Final Training Report

CPE390 & CPE490: Practical Training 1 & 2

Student Name: Yahya `Mohammad Khier` Khaleel Al-Mubaideen
Student ID: 83755
Student Mobile: 0785323313
Student Email: ymmubaieen10@cit.just.edu.jo
Department: Computer Engineering
Faculty of Computer & Information Technology

Submission date: January 29, 2018

Company Name: GCS (Geelong Computing Systems)
Company Address: Building No 100, Gate 5, Second Floor, Haroun Al Rasheed St., Arajan – Amman, Jordan
Employer Supervisor Name: Haneen Abu Rajouh
Telephone: 065688454
Email: hello@gcs.jo

Table of Contents

Introduction

Company profile

GCS is a software solutions company, GCS provides Electronic Medical Record (EMR), Hospital Management and Health Information System (HIS) to small private clinics, medical centers, hospitals and up to large national public health systems.

GCS provides simple, fast, and secure cloud solutions and on demand applications that can be accessed anytime from anywhere, that is including Electronic Medical Record (EMR), Hospital Management, Health Information System (HIS) and Enterprise Resource Planning ERP systems.

Comprehensive Cloud Medical System, Patient management system, Admission, Appointments, Prescriptions, Laboratory, Imaging and Hospitalization. Financial and Medical Stock Management System, all included in one 360 cloud medical solution.

World Health Organization (WHO) Standards, International Classification of Procedures in Medicine, WHO ICD-9-CM Volume 3 Procedure codes and WHO International Classification of Functioning, Disability and Health.

Resilient Technology and Platform with modern technology, high performance platform and resilient solutions rely on us to architect a strategic, dynamic and long-term ICT plan that aligns with any business core and support future growth.

Figure 1: GCS ERP capabilities

I did my training in the Software Programming department where all development of software features happens, with a small team of 5 developers, the other departments are Management, Sales/Marketing and Technical Support.

GCS was established in Jordan in 2014, located at Arajan one of the nearest districts to the center of Amman and most lively parts of it, where a lot of government sectors buildings reside and private companies too, GCS is privately owned by a Jordanian entrepreneur (Mr. Fadi Samarah) living in Australia, and it is a small size company of about 10 employees, running the Health ERP system and some outsourcing for a Swiss company called Additiv AG, where it does some of the development of Additiv AGs financial web application.

Outline of training program

Starting 23/10/2017, The main topics that were covered during the training were:

1. Front-End software development
2. Back-End software development
3. Source code control management
4. Understanding Agile software development practices
5. Database management

- Technologies: Microsoft Visual Studio 2015, ASP.NET MVC 5, C#, HTML, CSS, JavaScript, Microsoft Team Foundation Server with Git, Microsoft SQL Management Studio, Bootstrap responsive design framework, jQuery.

6. Extra task: Find a suitable method to deploy Automated testing, giving back a report about findings.

Training program Monthly analysis

First month

In the first month I as a new developer was given the task of understanding a list of topics (detailed below) as the minimum knowledge needed to work in the company, and although it is an itemized list, there is no other way of reporting back that which I learned in that period of my training because I was going back and forth between them and I don't remember the exact chronology:

1- Review the C# programming language, which I knew before in many courses.
2- ASP.NET MVC5, MVC stands for Model-View-Controller and it is a software architectural pattern, one of the developers gave me MVC Video Tutorials done by a well-known YouTube channel for .NET development, with a total of 100 lessons that span a condensed 20 hours of pure coding and concepts, going along with coding each exercise in the videos is what I did for about 2 to 3 weeks, this answered the main question of how to make dynamic websites. This has no equivalent in any course I took throughout my study in University.
3- Review MS SQL Database Management Solution, which I used before in one occasion while doing the software project course.
4- Review HTML, CSS, and JavaScript, these are the web fundamentals and knew before in a given course.
5- Know how to use Visual Studio IDE, this one is well known and used in many courses starting from first year Introduction to programming course.
6- Team Foundation Server (TFS), this is a software management solution, but since a developer only uses some part of it, I learned that which is how to connect code with main code base, and some other details. I didn't have any courses that show something like it.
7- GIT version control system, which is a popular way of connecting many developers and not have conflict code and allow management to see the differences done with each submit by each developer, this has no courses in University but I knew before on my own and so I only had to do a review.
8- Repository Pattern and Unit of Work implementation, this is a software architectural pattern that is used to simplify large project into specific layered projects, no University course related.
9- Entity Framework and Entity Mappings using Fluent API, this is a special way of connecting ASP.NET projects to the Database, no University course related.
10- SQL Server Management Studio, the program used to manage databases remotely, no University course related.
11- ReSharper: Visual Studio Extension for .NET Developers to improve the code quality, no University course related.
12- Domain Driven Design (DDD) an approach to coding with emphasis on business entities and making the code flexible to change no matter how complex and big it gets, no University course related.
13- Dependency Injection, one way to achieve a Loosely Coupled Components in code, no University course related.
14- Don't Repeat Yourself (DRY) principle, no University course related.

I was happy with what I learned during the first month, because it connected many of the things I knew before into something that can give fruit (seen output): a fully working dynamic website with a clean and maintainable code.

One of my main problems that I had during this period was time, I had to go to University on Sunday, Tuesday and Thursday, with the rest of the week for training, namely Monday, Wednesday, Friday and Saturday, I also had to think about my Graduation project, while having problems in my daily life, things soon came to a total chaos, most of the training days I would arrive late by up to an hour at times, which I had to make up for by not taking a full one hour launch break, this will carry on to the first two weeks of the second month as I still didn't give up on the University course and Graduation project.

Second month

Finished Learning ASP.NET MVC 5 & Others listed in first month during the first week of the second month

Then I was given a demo solution that contains projects with missing code that I was told to fill in: 1- Create SQL DB file with some data (including schema). 2- Make local git branch. 3- Dynamically list table of Index page as Bootstrap table by JavaScript (given but have to get the code to run, solving the errors). 4- Connect the DB and edit model classes to match. 5- Do the CRUD (Create, Read, Update, Delete) functionalities in Index & Edit for Client class (with business logic). I finished all of which in roughly a week and half.

At that time the tasks scheduled for my training finished yet only half of the total time was (around 150 hours), and the Team leader was so busy that made me feel as if I say "I need new tasks", it would put her in an even worse position and I would become a total burden, this is the same time I learned that the project they're working on is financial and I can't be given a task on it unless I'm an employee (not a trainee). So, after I proposed it, I was given the task to learn about Automated Testing. This is something that was on my mind for the past 3 years, yet never was taught about in University, I think there are courses related in the Software Engineering Department but not CPE.

Anyway, here things started to become monotone, I would arrive almost daily at 8AM sharp, leave at 5PM or a little later, doing nothing but pure research on Automated testing and especially for ASP.NET MVC web technologies which I will talk more about in the third month. Contrary to the first month I solved the problem of time using the only option I could do, by giving up on the University course and Graduation project, because my daily life problems would never cease.

I wasn't totally happy with what I learned during the second month, because I couldn't get my hands on the actual project, rather a demo, and everybody knows that demos contain basics only.

Third month

During the third month, as I started with in the past two weeks of the second month, pure research of Automated testing, this is totally new to me in both theory and practice, never learned in a University course, or on my own before, only knew the existence of, and the task was very simple in words but hard to achieve "Find the best and most suitable way to add and implement automated testing to an existing MVC web application".

I started by searching on the Internet about what are the ways to test code, and things started to make sense after three to four days, I read many articles about the different types of tests, some of which are unit tests, which test a defined piece of code, usually a method or class, others have different names by different people and by how they are done, but the most known is functional tests, which tests one scenario of how a user would interact with the program to achieve something, and other articles about how to implement them in MVC, and came across a lot of technical jargon, for example: using Mocks, Test Drivers, Sync tools, Visual difference tools, Test Manager, Test Feedback, Boilerplate, CI Build, Test harness, just to name a few.

