LAVISHED – NAILS BY LOLLY

A BUSINESS ANALYSIS
PROJECT

MOJALEFA THAPO

TABLE OF CONTENTS

Version	**Date**	**Description**
Inception: Iteration 1	22 April 2022	Provides an overview of the system proposed to *Lavished Nails by Lolly* nail salon, including its functional requirements, actors and goals, 3 detailed use case and diagrams.
Elaboration 1: Iteration 2	07 June 2022	Including the overview and all details of the system proposed to *Lavished Nails by Lolly* in the last iteration, Version 2 of the document expands on the system's core use cases in fully dressed format, with the inclusions of the Sequence Diagrams per realised use case, State Machine, Domain and Data Models, Business Process Modelling, updates to the Activity Diagram and further use case explanations. 'Generate Report' and 'Delete Appointment' (administrator's side) was removed as core use cases and replaced with 'Update Appointment' (administrator's side) and 'Delete Booking' (client's side) respectfully.
Elaboration 2: Iteration 3	09 September 2022	Including all the components and models in the last iteration, Version 3 of the document expands on all six of the system's core use cases in fully dressed format, with explanations provided. Sequence Diagrams per realised use case are included. All remaining maintenance use cases are done in brief format.

DOCUMENT OVERVIEW

This document contains all information pertaining to the use cases, diagrams, and models of the proposed system to be developed. The document will include a description of the system boundary, feature list, an overview of the actors and goals of the system, and shall continuously update on:

- The Use Case Model:
    - Six core use cases and maintenance use cases, with three expanded into fully dressed use cases and the remaining three planned for expansion in Iteration 3.
    - The Use Case Diagram.
- The Sequence Diagrams.
- The State Machine Diagram.
- The Domain and Data Models.
- The Business Process Modelling.

1. SYSTEM FEATURE LIST

- Client profiles and data thereof.
- Capturing of appointments and services rendered.
- Payment authorisation (credit, debit, EFT).
- System administration of clients and bookings pertaining to the Nail Salon.
- Communication between the business and client.

2. OVERVIEW OF ACTORS AND GOALS

Actors:

The primary actors using the system are identified as the clients, who shall be making appointment bookings and managing them, and the business administrator who shall manage the clients' appointments. The secondary actor is identified as the system.

Goals:

The proposed system will be a Progressive Web Application (PWA), specifically an automated online booking system helping to streamline business processes so that they are less time-consuming for the business and its clients. The goals of the system will include relieving the owner, who is currently the sole employee, from the burden of over-communication with clients and improve the business's financial and booking administration. The system will also be able to include push notifications and reminders so that both the salon employee and its client will be notified of a confirmed booking and sent a reminder of a set time before an appointment. Another goal of the system is to create user profiles, allowing the nail salon business to build a Business Intelligence database through built-in functionality. This allows the business to track its day-to-day activities, customer trends and overall business information.

Actor-Goal Table

Actor	**Goal**
Client	Update Client Details Create Booking View Booking Delete Booking Pay Deposit Receive Confirmation
Administrator	View Client Details Receive Confirmation View Appointment Update Appointment Delete Appointment View Report
System	Capture Client Details Capture Booking Send Confirmation Generate Report

3. SIX CORE USE CASES:

The following use cases were selected to show the user processes from different angles of the system, and shall be expanded on in use case narrative descriptions:

1. **Register on System** expands on the registration aspect of the system from the client's side, as the system's starting point.
2. **Create Booking** explores the main aspect of the appointment booking tracking system, creating a booking from the client side.
3. **Delete Booking** deals with the deletion of an appointment to show convenience from the client's side.
4. **Update Client Details** is intended to show an update to information in the system's database.
5. **Update Appointment** is intended to show an update to information regarding a client's appointment in the system's database, as well as interaction from the administrator's side.
6. **Complete Payment** shall show the system's interaction with a third-party payment service.

4. EXPLANATION OF USE CASE IMPLEMENTATION

Use Case 1: Register on System

As the starting point in the booking process, Register on System is implemented as it entails the beginning of every activity on the automated booking system. To create, update, delete or view a booking, a client would need to be logged in and exist on the system with their respective ClientID.

This is all conducted to ensure referential integrity within the database tables, and that there are as minimal errors as possible when using the system. With each registration, it is ensured a unique ClientID is generated to avoid unnecessary overlaps. Register on System is thus chosen as the first core use case to be implemented, as the ClientID generated in this use case is vital for the support, functionality and ease-of-use for the system.

Regarding the user's experience, 'Register' is one of the first and most prominent aspects of any system or website the user is exposed to. Hence, Register on System is given the same level of importance within the automated booking process, as this use case can also be where the user decides to forgo any further interaction should the process be unpleasant. This use case was implemented as it is an important aspect to the business's clients, as well as to the keeping of the system itself.

Use Case 2: Create Booking

Upon browsing, registering and logging in, a client shall wish to receive a service offered by the business. As this is an automated booking system, Create Booking is selected as the next use case to be implemented as it falls part of the system's core functionality and purpose, and is next in the user's sequence. To update, delete, view, receive services or receive revenue for services rendered at all, a booking must be created between the client and the business.

As the current process of manual recording is tedious and leaves much room for error, miscommunication, unnecessary time spent on liaison and even frustrations between the business and client, the Create Booking use case ensures the process is client-based and eliminates the need for stringent communication. User experience is another vital aspect taken into consideration for this use case as, should the process be confusing or unpleasant, prospective clients may decide to forgo further interaction with not only the system, but the business in its entirety.

This use case was implemented as, being a core and vital function of the system with many factors to consider, it is to be planned and developed in as much detail as possible with the sequence of how a client may navigate and experience the system in mind.

Use Case 3: Delete Booking

One of the most arduous tasks experienced by the business is handling booking cancellations. Clients have promised appointments, yet do not pay the required deposit and do not appear for their booking, wasting time spent on administration and further waiting of the technician to render the services the client had wished for. The Delete Booking use case is implemented with the aim to reduce frustrations, namely, by allowing clients to delete their own bookings and having these changes automatically update on the system's calendar.

Although the use case is designed with the business in mind, clients will also find this useful as it provides an efficient way to communicate their desires without direct contact with the business and can be done in the client's own time.

The Delete Booking use case also provides essential information needed for the Cancellations report with its feedback function. With this information, the business can deduce new strategies to be implemented to reduce cancellations in a specific area, and have a reading on their market and competitors should bookings be deleted due to clients having an appointment elsewhere. This use case was thus implemented as it is an important area to the beginning of the salon's Business Intelligence operative, while also addressing existing frustrations in the current business process.

Use Case 4: Update Client Details

Another arduous task of the nail salon’s current operation is manually recording and changing all client details. Details may get lost in the abundance of direct messages the business receives or the client could decide not to notify the salon of their new details at all due to a lack of a system, allowing for future miscommunication between business and client. It may pose privacy risks as well, as the client could be uncomfortable sharing their personal information while any person could access it, especially since social media accounts are prone to hacking.

Taking this into consideration, an online booking platform where clients may register and update their details at any time would be most ideal in giving clients a more reliable platform for their information and would allow a more enthusiastic providing of their correct and updated details, as they can directly do the change themselves. This enthusiasm would also be substantiated by clients now interacting with a system with an easy-to-use and appealing user experience, instead of a direct messaging platform where information is not categorised.

Use Case 5: Update Appointment

As with Update Client Details, any updates would be conducted through direct messaging on the salon’s social media. These may be missed due to the influx of direct messages the salon’s account receives daily, as well as miscommunication or communication redundancy should the client wish to change their appointment date and/or time to an already-scheduled slot or a slot where the technician is unavailable. This may also cause further frustrations between the business and client as the client has no way of knowing the validity of the scheduled slot and may be adamant in requesting this date and time.

