

StayChecked

Accommodation Management System

-PROTOTYPE-

Last Updated at: 13/07/2025

SYSTEM GUIDE & DOCUMENTATIONS

Table Of Contents

1. Overview.....	3
1.1. Main Features.....	3
1.2. Application Flow.....	3
Guest User.....	3
Accommodation User.....	5
System Admin User.....	6
2. Test Documentation Production-Side.....	7
2.1. Feature/System Module Test.....	8
General Systems.....	8
Accommodation Verification - System Admins.....	15
Booking Verification - User.....	16
Support Ticket Management - Guests and Accommodations.....	19
Lookup System - (View Only).....	24
RAG-Powered AI Assistance.....	26
2.2. System Logging Processes.....	27
Terminal-Based Logging Processes.....	27
File-Based Logging Processes.....	27
3. System Documentation Development-Side.....	28
3.1. System Overview and Architecture.....	28
3.2. Core Components.....	29
Data Management (DataStore & DAOs).....	29
Main Data Flow Processes.....	29
Application Entry Point.....	30
User Authentication.....	31
Booking Management.....	32
Ticket Support System.....	33
RAG AI-Powered Chatbot.....	34

1. Overview

StayChecked is a comprehensive JavaFX-based accommodation booking and support management system designed to facilitate interactions between Guests and Accommodations. The application provides a centralized platform for booking verification, support ticket management, and accommodation oversight with an integrated AI-powered help system.

1.1. Main Features

- User Management System
- Booking Management & Verification
- Support Ticket System
- AI-Powered Help System through RAG Approach

1.2. Application Flow

1.2.1. Guest User – Step-by-Step Feature Flow

A. Login / Registration

- A new user can register as a Guest by filling in personal information.
- After registering, they can log in using their email and password.

B. Access Guest Dashboard

- Upon login, the user is redirected to the Main Landing Page with access to guest-specific features.

C. Booking Verification

- The user can verify a booking by entering the Booking ID and their last name.
- If the details are correct, the system confirms the booking and displays it in the booking list.
- Users can view the status of each booking (verified, closed).

D. Submit Support Tickets

- Guests can create new support tickets linked to a booking. To do so, they must:
 - Select the relevant booking
 - Choose a support category
 - Write the subject and inquiry description
- The system validates the form before submission.

E. Manage Existing Tickets

- Guests can view open and closed tickets.
- For open tickets, they can send follow-up messages or close the ticket once resolved.
- Closed tickets remain viewable but are no longer editable.

F. Browse Accommodations (View Only)

- Guests can access a list of all approved accommodations, including details like:
 - Accommodation Name
 - Email, Phone Number
 - Address

G. Use AI-Powered Help System & FAQs

- The guest can go to the Help Page to:
 - View FAQs
 - Interact with the AI Chatbot, which provides contextual responses using RAG (Retrieval-Augmented Generation) technology.

1.2.2. Accommodation User – Step-by-Step Feature Flow

A. Login / Registration

- A user can register as an Accommodation, entering extra verification details.
- Their account is pending verification by a System Admin.
- Once approved, they can log in to access their dashboard.

B. Access Accommodation Dashboard

- After login, the accommodation user is taken to the Main Landing Page, where they can access provider-specific features.

C. View Accommodation Information

- Users can view their own accommodation data, such as:
 - Booking ID & Room number
 - Guest last name
 - Approval Status

D. View Guest List

- Accommodation users can view a list of guests who booked their property, including:
 - Guest ID & Name
 - Email and Phone Number
- This helps with communication and guest management.

E. Manage Support Tickets

- Accommodations can view and respond to support tickets submitted by guests.
- For each ticket, they can:
 - Read and reply to messages
 - View status (Open/Closed)
 - Track conversation history

F. Use AI-Powered Help System

- The Help Page includes an AI Chatbot for asking questions and retrieving system help.
- FAQs are also available for quick reference.

1.2.3. System Admin User – Step-by-Step Feature Flow

A. Login

- Admins do not register through the interface. They use predefined credentials to log in.

B. Access Admin Dashboard

- After logging in, the Admin is taken to the Main Admin Panel, which provides access to admin-level controls.

C. Verify Accommodations

- Admins can view newly registered accommodations and:
 - Review their registration details
 - Approve or reject the application
- Only approved accommodations become visible to guests.

2. Test Documentation | Production-Side

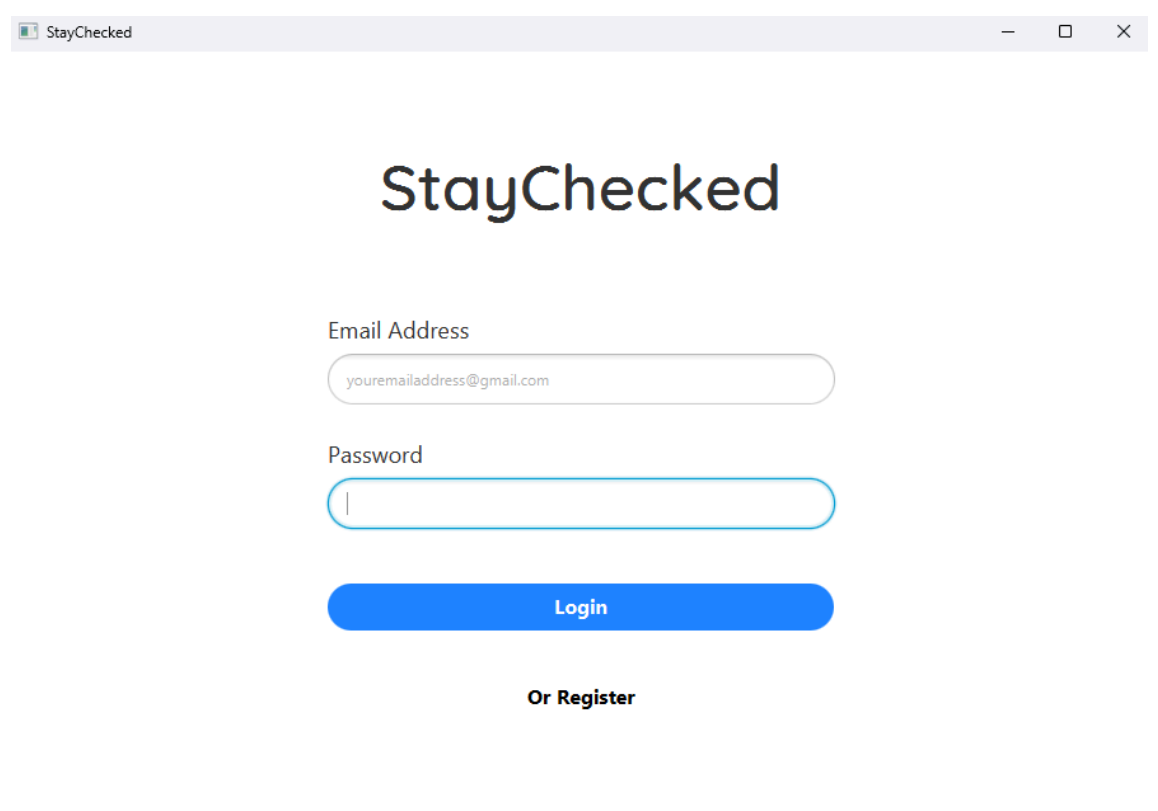
The following section is comprised of test documentation details for the production-ready prototype. It showcases the functionality and responses on major features, illustrating how users will be able to use the main features of the app as well as the main behavior of it.

