via email so I can use it as a reminder.

As a user, I would like to be able to retrieve a forgotten password directly from the web application.

As a user, I would like to be able to have my credentials remembered upon login, so that I dont have to log in everytime.

As a user, I would like to be able to have my credentials remembered upon login, so that I dont have to log in everytime.

As a user, I would like to be able to see my loyalty points, so that I know whether I have enough for a reward.

As a user, I would like to be able to see the menu of the restaurant before booking, so that i know whether I want to eat there.

As an admin, I would like to be able to delete a customers booking, in case they cannot attend anymore/overbooked.

As an admin, I would like a chronological view of every booking made at the restaurant.

As an admin, I would like to be able to see every customer that has signed up to the web application, so I can add personal comments.

As an admin, I would like to be able to send promotional emails to customers directly from the website.

As an admin, I would like to be able to set the max occupancy of the restaurant to prevent over congestion.

As an admin, I would like to be able to change the opening hours of the restaurant in case we change opening hours or it's a holiday.

As an admin, I would like to be able to change booking intervals between time slots, in case the time slots are not long enough/too short

APPENDIX J

Requirement number	Requirement	Description	Priority
1	Sign Up page	A page where the user can create an account for the application	High
2	Login page	A page where a user can log in with the credentials they created on the sign up page	High
3	Booking page	A page where the user can set their booking information and confirm it for the restaurant.	High
4	User dashboard	A page where the user can view their existing bookings and personal information and current loyalty points.	High
5	User delete booking	The ability for the user to delete the booking they have made.	High
6	User edit information	The ability for the user to change their personal information for future bookings.	High
7	Confirmation email	Confirmation emails are to be sent to users after each booking is made.	Medium
8	Forget password retrieval	The ability for the user to change their password directly from the web application	Medium
9	Menu page	A page to display the menu for the restaurant.	Medium
10	Remember login details	A checkbox for the user to tick if they want their credentials stored for future log in attempts	Low
11	Select booking date via calendar	A calendar plugin is integrated which allows the user to select the day they want to reserve a table.	High
12	Select timeslot autogenerated from database	Time slots are generated from the database using time intervals and opening hours. The times are not hardcoded.	
13	Usage of REST API	All data requests are made via a REST API.	High
14	Contact page	A page for the user to directly message the restaurant.	Low
15	Admin Dashboard page	An admin dashboard page which is not accessible to users - admin privileges required.	High
16	View all existing bookings	All existing bookings made by customers can be viewed on this page	High
17	Delete bookings	Bookings can be deleted directly from the admin dashboard booking page.	High
18	View all customers	All customers that have signed up to the website can be viewed on the admin customer page.	High
19	Add comment to user	The admin can add a comment attached to the user with whatever information they please.	Medium
20	Update Loyalty points	The admin can directly update the loyalty points associated with a customer.	Medium
21	Send promotional emails to customers	The admin can send promotional emails to customers directly from the dashboard.	Medium
22	Tweak booking slot intervals	The admin can change the time interval between booking slots.	Medium

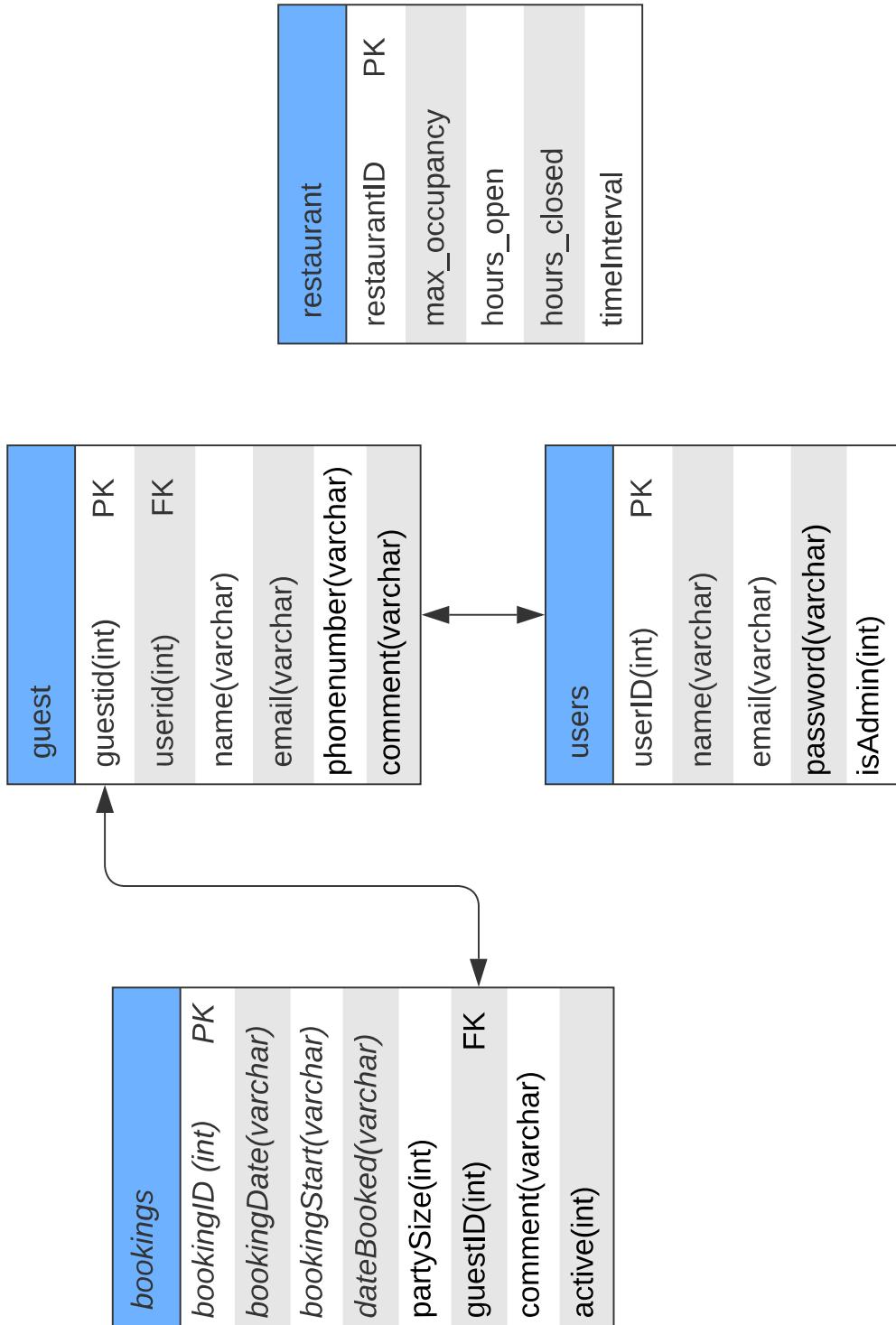
23	Set max occupancy of restaurant	The admin can set the max occupancy of the restaurant so that customers cannot overbook.	High
24	Set opening and closing hours of restaurant	The admin can set opening and closing hours of the restaurant so the customer can only book between these hours.	High
25	Mobile Responsive	The web application responds well on smaller devices on all pages.	High

APPENDIX K

Requirement Number	Rerequirement	Description	Priority
1	Security	Best security practices will be implemented to secure all information inputted by the user.	High
2	Scalability	Code will be split up into class components so that they are reusable and scalable.	Medium
3	Usability	The product will be very easy to use and the overall experience for the user will be pleasant	High
4	Maintability	The code will have high maintainability and error and exception handlers will be implemented to help debug code.	High

