Automated UI Testing

Ara Institute of Canterbury | Te Pūkenga ICT Division.

Daniel Wild
d.j.wild2996@gmail.com

Bachelor of Information & Communication Technologies
Semester 1 2023

Introduction

The Ara Te Pūkenga ICT Division wanted to improve its code release process by creating rapid testing that works directly from development to production deployment—developing a set of quick tests that require minimal interference to ensure that no release degrades the overall standard of the tested apps.

To help create a solution, Puppeteer was selected as the primary tool. Puppeteer is a Node.js library that provides a high-level API to control Chrome development tools protocols.

Methodology

Kanban was selected as the development methodology as the team used it to focus on their Continuous delivery (CD) and Continuous integration (CI) pipeline, which is essential for development teams to ship software fast while ensuring high quality.

Kanban is a method for managing workflow and controlling waste. A Kanban board is used to visualise the workflow and monitor the project progress by showing the development process activities and keeping WIP in control.

Learnings

ICT-Specific Skills

- Improved my technical skills in Typescript, testing, and version control.
- Developed new skills in XPATH, Pipeline deployment, and API.

General Skills

- Experienced Agile in a real-world environment.
- Improved communication and presentation skills in a professional workplace.
- Experienced project management practices.

Goals

Industry

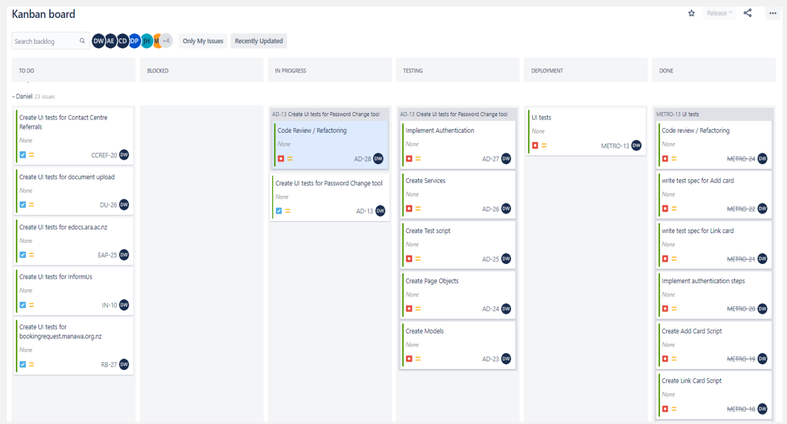
- Complete automated testing of react web application.
- Test failure email notifications.
- Minimal manual interference automated deployment pipeline.

Personal

- Complete testing code to industry standard.
- Expand my skills with real-world experience.
- Have a satisfied client in my work history.

Figure 1

Kanban Board Layout



Outcomes

- Six complete react app UI-testing suites.
- All tests are passing.
- Comparison between UI and API elements.
- Tests linted and code reviewed to industry standards.
- Tests integrated into the deployment pipeline.
- Architectural-level documentation created.

Benefits

Industry

- Direct development to production deployment for rapid testing.
- Increase the speed of testing over manual testing.
- Automated pipeline requiring minimal human intervention.

Personal

- Gain experience working in the IT industry.
- Plan, develop and manage an IT project.
- Expand my software testing skills.