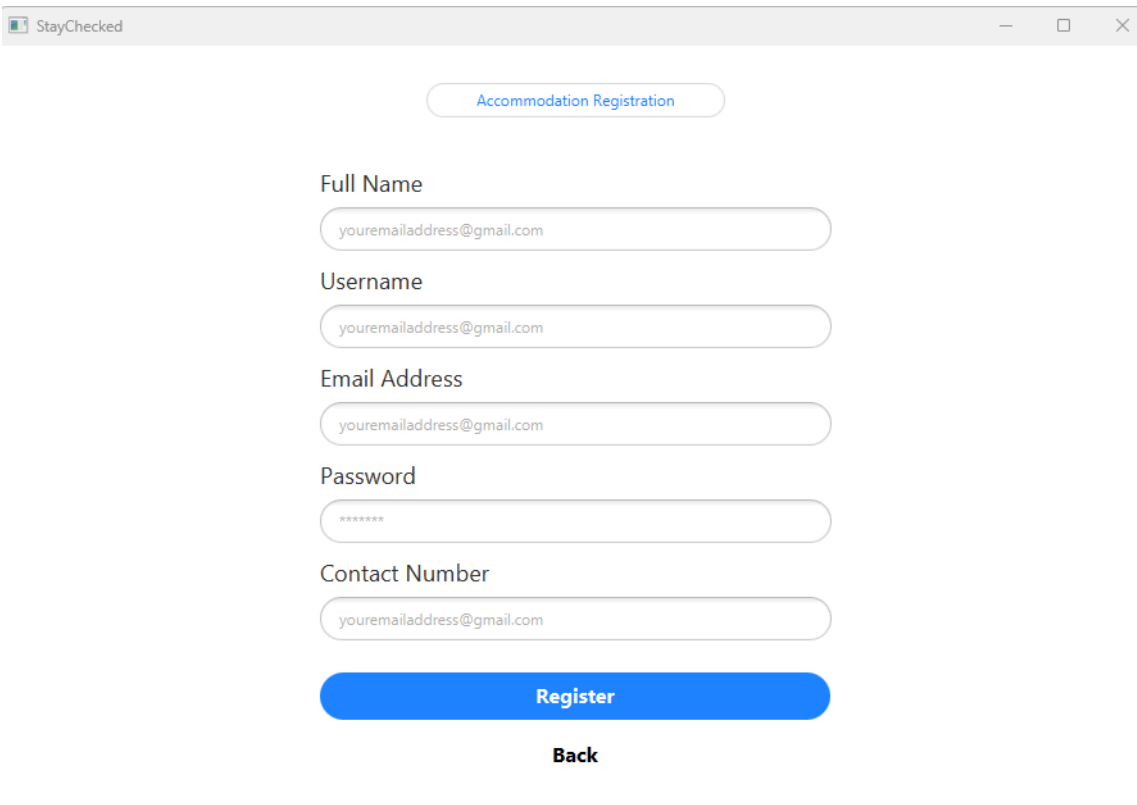
The section is split into two different test sections, comprising the Feature/System Module Tests and the System Logging Processes, showing how each major feature behaves as well as the effort made to optimize bug-finding and the debugging processes for development

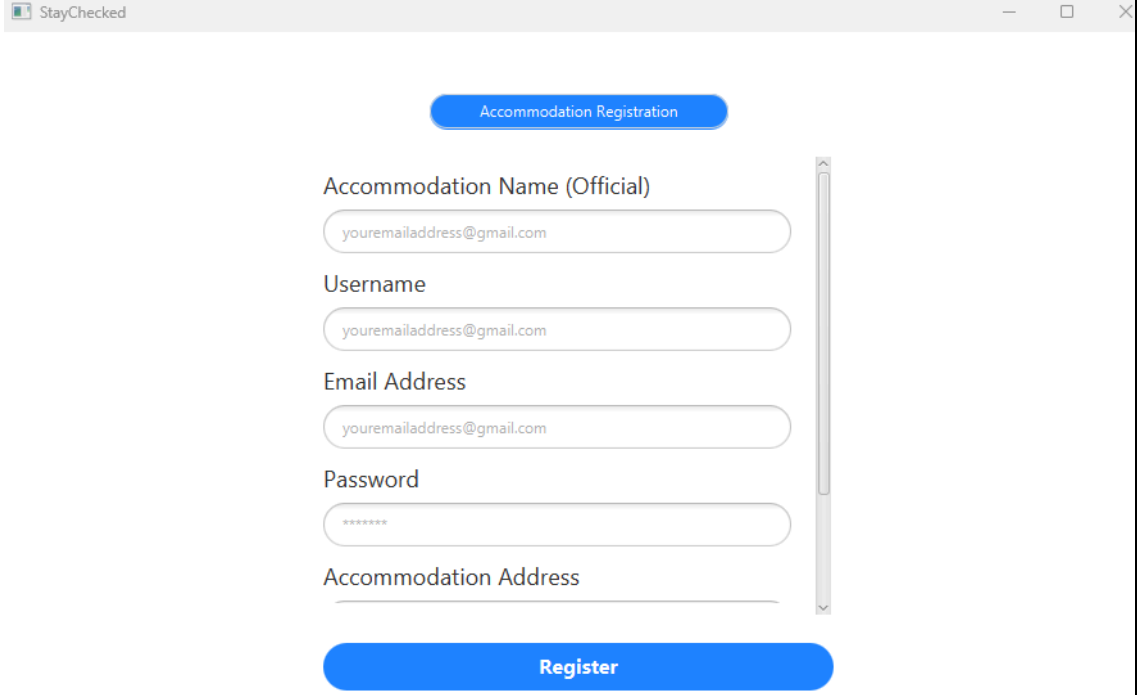
2.1. Feature/System Module Test

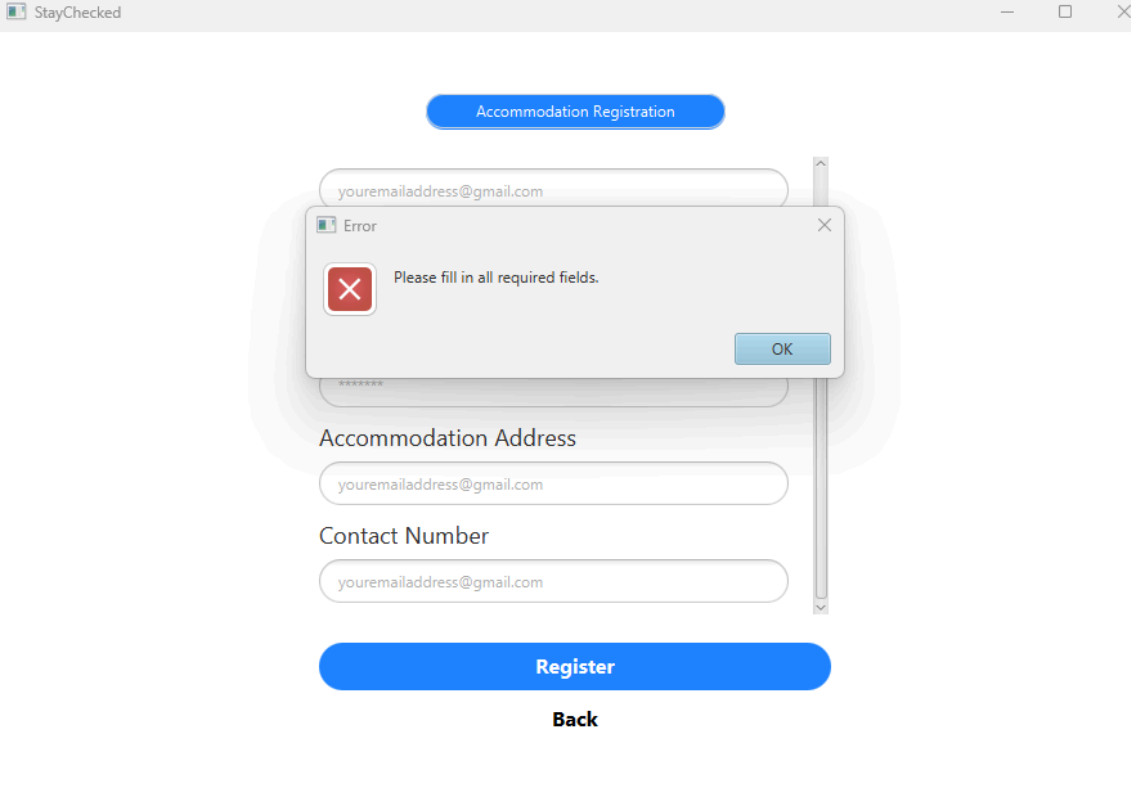
The following pages will showcase the practical capability of the app and how the Application Flow is designed to work, starting from Authentication Processes, App Navigation, and how each major component will behave and affect one another.

