

## Introduction

*1.1.1.1 This report summarises what I learnt in FIT5032 INTERNET APPLICATIONS DEVELOPMENT S2 2019. It includes a self-assessment against the criteria described in the unit outline, a justification of the pieces included, details of the coverage of the unit's learning outcomes, and a reflection on my learning.*

## Overview of Pieces Included

This section outlines the pieces that I have included in my portfolio...

[Add a list of all of the **pieces that you are** including. Write a sentence to briefly describe each.]

1. Week 1 Task 1.1 Setup eFolio as per the specification and Run Visual Studio 2019 Community Edition IDE
2. Week 1 Task 1.2 Research and list five different IDEs currently used in the Agile software development industry and Briefly discuss five core features of Visual Studio 2019 Community Edition as an IDE
3. Week 2 Task 1.1 CSS example and Demonstrate responsiveness using Bootstrap
4. Week 2 Task 1.2 Database First Development of ASP.NET MVC Web Application using Visual Studio Scaffolding
5. Week 3 Task 1.1 Introduction to C#
6. Week 3 Task 1.2 version control
7. Week 4 Task 1.1 learning Model First Development of ASP.NET MVC Web Application using Visual Studio Scaffolding
8. Week 4 Task 1.2 learning Code First Development of ASP.NET MVC Web Application
9. Week 5 Task 1.1 JavaScript example – Notify.js and Understand the difference between defer and async
10. Week 5 Task 1.2 Use of Bootstrap datepicker
11. Week 6 Task 1.1 Using Data Annotation for view model validation
12. Week 6 Task 1.2 Understand jQuery unobstructive validation
13. Week 7 Task 1.1 Using ASP.NET Identity and learning security method
14. Week 7 Task 1.2 Setup MS Identity to sign in with Google Account
15. Week 8 Task 1.1 Obtain API key from SendGrid and Integrate SendGrid into ASP.NET application for sending emails
16. Week 8 Task 1.2 Research and understand the key features of 3rd party emailing tools and Briefly discuss the advantages and disadvantages of using a 3rd party emailing tool
17. Week 9 Task 1.1 Optimize your project
18. Week 9 Task 1.2 Introduction to .NET Core
19. Week 10 Task 1.1 Creating First Web API using .NET Core
20. Week 10 Task 1.2 Creating a Simple AngularJS Application
21. Week 11 Task 1.1 Explore differences between Continuous Integration and Continuous Deployment and Deploy an ASP.NET MVC Project in the Azure Cloud
22. Week 11 Task 1.2 Research and list five different Software Testing approaches used in the Agile software development industry and Create Unit Tests for ASP.Net MVC

## Coverage of the Learning Outcomes

This section outlines how the pieces I have included demonstrate the depth of my understanding in relation to each of the unit's learning outcomes.

[For each Learning Outcome, describe the work you have included in your portfolio that demonstrates your ability in relation to the required outcomes.]

### LO 1: Setting up development environment

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Share and include eFolio URL and Submit Visual Studio 2019 Community Edition IDE IIS localhost screenshot
- **Task 1.2** Submit a list of five different IDEs and Brief discussion on features with screenshots of Visual Studio 2019 Community Edition as an IDE

### LO 2: CSS, Bootstrap and Visual Studio Scaffolding

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Screenshot of a website using some CSS examples and code and Screenshot of a website using some Bootstrap responsiveness examples and code
- **Task 1.2** Screenshots of a running ASP.NET MVC Web Application along with database schema and code

### LO 3: Introduction to C# and Using Git as a version control tool.

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Screenshot of the instructions and Project zipped and submitted on eFolio under Week03
- **Task 1.2** Document containing your GitHub username and project URL

### LO 4: Approaches to using the Entity Framework

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Screenshots of a running ASP.NET MVC Web Application along with database schema and code
- **Task 1.2** Screenshots of a running ASP.NET MVC Web Application along with database schema and code

### LO 5: Basics of JavaScript and jQuery and How to integrate the above with Asp.net

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Screenshot of a website using Notify.js and code and A document explaining the difference between defer and async, provide examples of scenario of using it
- **Task 1.2** A screenshot demonstrating uses of Bootstrap datepicker (with 20th August set as a default date)

### LO 6: What a ViewModel is and how Visual Studio simplifies the creation of forms. And How to use data annotations for model validation

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Screenshots showing your week 6 application running with validation on "Create" page and Screenshot of your code for FormOneViewModel class.
- **Task 1.2** A document explaining what jQuery unobstructive validation is and how it works in combination with ViewModels.