With the proposed system, this process would be easier as the calendar page shall indicate the days the salon’s technician is unavailable prior to the client finalising the appointment. Once the appointment is finalised, the client has the option to contact the salon to change their appointment date and/or time should they no longer be available for their initial time. After liaison with the salon’s administrator, the administrator shall update these details on their side, of which the system notifies both the client and salon – reducing miscommunication.

Use Case 6: Complete Payment

One of, if not the most, frustrating aspect of the salon’s current system is lack of communication over whether clients have paid the 50% non-refundable deposit required prior to the appointment, and that the business records and schedules appointments without receiving this deposit at all. According to the business rules, clients should pay this deposit upon the salon recording the appointment, as this takes time slots and stock allocation that could go to another client. Lack of control over this deposit has caused the salon much loss of income.

The proposed system solves this by having the Complete Payment use case, whereby this problem is addressed through the salon's administrator receiving appointment notifications only after the non-refundable 50% deposit is paid – recognising the appointment as confirmed and finalised. Until this deposit is made, it is clearly communicated to clients that they would not receive services unless the deposit payment is recognised. This process steadily reduces communication frustrations and the salon losing potential income.

5. CORE USE CASES

5.1 USE CASE 1

Use Case Name	Register on System
Scope	Client Appointment Tracking System
Triggering Event	The client clicks 'Register'.
Brief Description	A client wishes to undergo a service offered by the Nail Salon but must first register on the system to create their profile linking them to the services they require. The client will select the 'Register' button on the Home Page and will be taken to the Registration Page. The client is then prompted to enter their details and will also need to enter and confirm their password. The client will then select the 'Proceed' button and the system will verify all details entered. Once the validation is successful, the system will display a confirmation message to the client stating that they are registered. All the client's details are saved on the system and a ClientID is generated.
Actor(s)	Client (Primary)
Related Use Cases	Login (extends to Register)

Stakeholders & Interests	Client: Wants to successfully register on the system.	
Pre-conditions	• The system must be fully functional, operational and running. • The user must not have an existing account on the system.	
Post-conditions	• Client is successfully registered on the system. • ClientID is generated, assigned and stored to the Client Table. • The Client Table is updated with the client's details.	
Flow of Activities	**Actor**	**System**
	1. Client indicates the start of the Registration process by selecting 'Register' on the Login Page.	1.1 Initiates the client Registration process. 1.2. Displays the Registration Page.
	2. Client is prompted to enter their details (ClientFirstName, ClientLastName, ClientEmail, ClientPhoneNumber, ClientPasswordHash). They also confirm their password.	2.1 Captures the client's details to the Client Table after all details are entered.
	3. Client selects 'Register'.	3.1 System validates all details entered.
	4.1. Client receives successful registration confirmation.	4. Sends confirmation message to the client.
Exceptions	3.1.1. Receives error messages.	3.1. Display error messages ("All fields must be completed", "Invalid inputs", "Passwords do not match"), if any invalid results return.
	Any time before the registration process is complete:	
	1. The client can select 'Back'.	1.1 Takes client back to the Home Page.

5.2. USE CASE 2

Use Case Name	Create Booking	
Scope	Client Appointment Tracking System	
Triggering Event	The client selects ‘Book Now’.	
Brief Description	Upon browsing the services the salon has on offer, the client wishes to create an appointment booking to receive such services. The client logs into the system, selects the ‘book now’ option and then selects the details of the service they desire, repeating until satisfied. The client then chooses the date and time of the appointment and the preferred payment method. An appropriate confirmation message is generated and sent to the client.	
Actor(s)	Client (Primary)	
Related Use Cases	Complete Payment Manage Booking	
Stakeholders & Interests	Client: wants to book an appointment to receive services from the salon.	
Preconditions	• The client should be logged in to the system. • The system should be functional.	
Postconditions	• The booking is created, and details are stored in the Appointment Table. • InvoiceID is created. • Appropriate confirmation messages are sent to the client.	
Flow of Activities	**Actor**	**System**
	1. The use case begins once the client selects ‘Book Now’.	1.1. Initiates the Booking Process. The booking is created, assigned an AppointmentID and stored in the Appointment Table. 1.2. Displays the Booking Details page.
	2. Client selects the services they would like to receive, based on the available nail sizes. The client selects the price of the service(s) they require, then clicks “confirm”.	2.1. Generates ServiceRenderedID, records the service details selected and adds the relevant ServiceID to the ServiceRendered Table.

		2.2. Once ‘confirm’ is selected, the system will display the calendar page.
	3. Client selects an appropriate time and date for their appointment (AppointmentDate, AppointmentTime), and clicks the ‘confirm’ button.	3.1. AppointmentDate and AppointmentTime is captured to the Appointment Table. 3.2 Returns a confirmation message that shows that the client’s chosen slot is valid
		4. The appointment booking page is displayed, which shows the total amount and summary of details for all services the client has selected. 4.1 InvoiceID is created.
	5. Client selects ‘Pay Now’ to pay the 50% deposit.	5.1 Redirects the client to an external payment system. 5.2 Saves transaction_id and status of the payment 5.3 Creates an appointment_id on the appointment table 5.4 Use case ends.
	[Optional] 6. Client decides to pay later, clicks “Later”.	6.1 Returns a message stating “For your appointment to be considered, a 50% deposit is required”
	6.2 Client still decides not to pay, clicks “Later”	6.3 The appointment is saved into the pending_appointment column in the appointment table and use case ends.
Exceptions	**Actor**	**System**
		Validation: Should any information be incomplete, the system will display the

		relevant error messages (“Please log in to proceed”, “Please select a nail size”, “Please select a service”).
	Any time before payment is successful: 1. Client may remove one or more of the services.	1.1. Booking details are not saved on the system and the use case ends.
		2.1. Current booking details are saved on the system but not confirmed as a finalised appointment, and the use case ends.
		3.1. Removes the specified service from the booking details.

5.3. USE CASE 3

<table>
<tr><td><b>Use Case Name</b></td><td colspan="2">Delete Booking</td></tr>
<tr><td><b>Scope</b></td><td colspan="2">Client Appointment Tracking System</td></tr>
<tr><td><b>Triggering Event</b></td><td colspan="2">The client wants to delete their booking from the system.</td></tr>
<tr><td><b>Brief Description</b></td><td colspan="2">The client navigates to their relevant booking that was made and saved on the system that they wish to delete. The client will select their booking and delete it from the system. The system will then display a message to the client asking the reason as to why they chose to delete their booking. The client will then choose a reason from the options provided in the list and this feedback will be saved to the system’s database.</td></tr>
<tr><td><b>Actor(s)</b></td><td colspan="2">Client (Primary)</td></tr>
<tr><td><b>Related Use Cases</b></td><td colspan="2">Manage Booking</td></tr>
<tr><td><b>Stakeholders & Interests</b></td><td colspan="2">Client: Wants to successfully delete their booking from the system.
Administrator: Wants to receive feedback from the client about the reason for the cancellation of the appointment.</td></tr>
<tr><td><b>Pre-conditions</b></td><td colspan="2">• Appointment must exist in the Appointment Table.
• Client must exist in the Client Table.
• The system should be functional.
• Client should be logged in on the system.</td></tr>
<tr><td><b>Post-conditions</b></td><td colspan="2">• The client’s booking is successfully deleted.
• The Appointment Table is updated.</td></tr>
<tr><td><b>Flow of Activities</b></td><td>Actor</td><td>System</td></tr>
<tr><td></td><td>1. Client opens the hamburger menu and selects appointments.</td><td>1.1 Displays bookings under the Appointment Page.</td></tr>
<tr><td></td><td>2. Client selects relevant booking that needs to be deleted.</td><td>2.1. Displays the booking details for the relevant appointment.</td></tr>
<tr><td></td><td>3. Client clicks ‘Delete’.</td><td>3.1. Displays message “Are you sure you want to delete your appointment?” on the Appointment Page.</td></tr>
<tr><td></td><td>4. Client clicks ‘Yes’.</td><td>4.1. Deletes the relevant appointment.</td></tr>
<tr><td></td><td></td><td>5. Sends confirmation message “Appointment deleted”.</td></tr>
</table>