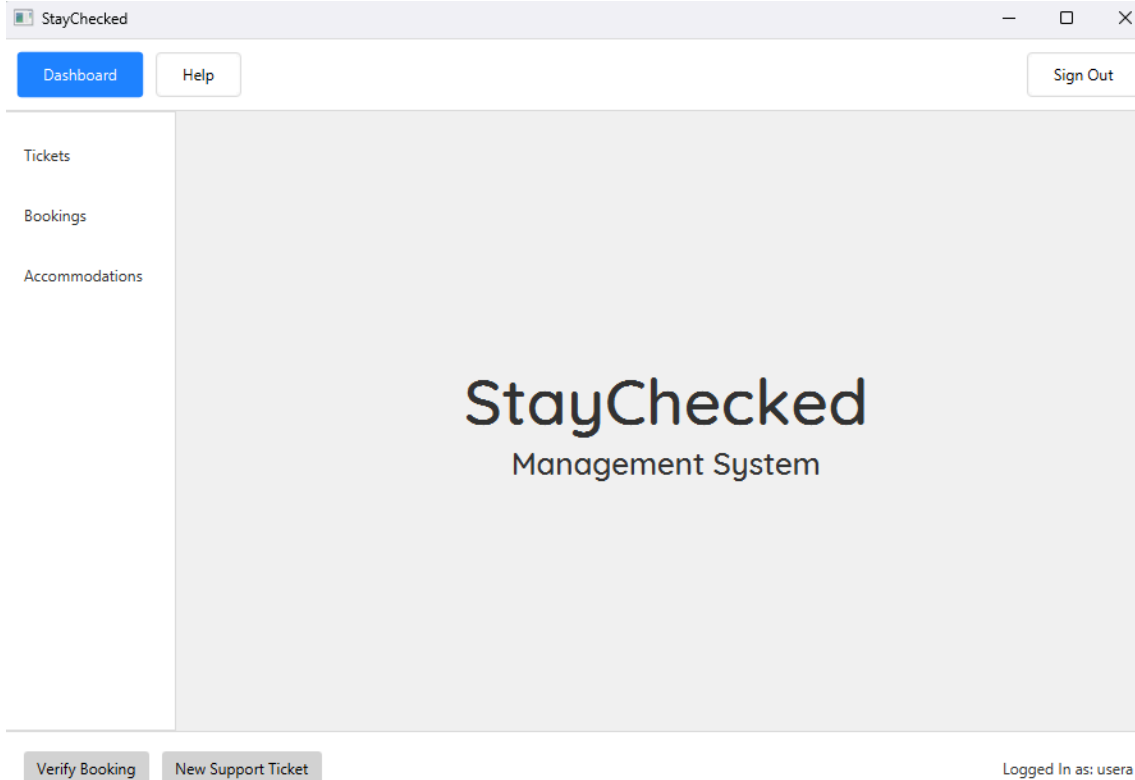
General Systems

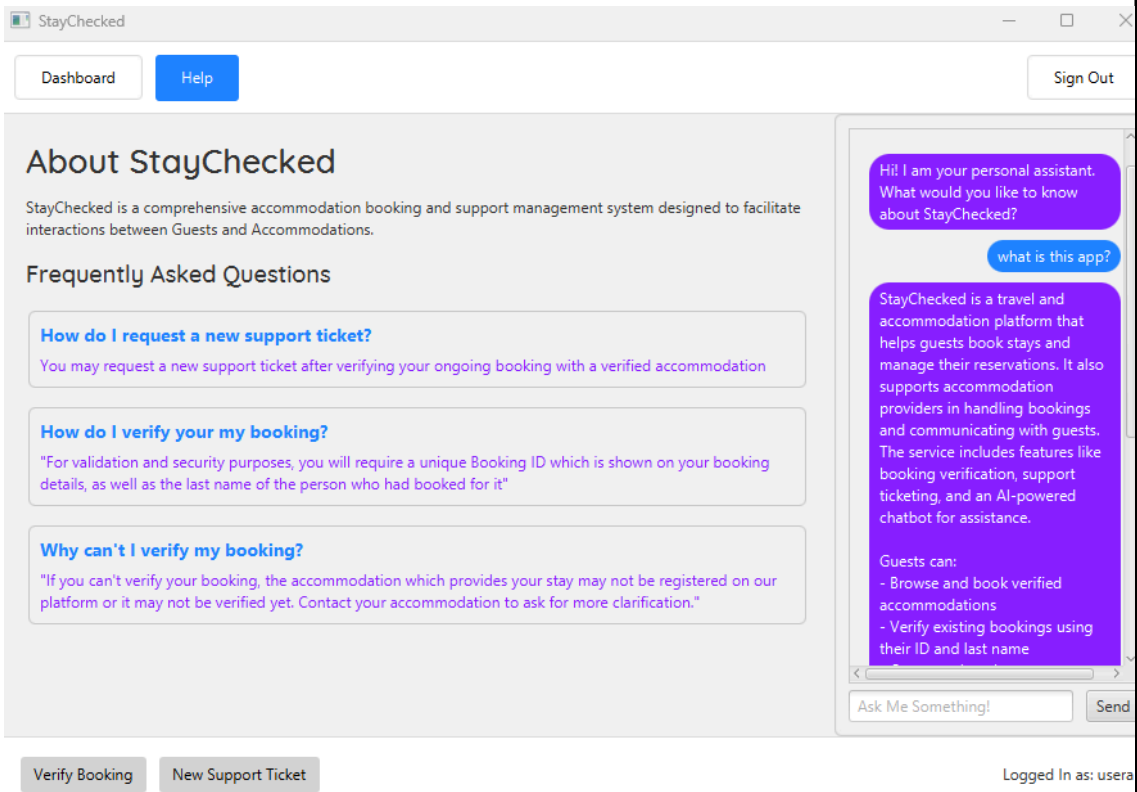
ID // Feature	Demonstration	Details/Remarks
1.0 User Login		The Login Page is the interface shown every time the user launches the app through the launcher.

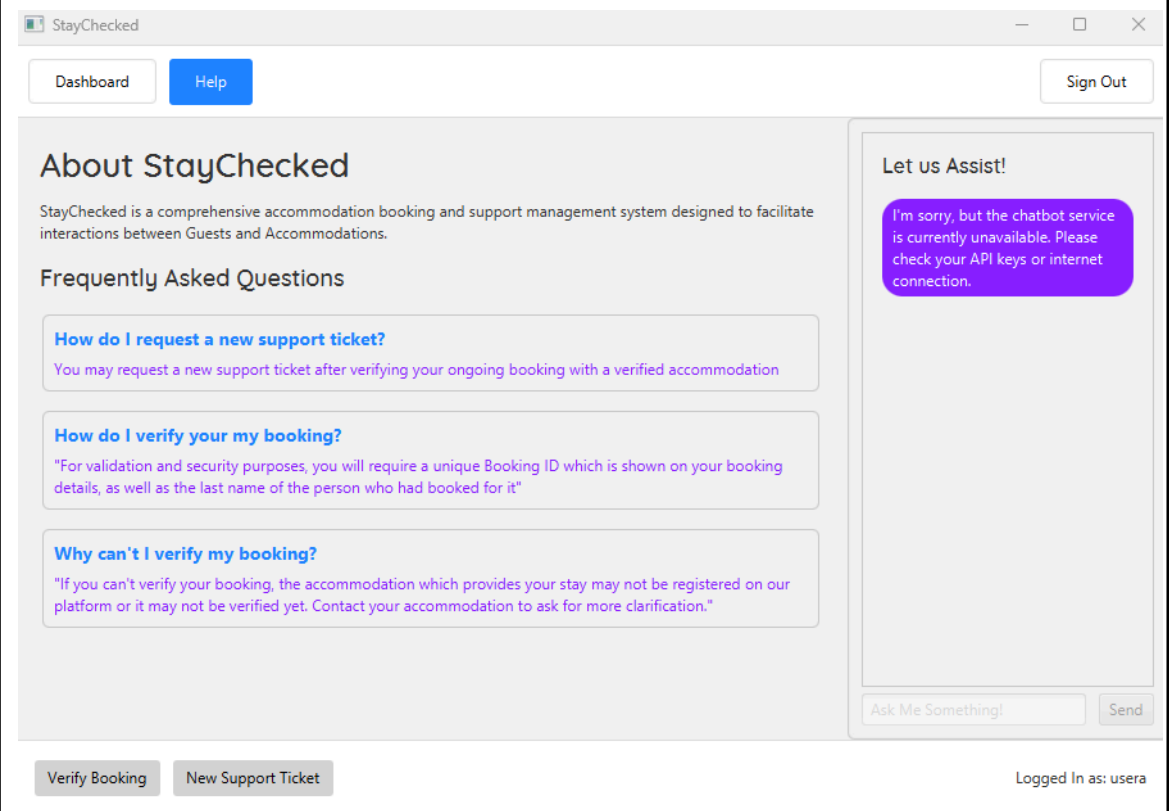
ID // Feature	Demonstration	Details/Remarks
<p>1.1 User Registration (Guests)</p>	 <p>The screenshot shows a web browser window titled 'StayChecked'. Inside, there is a section titled 'Accommodation Registration'. Below this title are five input fields: 'Full Name', 'Username', 'Email Address', 'Password', and 'Contact Number'. Each field contains the placeholder text 'youremailaddress@gmail.com'. The 'Password' field has asterisks for masking. At the bottom of the form is a large blue button labeled 'Register' and a smaller black text link labeled 'Back'.</p>	<p>Guests who are not registered on the platform yet may fill out the registration form to register themselves</p>