### LO 7: Security & ASP.NET Identity

The following pieces demonstrate my ability in relation to this LO:

- **Task 1.1** Powerpoint Slide and Submit screenshot of the page after and before login and zipped file containing the code
- **Task 1.2** Submit screenshot of the login page and Startup.Auth.cs file containing the code and Submit the screenshot of the page with all student details and the screenshot of the student detail page after logging in along with the code
- 
-

### LO 8: Gain a basic understanding of 3rd party emailing tools and Sending email using SendGrid API

The following pieces demonstrate my ability in relation to this LO:

- Task 1.1 A screenshot of your SendGrid console with at least one email delivered
- Task 1.2 A document detailing the advantages and disadvantages of using a 3rd party emailing tool

### LO 9: Optimize your project and Introduction to .NET Core

The following pieces demonstrate my ability in relation to this LO:

- Task 1.1 A screenshot of demonstration of using Responsive Images (in PDF document) • A document explains the difference between .woff and .woff2 font using your own word (in PDF document)
- Task 1.2 A document explains new feature of .NET Core and share 3 differences with ASP.NET Framework web application

### LO 10: Creating Web API with .NET Core and Understand a simple AngularJS application

The following pieces demonstrate my ability in relation to this LO:

- Task 1.1 Screenshots of the API response
- Task 1.2 Screenshots of a running AngularJS Application

### LO 11: Deploying in Cloud using App Services and Creating Unit Tests

The following pieces demonstrate my ability in relation to this LO:

- Task 1.1 A brief report on differences between CI and CD And Screenshot of webpage deployed in the Azure Cloud and URL
- Task 1.2 Submit a list of five different Software Testing approaches and Screenshot of Unit Tests for ASP.Net MVC

## Reflection

Reflect on your learning and discuss these areas. Read the suggestions in [ ] for each question. Write your reflections then delete the text in the [ ] and delete this box

The most important things I learnt:

1. W1, Visual Studio 2019 Community Edition offers a lot of features which help developers get their work tasks easier. For this topic, I learn how to use visual studio and know its advantages.
2. W2, CSS is critical in page design. But I only learned some basics.
3. W3, GitKraken is a nice client-server for GitHub
4. W4 There are two type of models that the developers can choose, code first and model first. Model first is easy for beginners.
5. W5 datepicker displays the date in view type, which is convenient for the user to select the date.
6. W6 Unobstructive validation is a usage of the jQuery Validate plugin. It is useful, and it allows users to validate attributes without any extract coding.
7. W7 The system can sign in with Google Account. This method is convenient for users to log in with their own account and increase the system platform connection. This provides data association for user data analysis.
8. W8 The system can use SendGrid library to support sending email. SendGrid lib offers different way to send email, such as only contact, with attachment, bulk email.

9. W9 There are many software implementations for image or video compression, which can reduce the size of the entire project.
10. W 10. The api type seems to be the user mobile APP data transmission. AngularJS application is single page web application, SPA. It contains many controller and app in one page.
11. W11 CI and CD use different methods to implement integration testing. Visual studio offers Unit Testing Framework usage to test coding.

#### The things that helped me most were:

1. GitHub is very easy to version control and code sharing
2. Unobstructive validation, it reduce coding about validate attributes.
3. datepiecker, it offers date selector, which can reduce user input error.
4. image or video compression, these reduce the size of project and reduce the page loading time.
5. CSS, it makes page look more friendly.