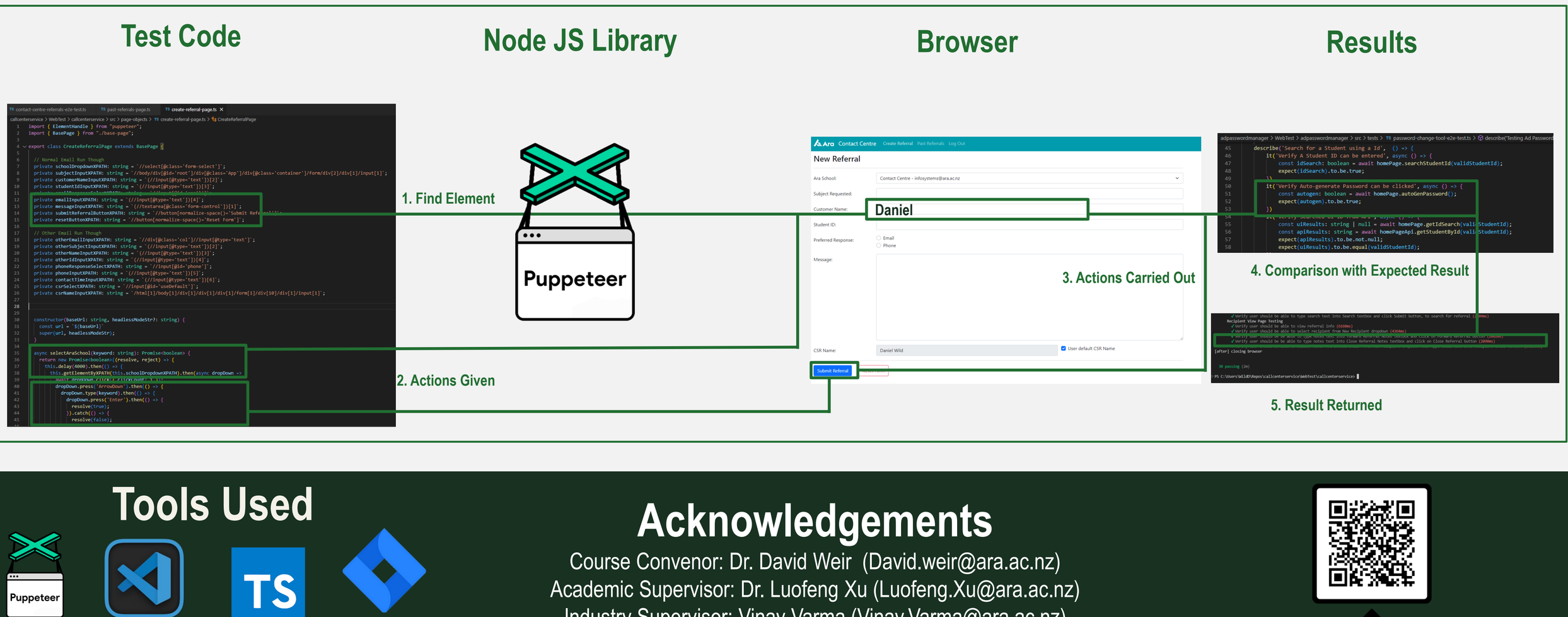
Puppeteer Workflow

As shown in the below Figure 2 Puppeteer works by doing the following steps during its workflow.

1. XPATH code is used to find each HTML element on the page.
2. Puppeteer code is then used to give the browser actions.
3. Input and actions are carried out in the browser.
4. Actions are passed to mocha chai tests to compare with the expected result Boolean.
5. A result is returned in the terminal.

Figure 2

Puppeteer Workflow Diagram



Conclusion

The project provided to be very beneficial to the industry helping to increase the deployment time and reliability of each of the tested apps. Automated testing of complex app workflows became quick and easy, which reduced the overall development cost in both time and money.

This project showed me the possibilities and power of software automation. The speed at which a large testing suite can be run compared to manual testing is impressive and shows how it can be used to maintain overall app quality. I would like to see what over things these tools could be used to automate.

Tools Used

Puppeteer · VS Code · Typescript · Jira

Acknowledgements

Course Convenor: Dr. David Weir (David.weir@ara.ac.nz)
Academic Supervisor: Dr. Luofeng Xu (Luofeng.Xu@ara.ac.nz)
Industry Supervisor: Vinay Varma (Vinay.Varma@ara.ac.nz)
Product Owner: Dean Patfield (Dean.Patfield@ara.ac.nz)

SCAN ME