How to Teach Test Automation in Software Testing

Minjie Hu
WelTec, New Zealand

Tony Assadi
Whitireia, New Zealand

Keywords

Software Testing, Agile Development, Problem Based Learning

Abstract

In recent years, software testing has been reflected from a software engineering practice into specific courses in Information Technology (IT) degrees, due to the demands for qualified professionals. In 2019, we introduced a set of software testing into the software development major of the degree. However, it also brought in many challenges with teaching test automation within available resources and infrastructure. This research aims to investigate how different teaching practices were used in a newly developed software testing course to match industry expectations and practice. We proposed two research questions. RQ1: Does the inclusion of software implementation enhance student learning in a software testing course? RQ2: How does the use of group work and project-based assessments impact students learning in a software testing course?

This study involved 15 participants who enrolled in the 2nd year of the Testing and Secure Coding paper in the degree. The majority of participants learned C# and HTML before joining this course. Some of the participants had experience with Java and Junit, however, a small number of the participants did not know C# and had very limited programming background. The course used in this study had two 2-hour sessions per week, which were a mixture of lectures and labs. To answer the research questions, a survey was used to collect data in the last week of the course.

The course design included two projects based software testing. One is Window based testing. Students were required to develop a Windows project in C# and then test it using conventional methods such as Black-box Test, White-box Test, Unit Test, and Coded UI Test. Since software development is the fundamental skills for our students, we believe that students gained experience of Test-driven Development through their project implementation.

Another is Web based testing. Students were required to develop a Web project using ASP.NET MVC Core and SQL Server. Besides the above mentioned test methods, test automation was emphasised using Performance Test, Load Test, and Live Unit Test. To reinforce teams work, we introduced the agile method with four sprints in the second Web project. Considering the diverse programming skills that the participants had in this study, we provided relevant course materials including step-by-step instructions for implementing sample projects so that students were starting from the same point.

This quality assured paper appeared at the 11th annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ 2020) and the 33rd Annual Conference of the National Advisory Committee on Computing Qualifications [Convened online in 2020], Oct. 7-8.

For test automation, we used the open-source plug-in to MS Visual Studio instead of commercial testing tools. In the Window project, we introduced Coded UI Test and Live Unit Test. In the Web project, we provided detailed information except for PBL in login function and further web layout development using Razor. Students were evenly divided into three groups in the agile method. Results indicated that participant attendance in the course was much better than that in the first project.

In the last week, eight students completed the online survey. Students were happy in developing the project and testing it based on examples. Some students commented that it was too hard to learn new functions based on online resources (i.e. PBL) for the web project as they had many other assignments. Others complained that extra workload was split to the rest of the members when someone dropped out. We concluded that guided project development not only supported students to understand test-driven development, but also assisted beginners to start software testing at the same level as others. Although PBL is commonly used in a team project, we found it is unsuitable to our second-year students in the development of Test Automation. For the future evaluation, we would remove any development without detailed guidance and change the Agile team to pair programming or individual task.

This quality assured paper appeared at the 11th annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ 2020) and the 33rd Annual Conference of the National Advisory Committee on Computing Qualifications [Convened online in 2020], Oct. 7-8.