# CONTENT OF COURSES

## IT242

We were not dealing with HTML or CSS or JS in our project. This course was largely self-directed except for the tutor sessions each class. In the mean time we were expected to practice and know the techniques that were being taught.

I will talk about my experience of team work in our assignment. I remember that we had differing opinions about what we should develop for our product. It was required to set up a project proposal from the outright, so knowing the product we were developing was important. I would never have known that it was the issue of opinion about development that would be in our project too!

## IT252

The IT252 course curriculum was used in our editing of design elements on each game.

The actual process of designing in IT252 was self-directed and part of managing a product from start to finish.

The techniques that I used in my child prodigies were used quite extensively in the project. For instance, in the course we used the complex tools for selecting areas on an image either by colour inversion, or lasso tools that allowed you to cut out areas of a picture. This was helpful in producing transparent backgrounds to each of the animals/items that we developed for the project.

In addition to this, what I did with sound was used in the 80 or so samples that were needed for the project - testing to see whether the samples were satisfactory for use in the program. And also knowing where you can make compromises, e.g with file type and sampling size. We were dealt with a sound player that had a lot of constraints. One was that the default only uses wav files. When we realised how big the sound files would be as wav, we decided to search for a format that was more compressed and that could be played easily. We decided on wma.

Being able to critically look at the benefits of it was important for the maintainability of the project (could you imagine having this many sounds in a project?)

We produced storyboards for the games, so that we knew roughly what each screen would look like. We didn’t use storyboards to great effect on the project. Mostly our collaboration about elements on each page was enough - conversation - since we were always in the same room and easy to get feedback from each other. We didn’t use storyboards for the design process, more for the planning stage than actual implementation of the ins and outs of the game – which is something storyboards can be helpful for in producing code.

## IS201

We used the agile methodology for software development. Applying what we know of Agile and XP is in the methodology essay. When we started the project, determining the methodology used was recommended by our Academic Supervisor. In our first meeting it was laid out how we were going to do it, one person Project Leading, the other coding and the third testing. As far as content is concerned, the course was well organised to understand how different methodologies differ in their implementation. The assignments reinforced the differences between methodologies. Both assignments dealt with finding out about certain methodologies.  
  
I did not analyse our use of the methodology while doing the project. I am taking what I know and applying it to the way we used the methodology in the methodology essay.   
  
A point of note, when doing my level 1 papers we did read code complete, which does have a chart that structures project based on different project requirements. This was covered in the SE102 course as well. In it we understood the importance of upfront design and planning before starting the project. However, the way agile works this design and planning takes place in the first sprint, and at regular intervals during each sprint.   
  
It wasn’t until I put it into practice that I found out how practically useful developing in agile had on developing projects. I also found that it was easy to adopt the methodology but not easy to see it completely maintained. I remember having such a passion for project management that when I started doing Alternative Modelling, that I also wanted to get into it as well. Those factors have not been included in the course.   
  
As an aside, I found that my use of methodology in my other classes was not really well maintained, until I started my project. I have had good experiences too – but this was outside of class. I think it came down to the stage of learning I was at with regard to my programming papers. It was easier just to work on my own! This isn’t because I hadn’t had teamwork experience. Rather trying to voice my opinion in assignments led to some weird situations. I’ll talk more about that in the other sections.  
  
So in analysis, I conclude that the improvements that I would make to the class are on the basis of practical working on projects, because the content is fine.

## IS202

From a personal perspective, I found this course to be a good step to Management of ICT, which included concepts such as PERT charts for evaluating a project, and Gantt charts, for seeing a project broken down into chunks. These ideas are kind of what Agile uses for the same purpose, though using different tools (such as sprint logs, and burndown charts) as well as seeing the management of projects in terms of people. We didn’t use PERT charts or GANTT charts, but we did use scrum tools extensively on a daily basis. We didn’t need to use usability testing so much, as the scope of our project did not include testing with an actual audience of users (other than in-house)

It came unstuck for me when I was going to the next level – which added RISK management, quality management, and how to present IT technologies to an actual client. I can’t recall the purpose of what we were doing as it applied to the IS papers. It’s easy to look back in hindsight and say that they were all connected, but I didn’t make that observation while going through the course. For instance, it’s good to be taught that the number one resource you have as a project leader is people, but as far as the issues that we faced, it wasn’t until we were in a team with people that I understood why Malcolm talked to us about that. Also, that if people are our main resource, doing more people-oriented activities rather than writing a report about someone else’s implementation of a project management methodology.

The assignments in the course also helped to make the course real, such as analysing websites with usability criteria, and looking at an actual system that was using a methodology for project management.

I haven’t needed to critically analyse the project management model in our project, but will be recorded in the methodology essay. The iteration of concepts has been good for training in the IT industry, as I find it relevant to understanding the big picture that we are a part of. Our project has made it possible for us to see the benefit of following a methodology.

Also, looking back at the assignments I did for this paper, how atrocious!

## IS301

I really appreciated the assignments in this paper. They really made a difference to the way I understand IT projects. Frankly, I think that it was at a level that I needed it to be at for the end of my degree.