Restaurant Table Reservation System ER Diagram

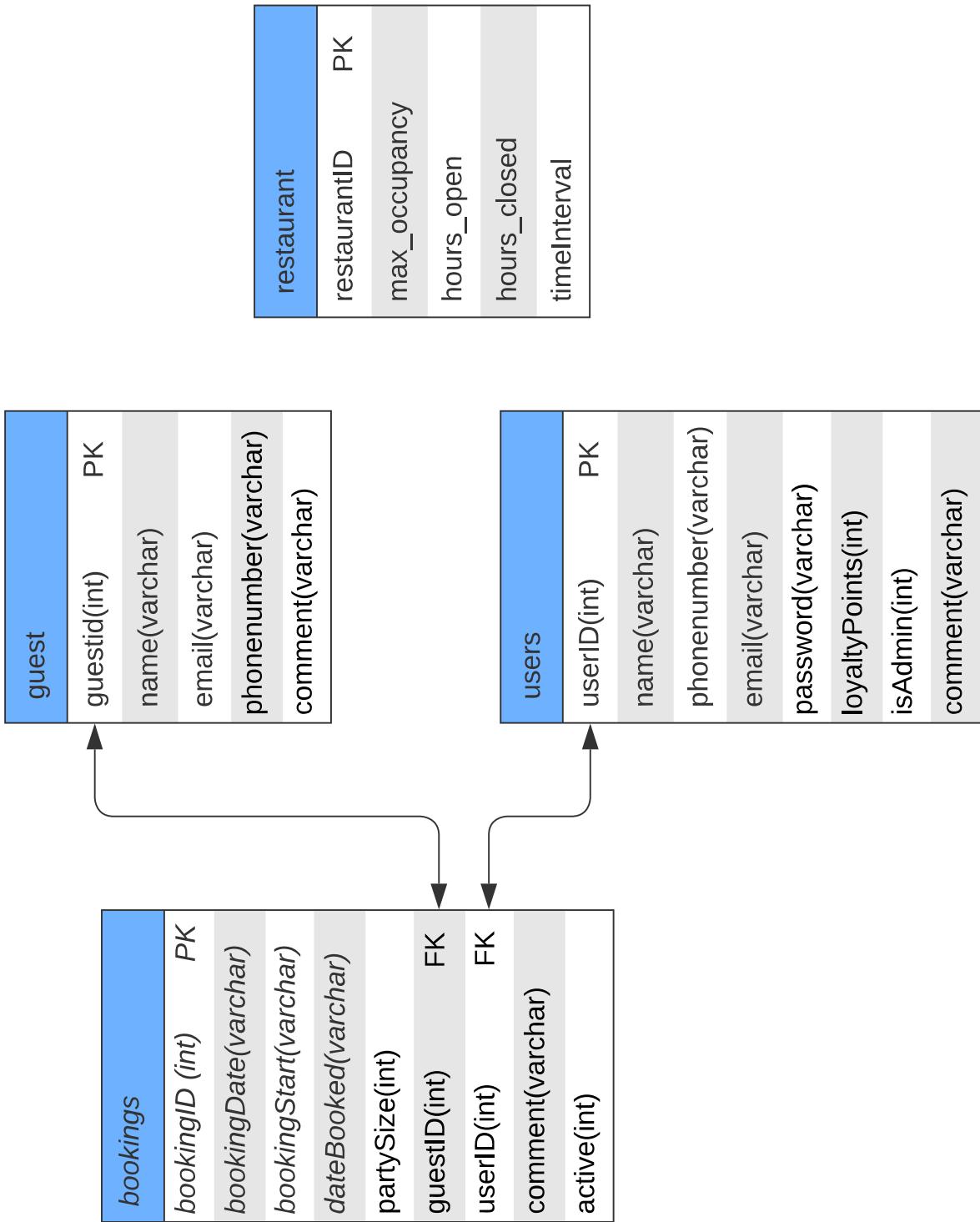
Scott | May 3, 2022



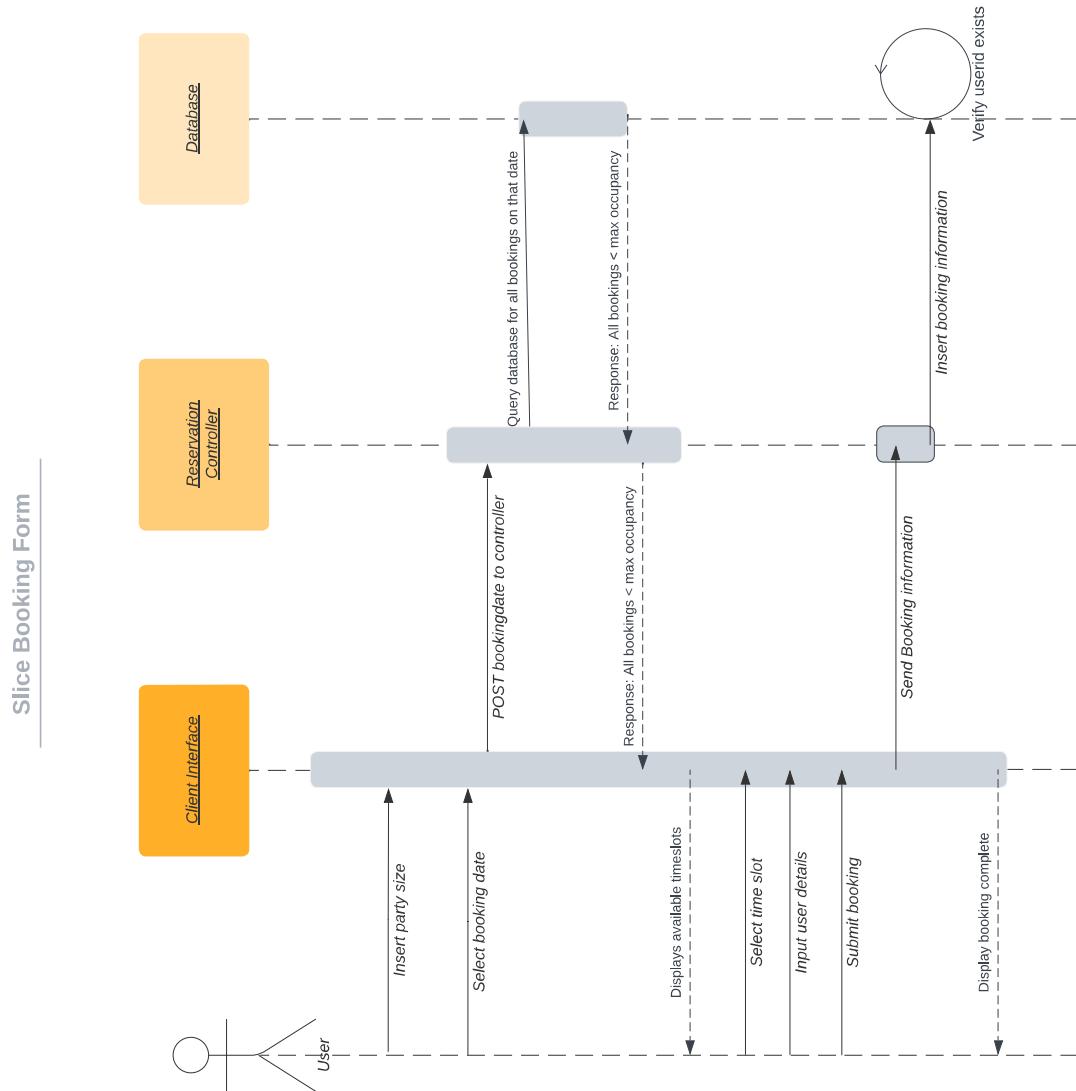
Restaurant Table Reservation System ER Diagram