ID // Feature	Demonstration	Details/Remarks
1.2 User Registration (Accommodations)	 <p>Accommodation Registration</p> <p>Accommodation Name (Official)</p> <p>youremailaddress@gmail.com</p> <p>Username</p> <p>youremailaddress@gmail.com</p> <p>Email Address</p> <p>youremailaddress@gmail.com</p> <p>Password</p> <p>*****</p> <p>Accommodation Address</p> <p>Register</p> <p>Back</p>	Accommodations have a slightly different interface due to the necessary details they need to input for verification purposes

ID // Feature	Demonstration	Details/Remarks
<p>1.3 Invalid Forms</p> <p><i>Error Handling</i></p>	 <p>The screenshot shows a web browser window titled 'StayChecked'. It displays a registration form with the following elements:</p> <ul style="list-style-type: none"> A blue button labeled 'Accommodation Registration' at the top. An input field containing 'youremailaddress@gmail.com'. An 'Error' dialog box in the center with a red 'X' icon and the message 'Please fill in all required fields.' and an 'OK' button. An input field containing '*****' (password). A label 'Accommodation Address' above an input field containing 'youremailaddress@gmail.com'. A label 'Contact Number' above an input field containing 'youremailaddress@gmail.com'. A blue button labeled 'Register' at the bottom. A 'Back' link below the 'Register' button. 	<p>For all the authentication interfaces, upon clicking the submission buttons, the data will be processed and return an error dialog when invalid.</p>

ID // Feature	Demonstration	Details/Remarks
2.0 Main Landing Page		<p>After all Authentication Processes are done, successful authentication will be redirected to this page.</p> <p><i>For all users, the page appearance will be similar, with differences only in the selection of features they may be able to navigate to</i></p>

ID // Feature	Demonstration	Details/Remarks
2.1 Help Page		<p>The Help Page consists of a short overview of StayChecked, with a list of FAQs, and a private AI-Powered ChatBot integrated to it.</p>

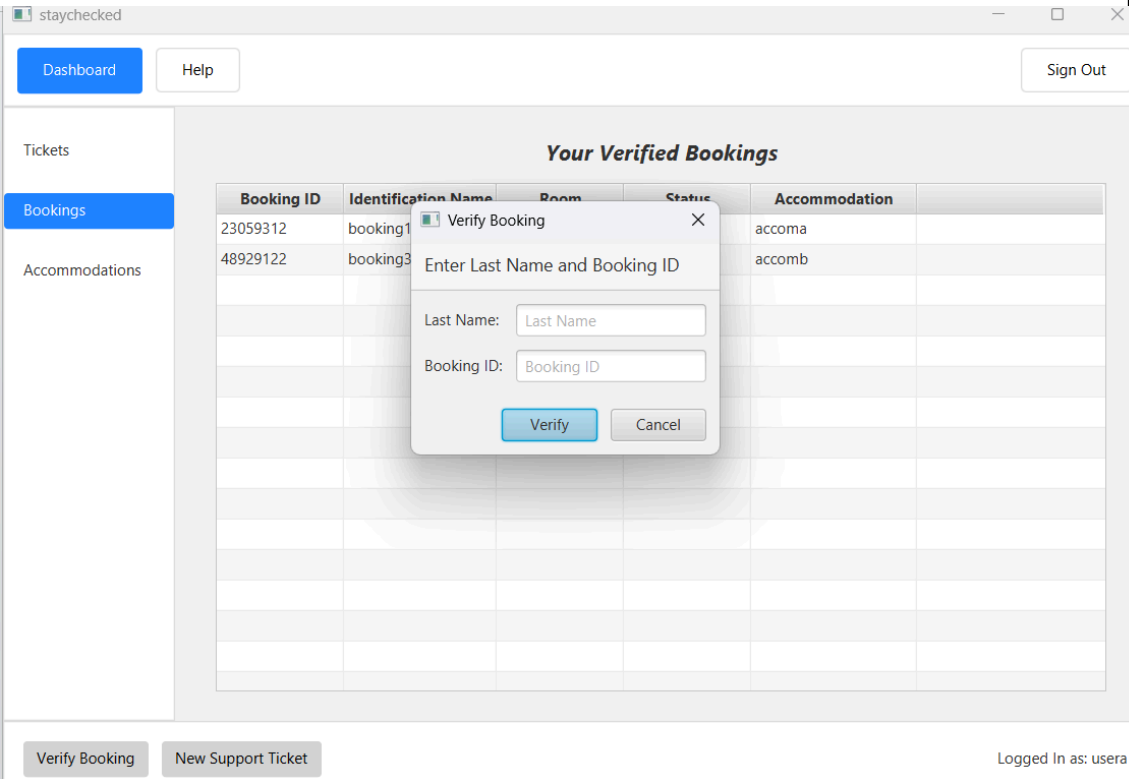
ID // Feature	Demonstration	Details/Remarks
<p>2.1.1 Help Page with disabled Chatbot Integration</p> <p><i>Error Handling</i></p>	 <p>The screenshot shows a web application window titled 'StayChecked'. At the top, there are buttons for 'Dashboard', 'Help' (highlighted in blue), and 'Sign Out'. The main content area is titled 'About StayChecked' and describes the system as a comprehensive accommodation booking and support management system. Below this is a 'Frequently Asked Questions' section with three items: 'How do I request a new support ticket?', 'How do I verify my booking?', and 'Why can't I verify my booking?'. Each item has a brief description. On the right side, there is a chatbot interface titled 'Let us Assist!'. A purple message bubble states: 'I'm sorry, but the chatbot service is currently unavailable. Please check your API keys or internet connection.' At the bottom of the chatbot interface is a text input field with the placeholder 'Ask Me Something!' and a 'Send' button. At the bottom of the page, there are buttons for 'Verify Booking' and 'New Support Ticket', and a status indicator 'Logged In as: usera'.</p>	<p>If, by any chance an error occurs to the ChatBot System, the page will still load with a disabled chatbot interface as part of error handling</p>

Accommodation Verification - System Admins

ID // Feature	Demonstration	Details/Remarks																
3.0 Accommodation Management System	<div><div>StayChecked</div><div><div>Dashboard</div><div>Help</div><div>Sign Out</div></div><div><div>Verifications</div><div><div>Registered Accommodations</div><table><thead><tr><th>Accommodation Name</th><th>Email</th><th>Status</th><th>Address</th></tr></thead><tbody><tr><td>AccommodationA</td><td>accoma@gmail.com</td><td>Verified</td><td>AccommodationALoc</td></tr><tr><td>AccommodationC</td><td>accmc@gmail.com</td><td>Verified</td><td>AccommodationCLoc</td></tr><tr><td>AccommodationB</td><td>accomb@gmail.com</td><td>Verified</td><td>AccommodationBLoc</td></tr></tbody></table><div><div>Accommodation Details</div><div>Accommodation Name: AccommodationA</div><div>Email Address: accoma@gmail.com</div><div>Address: AccommodationALoc</div><div>Status: Verified</div><div>Close</div></div></div></div><div>Logged In as: admina</div></div>	Accommodation Name	Email	Status	Address	AccommodationA	accoma@gmail.com	Verified	AccommodationALoc	AccommodationC	accmc@gmail.com	Verified	AccommodationCLoc	AccommodationB	accomb@gmail.com	Verified	AccommodationBLoc	<p>Accommodation Management System is a feature only accessible by System Admins.</p> <p>When a new Accomodation registers themselves, System Admins will need to verify them before they are officially registered on the application (visible to Guests)</p>
Accommodation Name	Email	Status	Address															
AccommodationA	accoma@gmail.com	Verified	AccommodationALoc															
AccommodationC	accmc@gmail.com	Verified	AccommodationCLoc															
AccommodationB	accomb@gmail.com	Verified	AccommodationBLoc															

