Sri Lanka Institute of Information Technology

IT4100 - Software Quality Assurance

Assignment

Modeman Web Application Report

2025-WD-06

SQA Test Report

Group Member Details

Registration Number	Student Name	Workload of the Web application
IT21172878	D.P. Liyanagama	Login
IT21177132	Madhuranga M.A.H	Reset - Password
IT21156588	Withanachchi S.S	Group Registration
IT21179594	Mallawaarachchi M.A.P.D	Registration

Introduction

Overview of the web application:

MODMAN is a web-based Module Management System built using the MERN stack (MongoDB, Express.js, React, and Node.js). It enables educational institutions to manage academic modules, assign lecturers, and handle student registrations efficiently. The system supports role-based access for Admins, Lecturers, and Students, each with specific functionalities. Its goal is to streamline module-related tasks through a centralized, user-friendly platform

Purpose and scope of testing:

The purpose of testing MODMAN was to verify that all core features function correctly across different user roles. Testing aimed to ensure the system's reliability, security, and usability. The scope included module creation, user role access, scheduling, and overall system responsiveness under typical usage conditions.

Test Plan:

Functional and UI Testing
Manual Testing and real user cases
Automated testing through the Cypress for core functionalities

Objectives of Testing

The main objectives were to validate system functionality, detect bugs, and ensure smooth user experience across Admin, Lecturer, and Student roles. Testing also aimed to verify secure role-based access and consistent performance across different devices and browsers.

Testing Methodologies Used (Manual and Automated)

Manual testing was used to verify user interfaces, navigation flows, and data handling across roles. Automated testing was performed using Cypress to test backend operations, authentication, and API responses. This combination ensured both detailed observation and efficient coverage.

Tools and Technologies Used

Cypress was used for automated end-to-end testing, while Chrome DevTools supported manual testing and debugging. The application was built using MongoDB, Express.js, React, and Node.js, with JSON Web Tokens (JWT) handling secure user authentication.

Manual Testing Documentation

https://docs.google.com/spreadsheets/d/1Woi8QxZO48r9cEqj_V7JUJFX6tjOGjzRRB1Im-4AXaE/edit?usp=sharing

Automated Testing Documentation

- ● **Selected automation tools and reason**
   Cypress was selected as the automation tool due to its modern architecture tailored for testing modern JavaScript applications like MODMAN, which is built on the MERN stack. It provides fast, reliable end-to-end testing by running directly in the browser, allowing real-time reloading and debugging. Cypress also offers an intuitive syntax and powerful DOM manipulation capabilities, making it easier to simulate real user interactions and validate dynamic content. Its built-in support for asynchronous operations and automatic waiting reduces flakiness in test execution

- ● **Automated test cases with scripts**
   Automated testing for MODMAN was carried out using Cypress to validate core functions like authentication, module registration, and role-based access. The test scripts replicated real user actions for Admin, Lecturer, and Student roles, ensuring accurate behavior under various conditions. Cypress enabled real-time browser-based testing with visual feedback and easy debugging. This improved test reliability, reduced manual workload, and increased system stability by catching issues early.

List of identified defects with priority and severity levels

Condition	Descriptions	Priority	Severity
Login Fails with Valid Credentials (Admin Role)	Cypress script shows a failure when valid admin credentials are entered; login fails	High	Critical
Module Creation: Missing Input Validation	Automated test passed despite missing required fields like module code or title.	High	Major
Student Module Registration Doesn't Update Timetable Immediately	Timetable still shows unregistered modules until page refresh.	Medium	Critical
Lecturer Uploads Unsupported File Type Without Error Message	Upload proceeds without user feedback when a non-supported file format is selected	Medium	Minor
Slow API Response During Registration Process (Student Role)	Delayed server response when registering for modules, observed in test results	Medium	Minor
Incorrect Role Redirect After Login (Lecturer Logged in as Student)	Occasional redirection to student dashboard after logging in as a lecturer.	High	Major

Conclusion

Summary of findings.