Scott | April 30, 2022

APPENDIX M



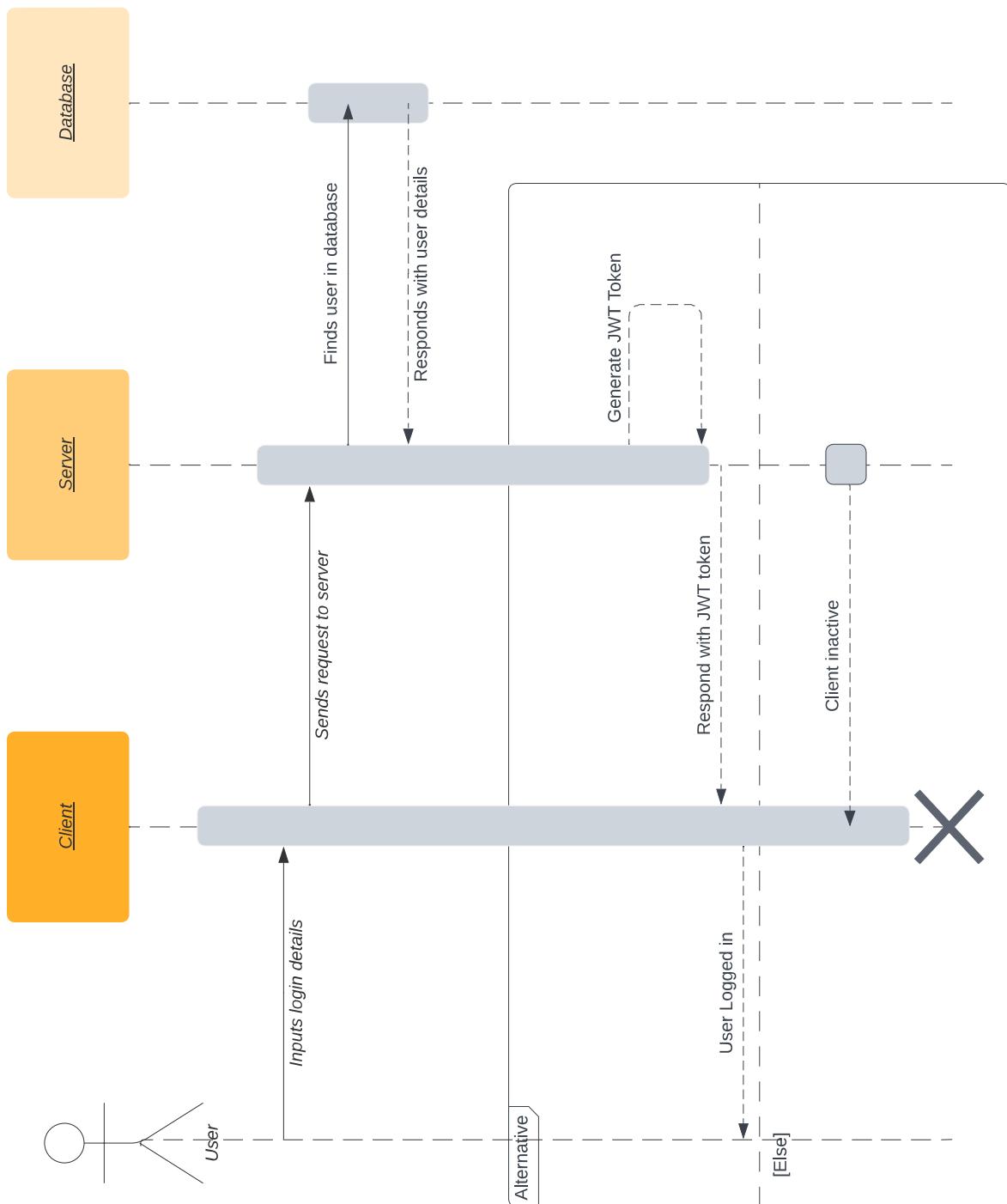
APPENDIX N



APPENDIX O

Slice Login Form

Scott | May 3, 2022



APPENDIX P

Task	Description	Expected Outcome	Actual Outcome	Resolution
handleSubmitLogin()	Login details are appended as form data and posted to the authenticate controller.	If login details are in the database, a success message appears. If not, then an appropriate error message will appear.	Expected outcome occurred	N/A
checkRestaurant()	This is called on booking form load which populates the timeslot buttons.	All time slots will be populated using the opening hours and time interval.	Results were not getting added to variable	Needed to add a [0] to the results to get the first object.
onDayPress()	This appends the max occupancy and booking date value as form data then posts it to the checktimeslots api to find out overbooked timeslots	Any slots over max occupancy will be removed from the time array.	Date value was being added with quotation marks which wouldn't work in the database.	"Moments" package was used to format the date.
handleSubmitBooking()	All data relevant to booking (partysize, bookingstart, bookingdate, phonenumber, email, name) are appended to formdata and sent to the reservation endpoint.	Booking will be added to database and a success screen will appear.	Expected outcome occurred	N/A
handleLogoutClick()	the JWT from local storage is removed from the browser.	The page refreshes and the user is no longer authenticated.	Expected outcome occurred	N/A
handleSubmitSignUp()	All signup details are tested against the REGEX and then appended to form data. It is then posted to the database.	User details are added to the database and available to log in with. A success page replaces the sign up page.	Expected outcome occurred	N/A
submitForgotPassword()	The email is appended to formdata and posted to forgot password endpoint. If it exists in the database then an email is sent to the user with a forgot password form.	An email is sent to the user. A unique login url is attached to the email which lets them update password. The password change is reflected in the database.	Password not updating properly in the database.	Couldn't solve this issue in time.

UserRemoveBooking()	User clicks remove booking on the user dashboard and the booking is deleted.	The booking is removed from the database by posting the user ID and the bookingID	Expected outcome occurred	N/A
getBookings()	The admin visits the booking page and all bookings are visible	All bookings currently in the database are displayed.	Expected outcome occurred	N/A
getCustomers()	The admin visits the customers page and all customers are visible	All customers currently in the database are displayed.	Expected outcome occurred	N/A
filterSearch()	The admin types into the search box and the database results are filtered in what was typed.	All relevant database entries are displayed.	Expected outcome occurred	N/A
removeBooking()	Admin clicks remove booking on the admin dashboard and the booking is deleted.	The booking is removed from the database by posting the specific ID and JWT with admin credentials.	Expected outcome occurred	N/A
updateComment()	Admin clicks update comment next to a customer. Modal appears which allows the admin to input a new comment in string format	The comment is updated and the page refreshes reflecting the update.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type
updateLoyalty()	Admin clicks update comment next to a customer. Modal appears which allows the admin to input a new comment in integer format	The loyalty is updated and the page refreshes reflecting the update.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type
sendEmail()	Admin types a message in the text area at the top of the page and click submit.	An email is sent the customer with the attached message.	Email wasn't sending.	Needed to add my google email to SMTP protocol and then accept it in my google mail account
handleEmail()	The admin presses a checkbox next to a customers email.	The email gets added to the text area at the top of the page.	Only worked for 2 emails and then the required comma wouldn't appear.	Need to add an if statement on submission checking the variable type
updateClosingTime()	Admin clicks update opening time. Modal appears which allows the admin to input a new time in time format	The opening time is updated in the database and the page refreshes to reflect this.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type
updateClosingTime()	Admin clicks update closing time. Modal appears which allows the admin to input a new time in time format	The closing time is updated in the database and the page refreshes to reflect this.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type

updateMaxOccupancy()	Admin clicks update max occupancy. Modal appears which allows the admin to input a new max occupancy in integer format.	The max occupancy is updated in the database and the page refreshes to reflect this.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type
changeInterval()	Admin selects a new time interval from the drop down menu and clicks update.	The interval is updated in the database and the page refreshes to reflect this.	Any type of value was getting submitted	Need to add an if statement on submission checking the variable type

APPENDIX Q

Task	Description	Expected Outcome	Actual Outcome
User makes a booking using the guest booking form	User goes through the booking form without being authenticated.	Booking will be created and sent to the database. A success page showing their booking details will display.	
User Sign Up with incorrect credentials	User enters sign up credentials without following the validation procedures	The form will alert the user to which part is incorrect and they will not be able to sign up	
User Sign Up with correct credentials	User enters sign up credentials while following the validation procedures	User successfully signs up and will be redirected to the home page showing that they are authenticated.	
User Sign In with incorrect credentials	User enters incorrect login details that do not exist in the database.	If credentials are incorrect then an unauthorised error message will appear.	
User Sign In with incorrect credentials	User enters existing login details present in the database	User successfully signs in and will be redirected to the home page showing that they are authenticated.	
Forgotten Password	User clicks forgot password on the login form which redirects them to a page to enter their email. They will then be sent an email which directs them to a form to reset their password	User follows the forgot password procedure and successfully changes their password.	
User makes a booking using booking form	User goes through the booking form while being authenticated.	Booking will be created and sent to the database. A success page showing their booking details will display.	
User makes a booking while they already have one existing in the system	User goes through the booking form while being authenticated after already making a booking previously	An error message will appear saying that a booking already exists in the system.	
User deletes booking in user dashboard	User deletes the booking they have just made from the the user dashboard.	The booking will be deleted from the database and will no longer show in their dashboard.	
User changes personal details in user dashboard	User changes their personal details in the user dashboard	User details will successfully change within the database and show the new information on the dashboard.	
User signs out	User clicks sign out button in navbar.	User will no longer be authenticated and will not have access to user dashboard.	

APPENDIX R

Task	Description	Expected Outcome	Actual Outcome
Admin sign in	The admin signs in to the system	Admin is logged in and navigated to the home page. They will have access to the Admin dashboard via navigation bar	
Admin delete booking	Admin clicks delete booking on a booking slot in the admin page.	The booking is deleted from the database and the page refreshes showing that the booking has gone.	
Admin Update Customer Comment Correct	Admin updates the comment attached to the customer on the customer page.	Admin clicks update, a modal shows up and the user types in anything they want. The page will refresh reflecting the update	
Admin Update Customer Loyalty Points correct	Admin updates the Loyalty attached to the customer on the customer page	Admin clicks update, a modal shows up and the user types in integer value. The page will refresh reflecting the update	
Admin Update Customer Loyalty Points incorrect	Admin updates the Loyalty attached to the customer on the customer page but uses a value other than integer	Admin clicks update, a modal shows up and the user types in string value. The update will not occur and a box will remind them to use integers	
Admin add email to email registry	Admin selects a checkbox located next to a customers name.	The customers email address gets added to the text field area at the top of the page.	
Admin send email to customers	Admin types a message in the text area underneath the email registry. Clicks submit	The message typed in the text area gets sent to the customers email.	
Admin Booking intervals	Admin chooses a booking interval in the settings page and clicks update.	The page refreshes and reflects the updated booking interval	
Admin set max occupancy correct	Admin updates the max occupancy on the settings page and clicks update.	Admin types an integer value and the page refreshes updating the change	
Admin set max occupancy incorrect	Admin updates the max occupancy on the settings page and clicks update.	Admin types a string value and the modal notifies the admin to use an integer	
Admin set Opening closing hours correct	Admin updates the opening and closing hours on the settings page and clicks update.	Admin types a time format and the page refreshes reflecting the update	
Admin set Opening closing hours incorrect	Admin updates the opening and closing hours on the settings page and clicks update.	Admin types a string value and the modal notifies the admin to use an time.	

