# Evaluation and Recommendations of Level 200 & 300 Courses Completed for the BICT

## BCIS285: Software Application and Testing

* It was good to learn about inspections and code reviews. It is interesting that it is possible to conduct a code review even if you do not know a particular framework (such as Angular), but know the general language, such as Javascript.
* It was useful to learn about the different UML paper prototyping modelling techniques for creating and testing the usability of UI designs.
* The lab about black-box testing and TDD was helpful. I had not properly practised TDD before.
* Using the Selenium IDE was interesting. It made me realise how much I prefer using Selenium WebDriver. For example, there appears to be no built-in functionality in the IDE for checking whether an element is enabled or not. However, it is possible to create a script to accomplish this task in the IDE, though.
* Basically, everything in this course helped me with my current project. The material we studied helped me to put aspects of my current project into perspective.

## BCPR301 – Advanced Programming

* It was definitely worthwhile to have studied design patterns, refactoring, bad code smells, and the Zen of Python. All of these topics are extremely relevant to my current project. I did not refactor the code in the framework according to any of the design patterns I studied in this course. Rather, I refactored to improve the implementation of the Page Object design pattern which is used to structure the code in the framework.
* BCPR 301 assignment 1 (Pulp Alley): Working in a group was a good experience: as a group we had to do a wide range of programming activities (for example, serialisation, unit testing). It was a shame that we had to do so much work for only 10%. The unit testing was relevant to my project.
* I think it was good to have done the Kivy assignment, even though it was just a small assignment. In this assignment it was good that we had to identify in the code examples of bad smells, design patterns, and the Zen of Python principles.
* Improvements: I found the project involving legacy code interesting. It involved re-engineering a Cobol program. However, it may be more useful for students to do an assignment which more specifically involves using design patterns and / or refactoring techniques.

## BCPR282 - Best Programming Practices in Java & 283 – Best Programming Practices in .NET

* These course are all about learning about how to be a programmer, and so are very useful.
* I learned how to do unit testing in these courses which was extremely relevant to my project at Telogis.
* I enjoyed making the games in these courses: a fun and good way to learn about OO programming skills and the programming frameworks (C# and Java). In BCPR283, I learned how to program in C# which very useful for the current project.
* In BCPR282, we made an Android app (Sokoban) which was challenging and good.
* Also, in one tutorial for BCPR282, we discussed class and object responsibilities and play-acted a scenario relating to this. This was an important lesson, and was relevant to the issue about the style of coding which Chris wanted to me to introduce into the UI testing framework.
* Improvements: possibly teach more advanced programming technologies, such as lambdas?

## BCPR294 – Server Side Web Programming

* I learned a lot about good OO principles - SOLID, DRY.
* My project did not involve any server-side programming.
* However, my industry supervisor was performing load testing, which would involve server-side programming (asp.net). Therefore, this course would still be relevant to certain types of automation testing.
* Improvements: would it be worthwhile to get students to use a PHP framework in this course, such as Drupal?

## BCPR203 – Database Management Systems

* A worthwhile course. However, my project did not involve manipulating or using databases.

## BCPR280 – Software Engineering 2

* We had to write test plans for the PSP assignments; this was the first time I had written a test plan for testing an application.
* Learning Javascript was useful: important for web applications. Telogis Platform uses the Ext JS framework, which I had to modify occasionally, to add ids or classes to js files.
* I enjoyed making the Maori Language app, which was modelled on the Hawaiian Tutor Android app. It was interesting (and challenging) writing a Javascript program which did not use classes. However, it would be better and probably easier to have used classes, and a MVC model. I had not learned about MVC at this stage of my degree. I would only learn about MVC half-way through my degree, during the summer holiday Rocket Reading project.

## BCIS301 - Management of ICT

* This course raises many important management issues which are important for IT companies. The planning hierarchy is important for defining a company’s mission and vision statements and the accompanying goals. The mission or vision, goals, strategies, policies, decision and actions are driven by the values a company or organisation has.
  + In regard to the Telogis project: a team in a company needs a clearly defined mission statement and suitable corresponding goals to ensure that the work they do (for example, providing test automation for the rest of the organisation) will be planned in the most appropriate way. Management issues need to be clarified as much as possible in order to ensure that each team in an organisation knows what its role is and the actions it needs to perform in order to achieve its goals.
* It was interesting and challenging to have to write an IT plan.
* Also, it was worthwhile studying the risk management frameworks, which was useful for the project.
* This was quite a difficult course. You are not really sure whether you are doing the right thing a lot of the time. Still, it is a worthwhile course to do. Probably the most important skill to gain from this course is critical analysis.

## BCIS201 – Alternative Modelling

* It was good to learn about the different development methodologies: SCRUM (cross-functional teams, and continuous improvement); Waterfall; Spiral model. Useful for project management.
* It was a good exercise having to write essays. Malcolm Wieck was quite concerned about good grammar, which is important if you are wanting to write a professional report.
* It may be useful to learn about domain-specific languages in an IS course, to complement programming activities.

## BCIS202 – Systems Design and Implementation

* We looked at project and risk management, quality assurance, leadership, motivation. All of these issues provided me with insights into working in an IT environment.
* The course felt a bit light: we looked at PowerPoint slides a lot, but it was still worthwhile. (Malcolm was taking this course for the first time that semester). Once again, this course helped to introduce students to the realities of working in a software development team or project.
* The course seemed like a continuation of IS201. For example, we continued on with the study of different methodologies which was begun in BCIS201. We investigated ideas such as synergy, which was relevant to the current project.
* The introduction to usability in this course was interesting and useful.

## BCCS299 – Systems Administration

* A good course, because I learned about using Linux as an administrator, and about using Linux servers. As a result of doing this course, when I heard the mention of things like NAS at Telogis, I knew what they were talking about. Also, Telogis are moving away from using Microsoft servers to using Linux servers. As a result, this would have been a useful course to have done in preparation for potentially dealing with Linux servers.
* Hopefully Centos, from whom CPIT is apparently now getting educational resources, will have better resource material than that which was provided by Red Hat, because the material from Red Hat contained a lot of errors. This made the whole learning process tricky and challenging. I had to verify the content of a lot of the material through actual practical work (which was not a bad thing to have to do).

## BCIT242 – Website Development

* It was a good experience building and creating a website. This was the first time I had made one.
* I built my website using the Pure framework which made the task challenging and interesting. It was still quite a new framework at the time. There was not a huge amount of information or tutorials on the internet for this framework. I had to use a lot of trial and error to get the website finally working.
* The investigation of the issues relating to the setting up of a website was worthwhile.
* This course was relevant to my project, because the project work involved working with and testing a web application. The skills I picked up in this class were definitely helpful.

## BCIS381 – Special Topic in ICT

* I learned important skills in research and analysis. This course involved the reading of a wide range of source materials.
* We had to write summaries of the research we had done through the project. It is useful being able to condense what you have learned during a week into approximately a page of writing.
* The long report I wrote for this course, about the three main mobile platforms, was good preparation for writing the final reports for my project.