	5.1. Client receives confirmation message of deleted appointment and clicks ‘Okay’. 6.1. Receives cancellation feedback message. 7. Client selects reason for appointment cancellation. 8.1. Receives confirmation message.	6. Sends message “Please provide a reason for your cancellation”. 7.1. Captures and saves the client’s feedback option. 8. Sends confirmation message “Thank you for your feedback”. 9. Returns to the home page
Exceptions	3.1. Client can cancel the delete process by selecting ‘No’ when the “Are you sure you want to delete your appointment?” message is displayed on the Appointment Page.	3.1.1. No changes are recorded. 3.2. Returns back to appointment page

5.4. USE CASE 4

Use Case Name	Update Client Details	
Scope	Client Appointment Tracking System	
Triggering Event	The client selects ‘Update Details’.	
Brief Description	Should a client wish to update their details on the system, they shall select the ‘update details’ option on the Client Page after logging in. They will make their necessary changes, then select the ‘save’ button, allowing the system to update the changes to the Client Table in the database. The process will be repeated until all valid details are entered. At any point before the process is complete, the client may select ‘cancel’ and no changes will be processed.	
Actor(s)	Client (Primary)	
Related Use Cases	-	
Stakeholders & Interests	Client: wants to change or update their details initially saved to the system. Administrator: wants the correct information about the business’s clients for appointments and communication.	
Preconditions	• The client should be logged in to the system. • The system should be functional.	
Postconditions	• The new details are stored in the Client Table. • Appropriate confirmation message is sent to the client.	
Flow of Activities	**Actor**	**System**
	1. The use case begins once the client selects ‘Update Details’ on the client’s profile page.	1.1. The system will display the Client Details page.
	2. Client edits their current details saved to the system (possible changes to ClientFirstName, ClientLastName, ClientEmail, ClientPhoneNumber, ClientPasswordHash), and selects the ‘save’ option once done.	
		3. Captures the changes.
		4. System saves all the changes in client details to the Client Table.

	6. Client receives the confirmation message, and the use case ends.	5. System sends the appropriate confirmation message of the client’s new details saved to the system.
Exceptions	**Actor**	**System**
	Any time before ‘save’ is selected: 1. Client may select ‘cancel’.	**Validation:** Should any information be incomplete, the system will display the relevant error messages (“Please complete all fields”.) 1.1. New Client Details or any changes made to the existing details are not saved to the system, and the use case ends.

6. USE CASES PLANNED FOR REFINEMENT:

The following use cases are part of the six core use cases and shall be implemented in the next iteration:

1. **Update Appointment:** If the administrator wishes to update a client's appointment, the administrator shall select the relevant appointment and make the required changes to the appointment details. Those changes will be updated and stored to the Appointment Table in the system's database. At any point before the process is complete, the administrator may select 'cancel' and no changes will be processed.
2. **Complete Payment:** Linked to the Create Booking use case, the client will follow the booking process as described. Upon selecting the 'Pay Debit/Credit Card' option, they will proceed to complete their payment to have their booking confirmed. The client shall then pay a 50% deposit at a third-party online payment service (refer to *Yoco* actor on *4.9. Use Case Diagram*). Once the payment is complete, the booking will be deemed as confirmed, and notifications will be sent out to both the client and the business administrator. At any point before the payment is successful, the client may select 'back' and their booking details will be saved to the Appointment Table and ServicesRendered Table in the system's database but is not confirmed as a finalised appointment.

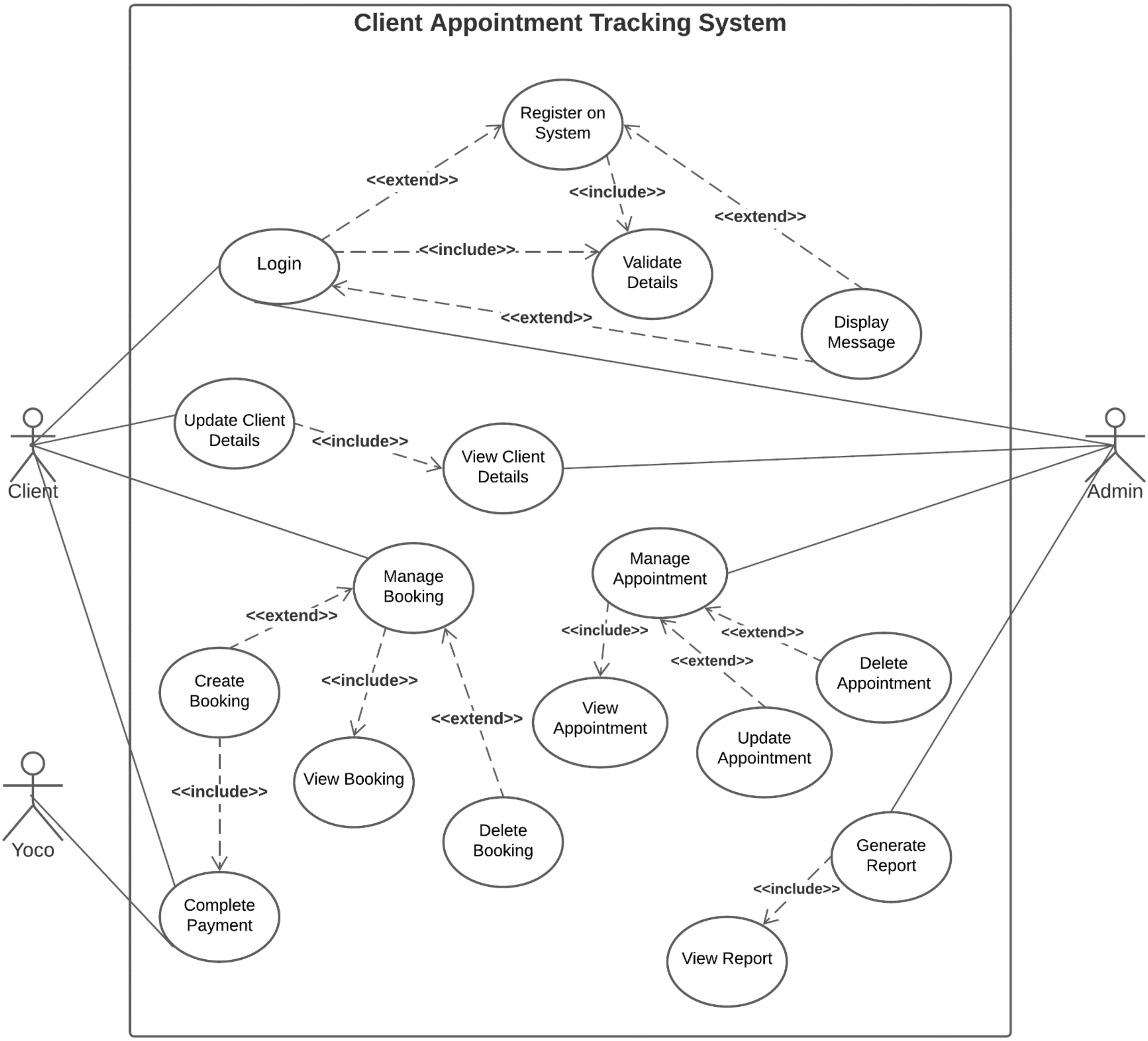
7. MAINTENANCE USE CASES

The following use cases are grouped to support certain core use cases and consist of create, view, update and delete processes, which all fall under the respective "Manage" maintenance case.