APPENDIX S

Use Case Descriptions

ID:	1
Title:	Register an Account
Description:	The user registers an account on the web application
Primary Actor:	Potential Customer
Preconditions:	None
Postconditions:	The credentials of the user are added to the database.
Main Success Scenario:	<ol style="list-style-type: none">1) The user clicks the sign up button located in the navigation bar.2) The application displays the sign up page, which reveals input fields to enter their name, email, phone number and password.3) The user types in their details.4) The system displays a success message and the user is navigated to the home page.
Extensions:	If the user enters invalid fields: <ol style="list-style-type: none">1) Application finds one or more invalid inputs.2) The system displays an error message indicating which field is incorrect.3) The user can re-enter their details until it is correct.
Frequency of Use:	Once per customer.
Status:	N/A
Owner:	N/A
Priority:	HIGH

ID:	2
Title:	Login Process
Description:	User tries to sign into the web application
Primary Actor:	User
Preconditions:	User has an existing account in the database
Postconditions:	The user is authenticated and can view the user dashboard.
Main Success Scenario:	<ul style="list-style-type: none"> 1) User clicks the login button located in the navigation bar. 2) The login page is displayed which reveals input fields to enter their email and password. 3) The user enters their details and clicks sign in. 4) The system checks the database for their credentials. 5) The system then authenticates the user and navigates them to the home page.
Extensions:	<p>If the user enters invalid details:</p> <ul style="list-style-type: none"> 1) The system does not find any relevant details in the database. 2) The application displays an error message telling them they are unauthorized. 3) The user can re-enter correct details.
Frequency of Use:	Often. Every time they use the application or when their JSON token expires.
Status:	N/A
Owner:	N/A
Priority:	N/A

ID:	3
Title:	New Booking Guest
Description:	The user successfully reserves a table at the restaurant
Primary Actor:	Customer (unauthenticated)
Preconditions:	The user is not logged into the system
Postconditions:	Reservation date, time and guest details are added to the database.
Main Success Scenario:	<ul style="list-style-type: none"> 1) The guest clicks the “Book a table” button on the home page 2) The system redirects them to a guest booking form. 3) Guest chooses the date they want and the time slots available. 4) The system presents them with input fields to enter their personal information. 5) The customer presses submit, and the system shows them a success message.
Extensions:	<p>User fails to enter credentials:</p> <ul style="list-style-type: none"> 1) User submits the booking form without typing in credentials. 2) The system presents them with an error message to fill in the fields.
Frequency of Use:	Frequently
Status:	N/A
Owner:	N/A
Priority:	HIGH

ID:	4
Title:	New Booking User (Authenticated)
Description:	The user successfully reserves a table at the restaurant
Primary Actor:	User
Preconditions:	User must be logged in and authenticated.
Postconditions:	Reservation date, time and user details are attached to the booking.
Main Success Scenario:	<ul style="list-style-type: none"> 1) The user clicks the “Book a table” button on the home page 2) The system redirects them to a user booking form. 3) User chooses the date they want and the time slots available. 4) The booking form populates the confirmation page with the user details from the database. 5) The customer presses submit, and the system shows them a success message.
Extensions:	<p>User already has a booking that exists in the system:</p> <ul style="list-style-type: none"> 1) User presses submit, and an error message is displayed saying they already have a booking.
Frequency of Use:	Fairly frequently
Status:	N/A
Owner:	N/A
Priority:	HIGH

ID:	5
Title:	Delete personal booking
Description:	The user has an existing booking that can be deleted in the user dashboard.
Primary Actor:	User
Preconditions:	User must be logged in and have an existing booking in the database.
Postconditions:	The booking attached to the user is deleted from the database.
Main Success Scenario:	<ol style="list-style-type: none"> 1) User has previously made a booking with the user booking form. 2) User navigates to their user dashboard. 3) System populates the user dashboard with their existing booking slot in the database. 4) The user clicks “delete booking” and the booking slot is deleted from the database.
Extensions:	N/A
Frequency of Use:	Fairly Frequently
Status:	N/A
Owner:	N/A
Priority:	HIGH

ID:	6
Title:	User update details
Description:	User updates the details in their user profile
Primary Actor:	User
Preconditions:	User must be logged into the system.
Postconditions:	The personal information that was changed in the dashboard has been updated in the database.
Main Success Scenario:	<ol style="list-style-type: none"> 1) User navigates to the user dashboard using the navigation bar. 2) User clicks "edit" next to the personal information detail they wish to update. 3) The system provides the user with an input field to change their information. 4) The user clicks update. 5) The system refreshes and the users details have been updated.
Extensions:	N/A
Frequency of Use:	Fairly frequently
Status:	N/A
Owner:	N/A
Priority:	Medium

ID:	7
Title:	Admin - Delete Customer Booking
Description:	The admin deletes customers booking from the admin dashboard.
Primary Actor:	Admin
Preconditions:	User must be logged in and authenticated as an admin role to access the admin dashboard.
Postconditions:	The booking slot is deleted from the database.
Main Success Scenario:	<ol style="list-style-type: none"> 1) Admin navigates to the admin dashboard. 2) The system populates the booking page with existing bookings within the system. 3) Admin locates the booking slot they wish to delete from the system. 4) Admin clicks “Delete Booking”. 5) The system presents them with a modal message to verify they want to do it. 6) The admin clicks “delete” and the booking is deleted from the system.
Extensions:	N/A
Frequency of Use:	Fairly Frequently
Status:	N/A
Owner:	N/A
Priority:	High

ID:	8
Title:	Admin – Update Customer Details
Description:	Admin updates customer details within the admin dashboard.
Primary Actor:	Admin
Preconditions:	User must be logged in and authenticated as an admin role to access the admin dashboard.
Postconditions:	The customers loyalty points/and or comment are updated in the database.
Main Success Scenario:	<ol style="list-style-type: none"> 1) Admin navigates to the customer page in the admin dashboard. 2) The database populates the customer page with details of existing customers. 3) The admin chooses whether they want to update their loyalty points or comment. 4) The system shows an input field which allows the admin to input new information. 5) The admin clicks update and the new information is updated in the database.
Extensions:	N/A
Frequency of Use:	Fairly frequently
Status:	N/A
Owner:	[Who owns this use case, in your project team?]
Priority:	N/A

ID:	9
Title:	Sign out
Description:	The user attempts to log out of the application
Primary Actor:	User, Admin
Preconditions:	User is logged in and authenticated.
Postconditions:	The JSON web token is destroyed and the user is no longer authenticated.
Main Success Scenario:	<ul style="list-style-type: none"> 1) The user clicks the “Sign Out” button in the navigation bar. 2) The system navigates the user to the home page.
Extensions:	N/A
Frequency of Use:	Fairly frequently.
Status:	N/A
Owner:	N/A
Priority:	High

ID:	10
Title:	Admin – send marketing email
Description:	The admin sends a promotional email directly from the marketing page.
Primary Actor:	Admin
Preconditions:	User must be authenticated as an admin
Postconditions:	Promotional email is sent to the customer(s)
Main Success Scenario:	<ol style="list-style-type: none"> 1) Admin navigates to the marketing page on the admin dashboard. 2) System populates the marketing page with existing customers emails in the database. 3) Admin ticks the checkbox next to the customer(s) they want to send an email to. 4) Text input field at the top of the page is populated with the emails of the checkboxes that have been ticked. 5) Admin types into the text field below with the promotional message they want to send. 6) Admin clicks “send email” and the message is sent to the customers.
Extensions:	<p>Email protocol is down:</p> <ol style="list-style-type: none"> 1) Admin clicks “send email”. 2) The admin is alerted that the email cannot be sent.
Frequency of Use:	Fairly frequently.
Status:	N/A
Owner:	N/A
Priority:	High

ID:	11
Title:	Admin – Update Booking Interval
Description:	The admin updates the time interval in the settings page.
Primary Actor:	Admin
Preconditions:	User must be authenticated as an admin
Postconditions:	The Time slots on the booking forms update.
Main Success Scenario:	<ul style="list-style-type: none"> 1) Admin navigates to the settings page. 2) Settings page is populated with the current time interval from the database. 3) Admin selects new time interval from the drop down. 4) The page refreshes and the new time interval has updated in the database.
Extensions:	N/A
Frequency of Use:	Fairly frequently.
Status:	N/A
Owner:	N/A
Priority:	Medium