Automated and manual testing confirmed that core features like login, module handling, and role-based access work as expected. However, a few issues such as inconsistent redirection, missing validations, and delayed updates were identified during testing**.**

Challenges faced

Challenges and improvements suggested some challenges included intermittent login failures, lack of real-time UI updates, and insufficient input validations. Debugging asynchronous behaviors and ensuring cross-role consistency also posed difficulties. It is suggested to enhance error handling, improve form validations, and implement real-time state updates across all user interfaces.
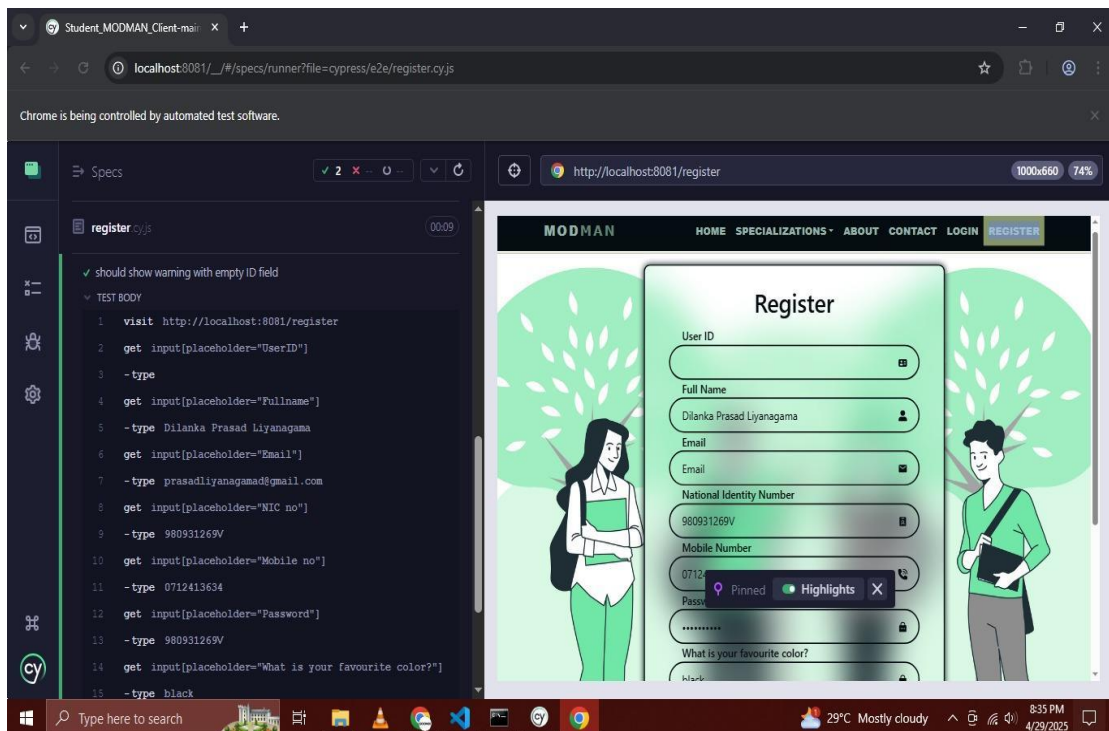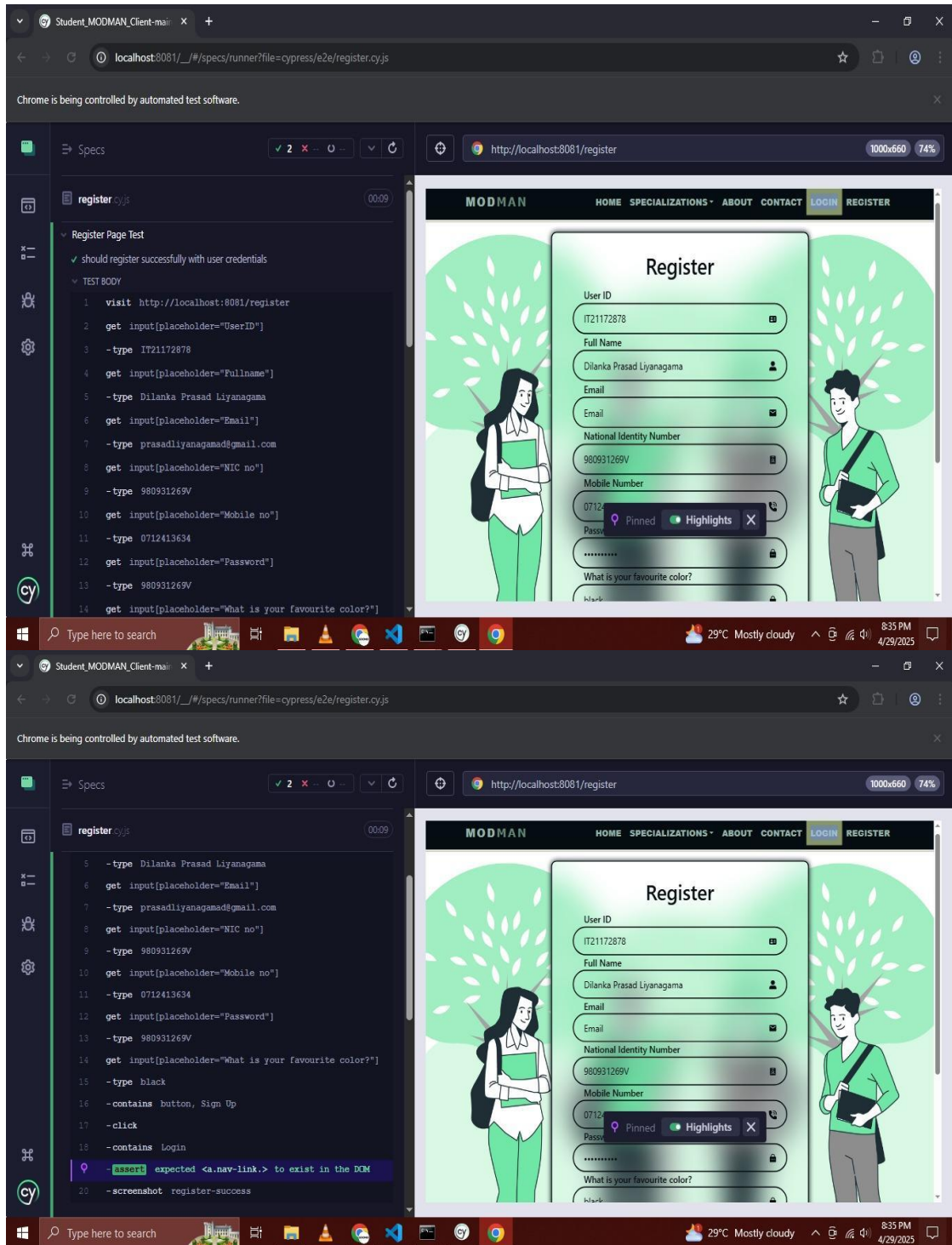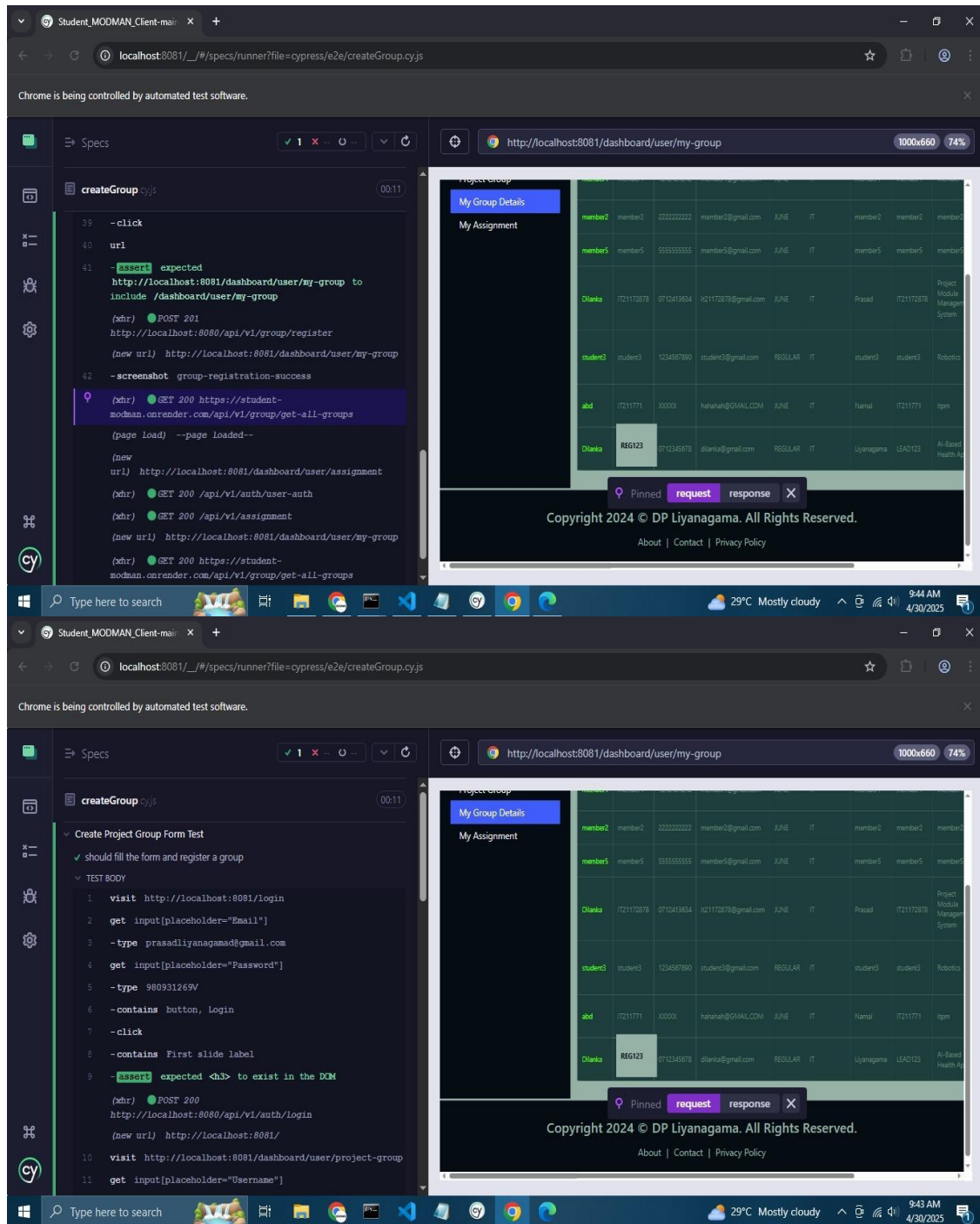
Appendix

