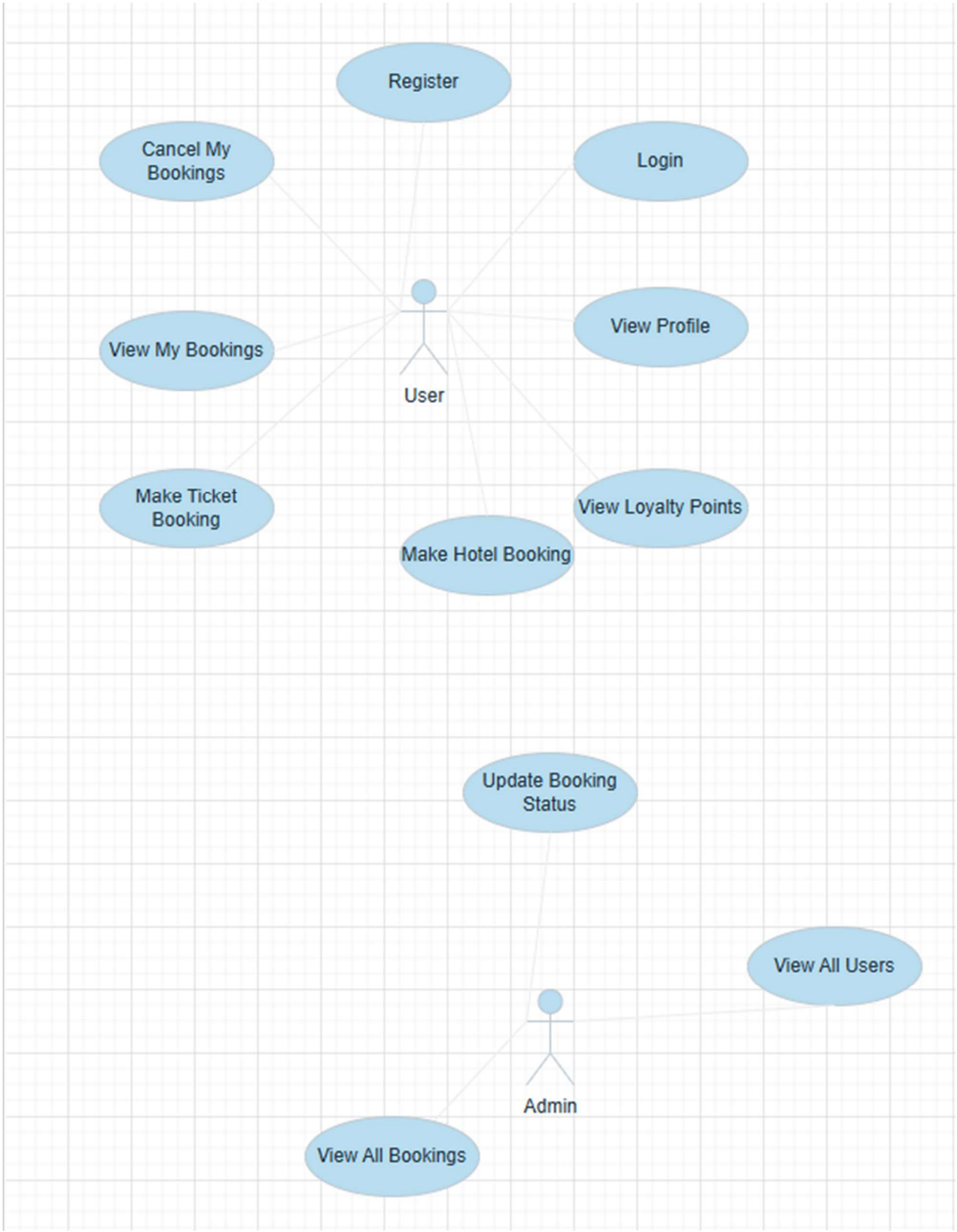
Riget Zoo Adventures – Task 1 – Design Document

Diagrams - Overview

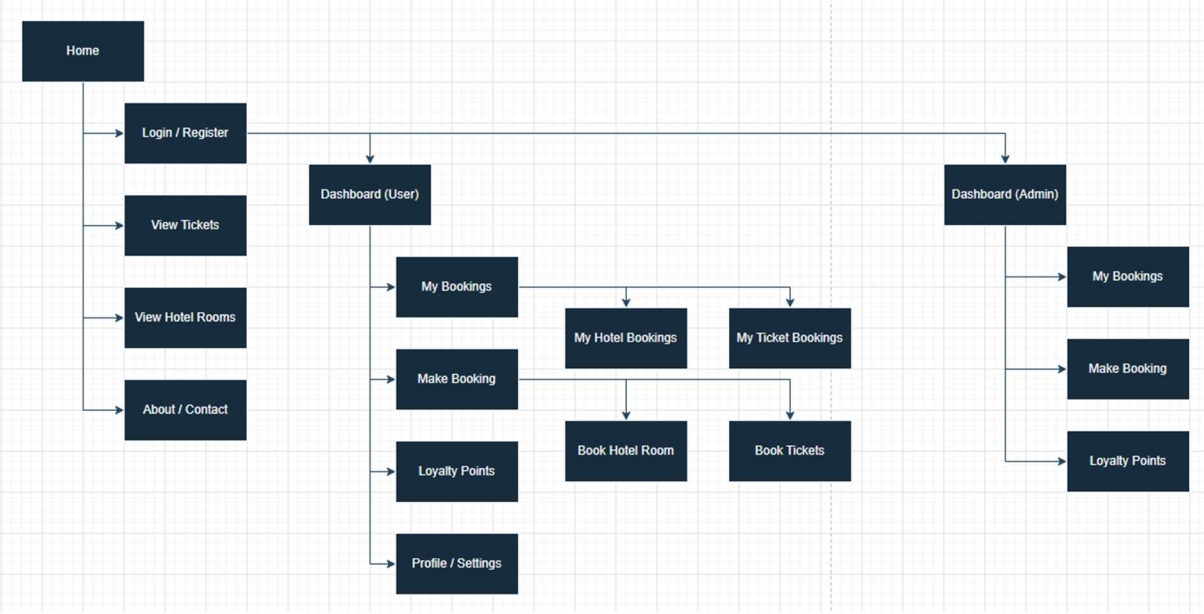
Use Case Diagram

The system supports two main actors: the User and the Admin. Users can register an account, log in securely, make and manage hotel and ticket bookings, and view their loyalty points. Administrators can monitor and manage bookings and user accounts. The use case diagram shows how each actor interacts with the system to perform these tasks.


Register
Cancel My Bookings
Login
View My Bookings
View Profile
User
Make Ticket Booking
View Loyalty Points
Make Hotel Booking
Update Booking Status
View All Users
Admin
View All Bookings

Hierarchy Diagram

This diagram outlines the application's navigation structure. Each page flows logically from the home page. Created using diagrams.net.