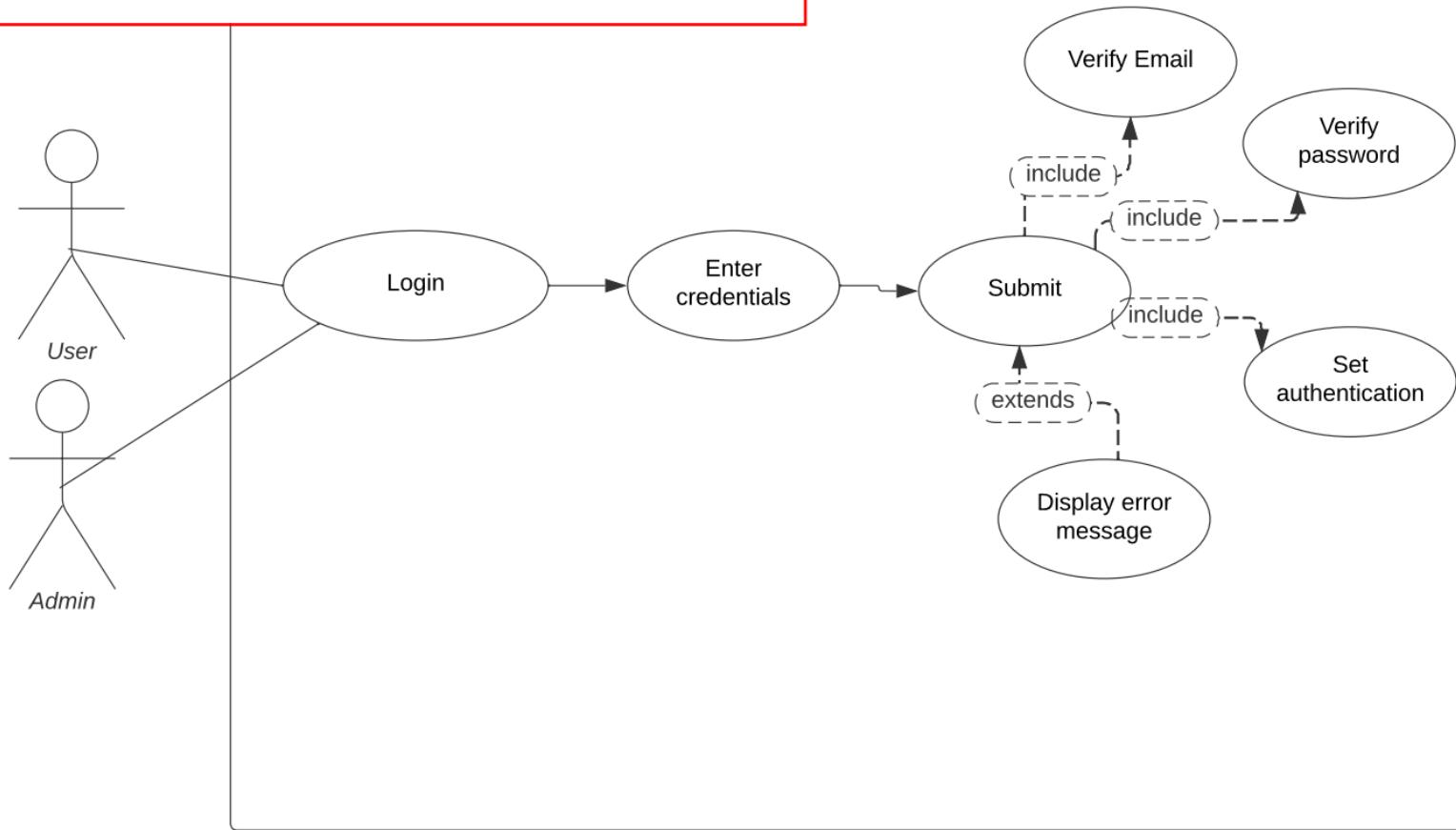
ID:	12
Title:	Admin – Update Max Occupancy
Description:	The admin updates the max occupancy on the settings page.
Primary Actor:	Admin
Preconditions:	User must be authenticated as an admin
Postconditions:	The Time slots on the booking forms update.
Main Success Scenario:	<ol style="list-style-type: none"> 1) Admin navigates to the settings page. 2) Settings page is populated with the current max occupancy from the database. 3) System presents the admin with an input field to type in new max occupancy. 4) The admin clicks update. 5) The page refreshes and the new time interval has updated in the database.
Extensions:	N/A
Frequency of Use:	Fairly frequently.
Status:	N/A
Owner:	N/A
Priority:	Medium

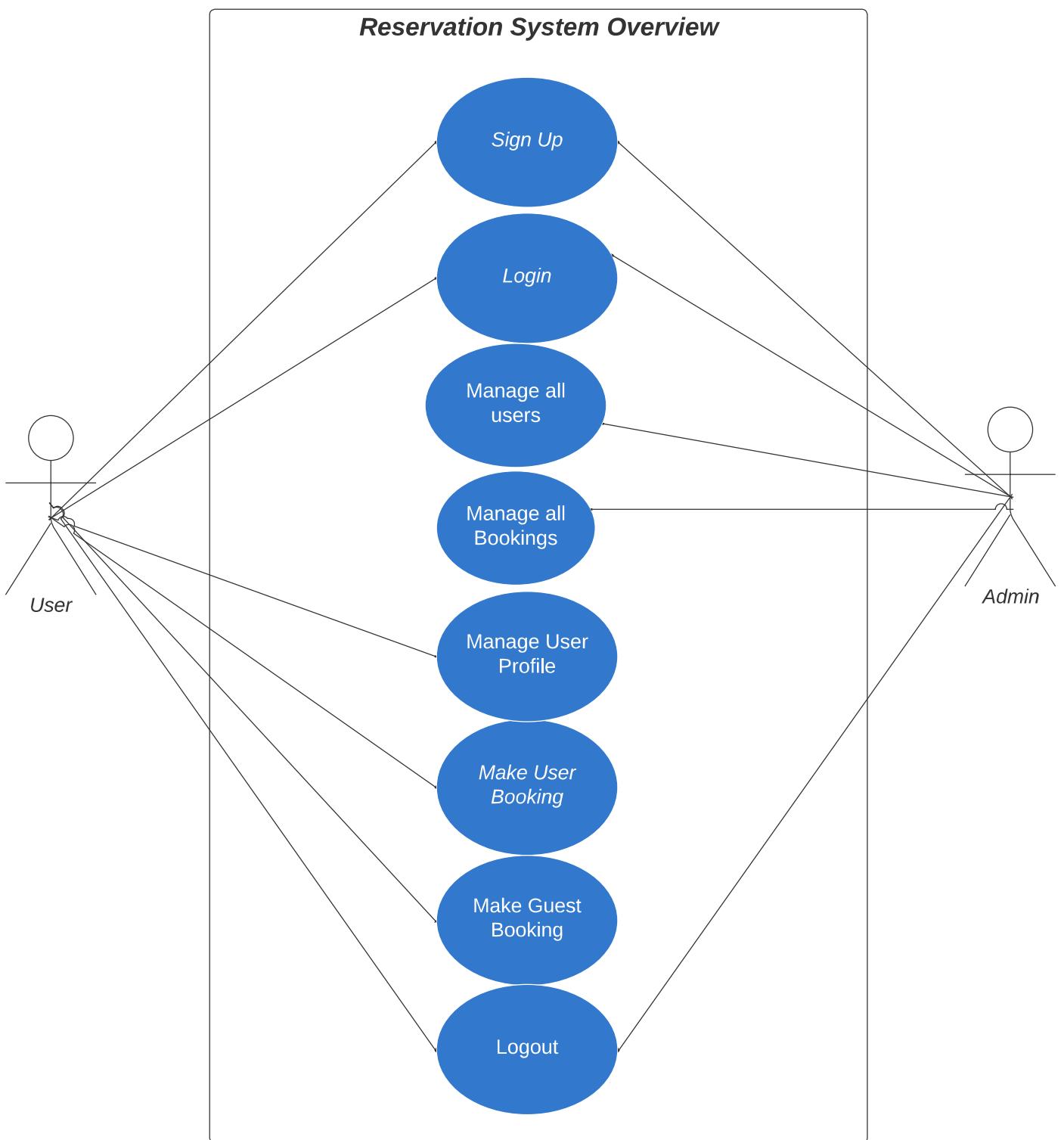
ID:	13
Title:	Admin – Update Max Occupancy
Description:	The admin updates the opening hours on the settings page.
Primary Actor:	Admin
Preconditions:	User must be authenticated as an admin
Postconditions:	The Time slots on the booking forms update.
Main Success Scenario:	<ol style="list-style-type: none"> 1) Admin navigates to the settings page. 2) Settings page is populated with the current opening hours from the database. 3) System presents the admin with an input field to type in new opening hours 4) The admin clicks update. 5) The page refreshes and the new time interval has updated in the database.
Extensions:	N/A
Frequency of Use:	Fairly frequently.
Status:	N/A
Owner:	N/A
Priority:	Medium