1. **Manage Booking**
    - Create Booking
    - View Booking
    - Delete Booking
2. **Manage Appointment**
    - View Appointment
    - Update Appointment
    - Delete Appointment

7.1. USE CASE DIAGRAM

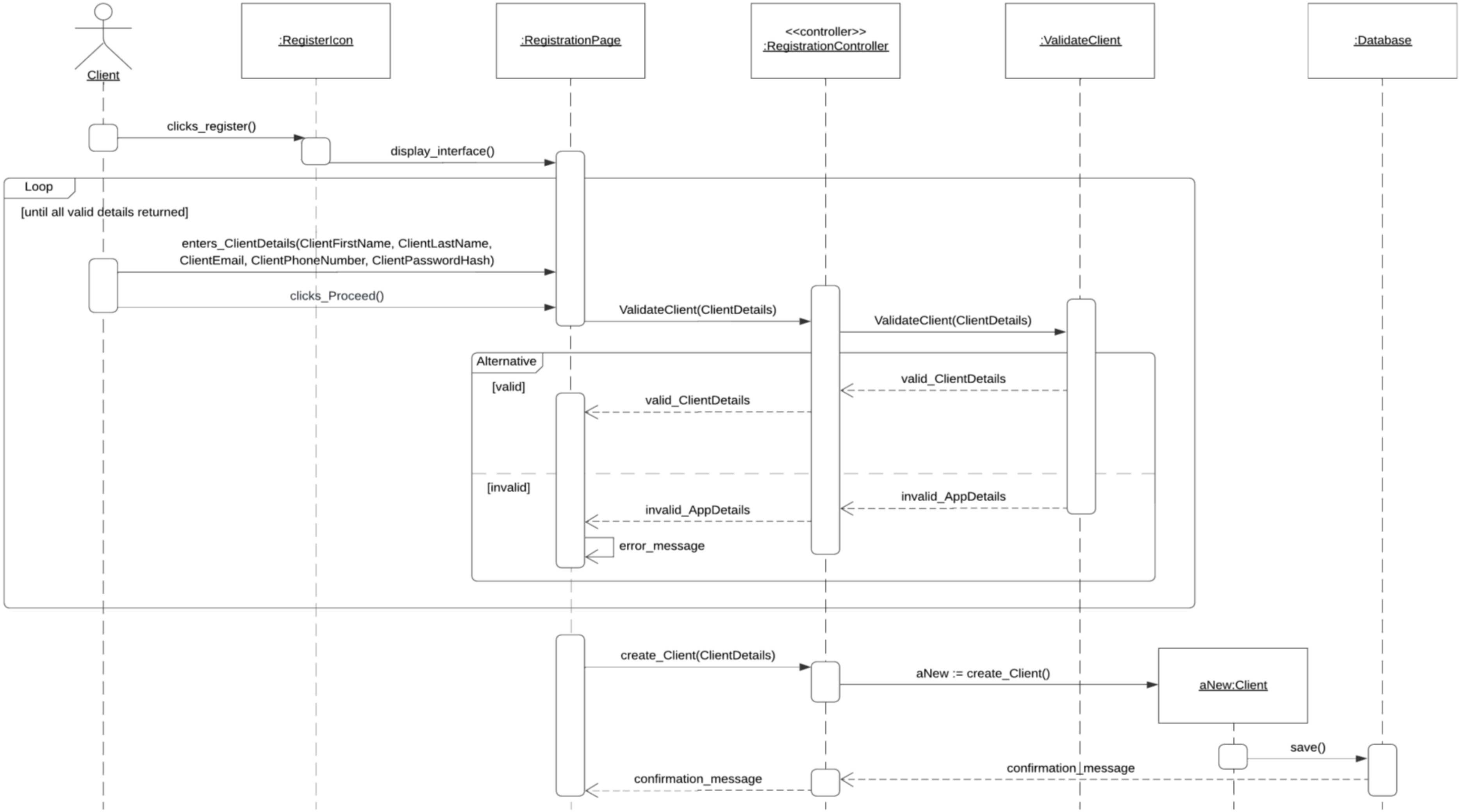
The following use case diagram shows the interactions between actors/users (LHS and RHS) and the system (middle) relating to the use cases of the proposed automated booking system. The diagram shows how a client would use the system to make and/or manage a booking, and how the business administrator would manage client appointments and record details thereof. There are also "Manage" use cases that consist of create, view, update and delete use cases, intended to fulfil the CRUD requirements of the system.



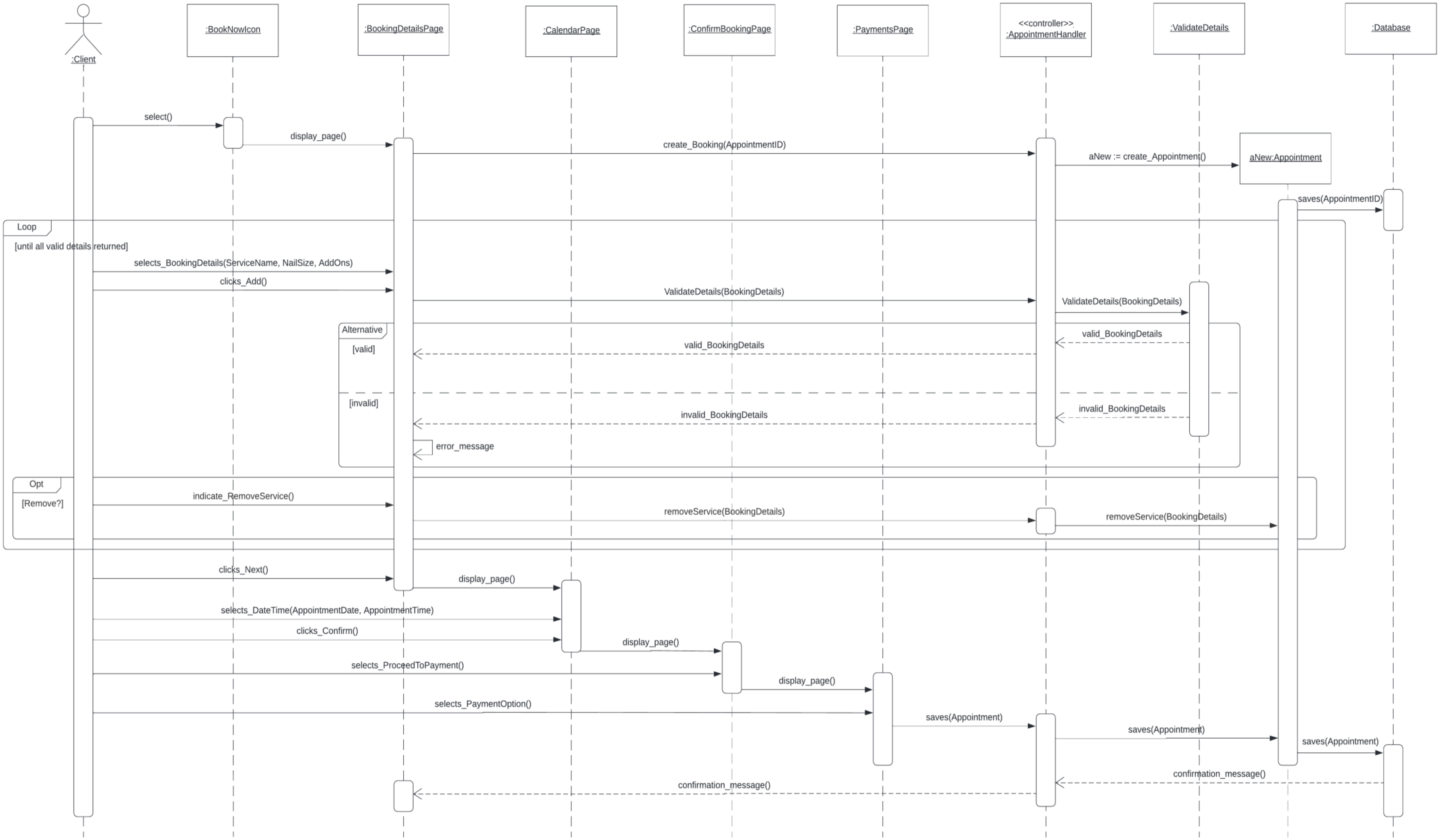
8. SEQUENCE DIAGRAMS

Below are the sequence diagrams pertaining to the first three core use cases of the system, to further illustrate the sequence of events involved, with explicit link to the flow of activities and design in the relevant fully dressed use cases.