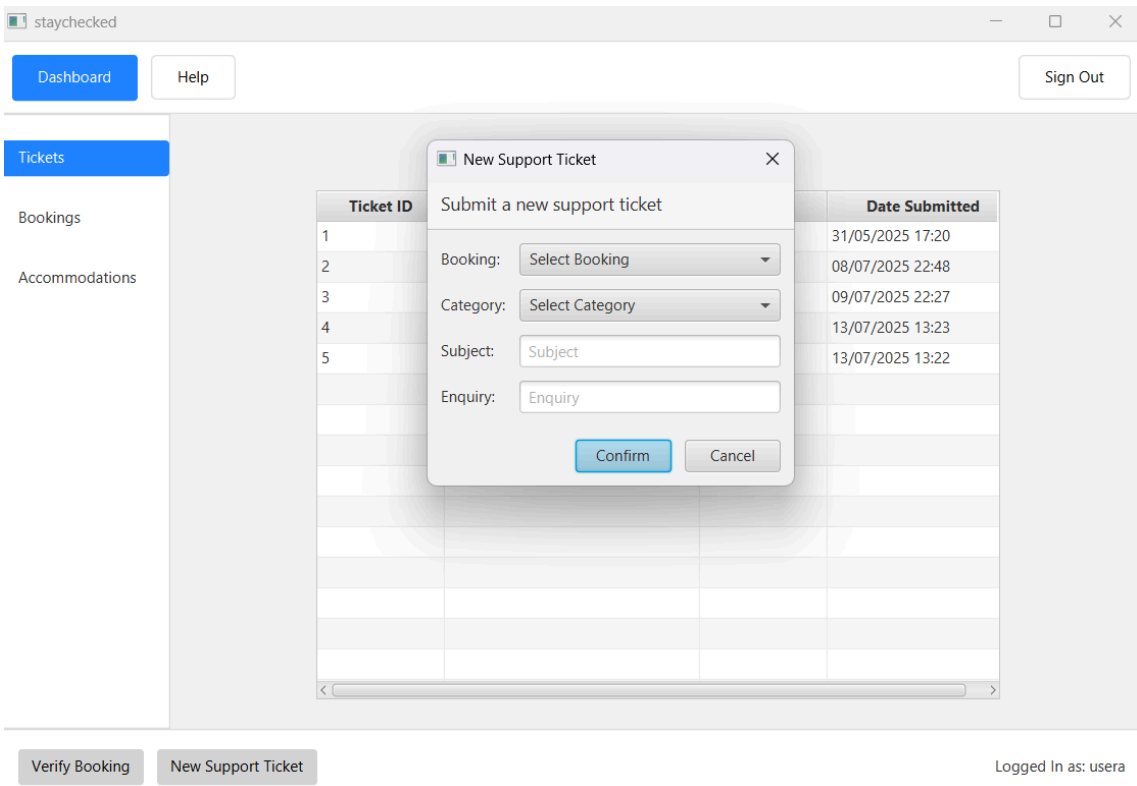
Booking Verification - User

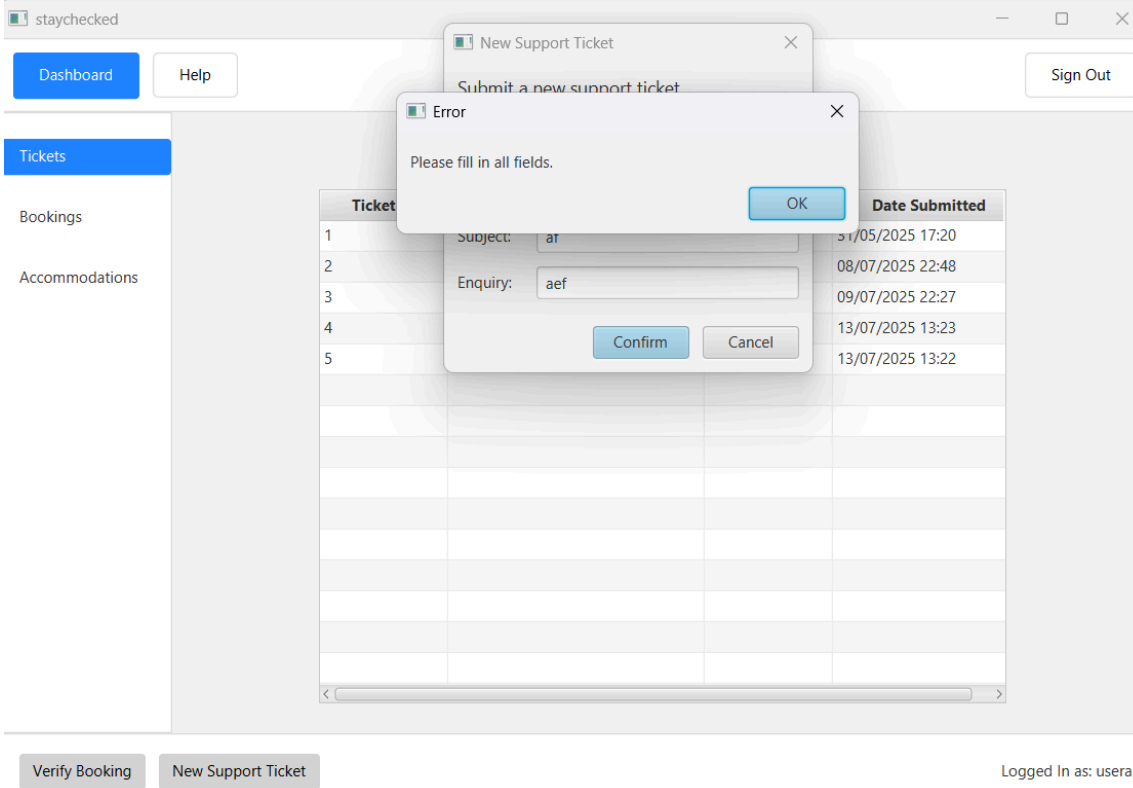
ID // Feature	Demonstration	Details/Remarks																																																																														
3.0 Booking Lists	<div><div>staychecked</div><div><div>Dashboard</div><div>Help</div><div>Sign Out</div></div><div><div>Tickets</div><div>Bookings</div><div>Accommodations</div></div><div><div><div>Your Verified Bookings</div><table><tr><th>Booking ID</th><th>Identification Name</th><th>Room</th><th>Status</th><th>Accommodation</th><th></th></tr><tr><td>23059312</td><td>booking1In</td><td>a11</td><td>Verified</td><td>accoma</td><td></td></tr><tr><td>48929122</td><td>booking3In</td><td>a2</td><td>Closed</td><td>accomb</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr></table></div><div><div>Verify Booking</div><div>New Support Ticket</div></div><div>Logged In as: usera</div></div></div>	Booking ID	Identification Name	Room	Status	Accommodation		23059312	booking1In	a11	Verified	accoma		48929122	booking3In	a2	Closed	accomb																																																														<p>The Booking Lists page displays a summary of all reservations that the user has previously verified. Each record displays the booking ID, associated identification name, room, status (e.g., verified, closed), and associated accommodation. This helps users track their bookings and statuses within the platform.</p>
Booking ID	Identification Name	Room	Status	Accommodation																																																																												
23059312	booking1In	a11	Verified	accoma																																																																												
48929122	booking3In	a2	Closed	accomb																																																																												

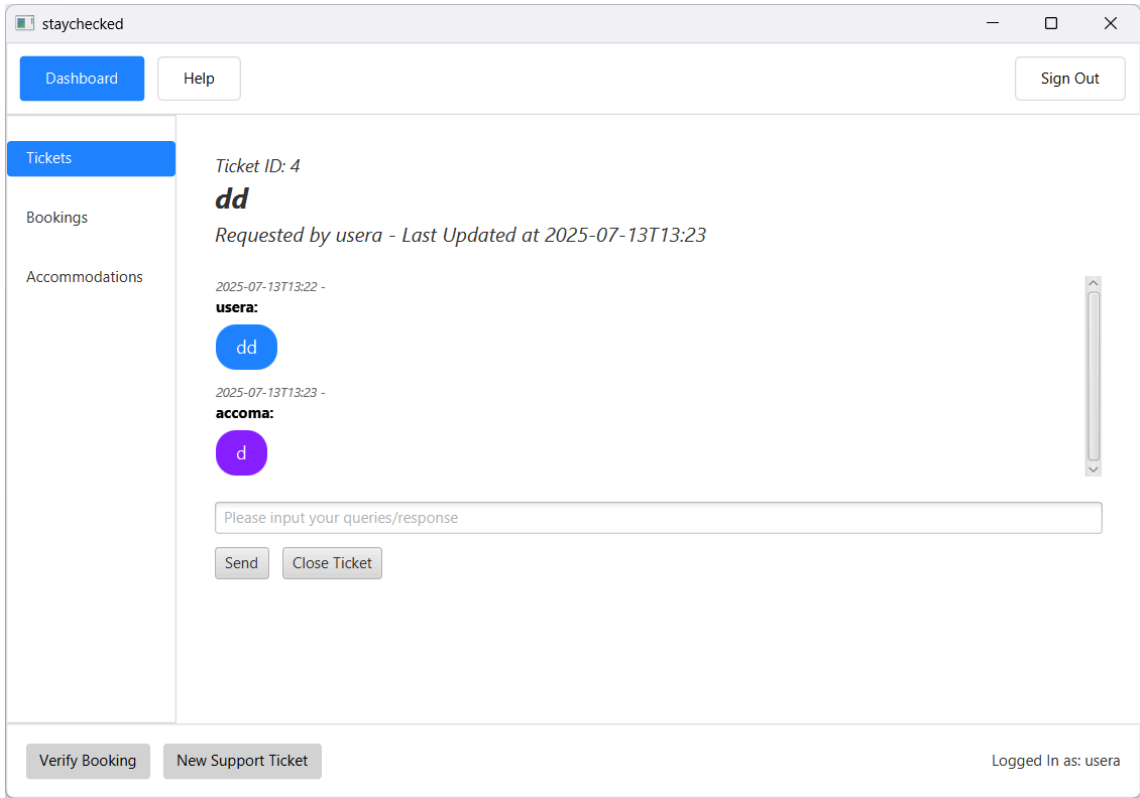
ID // Feature	Demonstration	Details/Remarks
3.1 Booking Verification System	 <p>The screenshot shows a web application titled 'staychecked'. It has a navigation bar with 'Dashboard' and 'Help' buttons, and a 'Sign Out' button. A sidebar on the left contains 'Tickets', 'Bookings' (highlighted), and 'Accommodations'. The main content area is titled 'Your Verified Bookings' and displays a table with columns: Booking ID, Identification Name, Room, Status, and Accommodation. Two rows are visible: one with Booking ID 23059312 and Identification Name booking1, and another with Booking ID 48929122 and Identification Name booking3. A modal dialog titled 'Verify Booking' is open, prompting the user to 'Enter Last Name and Booking ID'. It contains two input fields: 'Last Name' and 'Booking ID', and two buttons: 'Verify' and 'Cancel'. At the bottom of the application, there are buttons for 'Verify Booking' and 'New Support Ticket', and a status indicator 'Logged In as: usera'.</p>	Booking Verification for Users: Booking is verified by entering the last name and booking ID . Once verified, the reservation will be displayed in the reservation lists interface for tracking purposes. This function ensures that users can validate their booking records within the system and confirm their booking status.