#### I found the following topics particularly challenging:

1. the CSS, I have to learn a lot of rules to make sure CSS works, and I need to consider which page is user friendly.
2. for GitHub, If I don't use gitKraken, I don't know how to push entity progress in to github.
3. if the model is stable, model first is a good way to develop the system. But when the model changing, all coding need to update.
4. Bootstrap datepicker and jQuery UI datepicker both offer date selector. For what reason, choose one of them.
5. SendGrid, if I want to send email with attachment. I need to learn how to upload a file and attach it to the email.
6. If the project is a small one, CI is a good way to implement integration testing. Otherwise, CD is better.

#### I found the following topics particularly interesting:

1. The grid system is one of the most important functions in the bootstrap. It simply divides the "web page" into 12 defaults. The width of the column changes depending on the resolution of your "web page". This makes the page design more convenient.
2. Bootstrap datepiecker offers date type of data, jQuery datepicker can offer string type.
3. loading js file can choose async and defer, they offer different loading scenarios for different needs

#### I feel I learnt these topics, concepts, and/or tools really well:

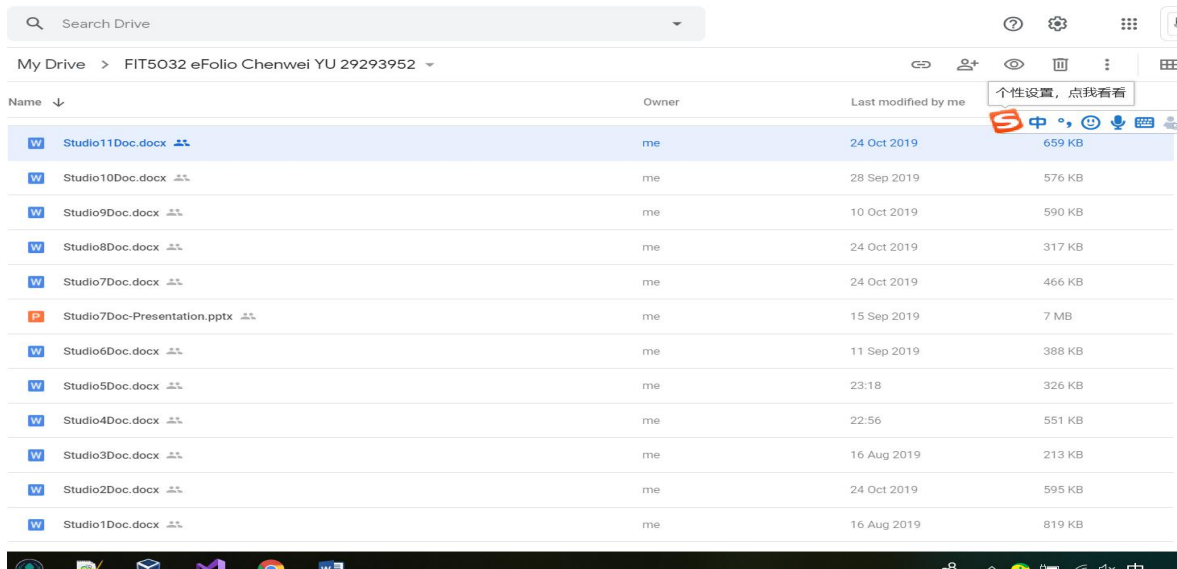
1. EF design from databse can generate the relation model from database, which is easier to understand for developers.
2. gitKraken is an easy to use version control tool

### I still need to work on the following areas:

1. I need to learn how to change the view of creating if some attributes do not need the user to input
2. I need to see more code about CSS in the future.
3. loading js file base on different requirement to choose different loading scenarios
4. Improve the ability to think aesthetically and comprehensively
5. Using different kinds of unobstructive validation, such as number, range.

### My progress in this unit was ...:

I learned a little bit of knowledge in each week's project, but there is no way to integrate them into a single project. Until I started doing assignment, I suddenly realized that they can be connected in series and interact.