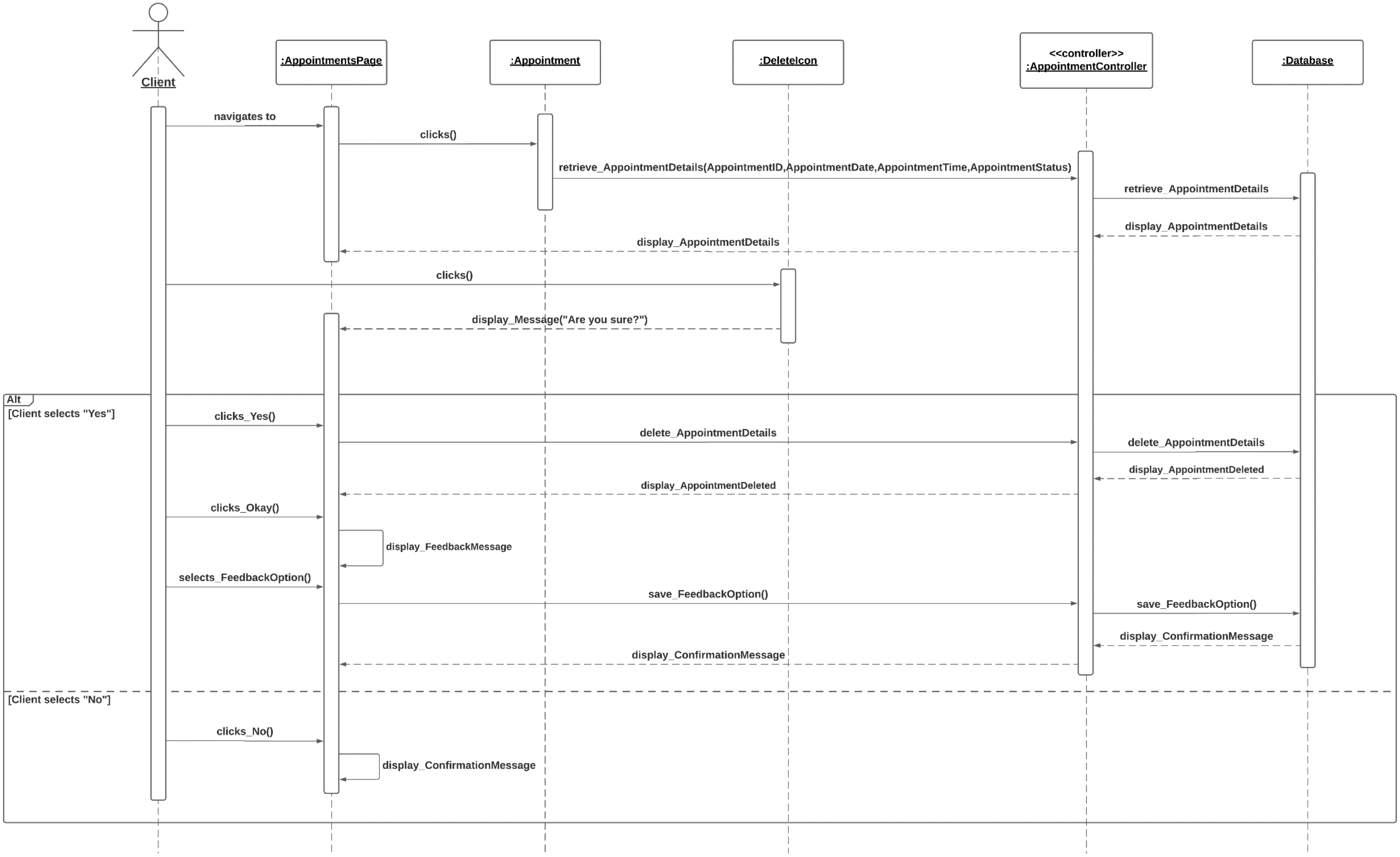
5.1. CLIENT REGISTRATION



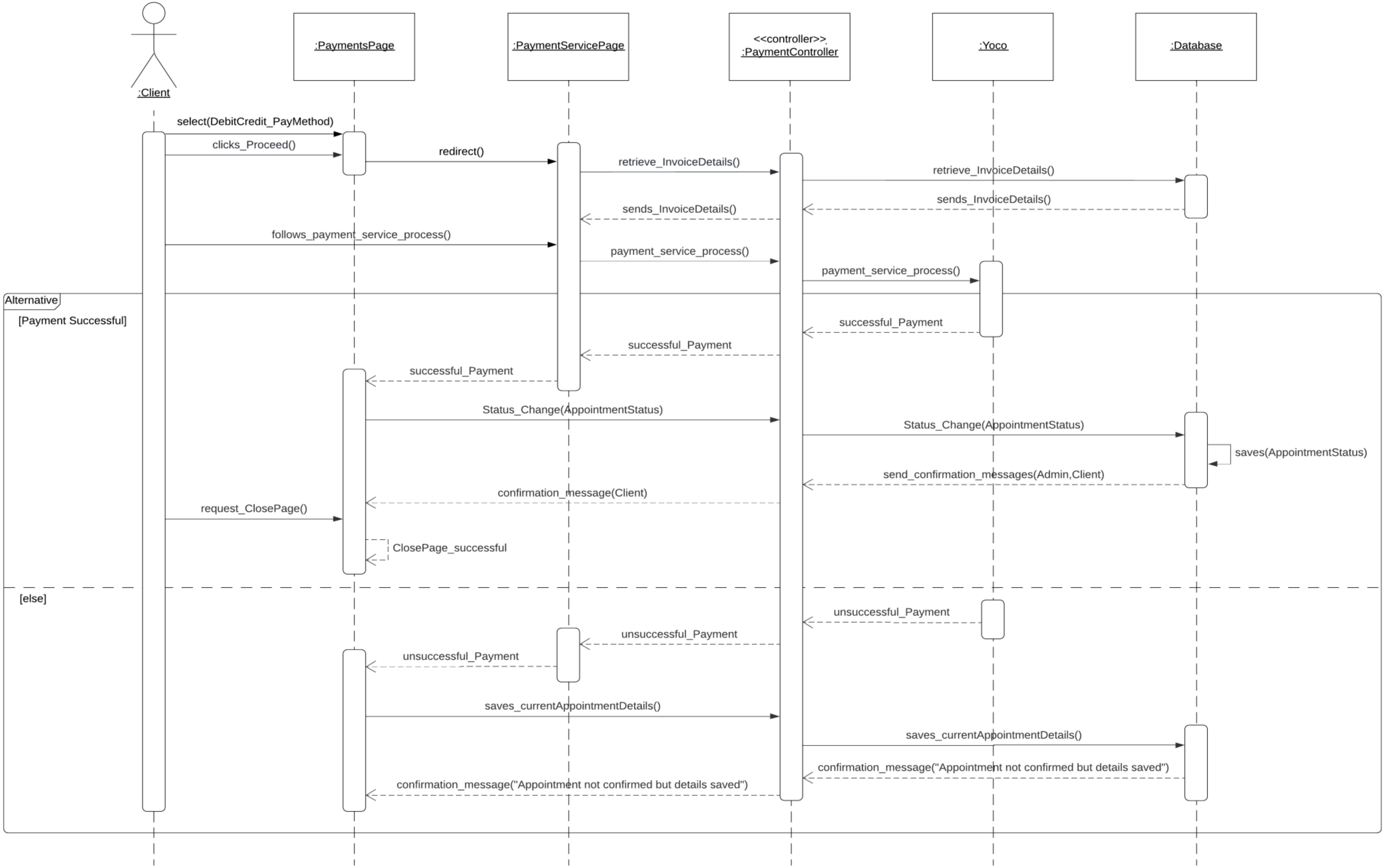
5.2. CREATE BOOKING



5.3. DELETE BOOKING

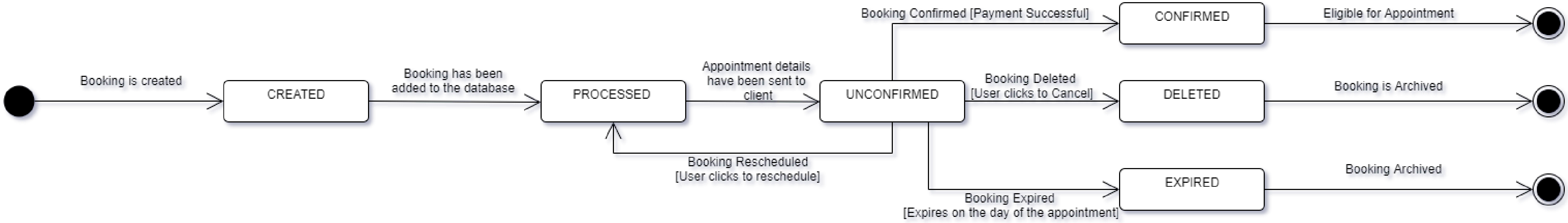

Client
:AppointmentsPage
:Appointment
:DeleteIcon
<<controller>>
:AppointmentController
:Database
navigates to
clicks()
retrieve_AppointmentDetails(AppointmentID,AppointmentDate,AppointmentTime,AppointmentStatus)
retrieve_AppointmentDetails
display_AppointmentDetails
display_AppointmentDetails
clicks()
display_Message("Are you sure?")
Alt
[Client selects "Yes"]
clicks_Yes()
delete_AppointmentDetails
delete_AppointmentDetails
display_AppointmentDeleted
display_AppointmentDeleted
clicks_Okay()
display_FeedbackMessage
selects_FeedbackOption()
save_FeedbackOption()
save_FeedbackOption()
display_ConfirmationMessage
display_ConfirmationMessage
[Client selects "No"]
clicks_No()
display_ConfirmationMessage

5.4. UPDATE CLIENT DETAILS


:Client
:PaymentsPage
:PaymentServicePage
<<controller>>
:PaymentController
:Yoco
:Database
select(DebitCredit_PayMethod)
clicks_Proceed()
redirect()
retrieve_InvoiceDetails()
retrieve_InvoiceDetails()
sends_InvoiceDetails()
sends_InvoiceDetails()
follows_payment_service_process()
payment_service_process()
payment_service_process()
Alternative
[Payment Successful]
successful_Payment