It made me learn that it is way more important than I thought of before and although it is hard to get into, the practice of using automated tests has more benefits to consider than cons to worry about, one such claim is that if a piece of code is covered by an automated test, there is no need to re-test manually each time new code that uses the old piece is introduced, another claim is that the full program need not be ready to make sure some of the code is fully working (think of the addition functionality in a calculator, it can be tested without the output text area and the other functionalities of the calculator).

I was happy with what I learned during the third month, because I pushed myself to one of my limits in coding, I threw myself in a new world, without any help, I was learning both theoretical and practical knowledge on my own, with no material to rely on, not a book, not a teacher, not a video tutorial but rather many, many articles and documentations.

My only negative experience at the time was when I became sick with a severe flu, I told the team leader that I will take the day off by email, which is the agreed way in GCS of reporting absence and other information, and I didn't know that It will take me many days to get back in good state. What happened is that since I'm not reporting the task immediately I thought it would be better if I stayed bed-ridden until I got my health back but once I did go back the attitude towards me changed completely, I was considered as a loafer, I couldn't get the colleague vibe again, and that was at the last few days, then they said to me goodbye.

To conclude, those three months of training were months that I will never forget about, I learned so much and hope that I get even better chances in the near future.

Evaluation and Recommendations

i. Why did you choose this company for training? I tried many companies (20+) but with no luck, then I came across this company while searching for a company that has workdays on Saturday, and this one had it and I was accepted, it wasn't far from my home too.
What did you expect to learn from this company? I expected to learn about Web development using new technologies.
Was it up to your expectations? Yes, to a degree, my only concern is that I wasn't given a "true" task because the application is financial and they didn't allow me since I wasn't registered as a team member with the management, but a trainee.

ii. Which courses did you find most useful in connection with your training? All Software Programming courses, Software Engineering courses, Database course and English language courses.

iii. Do you think that 300 hours were enough for training? From what I experienced it was more than needed, I had to propose the extra task of finding about Automated Tests, and not stay passive through around the last half of the period.
Should be more or less? I think it's good as it is, the problem is in the way companies think of trainees, think that "While building an empire there is never too many hands".
Should it be just canceled? No, stating the fact that it is one of the civil responsibilities of companies to allow unexperienced workforce to get the experience needed, although there are many problems while training is being handled, if it gets canceled then a fresh graduate would be in a very bad state of how to act and interact in a professional environment, he would lack the tact to go by, only a few have this kind of attitude before their first job, the training period is the substitute to get that.

iv. What was most important in the training, and in what areas did you learn more? As mentioned above "how to act and interact in a professional environment" is the most important thing in training, yes, I had bad situations where I had to feel bad about, yet I was reassured by the fact that I was a trainee nothing more nothing less, if those kinds of situations happened in my first job, I think I would have thrown the white flag and gave up because of pressure, and tried to get another job. I learned more about Communication skills than Web development, Automated Testing and DevOps.

v. Was it challenging training? Everything genuine in life is.

vi. Was the training a good experience for you on the social level? Yes, as already mentioned, also those who I had to interact with on a daily basis in the company, some of them became friends of mine.

vii. If you are offered a job in this company, would you accept it? Yes, that was what they wanted in the first place.

viii. Would you recommend this company for your friends who will be training in the future? No, because this company didn't accept me for normal training, they wanted me to become part of their workforce (an employee) after I graduate, they don't accept trainees, they gave me this opportunity as a lucky individual.

ix. What are the skills do you feel you should still develop before starting the search for a job? Nobody wants to show their bad side, but I think it's my judgmental attitude in everything and everyone that gets in my way of living it out there.

I was 70% satisfied with my training.

