The Industry Relevance of an IT Transition Programme

Yu-Cheng Tu, Ewan Tempero, Paramvir Singh, Andrew Meads
{yu-cheng.tu,e.tempero,p.singh,andrew.meads}@auckland.ac.nz
The University of Auckland
Auckland, New Zealand

ACM Reference Format:
Yu-Cheng Tu, Ewan Tempero, Paramvir Singh, Andrew Meads. 2021. The Industry Relevance of an IT Transition Programme. In The 10th Computer Science Education Research Conference (CSERC '21), November 22–23, 2021, Virtual Event, Netherlands. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3507923.3507957

1 INTRODUCTION

There is a shortage of qualified people in the IT industry in the world, including New Zealand. To address this shortage, transition programmes have been created that help people change to careers in IT. At the University of Auckland the Postgraduate Certificate in Information Technology (PGCertInfoTech) is one such. The software industry always strives to recruit quality software developers. To provide a useful transition programme, we must understand what these developers need in order to fulfil industry's expectations. Moreover, the IT industry undergoes continuous change, meaning a programme that was acceptable when it was first created may no longer be fit-for-purpose, that is, it no longer provides the necessary training to allow a change in career to what the industry needs today. Consequently we need to regularly review what the industry needs are in order to evaluate our programme. We summarise the results of a survey we conducted whose goal was to help us evaluate and, if necessary, update the PGCertInfoTech.

The PGCertInfoTech programme provides opportunities for people coming from a non-ICT background to gain essential software development skills. The programme is for students who have a bachelor’s degree in a non-ICT area and who want to upskill or pursue a career in IT. The programme consists of two courses, COMPSCI 718 and COMPSCI 719. The goal of our survey was to determine whether the topics taught in these courses are relevant to industry, and if not, what to change.

2 BACKGROUND

The PGCertInfoTech, introduced in 2015, is an intensive and fast-paced programme with the goal of providing industry-focused education to non-ICT students. The programme also provides an opportunity for people to gain a formal qualification in IT. The programme is equivalent to one semester (12 weeks) of full-time study, where students study 40 hours each week. Each lecture is 1–2 hours long. Students must study COMPSCI 718 and COMPSCI 719 to complete the programme.

COMPSCI 718, Programming for industry, aims to develop problem-solving skills in the context of software development. The course focuses on object-oriented programming using Java as the primary programming language. The course covers introductory and advanced programming concepts that are typically taught in computer science or software engineering first-year and second-year programming courses.

COMPSCI 719, Programming with web development, aims to expose students to modern web development technologies. The course covers front-end and back-end web development.

The part-time variation of COMPSCI 719 uses NodeJS instead of Java Servlets for the back-end development.

Table 1 summarises the topics covered and the lecture hours spent in each course.

| Topic | Hours |
|-------|-------|
| COMPSCI 718 | Object-oriented programming (OOP) | 25 |
| Unified Modelling Language (UML) | 3 |
| Recursion | 2 |
| Swing | 3 |
| Concurrency and SwingWorker | 6 |
| Design Patterns | 6 |
| Testing | 3 |
| Refactoring | 3 |
| Planning using Scrum | 1 |
| COMPSCI 719 | HTML | 6 |
| Cascading Style Sheets (CSS) | 12 |
| JavaScript | 9 |
| Servlets (or NodeJS) | 3 |
| Cookies and Sessions | 3 |
| Databases | 6 |
| Java Database Connectivity (JDBC) and Data Access Objects (DAOs) | 3 |
| Security and Deployment | 3 |
| Git | 3 |

3 SURVEY DESIGN

How we asked our questions was important. We could have asked participants’ perceptions of whether or not a topic is important, but this would have introduced a degree of subjectivity in participants’ responses. Instead our goal is to ascertain the extent to which our participants use that topic in their day-to-day employment. Doing...
so meant that participants’ responses were based on concrete experience rather than opinion. To this end, the majority of survey questions are of the form “In your employment, have you used [topic / subtopic] within [timeframe]?” or “Have you used [topic / subtopic] in your employment?”. Where further categorization or clarification is required, follow-up questions were asked. For example, after determining that a participant used version control, a follow-up question would be asked to elicit the specific version control tool used (e.g. git, svn). Open-ended free-text-entry questions were also added for each topic of the form “Please provide any further comments on your use of [topic / subtopic]”.