successful_Payment
successful_Payment
Status_Change(AppointmentStatus)
Status_Change(AppointmentStatus)
saves(AppointmentStatus)
send_confirmation_messages(Admin,Client)
confirmation_message(Client)
request_ClosePage()
ClosePage_successful
[else]
unsuccessful_Payment
unsuccessful_Payment
unsuccessful_Payment
saves_currentAppointmentDetails()
saves_currentAppointmentDetails()
confirmation_message("Appointment not confirmed but details saved")
confirmation_message("Appointment not confirmed but details saved")

9. STATE MACHINE DIAGRAM

The following State Machine Diagram explains the state changes that would occur to an appointment booking on the automated booking system. The State Machine begins from a state of *created* and ending with *confirmed*, *deleted* or *expired*, with the relevant guard conditions highlighting the process leading to each state.

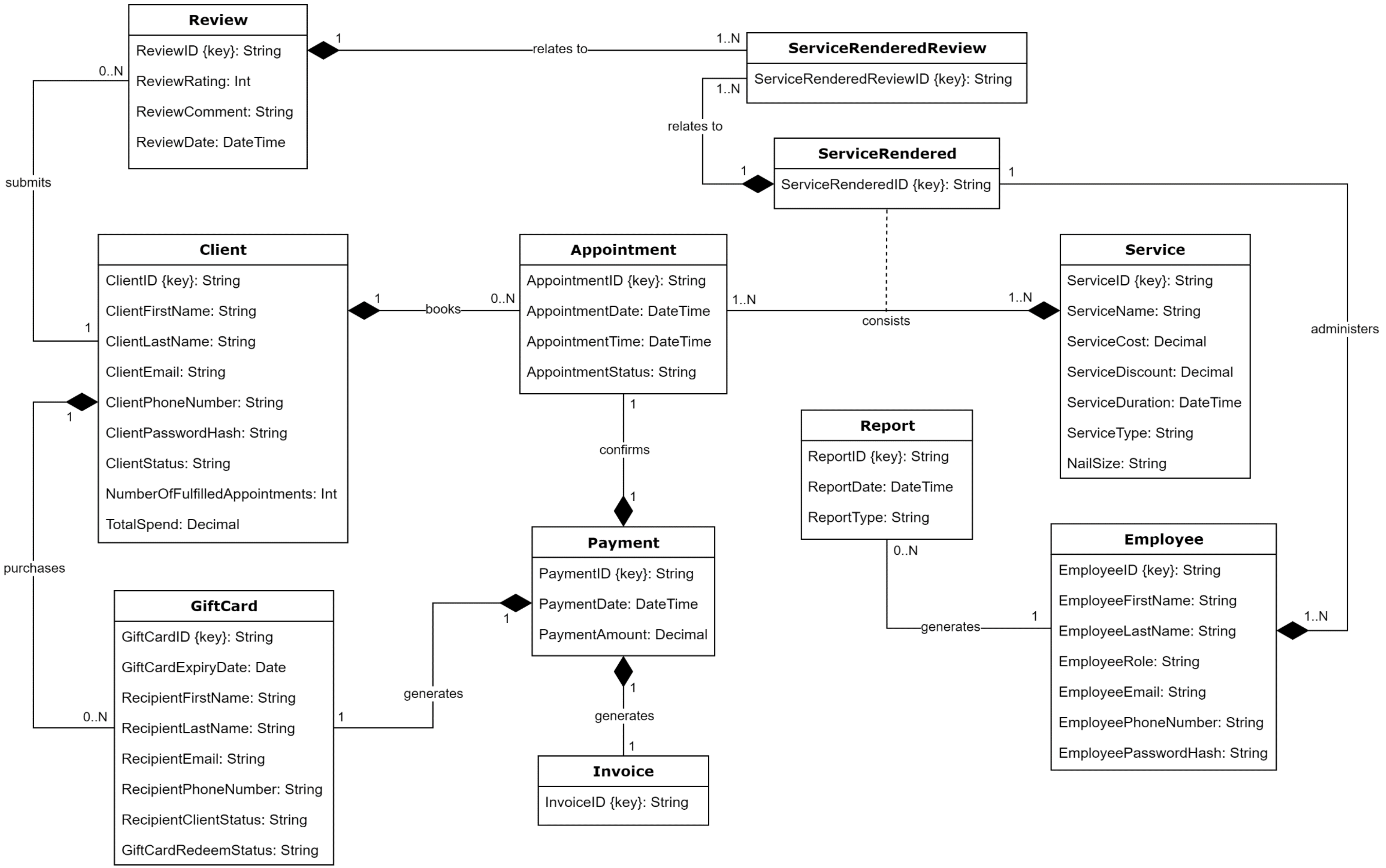

Booking is created
CREATED
Booking has been added to the database
PROCESSED
Appointment details have been sent to client
UNCONFIRMED
Booking Rescheduled [User clicks to reschedule]
Booking Confirmed [Payment Successful]
CONFIRMED
Eligible for Appointment
Booking Deleted [User clicks to Cancel]
DELETED
Booking is Archived
Booking Expired [Expires on the day of the appointment]
EXPIRED
Booking Archived