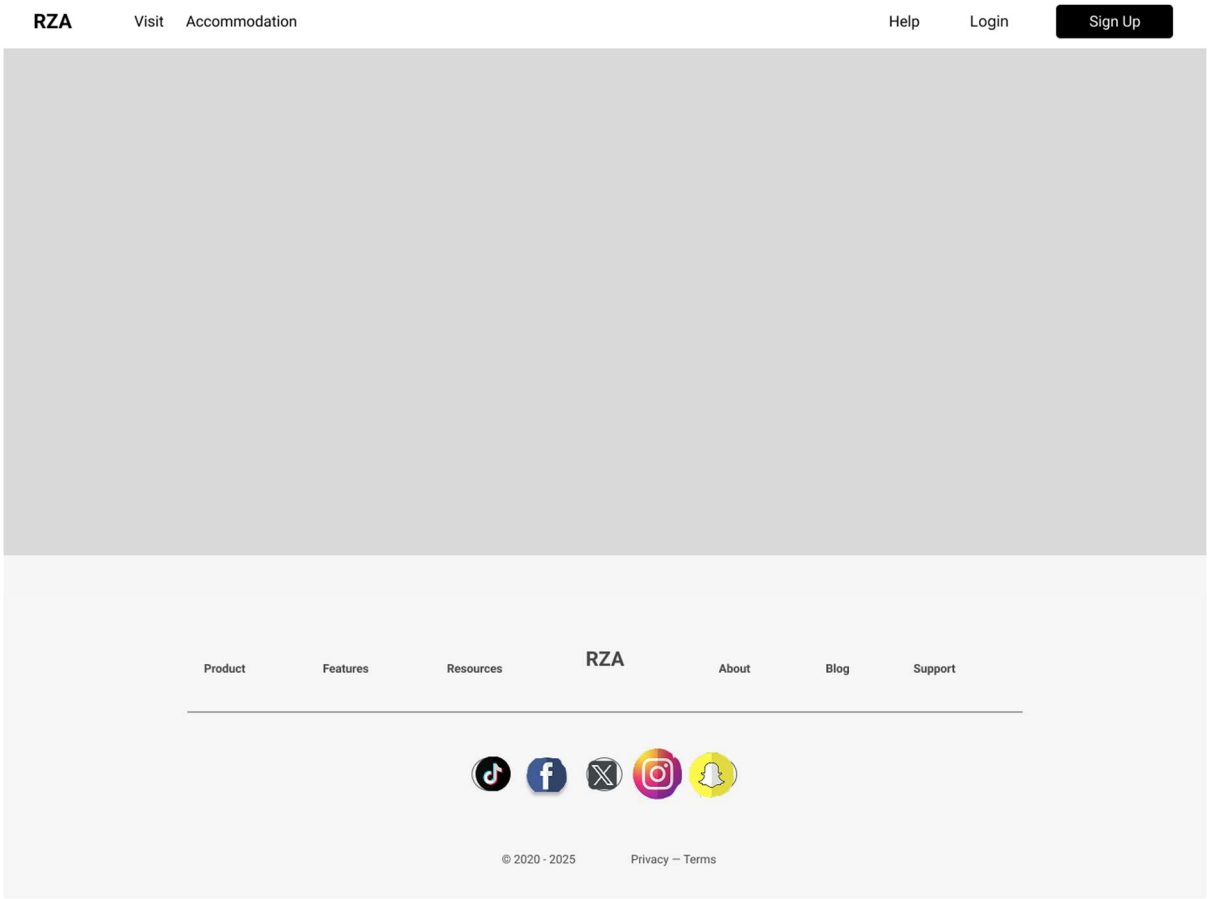
Wireframes -Overview

Wireframes provide a visual guide for layout and structure. They follow a hybrid style: clean layout, professional spacing, and minimal accent colours. Created in Figma with Lucide icons and Unsplash visuals.

Base

Includes header, footer, and general layout consistent across all pages.