The risk management portions of the course, in hindsight, were important, as part of our assignment 2. I am looking at how good it was to learn those aspects from a IT project perspective.

Being able to critically analyse options available did not really help our project, as we knew from the start what we were going to do – we didn’t do single-dimensional analysis of the varying programs we did, and present.

Because of what I’ve said already, the course was not that helpful to my project as we did not present to people what options we were going to suggest in our analysis of the project – Callaghan wanted us to look at the project and determine whether or not we wanted to be part of it at the start.

## PR203

We didn’t require the use of the database for the programs we developed. If we intend on making it available on the web, we would consider it, but it is not part of the scoped feature list.

## PR280

This course focused on developing apps for web, and though it was targeting this development, the aims of the course were still relevant to understanding how to develop a product. Such as PSP for using OO Javascript, and then analysing our learning based on the problems we encountered, how we were proceeding to budget, and being provided feedback for our work.   
  
For instance, each weekly report mentions the problems we face, the solutions to those problems, and areas that we were falling behind in, and what we found to be the most exciting aspects of our learning. As a learner, I want to be able to be in control of my learning and it was fascinating to be given an assignment in which we could learn about the ins and outs of a programming language.   
  
When working alone, you can critique your work very well, but when you are in a team, you need to be considerate of others to work as a team. This is a real game changer in terms of the assignments that we’ve been given up to now. For instance, I think the biggest problem that I faced in doing my javascript assignment was the realisation that I didn’t want to push myself because it was only going to be marked by a person. Where as when I was working in a team, and having to learn how to work in a team, the biggest problem was just because we knew about PSP, and ultimately, refactoring, we needed to make a decision based on how we should develop. As a result, I have to ask, PSP is individual? Or is it a group technique? And how do you implement PSP in a team environment? And how does it relate to refactoring?

Another point to note, is that previous study changes this course. I was able to do a project during the holiday of 2014-2015, in which we developed in JavaScript. This prepared me for the course.   
  
Others don’t get that opportunity, because they may be just learning. You have a whole set of new questions arising if you work in teams.

## PR283

This course was helpful on a syntax and in a development point of view. We developed all prototypes in c#, as it was the language of choice for developing on windows devices. Aidan had not known how to program in c#, but I was able through my knowledge of setting up the Theseus vs Minotaur game how to make a system. He knew MVC, so we went from there. This was a buzz for me!

Also, the analysis, of thinking about the language from algorithms, patterns and dependencies was useful in determining how we were going to do certain things in the winforms version. It was through the need to be able to handle many things on screen at once that we worked on how to implement threading.

Winforms had a very simple way of handling speech synthesis, which was included in the libraries. However unity did not have the same, so we ended up recording samples for the unity version. This was a decision made about the way it was going to be coded due to a design feature.

The focus on documentation has helped a lot. When developing Theseus vs Minotaur, we had to develop test cases for the model before it was made into a game. This focus I realised recently was the reason that we should have done test driven development! It is because the tests in a way show you the way to design them later. In fact, that is a good design philosophy right there!

We ended up using unit and integration testing models to ensure the product was tested adequately. I may not have documentation, but tests are a kind of documentation!

We didn’t do a lot in the class room as far as the technology that we needed to research for project.

I learned: events, states (I had experimented with states before, but not at length), unity. Unity was a framework which gave me insight into how to use c# outside of MVC. I also learned threading, but that was also learned by Aidan. Encapsulation!

Things that I took from class: Testing, MVC, win forms.

## PR294

We didn’t need to develop in php with our project.   
  
Developing a product from start to finish – including features and showing that they work, and implementing use cases – useful to consider with project as we were working with agile, and MOSCOW features, which meant the process of getting the product up to scratch needed to show the skills we learned in this class.  
  
Could not directly relate testing features by queries in the project. However, c# allows you to make test cases which in effect make feature testing possible.

Implementing of OO design principles – such as MVC, SOLID – very useful. When we started the project, MVC was helpful to teaching Aidan how to develop in Visual Studio. Determining what each part would do became a way to facilitate development of the product. E.g when it was known that the song controller would be responsible for singing lines, and displaying text, product development could go forward with regard to writing the code. Proper interfaces enabled the product to perform what it was meant for. When we developed in unity, the way it developed the prototype was different, so lots of what we had in the before code was either simulated, replaced or deleted. But we learnt more about how to use patterns in code to good effect.

Complex query automation – not directly applicable as we used c# as the programming language, and had no access to database. The idea of setting up data on a screen according to good parameter generation was a prime feature of the code in our game. However we did not analyse this while doing the project

Security issues – we are dealing with client, and client has code base for future development. Security was not a feature that we had to think about as we developed the prototypes.

Issues – problems we faced. It was good practice to highlight the problems we faced – this was because each day we were dealing with new challenges. According to agile we do a meeting at the beginning of the day which addresses problems we’ve faced. Some problems were encountered, dealt with and finished in a day, but others took longer to address. However problem management was a large part of it, and documentation of it in notes and weekly reports (on an individual basis)

Being able to identify OO principles including SOLID enabled us to write appropriate documentation, and to address problems we faced with code base.