State Machine Diagram: Confirming Booking

7. DOMAIN AND DATA MODELS

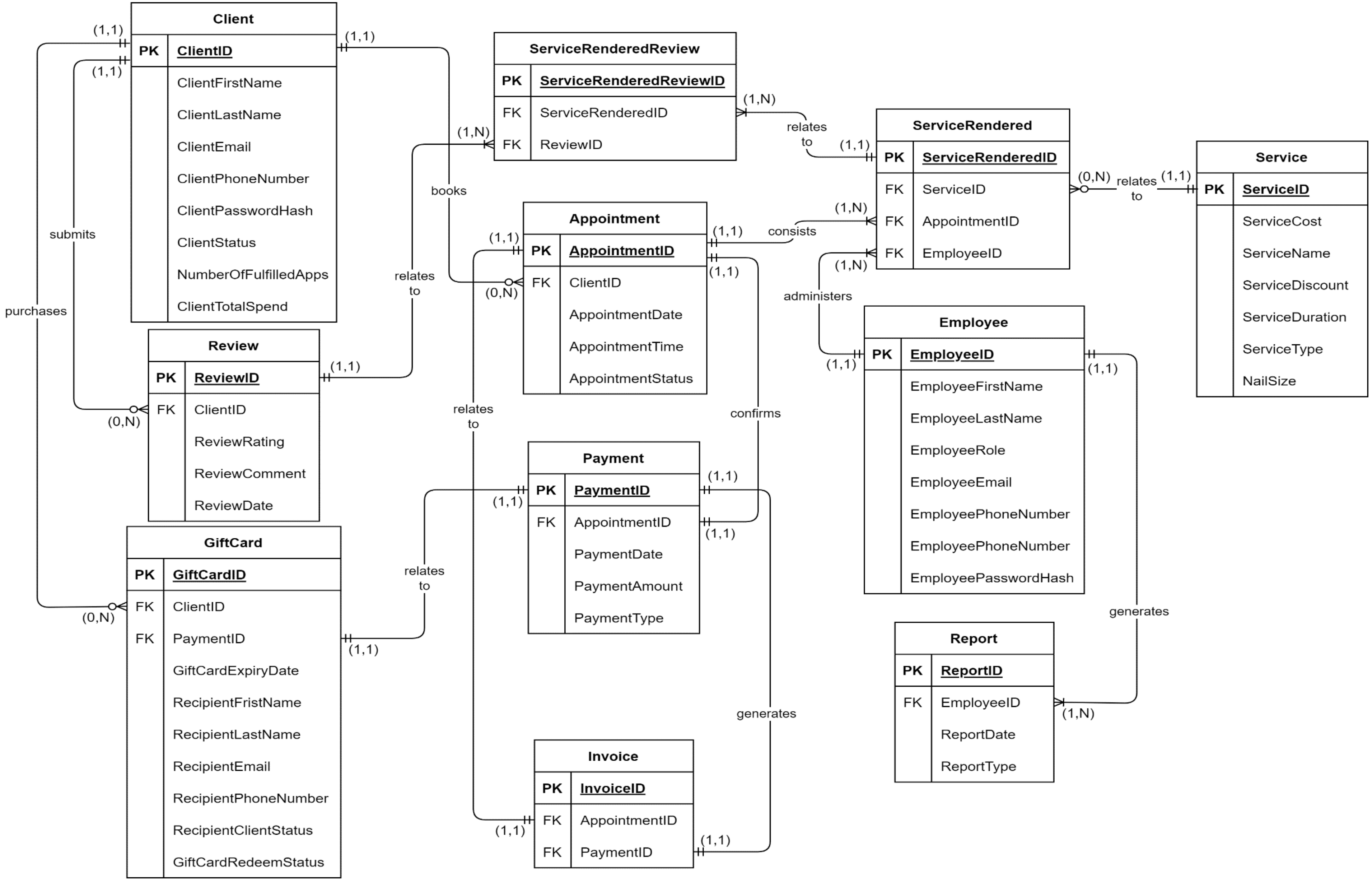
7.1. DOMAIN CLASS DIAGRAM

The Domain Class Model below indicates the strength of relationships between entities and pays a resemblance to the Entity Relationship Diagram.



7.2. ENTITY RELATIONSHIP DIAGRAM

The Entity Relationship Diagram below illustrates the relationships and rules related to the automated booking system.



Entity Relationship Diagram: Business Rules

Below are the business rules that outline the relationships between entities displayed in the Entity Relationship Diagram.

1. A **Client** may book zero or many **Appointments**. An **Appointment** may be booked by one and only one **Client**.
2. A **Client** may purchase zero or many **Gift Cards**. A **Gift Card** may only be purchased by one **Client**.
3. A **Client** may submit zero or many **Reviews**. A **Review** may be submitted by one and only one **Client**.
4. An **Appointment** may relate to one and only one **Invoice**. An **Invoice** may relate to one and only one **Appointment**.
5. An **Appointment** is confirmed by one and only one **Payment**. A **Payment** confirms one and only one **Appointment**.
6. One **Payment** generates one and only one **Invoice**. An **Invoice** is generated by one and only one **Payment**.
7. A **Gift Card** relates to one and only one **Payment**. A **Payment** may relate to one and only one **Gift Card**.
8. A **Review** relates to one or many **Service Rendered Reviews**. A **Service Rendered Review** relates to one and only one **Review**.
9. An **Appointment** consists of one or many **Services Rendered**. A **Service Rendered** is included in one and only one **Appointment**.
10. A **Service Rendered** relates to one or many **Service Rendered Reviews**. A **Service Rendered Review** relates to one and only one **Service Rendered**.
11. A **Service** relates to zero or many **Services Rendered**. A **Service Rendered** relates to one and only one **Service**.
12. An **Employee** administers one or many **Services Rendered**. A **Service Rendered** is administered by one and only one **Employee**.
13. An **Employee** generates zero or many **Reports**. A **Report** is generated by one and only one **Employee**.

8. BUSINESS MODELLING

8.1. BUSINESS PROCESS MODEL: CORE USE CASE INTEGRATION

The Business Process Model below (6.2.) illustrates the process *Lavished Nails by Lolly* follows regarding managing their bookings.

Currently, the business is without an automated system and receives booking requests through direct messaging on social media platforms, namely Instagram and WhatsApp. By integrating the automated booking system, the process of recording and keeping track of clients will be made easier through the **Register** functionality and use case, along with the **Update Client Details** use case, which absolves the business of unnecessary back-and-forth communication with clients.

Create Booking is also manually recorded by the business to date but, on the automated system, shall be a client-based experience. This shall free the business from tedious recording until the non-refundable deposit is paid by the client.