Automated Test Cases screenshots:

IT21172878 - D.P. Liyanagama
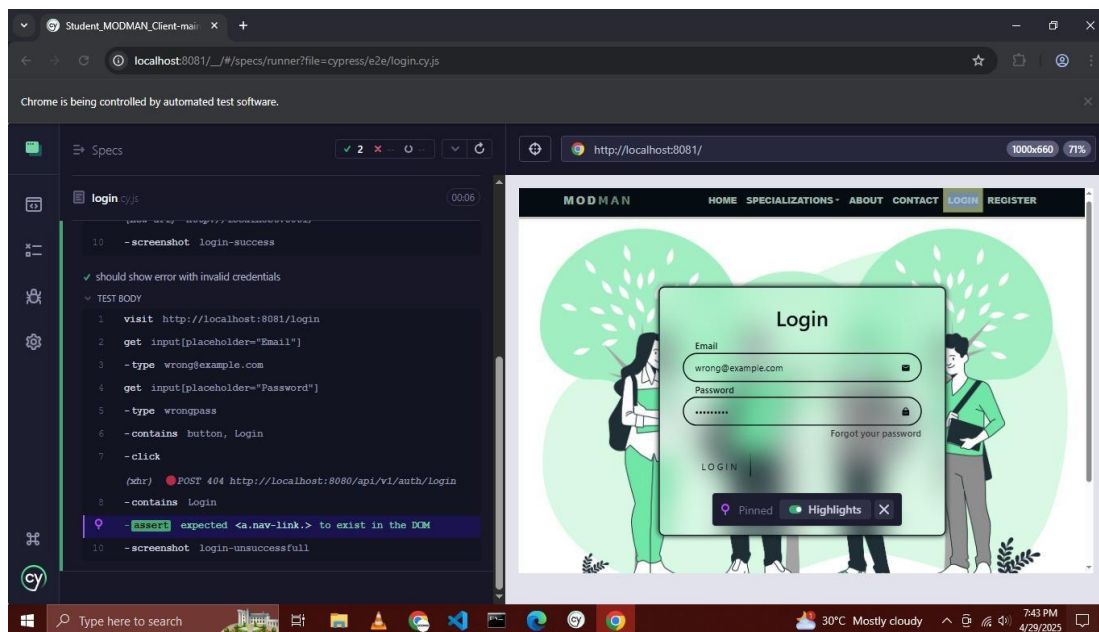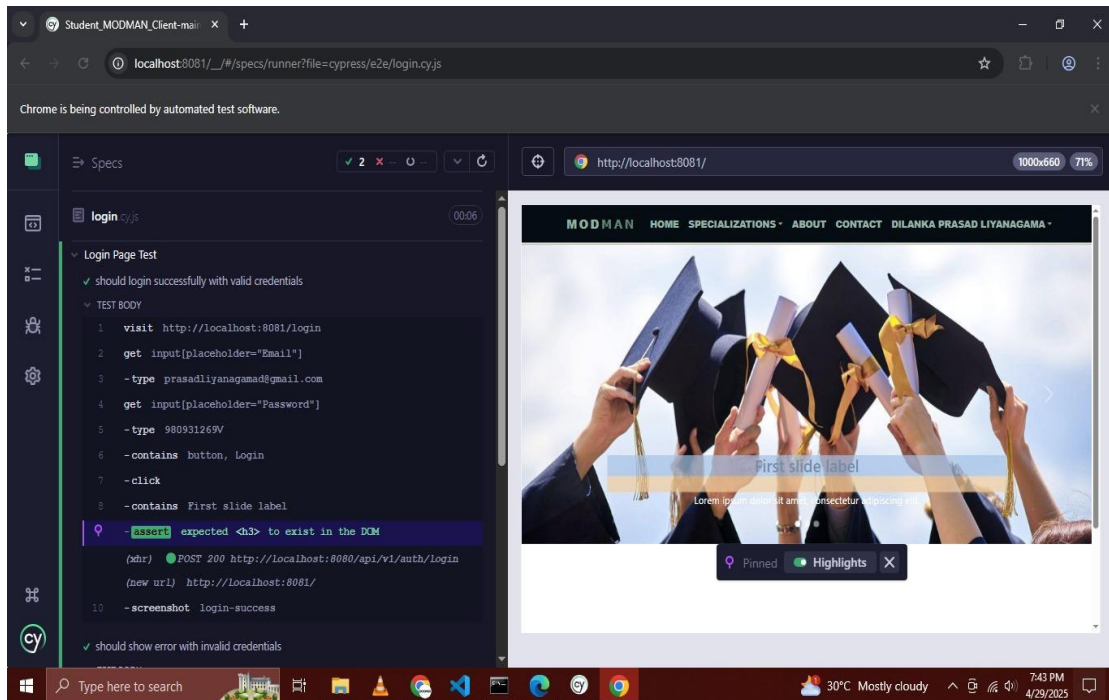
IT21179594 - Mallawaarachchi M.A.P.D

