Test specification Document

**LoginScreenTest.java:**

This document outlines the test cases for the LoginScreen class, specifically focusing on the behaviour of the login button and its interaction with the validateLogin method. The tests simulate user interactions with the UI and verify the expected outcomes.

**Test cases:**

A screenshot of a computer

AI-generated content may be incorrect.

**Please take this into account: The test cases are failing due to a technical fault otherwise the indicated error message is invalid as the name usernameField does indeed exists in the LoginScreen class.**

**Test Case 1: testLoginButton\_Success**

* **Description:** Tests the scenario where the provided username and password are valid.
* **Input:**
  + username = "validUser"
  + password = "validPassword"
* **Expected Behavior:** The login screen should close, indicating a successful login.
* **Test Steps:**
  1. Simulate entering a valid username and password.
  2. Simulate clicking the login button.
  3. Verify that the login screen closes.
* **Expected Output:** The login screen is no longer visible.

**Test Case 2: testLoginButton\_Failure\_WrongPassword**

* **Description:** Tests the scenario where the provided username is valid, but the password is incorrect.
* **Input:**
  + username = "validUser"
  + password = "wrongPassword"
* **Expected Behavior:** An error dialog should be displayed with the message "Invalid username or password."
* **Test Steps:**
  1. Simulate entering a valid username and incorrect password.
  2. Simulate clicking the login button.
  3. Verify that an error dialog is displayed with the correct message.
* **Expected Output:** An error dialog is displayed with the message "Invalid username or password."

**Test Case 3: testLoginButton\_Failure\_UserNotFound**

* **Description:** Tests the scenario where the provided username does not exist in the database.
* **Input:**
  + username = "nonExistentUser"
  + password = "testPassword"
* **Expected Behavior:** An error dialog should be displayed with the message "Invalid username or password."
* **Test Steps:**
  1. Simulate entering a non-existent username and password.
  2. Simulate clicking the login button.
  3. Verify that an error dialog is displayed with the correct message.
* **Expected Output:** An error dialog is displayed with the message "Invalid username or password."

**DoctorDashboardTest.java:**

**A screenshot of a computer program

AI-generated content may be incorrect.**

This document outlines the test cases for the DoctorDashboard class, specifically focusing on the behaviour of the viewBookingsButton and the getBookings method. The tests simulate user interactions with the UI and verify the expected outcomes.

**Test Cases:**

**Test Case 1:**testViewBookingsButton\_Success

* Description: Tests the scenario where a valid booking is found for the given date and time.
* Input:
  + date = "2023-10-15"
  + time = "10:00"
* Expected Behavior: The bookingsArea should display the details of the booking.
* Test Steps:
  1. Simulate entering a valid date and time.
  2. Simulate clicking the "Find Booking" button.
  3. Verify that the bookingsArea is updated with the correct booking details.
* Expected Output: The bookingsArea displays:

Booking Found:

Booking No: 1

Date: 2023-10-15

Time: 10:00:00

Patient Name: John Doe

Doctor Name: Dr. Smith

**Test Case 2:**testViewBookingsButton\_NoBookingFound

* Description: Tests the scenario where no booking is found for the given date and time.
* Input:
  + date = "2023-10-16"
  + time = "10:00"
* Expected Behavior: The bookingsArea should display "No booking found for the given date and time."
* Test Steps:
  1. Simulate entering a valid date and time.
  2. Simulate clicking the "Find Booking" button.
  3. Verify that the bookingsArea displays "No booking found for the given date and time."
* Expected Output: The bookingsArea displays:

No booking found for the given date and time.

**Test Case 3:**testViewBookingsButton\_InvalidInput

* **Description:** Tests the scenario where invalid date and time values are entered.
* **Input:**
  + date = "invalid-date"
  + time = "invalid-time"
* **Expected Behavior:** The bookingsArea should display an error message.
* **Test Steps:**
  1. Simulate entering invalid date and time values.
  2. Simulate clicking the "Find Booking" button.
  3. Verify that the bookingsArea displays an error message.
* **Expected Output:** The bookingsArea displays:

Error retrieving data: Invalid date or time format.