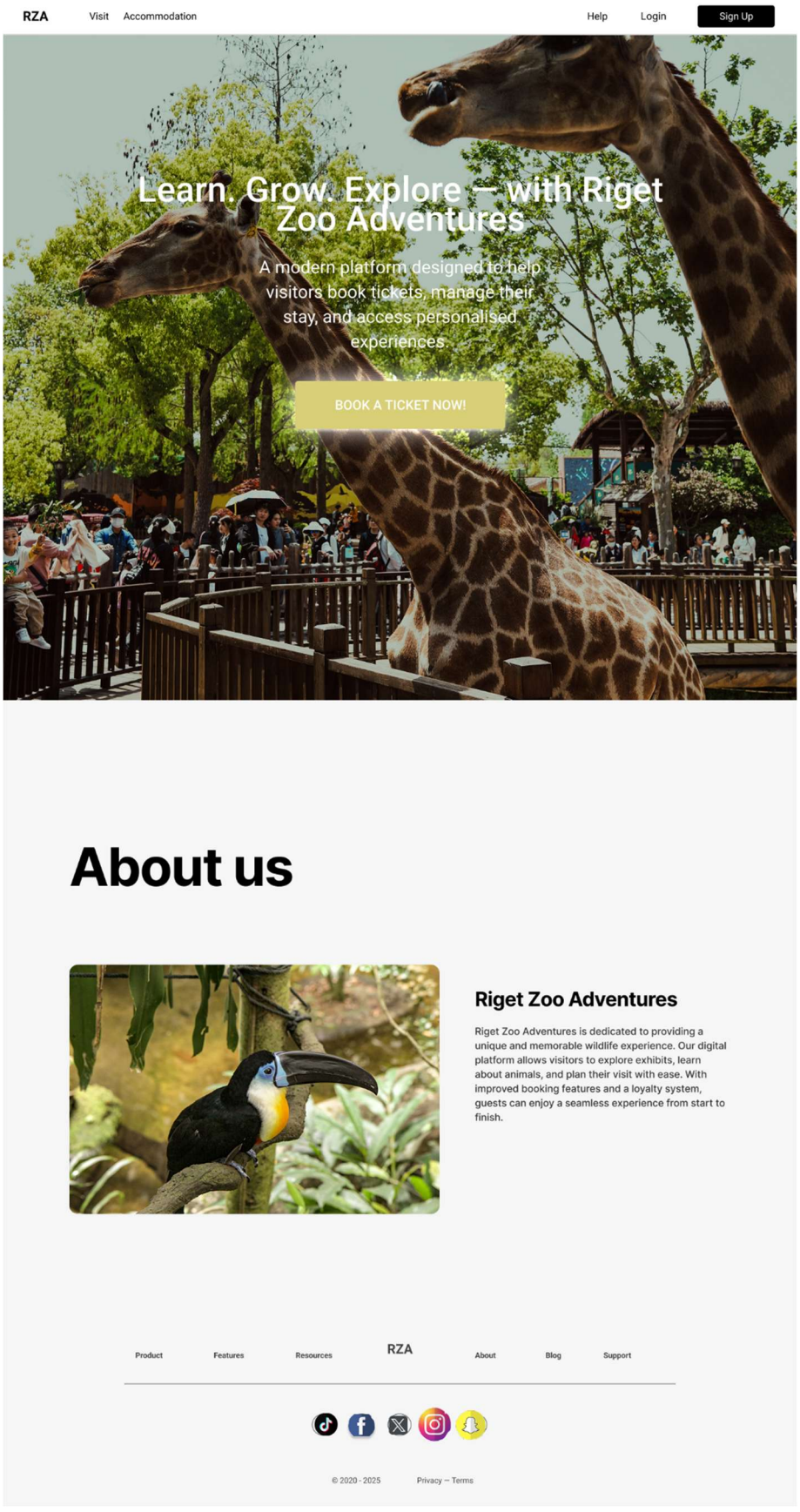
Header + Footer



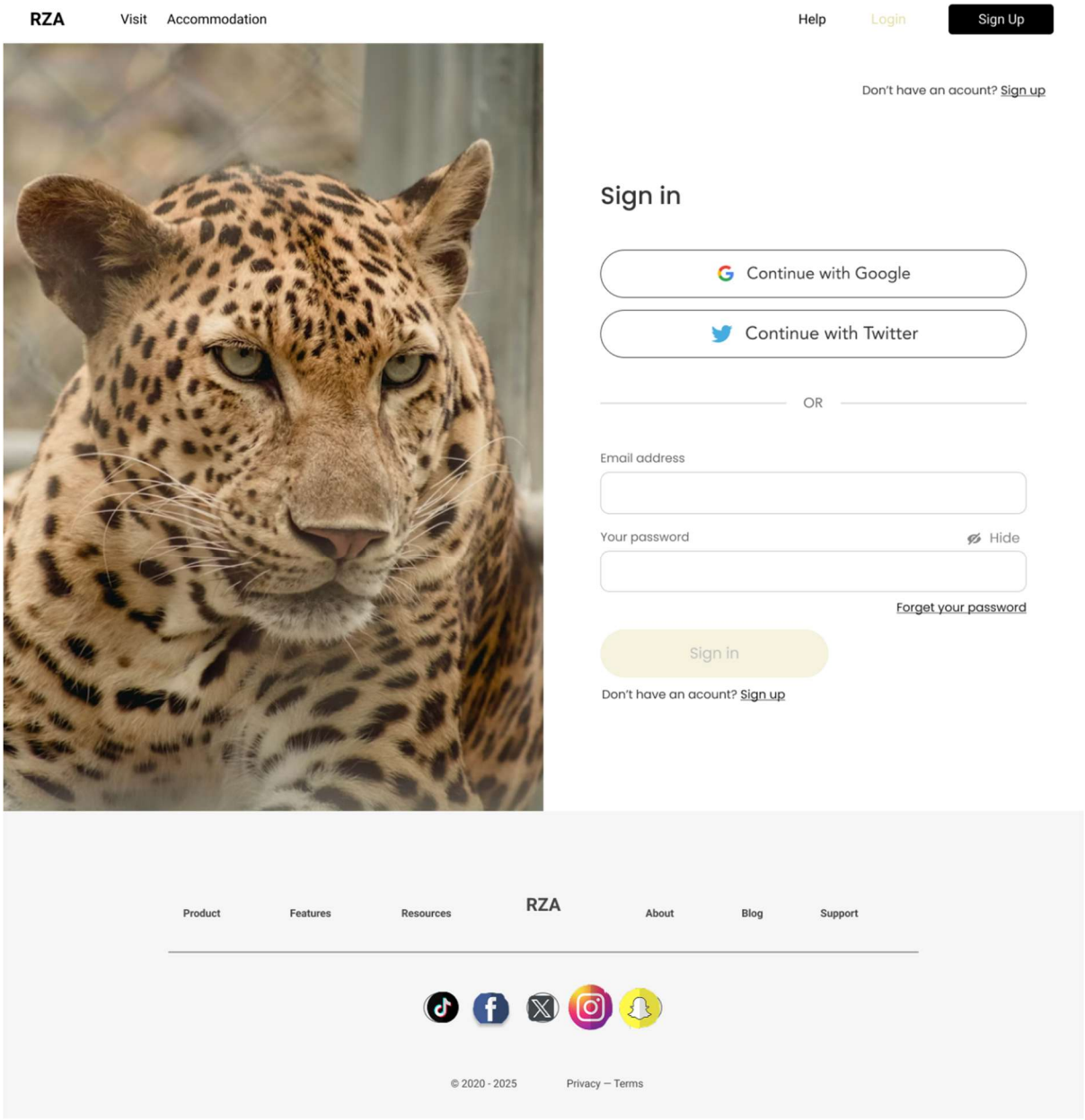
Constant

Pages that remain identical regardless of user type:

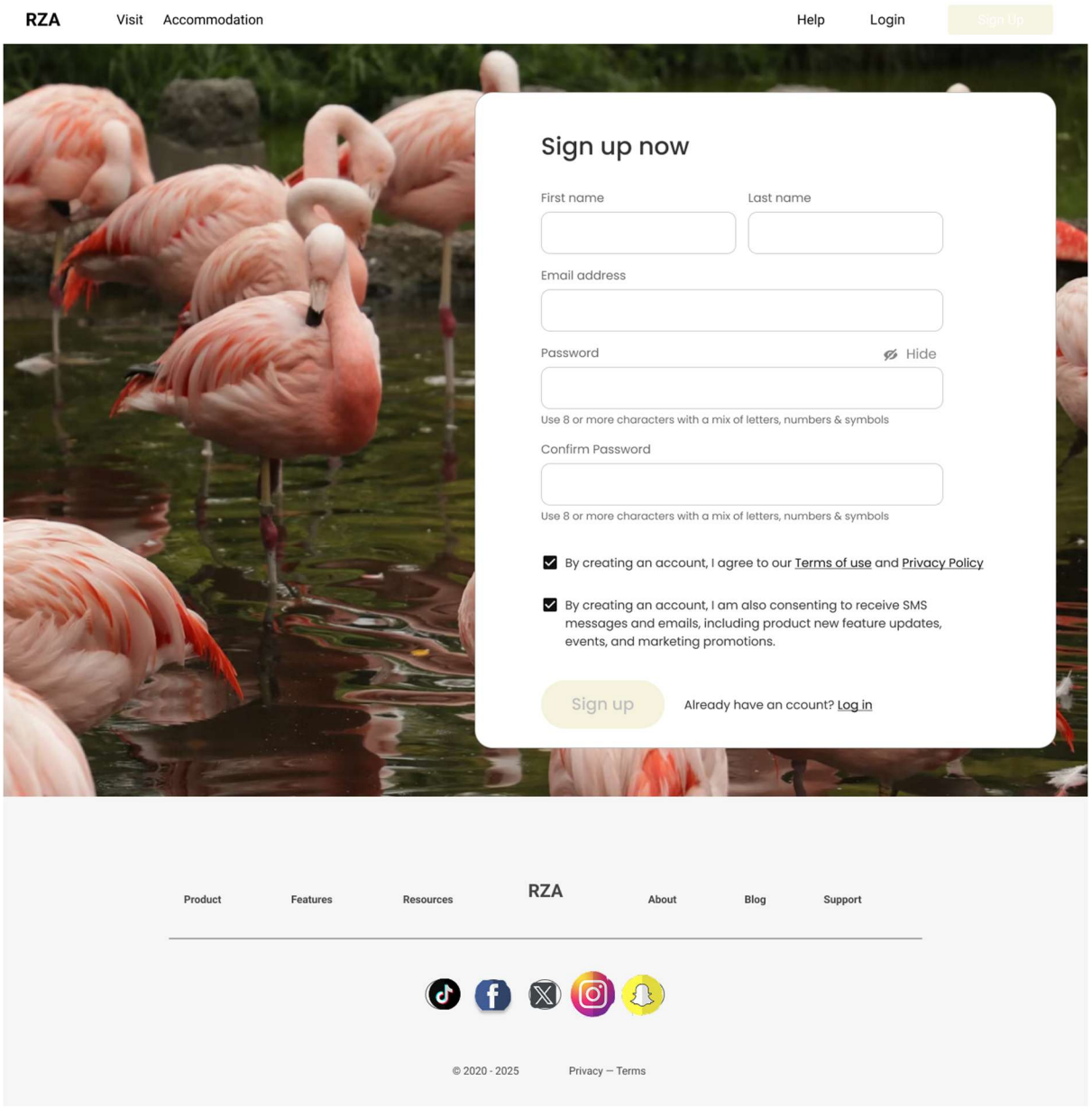
Home


RZA
Visit
Accommodation
Help
Login
Sign Up
Learn, Grow, Explore — with Riget Zoo Adventures
A modern platform designed to help visitors book tickets, manage their stay, and access personalised experiences.
BOOK A TICKET NOW!
About us
Riget Zoo Adventures
Riget Zoo Adventures is dedicated to providing a unique and memorable wildlife experience. Our digital platform allows visitors to explore exhibits, learn about animals, and plan their visit with ease. With improved booking features and a loyalty system, guests can enjoy a seamless experience from start to finish.
Product
Features
Resources
RZA
About
Blog
Support
© 2020 - 2025
Privacy – Terms

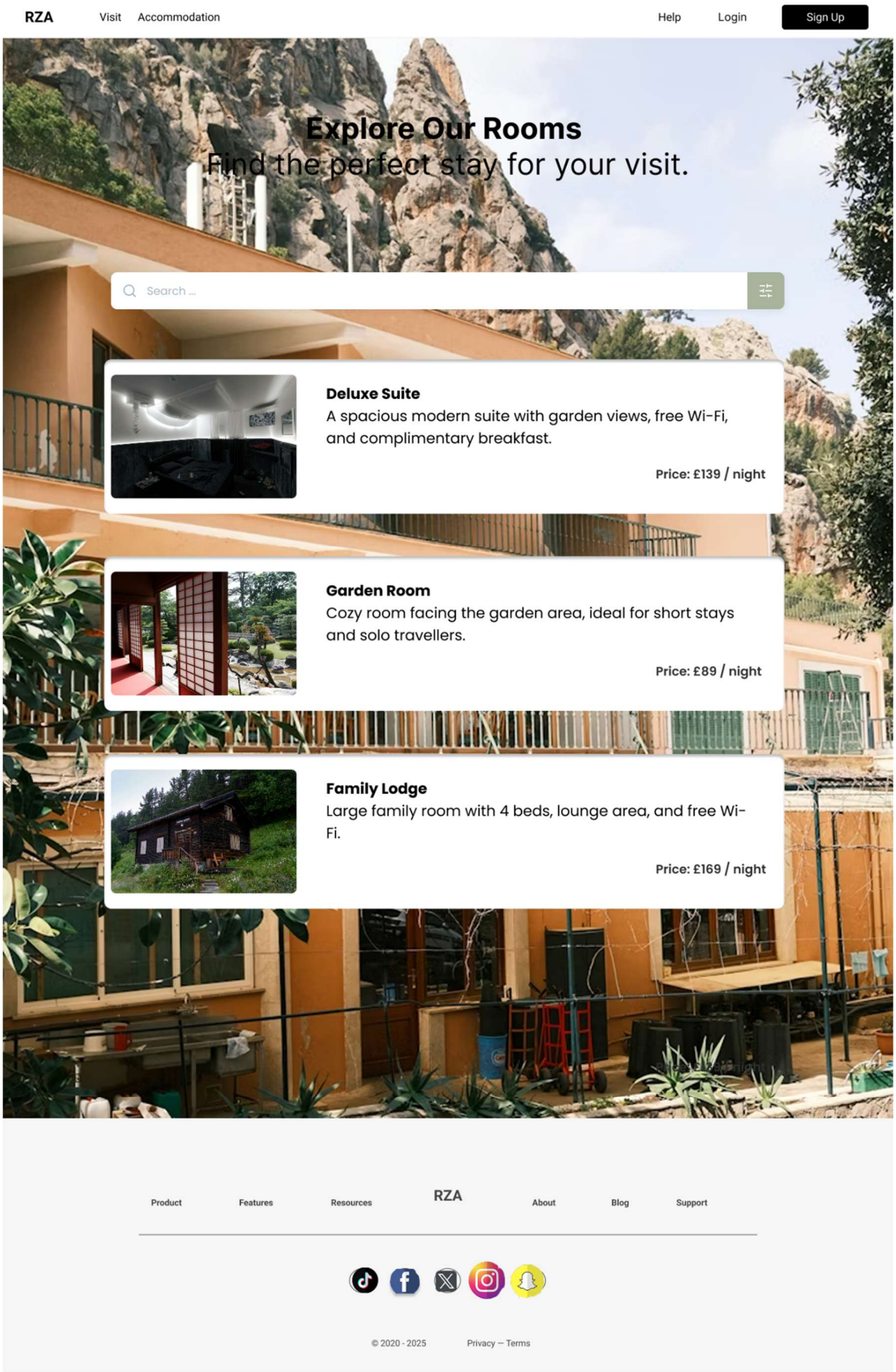
Login


RZA
Visit
Accommodation
Help
Login
Sign Up
Don't have an acount? Sign up
Sign in
Continue with Google
Continue with Twitter
OR
Email address
Your password
Hide
Forget your password
Sign in
Don't have an acount? Sign up
Product
Features
Resources
RZA
About
Blog
Support
© 2020 - 2025
Privacy – Terms

Register


RZA
Visit
Accommodation
Help
Login
Sign Up
Sign up now
First name
Last name
Email address
Password
Hide
Use 8 or more characters with a mix of letters, numbers & symbols
Confirm Password
Use 8 or more characters with a mix of letters, numbers & symbols
By creating an account, I agree to our Terms of use and Privacy Policy
By creating an account, I am also consenting to receive SMS messages and emails, including product new feature updates, events, and marketing promotions.
Sign up
Already have an ccount? Log in
Product
Features
Resources
RZA
About
Blog
Support
© 2020 - 2025
Privacy – Terms

Accommodations



Style Guide – Overview

To maintain a cohesive user experience, a consistent design system is applied:

- Colour palette selected using RealtimeColors
- Typography chosen for readability (recommended: Inter)
- Icons sourced from Lucide
- Visual spacing and components alignment remain uniform

Colour Scheme

Colours - Dark Mode

Neutral

True Black	Black	D.Grey	L.Grey	Grey-blue	Silver	White	True White
#000000	#151414	#2C2C2C	#8D8D8D	#D0E5F8	#E5E5E5	#ECE2EF	#FFFFFF

Branding

Primary	Secondary	Accent
#d9c731	#ddd99a	#9fb393

Color Shade

Primary Shade 1 #bfaf23	Primary Shade 2 #9e911d	Primary Shade 3 #8e811a	Primary Shade 4 #6c6314	Primary Shade 5 #4b450e
Secondary Shade 1 #948f32	Secondary Shade 2 #777228	Secondary Shade 3 #5a561e	Secondary Shade 4 #3c3a14	Secondary Shade 5 #1f1e0a
Accent Shade 1 #769166	Accent Shade 2 #647a56	Accent Shade 3 #516346	Accent Shade 4 #3e4c36	Accent Shade 5 #222a1d