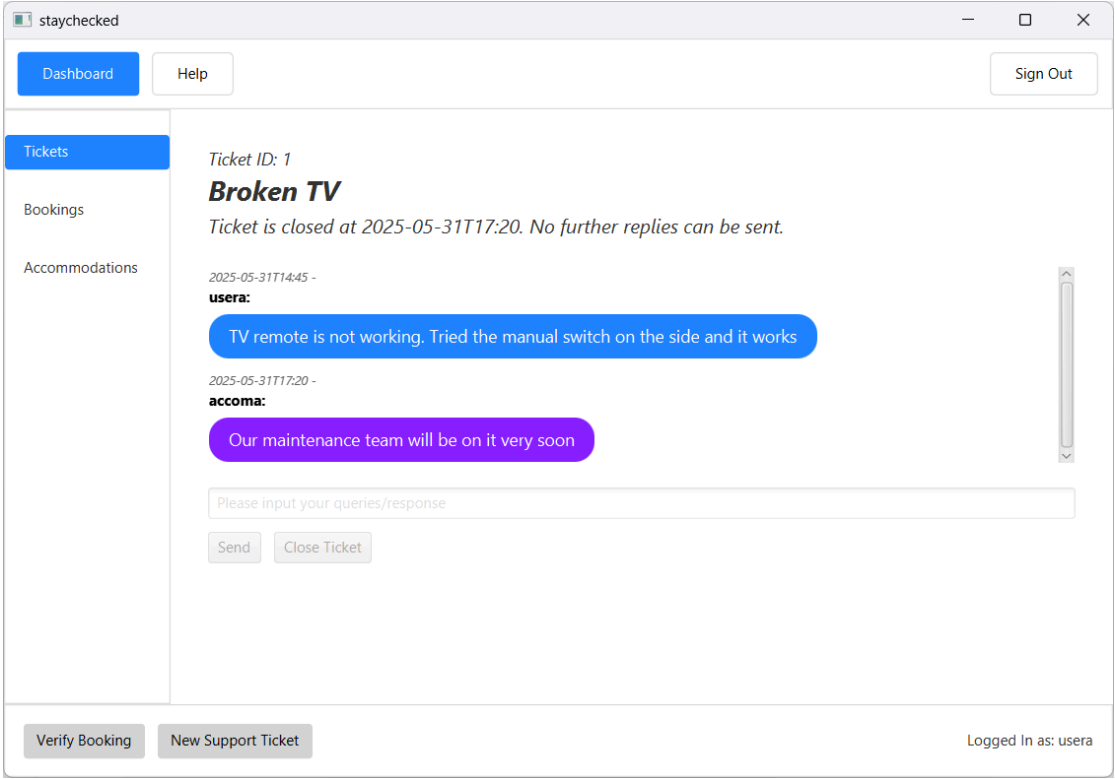
Support Ticket Management - Guests and Accommodations

ID // Feature	Demonstration	Details/Remarks																																																											
4.0 Support Ticket Lists	<div><div>staychecked</div><div><div>Dashboard</div><div>Help</div><div>Sign Out</div></div><div><div>Tickets</div><div>Bookings</div><div>Accommodations</div></div><div><div>Active Support Tickets</div><table><thead><tr><th>Ticket ID</th><th>Subject</th><th>Status</th><th>Date Submitted</th></tr></thead><tbody><tr><td>1</td><td>Broken TV</td><td>Closed</td><td>31/05/2025 17:20</td></tr><tr><td>2</td><td>Wifi Not Available</td><td>Closed</td><td>08/07/2025 22:48</td></tr><tr><td>3</td><td>Cleaning Request</td><td>Open</td><td>09/07/2025 22:27</td></tr><tr><td>4</td><td>dd</td><td>Open</td><td>13/07/2025 13:23</td></tr><tr><td>5</td><td>ef</td><td>Open</td><td>13/07/2025 13:22</td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr></tbody></table></div><div><div>Verify Booking</div><div>New Support Ticket</div><div>Logged In as: usera</div></div></div> <div><p>The Support Ticket List displays all active support tickets raised by the user. Each ticket record includes the Ticket ID, Subject, Status (Open/Closed), and the Date Submitted. This interface allows both Guests and Accommodations to monitor the progress of their inquiries and requests submitted to the Accommodation</p></div>	Ticket ID	Subject	Status	Date Submitted	1	Broken TV	Closed	31/05/2025 17:20	2	Wifi Not Available	Closed	08/07/2025 22:48	3	Cleaning Request	Open	09/07/2025 22:27	4	dd	Open	13/07/2025 13:23	5	ef	Open	13/07/2025 13:22																																				
Ticket ID	Subject	Status	Date Submitted																																																										
1	Broken TV	Closed	31/05/2025 17:20																																																										
2	Wifi Not Available	Closed	08/07/2025 22:48																																																										
3	Cleaning Request	Open	09/07/2025 22:27																																																										
4	dd	Open	13/07/2025 13:23																																																										
5	ef	Open	13/07/2025 13:22																																																										

ID // Feature	Demonstration	Details/Remarks
4.1 New Support Ticket (Guest only)		The new Support Ticket allows guest users to submit support inquiries related to their bookings. The user must select the Booking , specify the Category , provide Subject and describe the Enquiry before submitting.

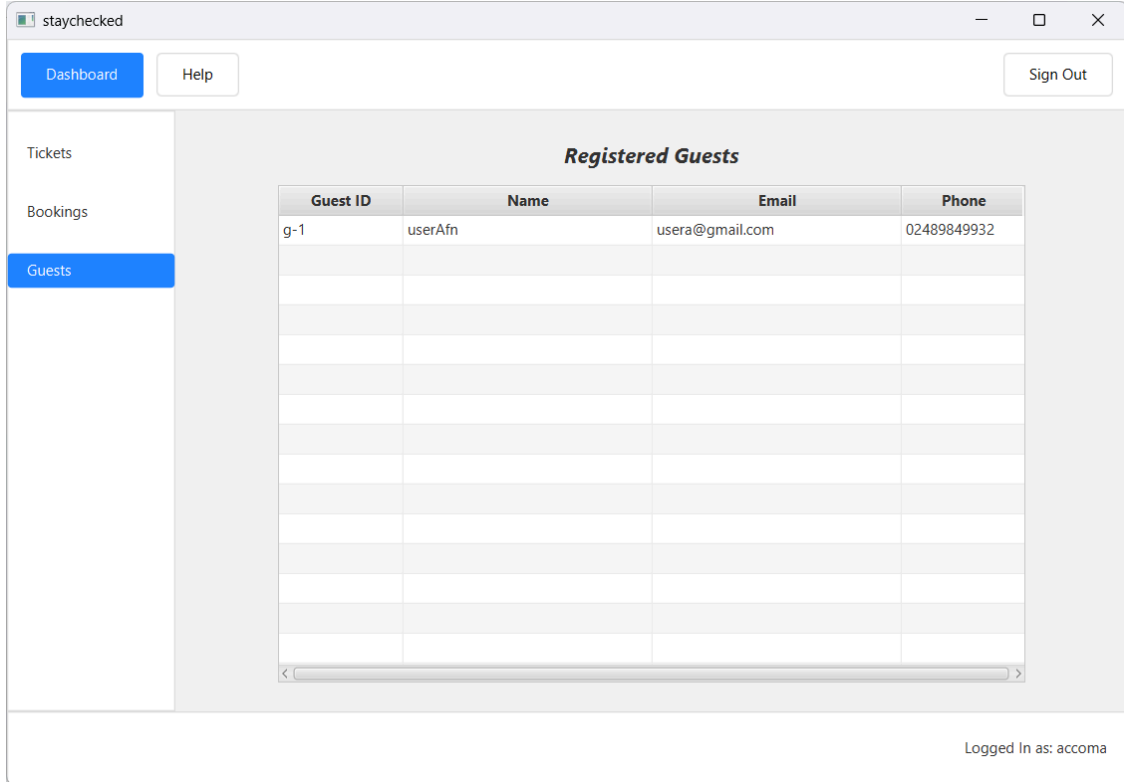
ID // Feature	Demonstration	Details/Remarks
<div>4.1.1 Invalid New Support Ticket Request (Guest only)</div> <div>Error Handling</div>		<p>If a user attempts to submit a new support ticket without completing all required fields, the system will trigger an error prompt indicating that all fields must be filled out. This verification ensures that sufficient information is provided to the support team to process the inquiry effectively.</p>