IT21156588 – Withanachchi S.S

Top screenshot:

Student_MODMAN_Client-main

localhost:8081/__/#/specs/runner?file=cypress/e2e/createGroup.cy.js

Chrome is being controlled by automated test software.

Specs

createGroup.cy.js 00:11

Create Project Group Form Test
✓ should fill the form and register a group
TEST BODY

```
1   visit  http://localhost:8081/login
2   get  input[placeholder="Email"]
3   - type  prasadliyanagamad@gmail.com
4   get  input[placeholder="Password"]
5   - type  980931269V
6   - contains  button, Login
7   - click
8   - contains  First slide label
9   - assert  expected <h3> to exist in the DOM
    (xhr)  ● POST 200
    http://localhost:8080/api/v1/auth/login
    (new url)  http://localhost:8081/
10  visit  http://localhost:8081/dashboard/user/project-group
11  get  input[placeholder="Username"]
```

http://localhost:8081/dashboard/user/my-group   1000x660 74%

My Group Details
My Assignment

Copyright 2024 © DP Liyanagama. All Rights Reserved.
About | Contact | Privacy Policy

29°C Mostly cloudy 9:43 AM 4/30/2025

Bottom screenshot:

Student_MODMAN_Client-main

localhost:8081/__/#/specs/runner?file=cypress/e2e/createGroup.cy.js

Chrome is being controlled by automated test software.

Specs

createGroup.cy.js 00:11

```
10  visit  http://localhost:8081/dashboard/user/project-group
11  get  input[placeholder="Username"]
    (xhr)  ● GET 200  /api/v1/auth/user-auth
12  - type  Dilanka
    (xhr)  ● GET 200  http://localhost:8080/api/v1/auth/users
13  get  input[placeholder="Registration Number"]
14  - type  REG123
15  get  input[placeholder="Mobile Number"]
16  - type  0712345678
17  get  input[placeholder="Email Address"]
18  - type  dilanka@gmail.com
19  get  input[type="radio"][value="REGULAR"]
20  - check  [force: true]
21  get  select                                    3
22  eq  0
23  - select  IT
24  get  input[placeholder="Enter Your Project Leader Name"]
```

http://localhost:8081/dashboard/user/my-group   1000x660 74%

My Group Details
My Assignment

Copyright 2024 © DP Liyanagama. All Rights Reserved.
About | Contact | Privacy Policy

29°C Mostly cloudy 9:43 AM 4/30/2025

IT21177132 - Madhuranga M.A.H