ID:	14
Title:	Forgotten password
Description:	User has forgotten password to enter the web application
Primary Actor:	User
Preconditions:	User must be logged out
Postconditions:	The users password will be updated in the database.
Main Success Scenario:	<ol style="list-style-type: none"> 1) User is logged out and clicks "Forgot password" on the login page. 2) System presents the user with an input field to type in their password. 3) Server sends a forgot password email link to the email that was entered. 4) User clicks link in the email which redirects them to a form to enter a new password. 5) User enters correct credentials. 6) System presents them with a success message.
Extensions:	<p>Email system is down:</p> <ol style="list-style-type: none"> 1) User enters email in the input field. 2) Error message presents them with a server error message. <p>User types in wrong password credentials:</p> <ol style="list-style-type: none"> 1) User doesn't enter valid password in the input fields. 2) System tells them to type in correct details.
Frequency of Use:	Not often.
Status:	N/A
Owner:	N/A
Priority:	Medium

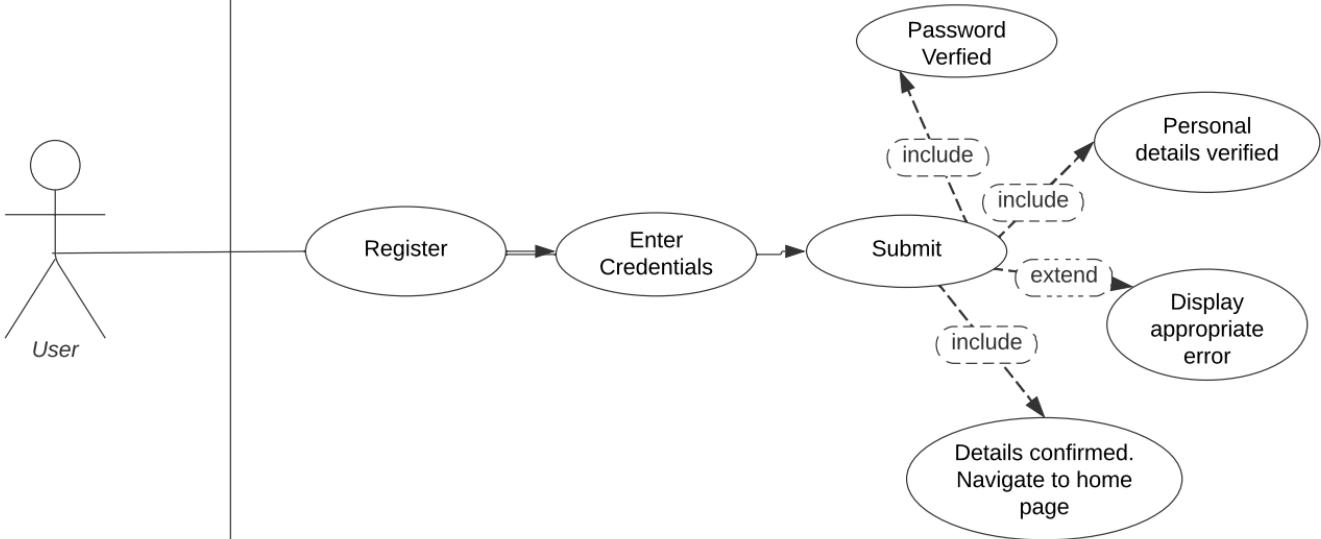
APPENDIX T

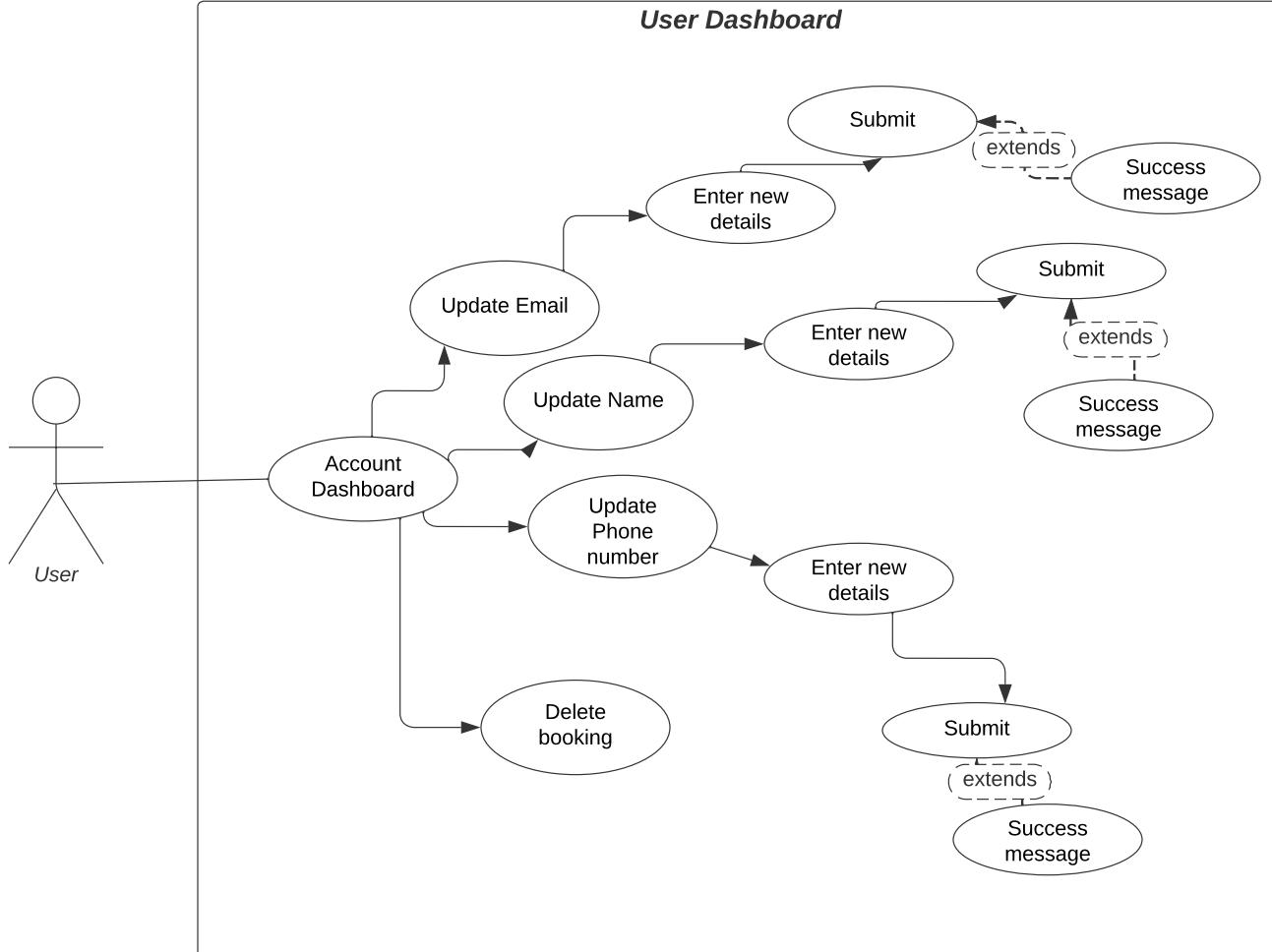