ID // Feature	Demonstration	Details/Remarks
4.2 Open Ticket Details		The Open Ticket Details allow users to view the full conversation history of an active support ticket. Users can send additional messages or inquiries directly. There is also the option to close the ticket when the user sees that the issue has been resolved. This facilitates continuous communication between the guest and the accommodation/service provider.

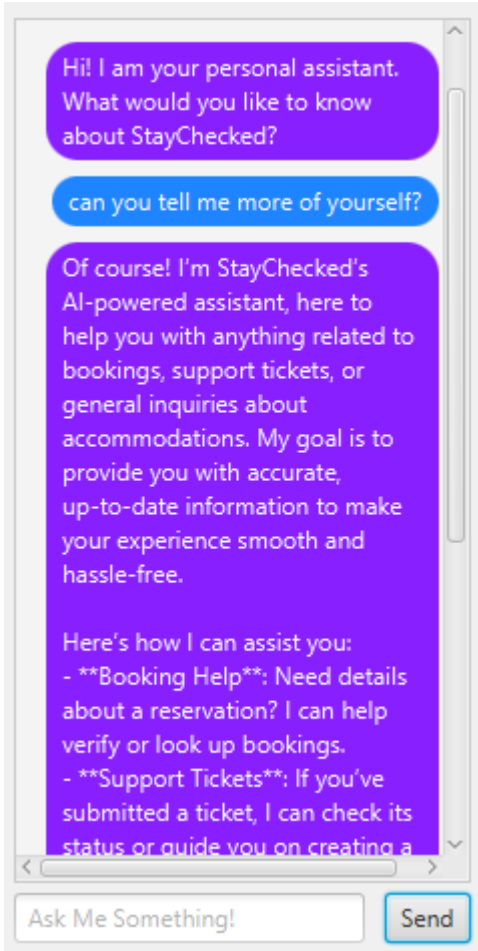
ID // Feature	Demonstration	Details/Remarks
4.3 Closed Ticket Details	 <p>The screenshot displays the 'staychecked' application interface. At the top, there's a navigation bar with 'Dashboard' (highlighted in blue), 'Help', and 'Sign Out' buttons. A left sidebar contains 'Tickets' (highlighted in blue), 'Bookings', and 'Accommodations'. The main content area shows 'Ticket ID: 1' with the title 'Broken TV'. Below the title, a message states: 'Ticket is closed at 2025-05-31T17:20. No further replies can be sent.' The conversation history includes a user message: 'TV remote is not working. Tried the manual switch on the side and it works' and an accommodation response: 'Our maintenance team will be on it very soon'. At the bottom of the chat area is a text input field with the placeholder 'Please input your queries/response', a 'Send' button, and a 'Close Ticket' button. The footer contains 'Verify Booking' and 'New Support Ticket' buttons on the left, and 'Logged In as: usera' on the right.</p>	Closed ticket details provide users with a view of the full conversation of the ticket marked as closed . Once the ticket is closed, the system disables additional responses and displays a closing time flag . This allows users to return to previous support interactions even after the solution is provided.

Lookup System - (View Only)

ID // Feature	Demonstration	Details/Remarks																																																			
5.0 Accommodation Lists (Guest only)	<div><div>staychecked</div><div><div>Dashboard</div><div>Help</div><div>Sign Out</div></div><div><div>Tickets</div><div>Bookings</div><div>Accommodations</div></div><div><div>Registered Accommodations</div><table><thead><tr><th>Accommodation Name</th><th>Email</th><th>Phone</th><th>Address</th></tr></thead><tbody><tr><td>AccommodationA</td><td>accoma@gmail.com</td><td>02489849932</td><td>AccommodationALoc</td></tr><tr><td>AccommodationC</td><td>accomc@gmail.com</td><td>03209575932</td><td>AccommodationCLoc</td></tr><tr><td>AccommodationB</td><td>accomb@gmail.com</td><td>08592389102</td><td>AccommodationBLoc</td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr></tbody></table></div><div><div>Verify Booking</div><div>New Support Ticket</div><div>Logged In as: usera</div></div></div> <div>Accommodation listings allow guest users to view a list of all accommodations registered within the system. Information displayed includes property name, email, phone number and address. This display only feature allows guests to search for available accommodations for reference or to make direct inquiries.</div>	Accommodation Name	Email	Phone	Address	AccommodationA	accoma@gmail.com	02489849932	AccommodationALoc	AccommodationC	accomc@gmail.com	03209575932	AccommodationCLoc	AccommodationB	accomb@gmail.com	08592389102	AccommodationBLoc																																				
Accommodation Name	Email	Phone	Address																																																		
AccommodationA	accoma@gmail.com	02489849932	AccommodationALoc																																																		
AccommodationC	accomc@gmail.com	03209575932	AccommodationCLoc																																																		
AccommodationB	accomb@gmail.com	08592389102	AccommodationBLoc																																																		

ID // Feature	Demonstration	Details/Remarks																																																																
5.1 Guest Lists (Accommodation only)	 <p>The screenshot shows a web application window titled "staychecked". It has a navigation bar with "Dashboard" (highlighted in blue), "Help", and "Sign Out" buttons. A sidebar on the left contains "Tickets", "Bookings", and "Guests" (highlighted in blue). The main content area is titled "Registered Guests" and displays a table with the following data:</p> <table><thead><tr><th>Guest ID</th><th>Name</th><th>Email</th><th>Phone</th></tr></thead><tbody><tr><td>g-1</td><td>userAfn</td><td>usera@gmail.com</td><td>02489849932</td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr><tr><td> </td><td> </td><td> </td><td> </td></tr></tbody></table> <p>At the bottom right of the application window, it says "Logged In as: accoma".</p>	Guest ID	Name	Email	Phone	g-1	userAfn	usera@gmail.com	02489849932																																																									Guest lists are available exclusively to accommodation users. Displays a list of all guests registered at the property in question. Information displayed includes guest ID, name, email, and phone number. This offer allows accommodations to manage or track their registered guests for communication or operating purposes.
Guest ID	Name	Email	Phone																																																															
g-1	userAfn	usera@gmail.com	02489849932																																																															