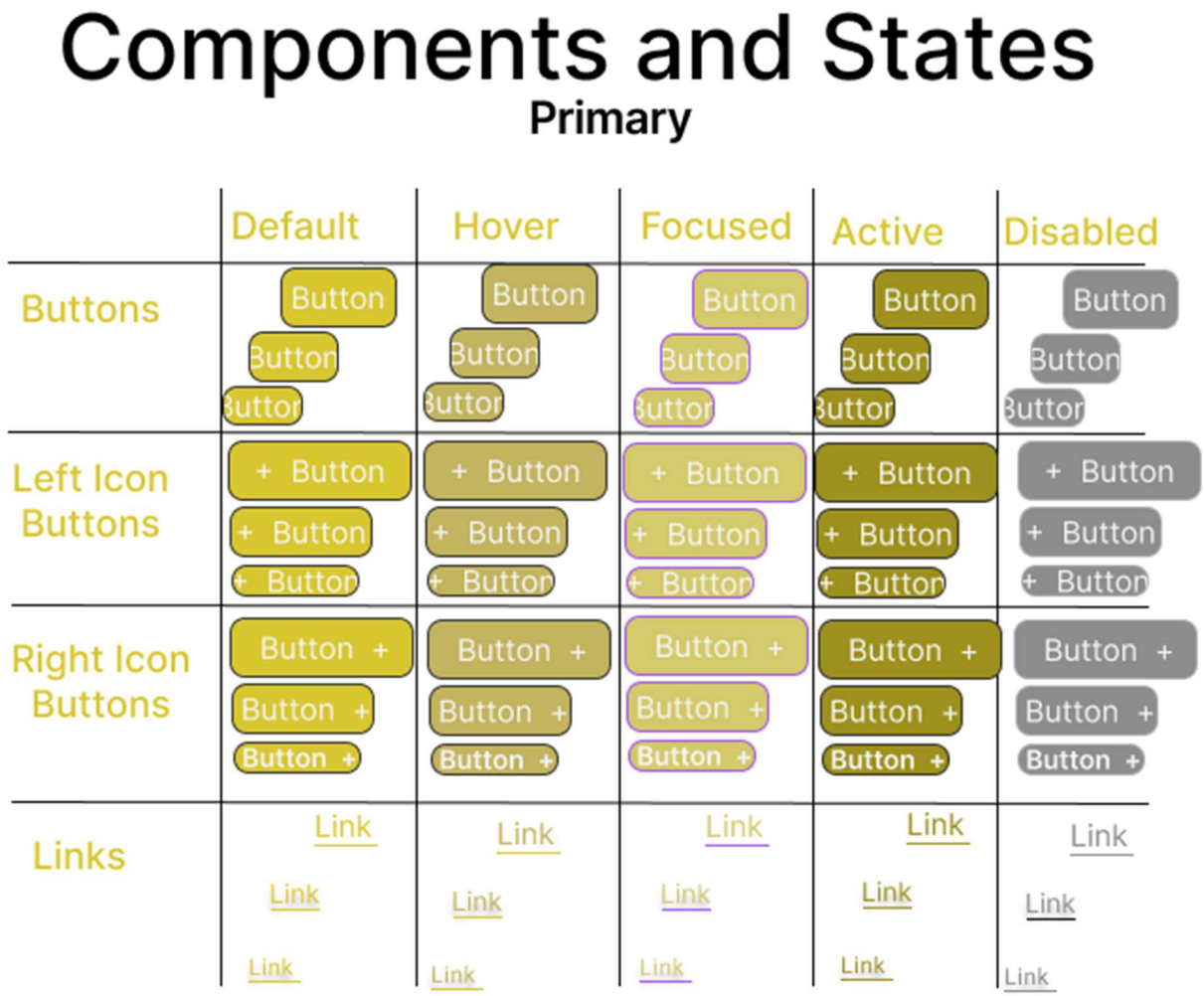
Color Tints

Primary Tint 1 #dfd052	Primary Tint 2 #e5d973	Primary Tint 3 #ebe294	Primary Tint 4 #f5f0c6	Primary Tint 5 #fefdf7
Secondary Tint 1 #bfb842	Secondary Tint 2 #cec86e	Secondary Tint 3 #ddd99a	Secondary Tint 4 #f1efd5	Secondary Tint 5 #fbfaf2
Accent Tint 1 #8ba37c	Accent Tint 2 #a9bb9f	Accent Tint 3 #becbb6	Accent Tint 4 #dce4d8	Accent Tint 5 #f1f4ef

Action

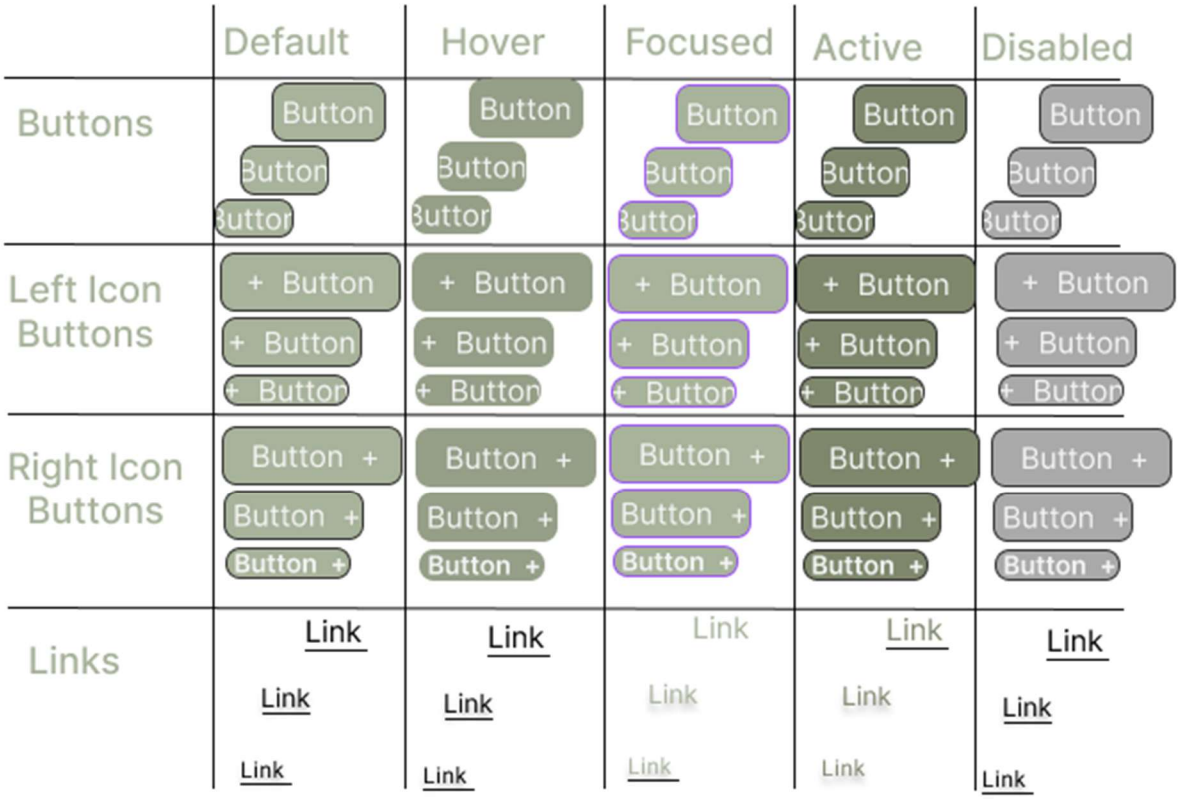
Warning	Error	Success	Info
#FACC15	#EF4444	#22C55E	#3B82F6