The survey contains 52 questions in total. Prior to releasing the survey, a small pilot was conducted where two participants in the target demographic filled out the survey in addition to providing feedback on the survey’s length and content. The feedback indicated that no changes needed to be made.

We have received ethics approval from the University to conduct the survey in January 2020. We distributed the survey to the 148 graduates of the programme at the time via an alumni mailing list maintained by the University of Auckland.

The distribution was performed by the institution administration. The authors were not involved in this process.

4 OUTCOME AND DISCUSSION

The results suggest that by and large the topics we already have are fit for purpose. The results suggest no changes to databases, frameworks, GUI, object-oriented programming, recursion, refactoring, security, and version control. Our results indicate that exposure to these topics are important in industry.

The results give us confidence to teach object-oriented programming in COMPSCI 718 as most of our participants have reported working with code written in object-oriented languages. We noticed only few reported using polymorphism which seemed to be inconsistent with the number of participants reported working with object-oriented languages. Given many participants have not much experience, we suspect that they probably have been using polymorphism at work without knowing, suggesting we may need to discuss this more in the course.

Regarding possible changes, the topics we should consider adjusting are concurrency, design patterns, and project management. The results suggest the use of asynchronous concurrency is more common than use of explicit threads. Currently, COMPSCI 718 covers concurrency with the use of explicit threads and SwingWorker. Based on the results, we think it would be appropriate to reduce the number of hours on explicit threads in COMPSCI 718, and introduce asynchronous concurrency in COMPSCI 719 as the topic is more applicable to web development.

The other topic to adjust is the design patterns taught in COMPSCI 718. The course currently covers adapter, composite, observer and template method patterns. The results show that while these design patterns are relevant to the industry, they were not the most frequently mentioned. Two people specifically mentioned MVC as important:

“MVC is the most important that everybody should know. The rest don’t need to be explicitly taught.”

From our results we conclude that introducing patterns such as MVC and Singleton would seem more appropriate for COMPSCI 718.

Currently, we spend one hour on planning using Scrum in COMPSCI 718. Based on the results, we think more time should be spent on project management in COMPSCI 718 as well as in COMPSCI 719. As most participants were exposed to at least one agile methodology, other agile methodologies such as Kanban should be introduced.

Half of our participants also indicated the use of automated testing at work, which is an important element of continuous integration. In addition, the results reveal a number of participants have seen or used DevOps at work. Based on the results, we need to rethink what we teach in COMPSCI 718 and consider at least exposing the concept of continuous integration and DevOps in either COMPSCI 718 or COMPSCI 719.

Documentation is a topic that we feel needs more investigation. The results do not give us a strong indication of the types of documentation that developers use at work. All but one participant had used some form of documentation, including inline comments. One participant made a comment on documentation:

“There’s never enough documentation. Generally documentation is stored in sharepoint, wiki or on a shared harddrive. Inline comments are a must for most jobs. GIT commits also a must”

It is worth commenting on our decision to ask about recursion. On the one hand, this is a fundamental concept in computer science and so would seem reasonable to teach, on the other hand there are a number of fundamental computer science topics that are not taught in the PGCertInfoTech so we wondered whether there was justification for including recursion and not these others topics. There was speculation within our team that it was not actually used in industry sufficiently to justify its inclusion and so we included a specific question on it to test this. The results clearly show that this is a concept worth covering in the programme. In fact, one participant implicitly indicated the importance industry attaches to this topic:

“Also had an interview question on recursion”

The last question of our survey asked whether there were topics used in their work that we had not asked about. Five answered this question, mentioning people skills, use of frameworks (2), performance issues, and use of the command line.

The comment about command line was particularly interesting. It was:

“I reckon it’s really important to know how to navigate commandline comfortably, as a huge majority of development tools are commandline only”

Our conclusion is that our programme, the PGCertInfoTech at University of Auckland is largely relevant to our graduates working in the industry, but we have identified areas for improvement. We hope our experience helps other institutions considering introducing, or evaluating a transition programme such as ours.

The full version of this paper is available at https://arxiv.org/abs/2112.11192.