Name	Owner	Last modified by me	Size
Studio11Doc.docx	me	24 Oct 2019	659 KB
Studio10Doc.docx	me	28 Sep 2019	576 KB
Studio9Doc.docx	me	10 Oct 2019	590 KB
Studio8Doc.docx	me	24 Oct 2019	317 KB
Studio7Doc.docx	me	24 Oct 2019	466 KB
Studio7Doc-Presentation.pptx	me	15 Sep 2019	7 MB
Studio6Doc.docx	me	11 Sep 2019	388 KB
Studio5Doc.docx	me	23:18	326 KB
Studio4Doc.docx	me	22:56	551 KB
Studio3Doc.docx	me	16 Aug 2019	213 KB
Studio2Doc.docx	me	24 Oct 2019	595 KB
Studio1Doc.docx	me	16 Aug 2019	819 KB

### This unit will help me in the future:

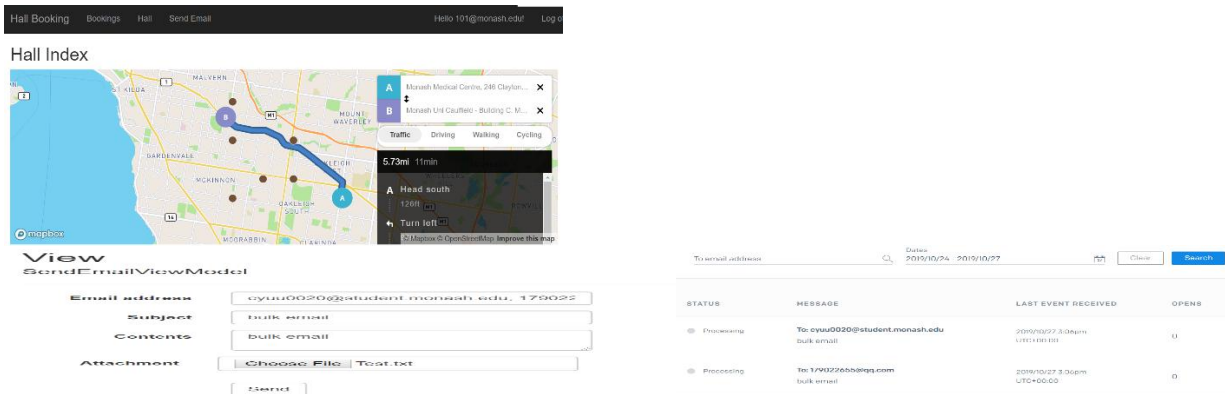
From learning a piecemeal knowledge point every week to completing a simple project, I found that learning knowledge does not mean how to use it, so it is necessary to apply knowledge to the actual project. I think that when I learn programming in the future, I should start a project and then apply the knowledge I have learned each time to the project. This not only masters the knowledge, but also the lack of knowledge points.

### If I did this unit again I would do the following things differently:

1. Learn and master the reasons for this operation every week, not just learn how to do it.
2. Communicate with classmates or tutors to broaden my design thinking, not just in my thoughts, and it also improves the rationality of the interface layout.

## Other...:

After interview, I modify map with two features and bulk email



The screenshot displays a web application interface. At the top, there's a navigation bar with links: 'Hall Booking', 'Bookings', 'Hall', and 'Send Email'. Below this, a 'Hall Index' section shows a map of a campus area with various locations marked. A route is highlighted on the map, and a small inset shows a detailed view of a specific location. To the right of the map, there's a 'View' section titled 'SendEmailViewModel'. This section contains form fields for 'Email address', 'Subject', 'Contents', and 'Attachment'. The 'Email address' field is filled with 'cyuu0020@student.monash.edu, 179022'. The 'Subject' field is filled with 'bulk email'. The 'Contents' field is filled with 'bulk email'. The 'Attachment' field has a 'Choose File' button and a text input field containing 'Test.txt'. Below these fields is a 'Send' button. To the right of the 'View' section, there's a table showing email processing status.

STATUS	MESSAGE	LAST EVENT RECEIVED	OPENS
Processing	To: cyuu0020@student.monash.edu bulk email	2019/10/27 3:06pm UTC+10:00	0
Processing	To: 1/P027655@ngs.com bulk email	2019/10/27 3:06pm UTC+10:00	0

## Declaration

I declare that this learning summary, eFolio tasks and the linked code are my individual work except group submission. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text and code, nor has any part of this submission been written for me by another person.

Signature: \_\_\_\_\_ Chenwei Yu