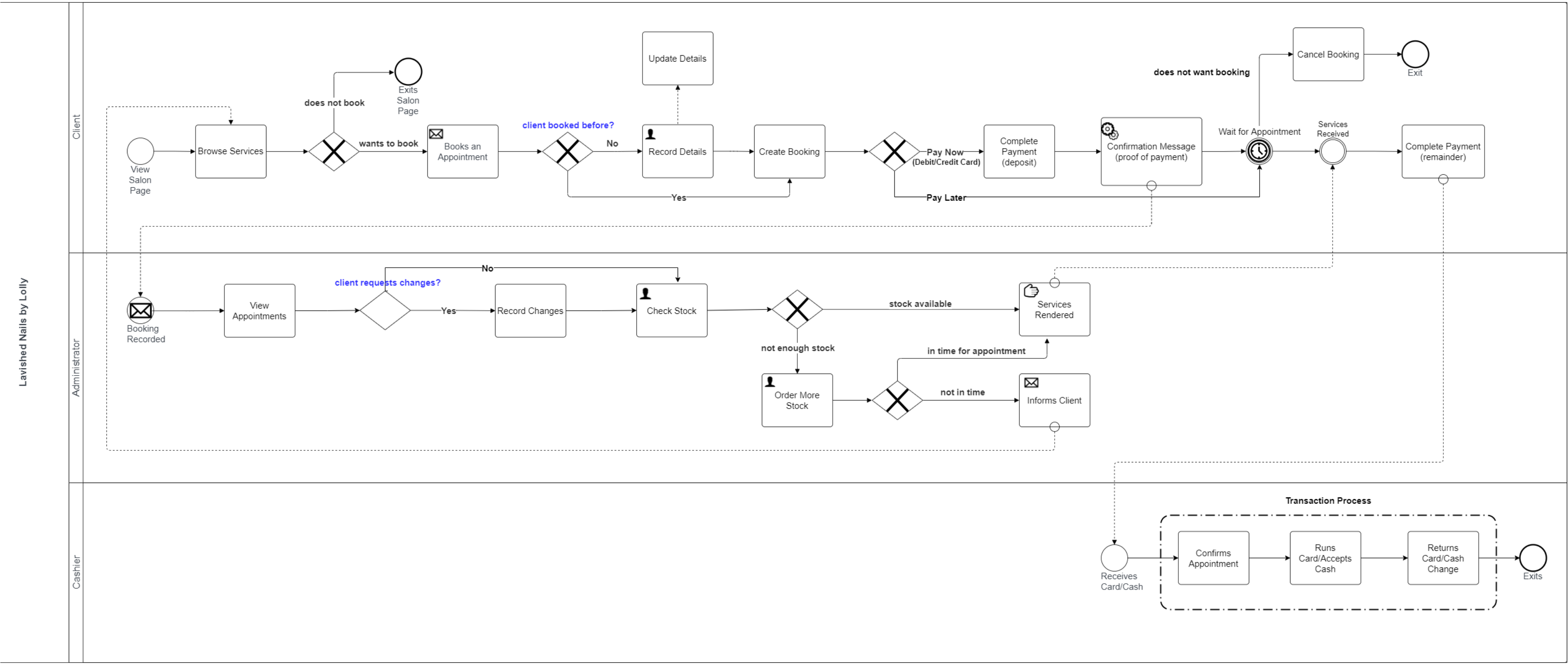
In terms of cancellations, the client will have the option of deleting their booking themselves at any time before the appointment on the automated system (**Delete Booking**) instead of manually contacting the business, as what currently occurs. With the automated system in place, the appointment cancellation is much more organised in the sense that the database will be immediately updated, and there is less room for error or time wasted in the current manual recording of said cancellations.

The administrator of the system is also able to change the dates of an appointment (**Update Appointment**) by utilising the calendar page should the client wish to reschedule, assisting in the tedious task of manually making the change in the business's own calendar and eliminating miscommunication between the business and client.

Another aspect of the system to be implemented, which will assist the business greatly, is the **Complete Payment** use case, whereby clients are required to pay the non-refundable deposit before their booking is confirmed and services are rendered. Currently, the business relies on communication with the clients to ensure the payment is made, leading to problems where clients do not arrive for their appointments without communicating a cancellation, and thereafter allowing the business to lose the deposit revenue.

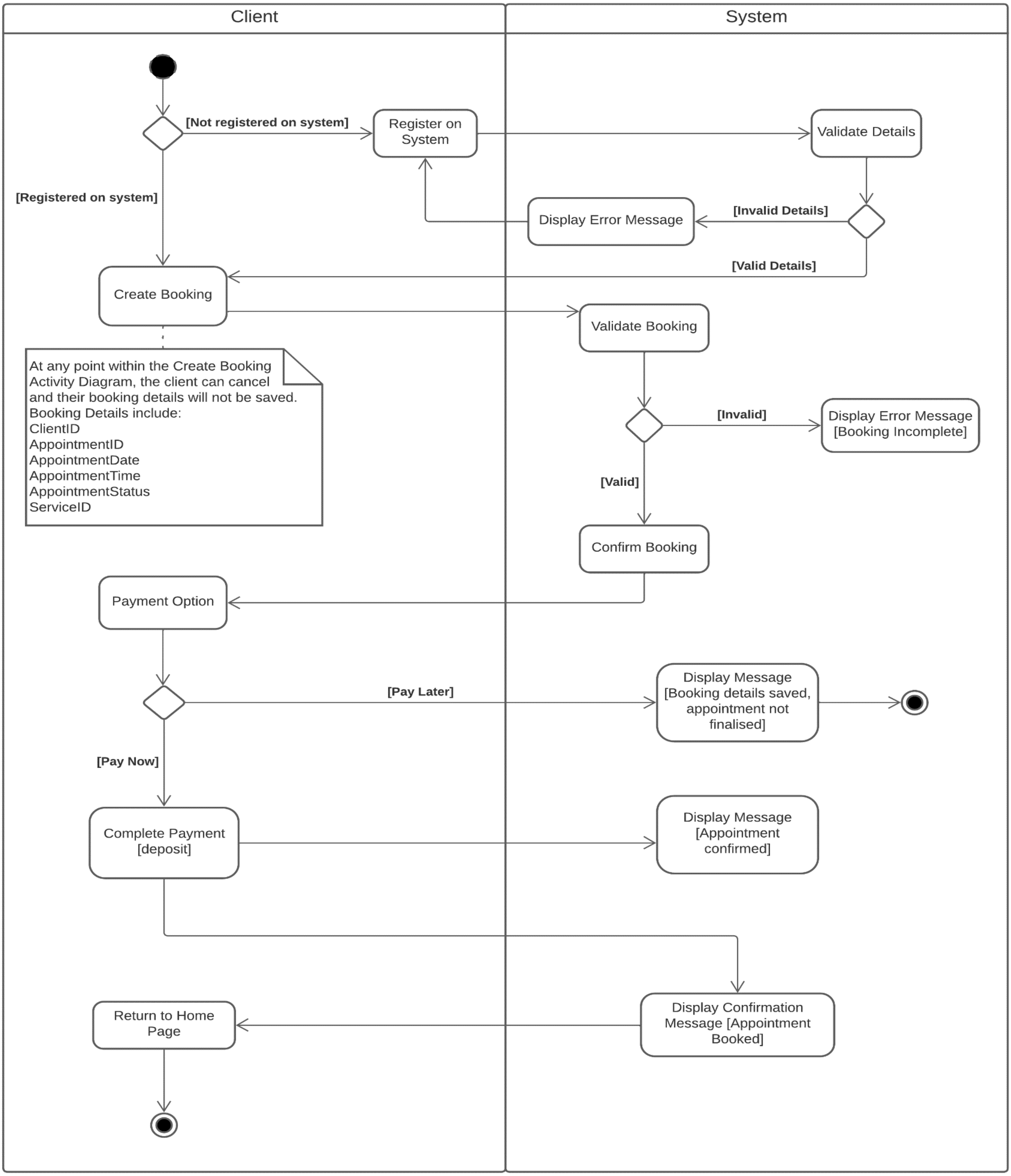
8.2. BUSINESS PROCESS MODEL

The following Business Process Model describes the current workflow of the *Lavished Nails by Lolly* nail salon, with specific attention to the process of booking an appointment.



8.3. ACTIVITY DIAGRAM

The following activity diagram for the Create Booking use case shows the workflow of activities and actions that a client will undergo to book an appointment on the automated booking system


Client
System
[Not registered on system]
Register on System
Validate Details
[Registered on system]
Display Error Message
[Invalid Details]
[Valid Details]
Create Booking
Validate Booking
At any point within the Create Booking Activity Diagram, the client can cancel and their booking details will not be saved.
Booking Details include:
ClientID
AppointmentID
AppointmentDate
AppointmentTime
AppointmentStatus
ServiceID
[Invalid]
Display Error Message [Booking Incomplete]
[Valid]
Confirm Booking
Payment Option
[Pay Later]
Display Message [Booking details saved, appointment not finalised]
[Pay Now]
Complete Payment [deposit]
Display Message [Appointment confirmed]
Return to Home Page
Display Confirmation Message [Appointment Booked]