RAG-Powered AI Assistance

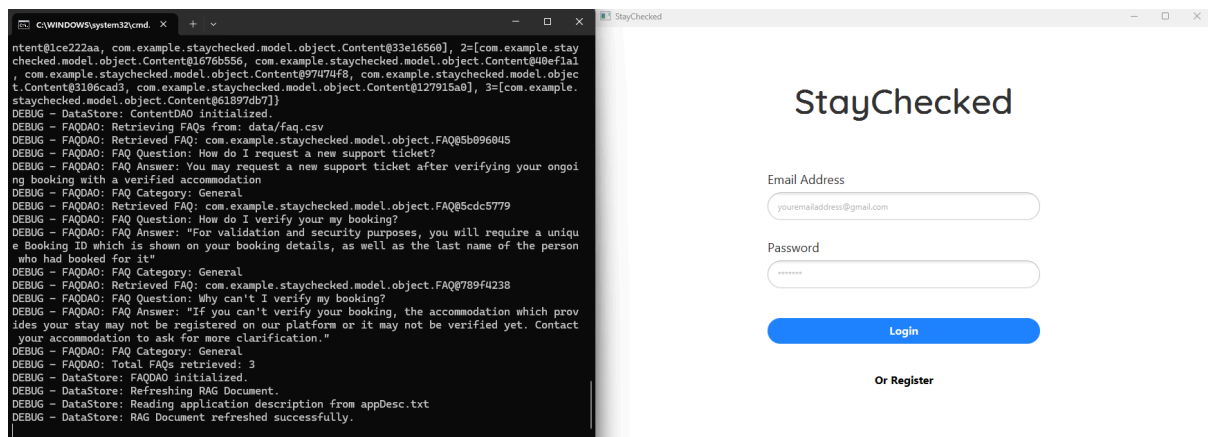
ID // Feature	Demonstration	Details/Remarks
5.1 RAG AI-Powered ChatBot	 <p>The screenshot shows a chatbot interface with a light gray background. At the top, a purple message bubble says: "Hi! I am your personal assistant. What would you like to know about StayChecked?". Below it, a blue message bubble says: "can you tell me more of yourself?". The chatbot's response is in a large purple bubble: "Of course! I'm StayChecked's AI-powered assistant, here to help you with anything related to bookings, support tickets, or general inquiries about accommodations. My goal is to provide you with accurate, up-to-date information to make your experience smooth and hassle-free." Below this, it lists its capabilities: "Here's how I can assist you: - **Booking Help**: Need details about a reservation? I can help verify or look up bookings. - **Support Tickets**: If you've submitted a ticket, I can check its status or guide you on creating a". At the bottom, there is a text input field with the placeholder "Ask Me Something!" and a blue "Send" button.</p>	The ChatBot is located inside the Help Page and is initialized/deployed with RAG systems connected to the in-memory database, allowing a level of access to the data stored inside the running application.

2.2. System Logging Processes

To support all activities related to troubleshooting, development, debugging processes, and many more, a System Logging utility feature is introduced to keep track of Application Activity both through the running terminal and a soft-copy of a .txt file which can be found within the source folder.

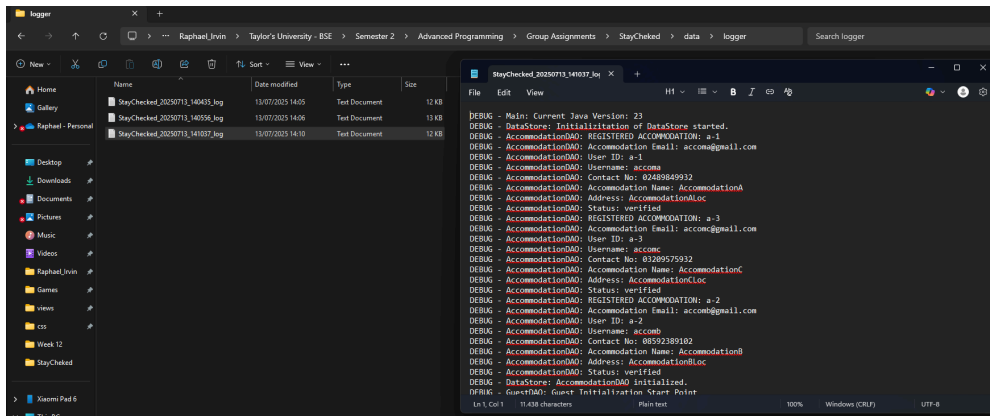
Terminal-Based Logging Processes

The Terminal, which is opened upon the launcher execution, will be kept open throughout the application process and will output the corresponding debug lines on detected application activity.



File-Based Logging Processes -> Source Folder/Data/logger

In order to support easy debugging processes which may be encountered by users, user will be able to easily find, access, and send debugging logs that are saved locally as well. This way, troubleshooting process will be easier to do by the development team by a simpler bug/trouble reproduction method and the possible causes.



3. System Documentation | Development-Side

3.1. System Overview and Architecture

The StayChecked backend is built on a multi-layered architecture using Java. It is designed to be a self-contained system that manages all data, business logic, and user interactions for a hotel booking and support platform. The architecture follows a pattern similar to Model-View-Controller (MVC), with clear separation of concerns into:

- **Data Access Layer (DAO):** Handles data persistence and retrieval from CSV files.
- **Model Layer:** Defines the core data structures (e.g., [User](#), [Booking](#), [Ticket](#)).
- **Service Layer:** Contains the business logic for all major functionalities.
- **Controller Layer (JavaFX):** Connects the backend services to the user interface.
- **View Layer (JavaFX):** Stores all pre-defined static **.fxml** files.

The system operates on an in-memory data model, loading all necessary data from CSV files into a central **DataStore** upon application startup. All modifications during a session are performed on these in-memory objects, which are then persisted back to the CSV files.

3.2. Core Components

Data Management (DataStore & DAOs)

DataStore (DataStore.java): This static class acts as a centralized, in-memory database for the application. It holds **ArrayLists** for all major data models, including **Users** (**Guests**, **Accommodations**, **Admins**), bookings, and support tickets. It provides static methods to access, find, and manage this data globally, ensuring a single source of truth during runtime.

Data Access Objects (DAOs): Located in the *dao* package, each DAO is responsible for the persistence of a specific data model.

The functionality is that they handle the reading of data from corresponding CSV files (**data/*.csv**) into the DataStore on startup (**initialize()** method) and saving the in-memory data back to the files (**saveAll...()** methods).

*Examples: **GuestDAO**, **AccommodationDAO**, **BookingDAO**, **TicketDAO**, **ContentDAO**.*

Main Data Flow Processes

Startup:

Main -> **DataStore.dataInitialization()** -> All **DAO.initialize()** methods are called.

Data Loading:

Each DAO reads its respective **.csv** file, creates objects, and populates lists in **DataStore**.

User Action (e.g., Login):

UI Controller (**UserAuthController**) -> **UserAuthService.loginUser()** -> **DataStore** (for data retrieval).

Data Modification (e.g., New Booking):

UI Controller -> Service Layer -> **DataStore** (to update in-memory list) -> **DAO saveAll...()** method (to persist changes to CSV).

Application Entry Point

Main (Main.java): This is the entry point for the JavaFX application. Its primary responsibilities in the `start()` method are:

1. **Initialization:** It triggers the entire data loading process by calling `DataStore.dataInitialization()`, which then calls the `initialize()` method of every DAO.
2. **UI Loading:** It loads the initial `LoginView.fxml` and injects its controller (`UserAuthController`), effectively starting the user-facing application.
3. **Centralized Logging:** It provides a static `debug()` method used throughout the application to print debug messages to the console and write them to a log file via `UtilService`.

User Authentication

Purpose: Manages user login, registration, and session management.

Function / Methods

UserAuthService -> `loginUser(email, password)`:

1. Searches for the user by email across all user types (`Guest`, `Accommodation`, `Admin`) in the `DataStore`.
2. If a user is found, it verifies the provided password against the stored (hashed) password.
3. On success, it sets the authenticated user in the `Session` class for global access.

UserAuthService -> `registerUser(...)`: Creates new `Guest` or `Accommodation` objects, ensures email uniqueness, and saves the new user to the `DataStore` and persists it via the corresponding DAO.

Booking Management

Purpose: Handles all logic related to booking verification and management.

Function / Methods

BookingAuthService -> `verifyBooking(bookingID, lastName):`

1. Finds a `Booking` in the `DataStore` using its ID.
2. Validates that the provided last name matches the guest's last name on the booking record.
3. If successful, it updates the booking status to "Verified" and associates the current session's guest with it.

Ticket Support System

Purpose: Manages the creation and lifecycle of support tickets.

Function / Methods

TicketService -> **postNewTicket(...)**: Creates a new **Ticket** object, linking a **Guest**, an **Accommodation**, and an optional **Booking**. It also creates the initial **Content** (the first message) for the ticket.

TicketService -> **replyToTicket(...)**: Adds a new **Content** object to an existing ticket's message history.

TicketService -> **closeTicket(...)**: Set the corresponding **Ticket** status to “Closed”

RAG AI-Powered Chatbot

Purpose: Provides RAG AI-powered assistance to users.

Technology: Integrates with an external AI provider (OpenAI) and uses a Retrieval-Augmented Generation (RAG) approach.

Function / Methods

DataStore -> `refreshRAGDocument()`: This method, called on startup and during any saving processes, aggregates data from the *DataStore* `ArrayLists` into a single document.

ChatbotService -> `getResponse(prompt)`: The RAG document is used as a knowledge base. The service sends the user's prompt and the knowledge base to the AI model to generate a contextually relevant answer.