Components


Components and States
Primary
Default
Hover
Focused
Active
Disabled
Buttons
Left Icon Buttons
Right Icon Buttons
Links
Button
+ Button
Button +
Link

Components and States

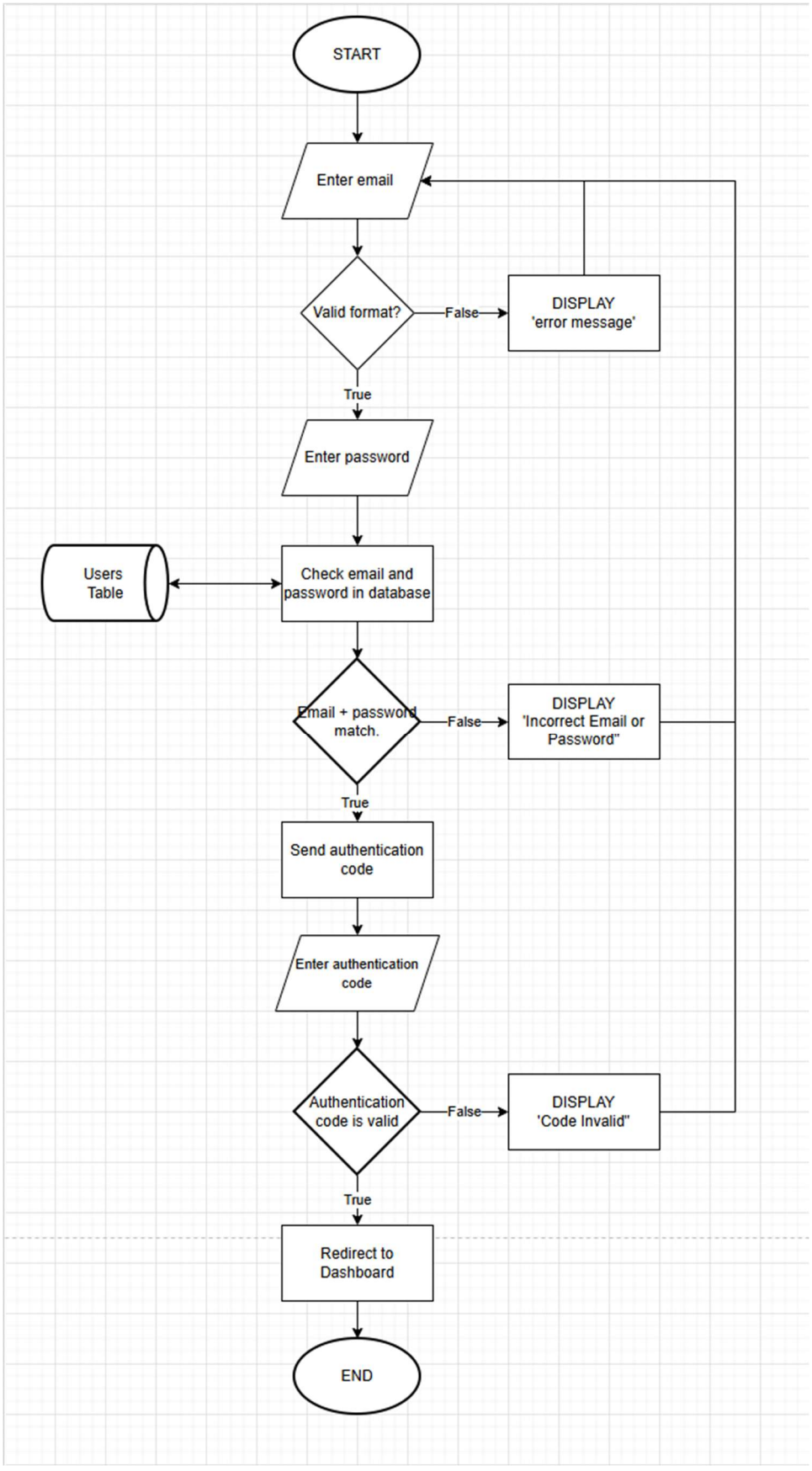
Secondary



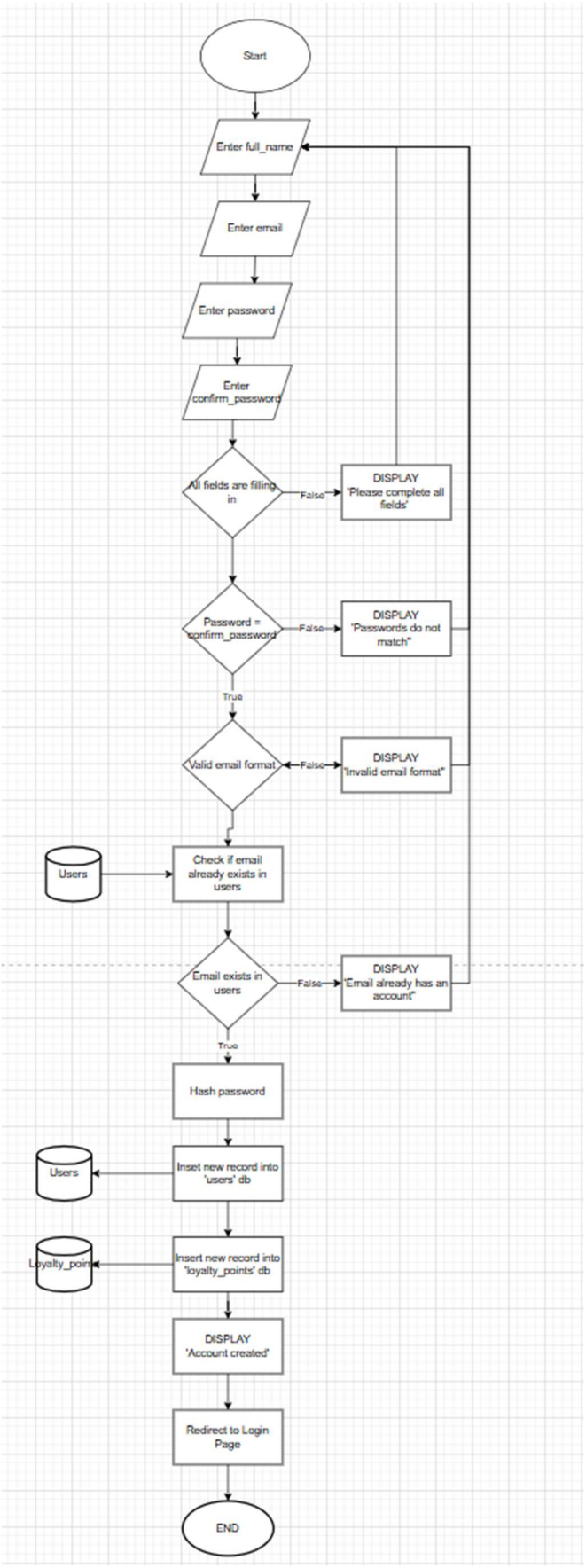
Algorithms/Flowchart

Flowcharts represent logic of major process. Created using diagrams.net.