Login Process

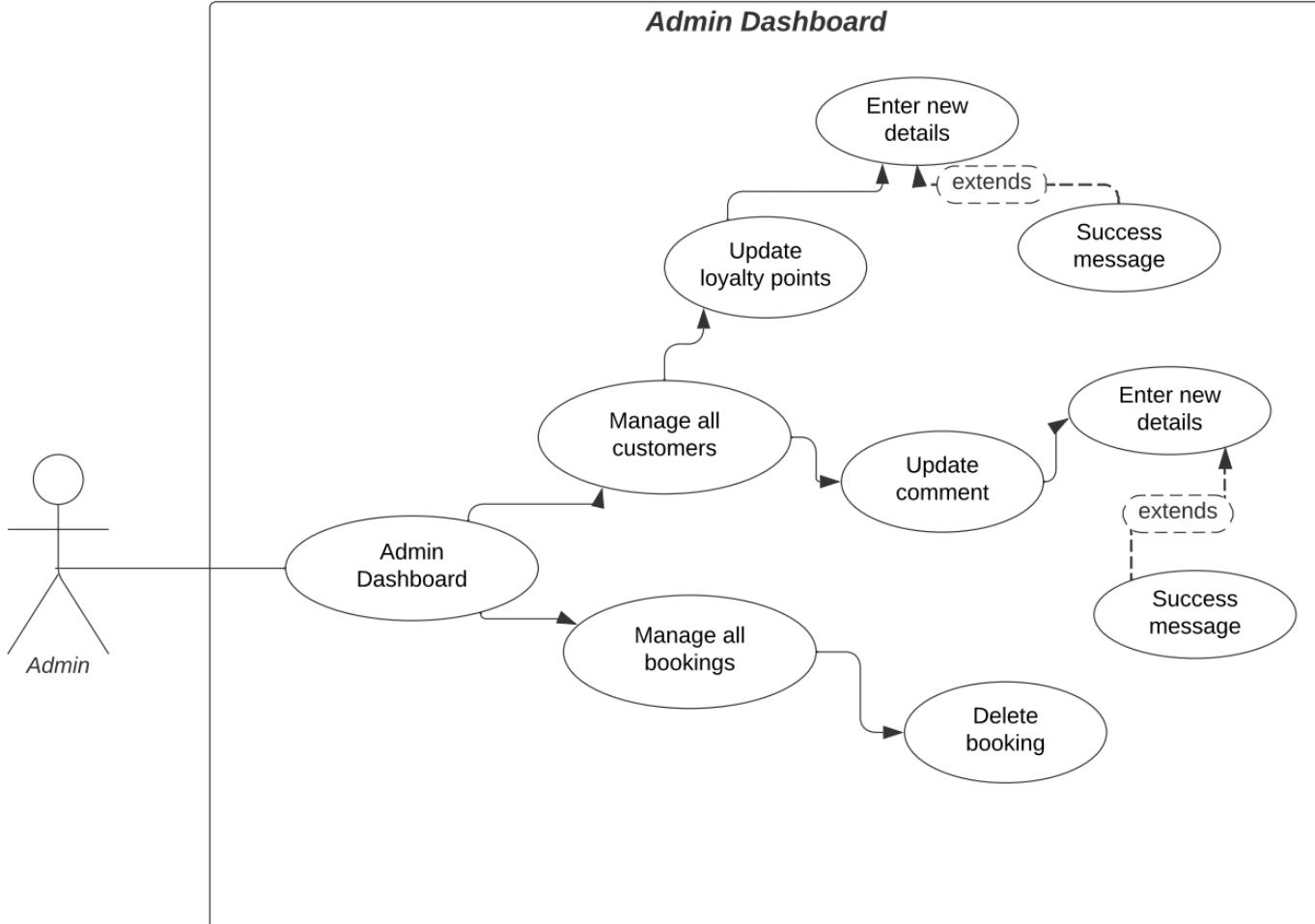


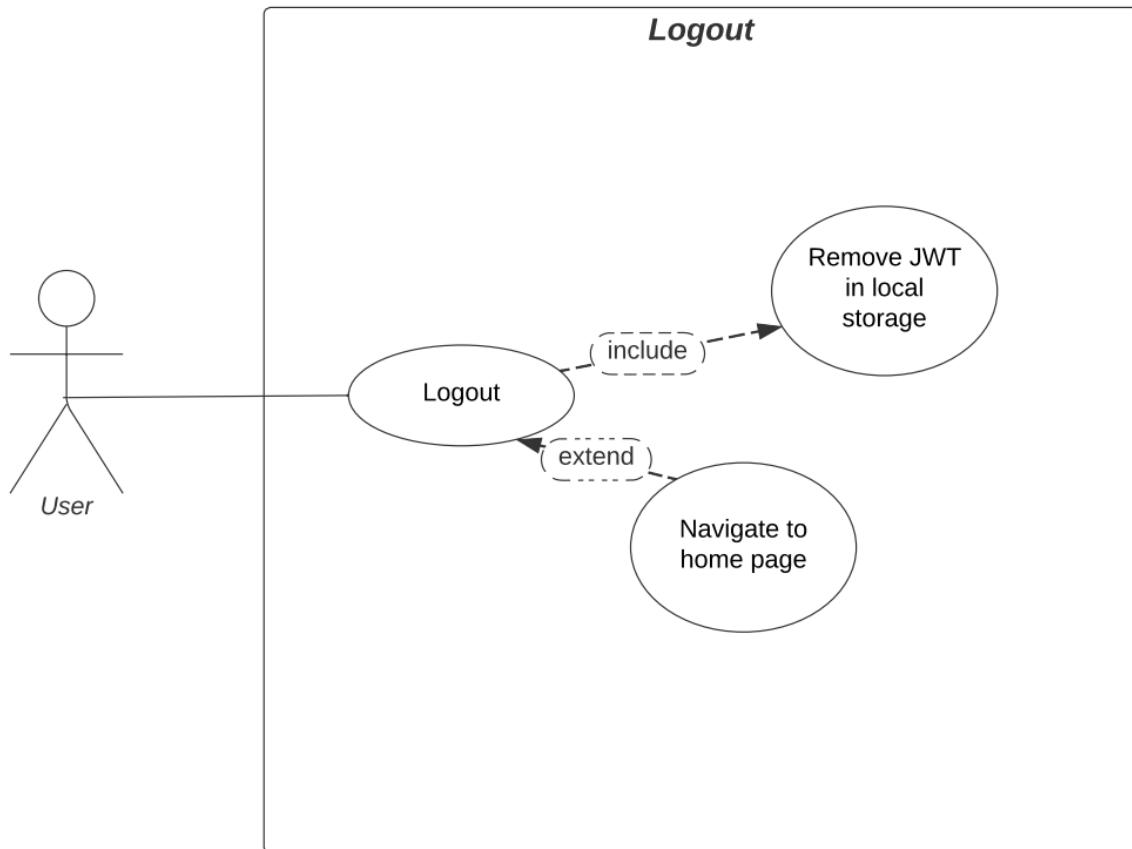


Sign Up process

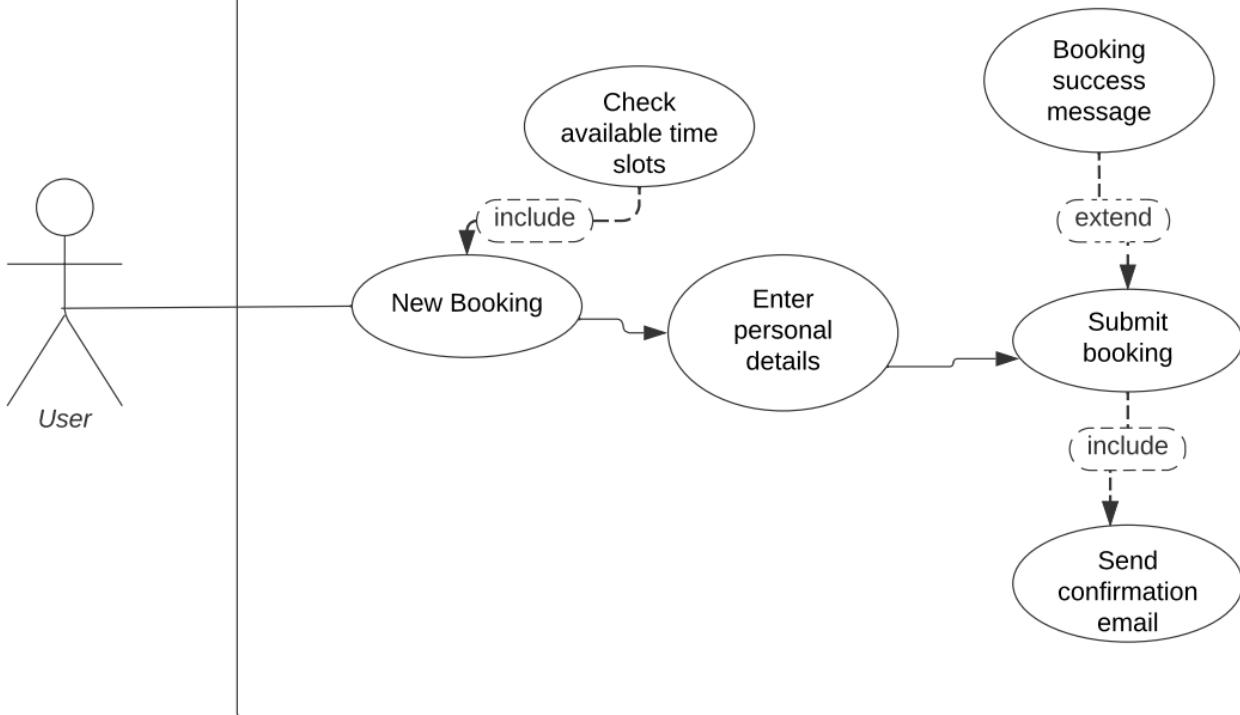








New Booking



APPENDIX U

SB

Slice Boro <scottmains4@gmail.com>

Wed 27/04/2022 21:10

To: You

Hello testing

Best wishes,

Slice Team

Email sent via [EmailJS.com](#)

Restaurant Logo

Home

Contact Us

Menu

Login

Sign Up

BACKGROUND IMAGE OF PIZZA

RESTAURANT NAME

DIRECTIONS

BOOK A TABLE

Restaurant Logo

Home

Contact Us

Menu



Your Account

Sign Out

RESTAURANT NAME

DIRECTIONS

BOOK A TABLE

BACKGROUND IMAGE OF PIZZA

Restaurant

Login

Sign Up

SIGN UP

Name:

Default input

Email:

Default input

Phone Number:

Default input

Password:

Default input

Confirm Password:

Default input

SIGN UP

CANCEL

Restaurant Logo

Home

Contact Us

Menu

Login

Sign Up

RESTAURA

DIRECTIONS

BOOK A TAB

LOGIN

Email:

Default input

Password:

Default input

Remember me

[Forgot password?](#)

LOGIN

CANCEL

[Home](#)[Contact Us](#)[Menu](#)[Login](#)[Sign Up](#)

Booking

Welcome! You are currently not logged in. Would you like to continue as a guest or log in and collect some loyalty points?