Here is a list of courses I wish to take and department didn't offer during my study at CPE:

1- Typing Course, at the start of any Computer related department and not only Computer Engineering exclusive course, if there is a typing course then writing this report, any other report and writing code would be faster for all students and not only those who did training on their own like me, I did it years ago and I feel the benefit of it every day, especially while coding, and coding is a big part of the Computer & Information Technology Faculty, I type with two hands full use of all 10 fingers with a good speed and less errors, which allowed me to have the edge in all labs that require coding, sometimes I look at others while they're finger picking the keys on the keyboard and think about "If only it was a mandatory course to know how to efficiently type on a keyboard in both English and Arabic".

2- Software programming best practices course, an advanced course about tools and methods used by good software developers to lessen the time needed to get an idea to an actual working code, there are written books on this topic for example: Code Complete, Design Patterns, Pragmatic Programmer. But not only theoretical knowledge, rather "good ways to code in general".

Appendix

Showing what I had done (I'm not allowed to show the code or the original work, so I used what I was allowed to show, the test captures):

The following picture shows the demo website after successfully implementing what was missing in the code and with a beautiful style as wanted.

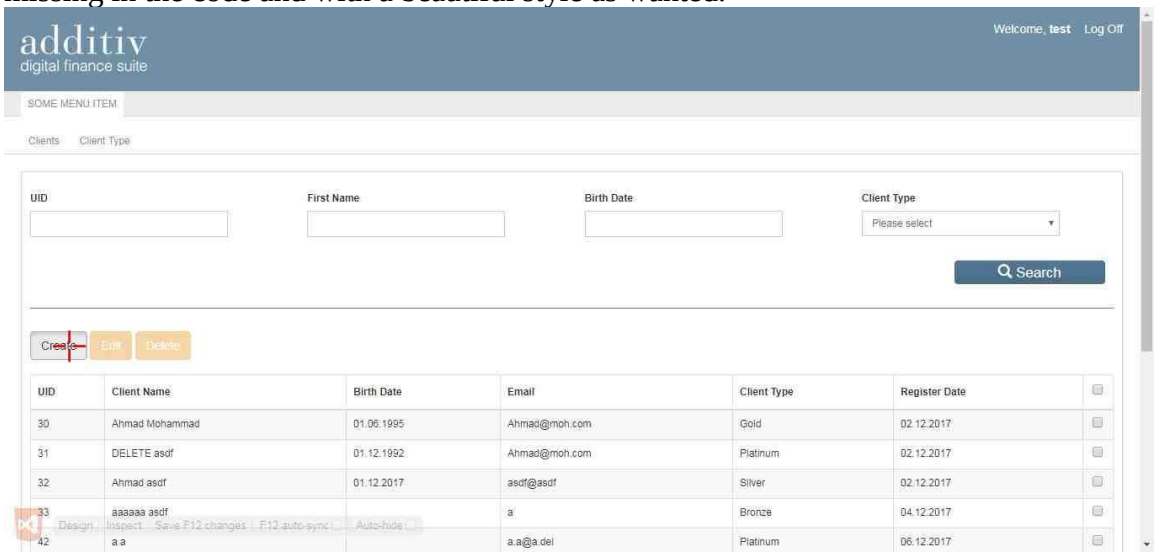

Figure 2: Demo - Client list and search page

When clicked on create this shows

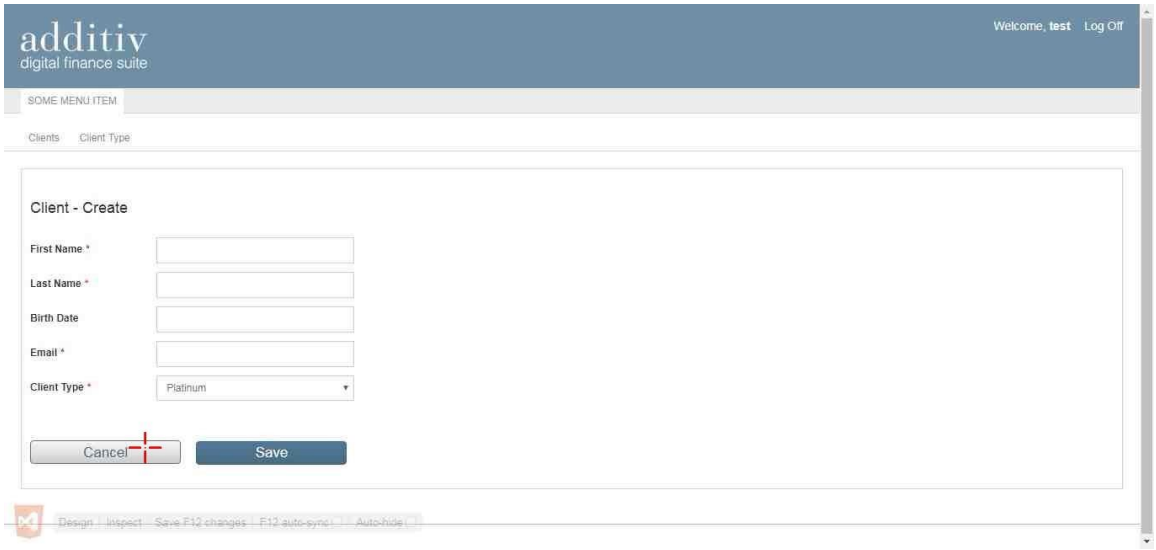

Figure 3: Demo - Client create page

Everything is working as expected and all fields allow logically sound entries (e.g. name only contains letters)

And the following shows a successful CI (Continuous Integration) build with tests.

It was very hard to do because it has its own type of knowledge called DevOps. This was part of the Extra Task and the most fun I had, because it took so much time but when It showed success I felt a huge accomplishment.

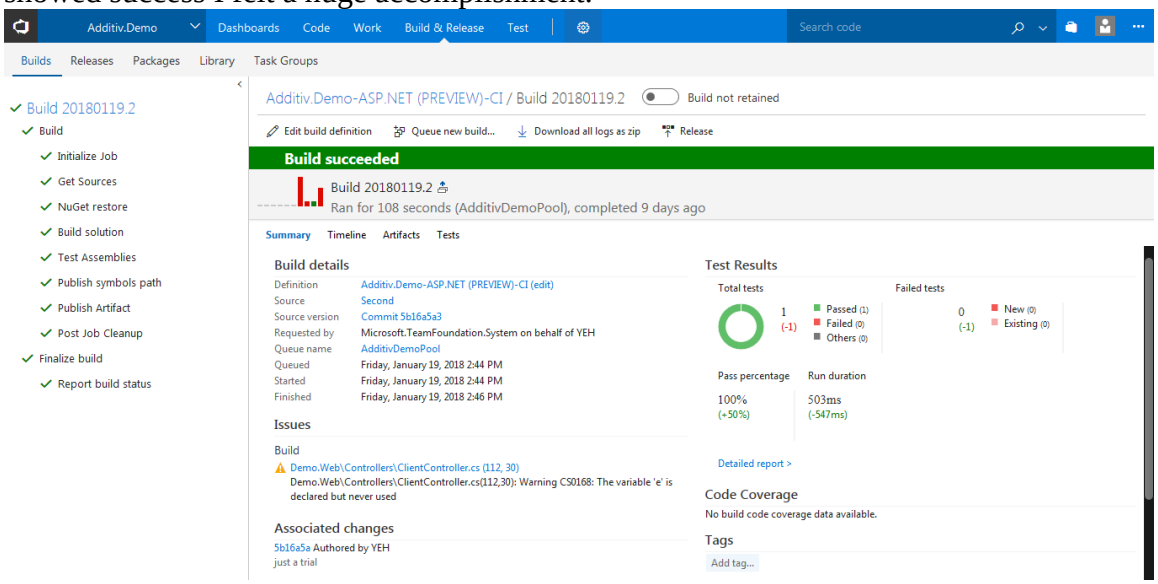

Figure 4: On-Premises TFS CI Build with Automated tests - Summary page

Thank God.