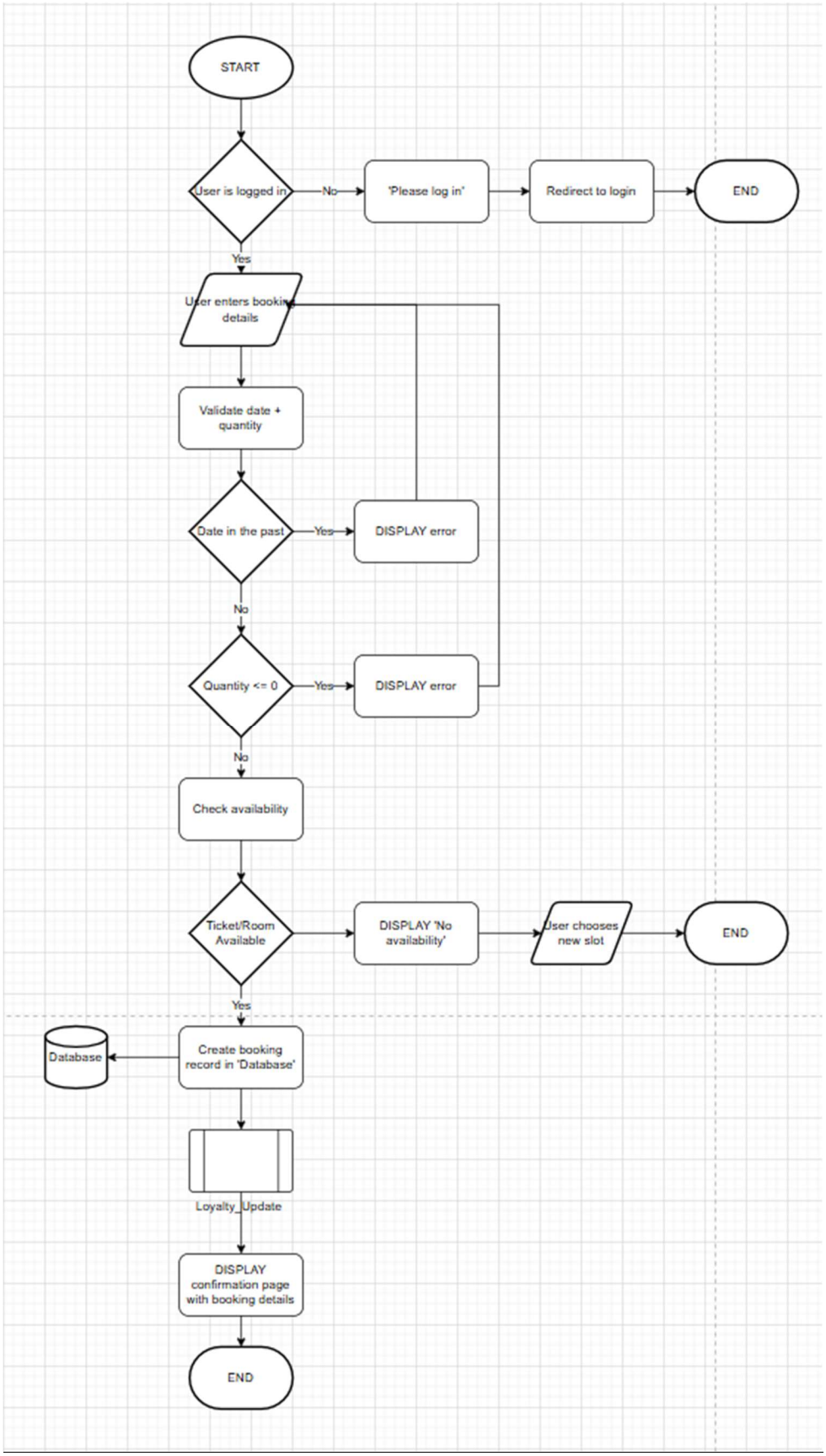
Login Flowchart



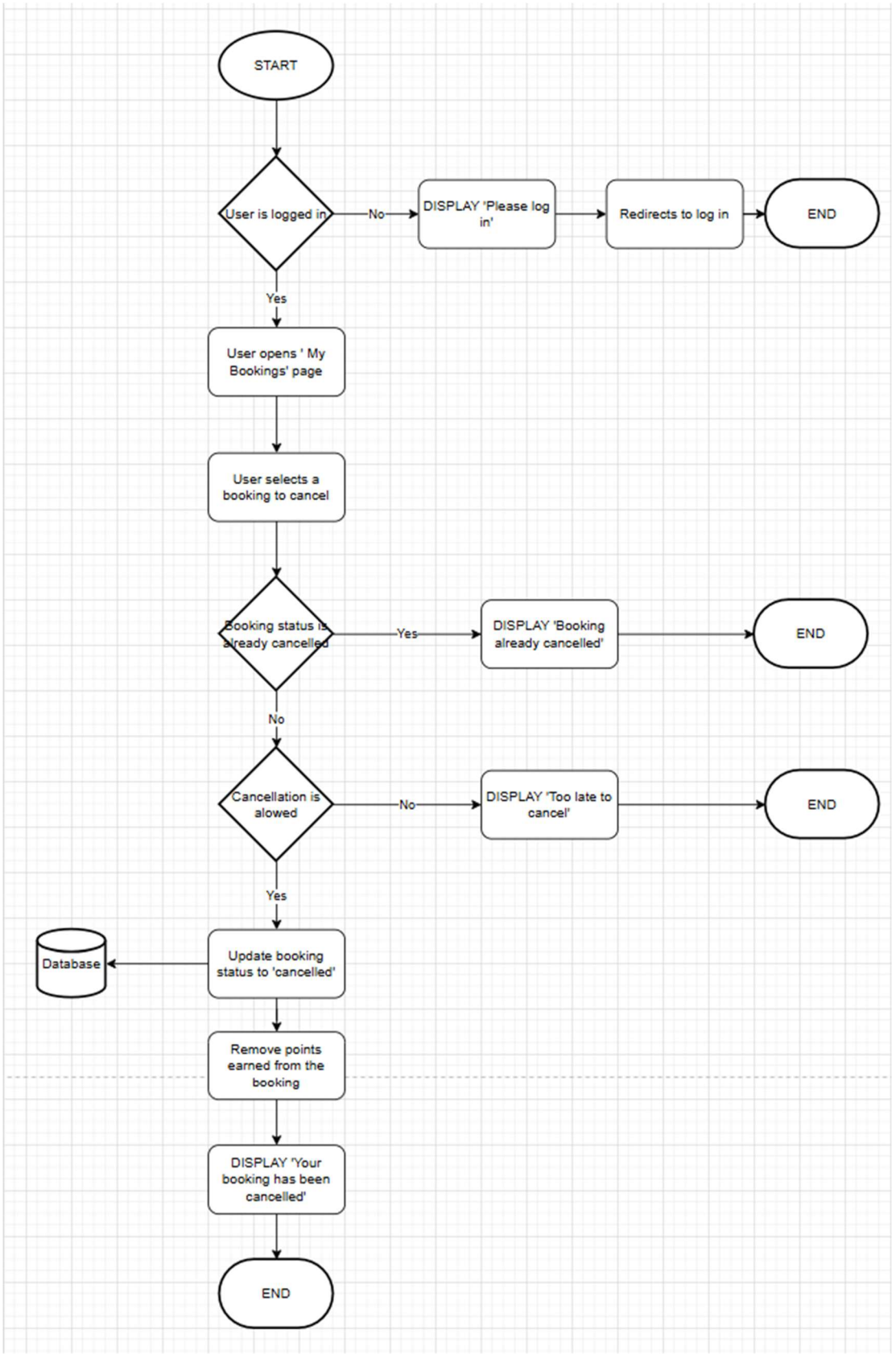
Registration Flowchart



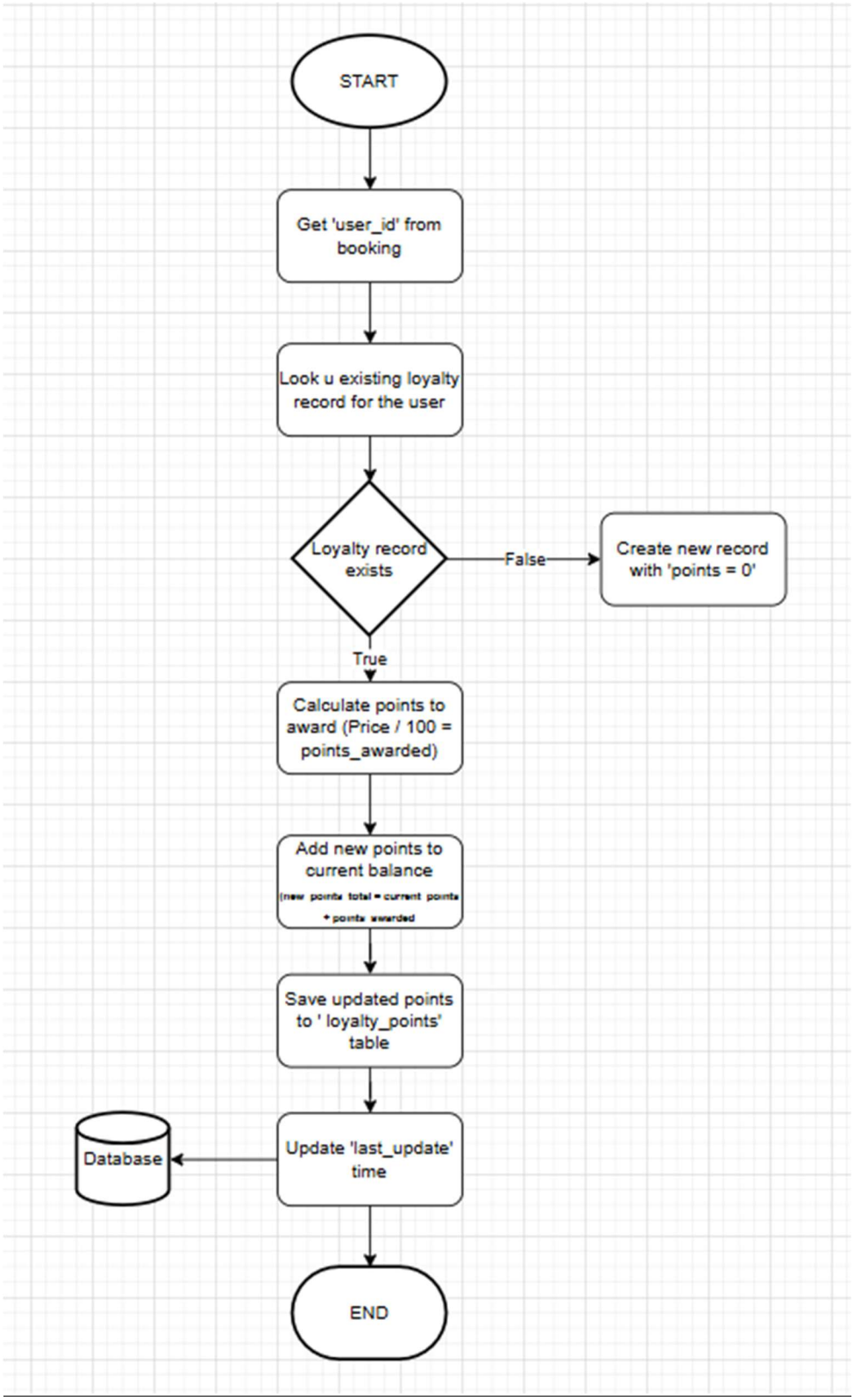
Booking (Hotel / Ticket) Flowchart



Cancel Booking Flowchart



Loyalty Update Flowchart



Data Requirements (UI-Level)

This section describes the data required by each page for proper UI operation.

Home Page

- **navigation_links** – labels for Home, Login, Register, View Rooms, View Tickets, etc

Login Page

- email_input
- password_input
- error_message
- remember me
- login_button_state

Register Page

- full_name_input
- email_input
- password_input
- confirm_password_input
- validation_message

Dashboard

- user_name
- loyalty_point_balance
- upcoming_bookings_list
- recent_activity_list
- quick_actions

Test Plan - Overview

For the successful implementation of the proposed features for the proposed solution, various types of data and methodologies will need to be used depending on what features is being tested, what data is being used to test, what data the feature can access and how the feature is implemented.

For testing of the data inputs within the application, various data types will be used to test the inputs and ensure they operate properly without any internal errors; while giving appropriate textual errors/alerts to the user so they can fix their inputs.

- **Normal data** – valid input that should be accepted
- **Erroneous data** – invalid input used to check validation works.
- **Boundary data**- values on the edge of the limited (e.g. 999, 1001,1001).
- **Extreme data** – very large or unusual values to test robustness