[Continue as Guest](#)

[Home](#)[Contact Us](#)[Menu](#)[Login](#)[Sign Up](#)

Party Size:

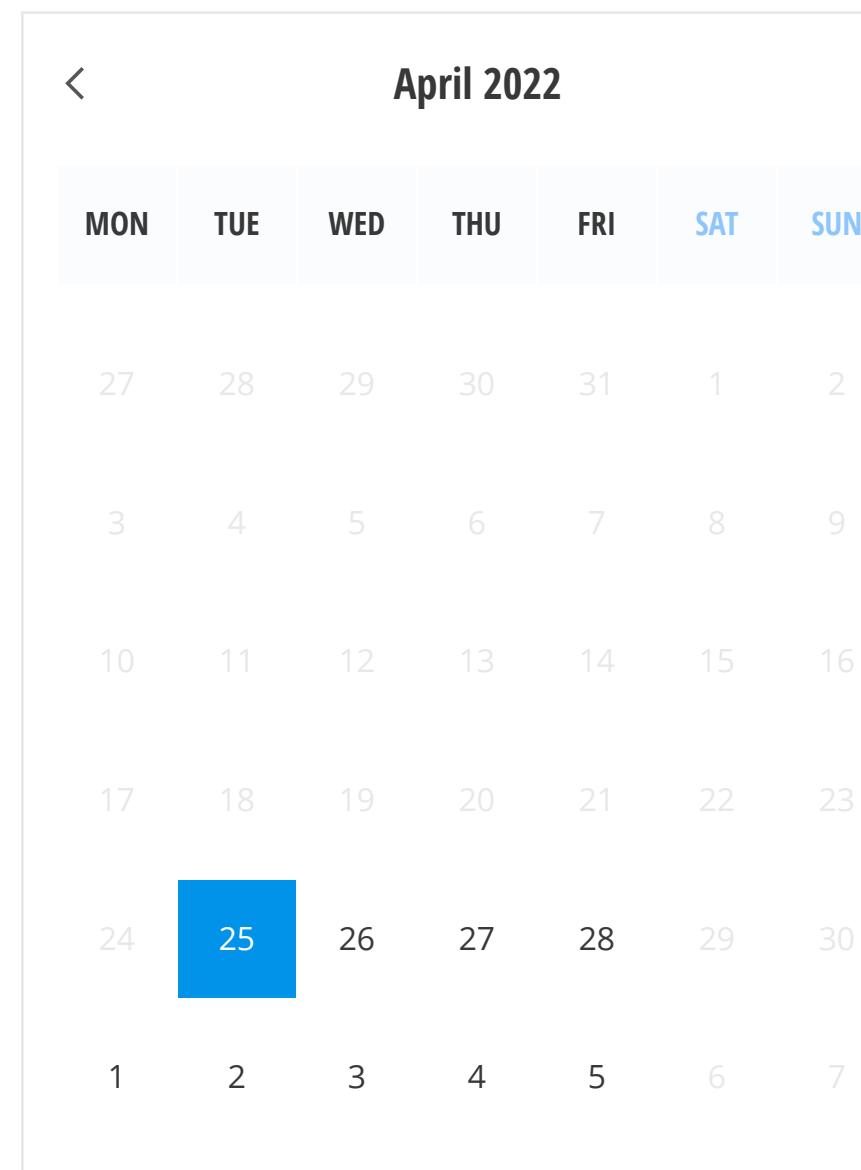
How many people will you be attending with?

1 ▾

[Continue](#)

Date

Please select the date you want to attend:



17:00 ▾

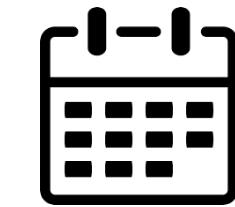
Back

Continue

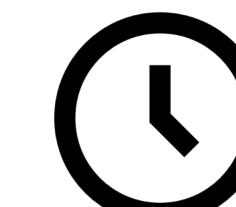
Booking Details:



5 GUESTS



24/04/2022



17:00

Name:

Email:

Phone Number:

Back

Complete Booking

[Home](#)[Contact Us](#)[Menu](#)[Login](#)[Sign Up](#)

SUCCESS!

You have booked a slot for 5 guests on 24/04/2022 at 17:00!

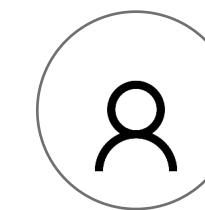
We have sent a confirmation email to example@gmail.com

Restaurant Logo

Home

Contact Us

Menu



USER DASHBOARD

USER DETAILS

BOOKING DETAILS

DELETE

REWARDS

Shows current loyalty points
with available rewards

BOOKINGS

Bookings

Search Bookings...

Booking details

DELETE

Customers

Marketing

Settings

Booking details

DELETE

Booking details

DELETE

Booking details

DELETE

CUSTOMERS

Bookings

Search Customers...

Customers

Customer details

Marketing

Customer details

Settings

Customer details

Customer details

MARKETING

Bookings

Customers

Marketing

Settings

EMAIL BOX

TEXT AREA TO WRITE EMAILS

SEND EMAIL

Customer email

Customer email

Customer email

SETTINGS

Bookings

Set Booking Interval

▼ 15 minutes

Update

Customers

Set Max Occupancy

50

Update

Marketing

Settings

Set Opening and Closing Hours

17:00

Update

APPENDIX W

Deadline(*)			Semester 1																Semester 2												Assessment Weeks	
Task	Detail	Time (Hrs)	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W1	W2				
Choose a supervisor	View available supervisors and email ones of interest	0.5																														
Confirm supervisor	Confirm supervisor and project idea and organise weekly meetings	0.5			*																											
RESEARCH AND PLANNING (124 hours)																																
Create Terms of Reference (TOR)	The terms of reference is a document that the project will be measured against.	10									*																					
Complete Ethics form	Online ethics form to be completed so that the project is inline with the university's standards and has ethical approval	3						*																								
TOR review meeting	Go through the TOR with supervisor and second marker	1								*																						
Revise TOR	If any suggestions come up in the TOR meeting, revise the TOR during this period	2																														
Create Literature Review	Create a literature review of all the reports that will be useful within my project so that I can get a better understanding of how I can create my system	40																														
Establish Client Requirements	Speak to client and note down all the things they require in the system.	5																														
Create initial storyboard and vision of product	Create initial storyboard using wireframes and present the vision of the product to the client	10																														
Persona(s) and Scenarios of target audience	Find target audience of the product and create persona(s) and scenarios.	10																														
User Requirement Document	Compile a user requirements document using all the above material and anything else that may be useful	10																														
Create draft chapters for the research and planning aspects	Start drafting up the chapters for the report in regards to the researching and planning sections	30																														
Submit chapters to supervisor for review	Send the chapters to supervisor and get his opinion on what needs to be changed	1													*																	
Discuss plans for Semester 2	Discuss in details on what needs to be done in the next semester to have a successful project	2													*																	
DESIGN AND BUILD (130 hours)																																
Create UML diagrams of the system		10																														
Create Low Fidelity Prototype		10																														
Create Entity Relationship Diagram and Data Dictionary		5																														
Create the database		5																														
Development of the server-side		20																														
Development of the client-side		60																														
Development of the CRM System		20																														
TESTING AND EVALUATION (40 hours)																																
Functional Testing		10																														
Usability Testing		10																														
Make changes based on testing processes		10																														
Evaluate Product		5																														
Evaluate the processes and own performance		5																														
WRITING THE REPORT (110 hours)																																
Abstract		3																														
Introduction		10																														
Research and Planning		25																														
Design, Implementation and Testing		25																														
Evaluation		25																														
Conclusions and Recommendations		10																														
Send for feedback		1																														
Make changes based on feedback		10																														
Submit report		1																										*				
VIVA (2 hours)																												*				
Organise Viva		1																										*				
VIVA Presentation		1																										*				