## PR301

I was in the middle of PR301 while doing my project. As such, the skills I was developing in PR301 were not used so much in the project. Also, I found out partway through the project that Unity is a Framework! Meaning it gives you the way to develop the system. It has hooks for those that want to make it do something, primarily in its component system. This allows you to make it behave the way you want, including giving script access for each object on the display. This meant developing in it in a totally different way than developing for winforms. We didn’t need to have so much connections between classes in the system, as Unity did all that in the back. An example of this is the update method, on any class you have. The update method allows the computer to evaluate code every 1ms. This meant we didn’t need to concentrate on the events and complex patterns we’d set up for winforms. (such as getting things to work together, in threading)

In this respect, the things I was learning in PR301 were incredibly practical.

PR301 indicates 15 reusability characteristics that are really important when dealing with OO systems. Though I didn’t use the patterns described, I still see the importance of them, which can freely be discussed. I want to learn more!

This class has been helpful in many ways:

Being able to discuss the concepts with my class mates

Being able to practice them myself in actual projects I have

The use of refactoring helped us to determine how we were going to solve the problem that we had.

## IS381

Because the course is different for each person, I will endeavour to speak on what I got from this course in terms of confidence and freedom to be able to choose my subject. I feel that it is the best option in evaluating the course and how it meets requirements. Because of the environment given, that meant, and still does, that I am passionate about Gamification. It was the learning that I had in this class which really was a boost to my own confidence in doing programming, and seeking to find a place of employment in the future. The course is structured in a way to ensure that you finish what you started. I found that the course is instrumental for me in critical thinking. This was actioned in the first week by setting up a proposal by which you can evaluate your outcomes and each week I was reminded of the steps to take to finish. This meant having a clear understanding of what I was achieving. In addition to this constructing a proposal from the first week enabled me to determine how much time will be spent on each task and be in control of my learning. This is something that is not so much given in other subjects. In fact, the biggest excitement I had about this was being able to present to peers on the subject of gamification. I went overboard! Mehdi provided a welcome encouragement all the way through the class. I will present the recommendations towards this course in my conclusion.

## CE301

The course assisted me in my project through the content that was provided.

I spent a day going over the risk management questionnaire to determine what might be an issue with the way our project was run.

Quality management for the project made it easy to determine what the issues were, which were a result of our CE301 academic supervisor giving us a methodology to work with. Milestones in the course also provided a way to see how far I was – as part of our course, we also had a group come in and evaluate the performance we were making. These things were useful, however we started late, and the structure of the course meant reevaluating the usefulness of the metrics that were provided – e.g how our team worked together.

I found the times that we got to talk with our Academic Supervisor to be useful – especially when we were nearing half way, and able to talk about the risk management and quality management aspects of the course.

We received some very helpful checklists at the start of the course that proved invaluable for determining whether or not the project would meet success criteria. In this light the information that we got was helpful.

## RECOMMENDATIONS

These are the main recommendations that I make concerning the 200-300 level courses, specifically as they relate to developing a software product.   
  
IS381 as I said earlier, provided a means for me to gain confidence in gamification, and results in me having confidence to seek employment in that direction. Where it came unstuck for me was in the last few weeks of the semester. What I was needing was a fresh reminder about the course requirements – it would have broken me out of a slump and lack of confidence regarding writing the report. I have some issues with my mental health and this meant that I didn’t quite have the clarity about the course at the end. I would recommend adding more involvement from my tutor so that I can continue thinking critically.   
  
My main block to achieving learning in this environment has been the ability to critically analyse what I’m learning. In earnest I’ve found that I learn best by given example, and practically having the chance to do that. As such, I would suggest providing group or team-work as a basis for learning. Like getting a BA, programmers, interface designers together for a project as a basis for achieving learning outcomes. I’m a big believer in team work as I’ve learnt so much from the project as it applies to teams.

Make opportunities for people to learn more, or get people interested in technologies being developed. I’ve been given enough opportunities, especially through the side-work that I’ve had outside of class. It has really put me on a solid footing for understanding courses.   
  
Documentation is really important. If I hadn’t decided on documentation as a legitimate extension of my project, then I think we could have just got away with occasional testing. But because documentation was the centre of the quality management, then I realised that our testing wasn’t up to scratch. This lead to developing proper documentation for each prototype. My observation is that I didn’t know that distinction until now. Especially when making things by yourself, documentation gets over looked because you’re working with no one. Make documentation important in the classes. I would be willing to come and talk about it.   
  
I found that I forgot risks when I didn’t know what I needed to do. This is an extension of the methodology used, and for good or bad, I want to see actual development in methodology practiced in the classes, as risks are really important but I don’t see them being taught or talked about in the classes, except at the end of the degree.   
  
I suggest having classes that are reward driven and allowing people to pass or fail based on their competency, motivation based rewards that get applied during classes. I don’t have any concrete answers, except that I used to get motivated by making databases that test my knowledge – tap into my learning potential and thirst for more.

It’s funny, how a word or song can trigger your memory. What is indicative of good habits? Goal-oriented learning. Challenges that are surmountable, progress that is seen, rewards that are real.