There are also various methods of testing the application that can be further used to ensure that the application works as intended and is secure in its operations. Below is a list of different types of testing methodologies and what their use cases are for.

- **Functional testing** - checks features work as intended
- **Performance / Stress testing** - check behaviour under heavy load
- **Security test** – check for vulnerabilities like SQL injection / XSS

Sample Tests

Description of test	Test data to be used (if required)	Expected outcome	Actual outcome	Comments and intended actions
Login (normal data)	user@example.com password123!	Login successful. Redirected to dashboard	To be completed during implementation.	
Login (erroneous data)	Abc123 Password	Login unsuccessful. Error message: “Invalid format”.	To be completed during implementation.	Ensure email field has correct validation.
Register (Boundary test)	**Password:** Abcdef67 (8 characters)	Account created successful. Redirect to login page	To be completed during implementation.	Check validation matches minimum password requirement.

Hotel Booking (Normal Data)	**Check-in:** 2025-06-10 **Check-out:** 2025-06-12 **Guests:** 2	Booking created, total_price displayed, appears in my bookings	To be completed during implementation.	
Ticket Booking (Erroneous Data)	**Quantity:** 0	Error message, 'Quantity must be greater than 0'	To be completed during implementation.	Confirm validation prevents illogical quantities.
Loyalty Points Update	**Booking total:** £100	Points increase by 100+	To be completed during implementation.	Ensure all loyalty calculations are applied consistently.