FIT5032 - Internet Applications Development

Portfolio Submission

Prepared by: Jian Liew, Yiwei Zhong & Chris Messom
Last updated: 25th October 2018

Executive Summary

This document will outline the requirements for this unit. It will describe the requirements for the components which are the **Design Report**, **ASP.NET MVC application**, **Research Report** and the **Learning Summary.** It will also describe the grading criteria with examples.

Estimated Reading Time : **~ 30 minutes ++**

Major Components & Grading Method

This portfolio work will be a major part of your portfolio submission for this unit. It will consist of up to 4 major components.

1. A Design Report
2. An ASP.NET MVC Application
3. Optional Research Report
4. Learning Summary

Depending on your target grade, there will be a different set of requirements. **You must approach this using a stepped approach.** In other words, you must first accomplish the C level requirements before you move onto the D requirements and so on. The various requirements will be outlined to you in this document. It is recommended for you to pick a target grade based on your expectations and the amount of time you are willing to invest in this unit.

Besides that, what we strive for in this unit is for **continuous improvement**. In other words, over to the course of the semester, you should constantly show your work to your tutor and he or she can advise you the various components in which you can improve upon. It is highly recommended to consistently show your work to your tutor.

If you are targeting a high grade, it is expected for you to constantly show your tutor your work during the tutorial or the consultations.

Design Report

The design report is due for feedback on **Week 5** onwards.

The Design report should provide a detailed overview of the functionality of your application. In addition, you should include mockups of your application (Screenshots of your application prototype or using WireFraming). Include user stories, that are driving your design decisions. In your design document include a completed checklist of site functionality.

You can use a wireframing tool like Sketch, Lunacy or Balsamic to accomplish this. You can also use any diagramming tools to generate the Mockups like LucidCharts, Draw.Io or even MS PowerPoint. [Some students (coders) use ASP.Net as a high fidelity Mockup tool (with screenshots of the application running), however, in industry, the other tools are normally used since they can be faster for producing low-fidelity mockups]

Your design report must demonstrate good communication skills, and present a well thought out application design. It must clearly communicate the abstractions being created and outline how the functionality is organised.

The methodology for the construction of both the design report and the ASP.NET MVC application is based on "User-Driven Development (UDD)". More information can be found here. In short, it must be "Use Case Driven".

You are given the freedom to select the choice of the web application you would like to make, however, you are bound to a set of requirements. This set of requirements will be based on this document specification and your tutor.

The theme for the web applications this year is:

- Web application for people who are into nutrition and health and well being

You can also use any form of the open dataset you can find in your application, however, this is not a requirement for this unit (FIT5032). However, using data in web applications will be good preparation for your Industry Experience Project. Australian open data can be found here.

Contents of Design Report

Your design report should consist of the following. Please remember you are only allowed to proceed beyond each range once your tutor has checked your work on Doubtfire or shown to your tutor. (The task must be marked off as "Completed" on Doubtfire)

The template for the Design Report is available on Moodle. This template should only serve as a rough guideline on how you should layout your design report. As long as the major components are present, you are given the liberty to improve upon your design report to your liking. (You can make the document as "professional" as possible if you wish)

Turnitin will be enabled for this submission (on Moodle).

Credit Level Requirements

- The name of your application.
- Overview of your application's goals
- User Stories that are driving your design decisions
- Functional diagram illustrating how the core aspects of the application fit together. (You can provide either a functional diagram or a System Use Case diagram)
- Usability Design Review (You can use either the **Flow Bohl usability principle** or **Donald Norman Principles**)
- A Checklist of the site functionality

Additional D Level Requirements

- Your selected approach when constructing the application. **(Model First, Code First or Database First)**
- If you are using the Model First or Code First approach, you will need to provide a class diagram. An Entity Relationship Diagram (ERD using Crows Feet or Chen Notation) should be provided if you decide to use the Database First methodology.
- Data Dictionary. (This includes the justification of the data types which you use)

Additional HD Level Requirements

- A description of your development methodology for this portfolio submission. (For example, did you approach the portfolio submission using a Code and Fix method or using a Test Driven Development). Most students are expected to use the Code & Fix model. (This is completely normal for all assignments)
- Description of how you handled the versioning of your project. For example, if you have used Git, Mercurial, CVS, or SVN. Please provide evidence of your repository if you have done so. (If you did not use it, justify why). **It is not a requirement to use these tools but if you have done so, please state it.**

ASP.NET MVC Requirements

- The assignment must be completed with ASP.NET MVC. Using any other technologies unless otherwise specified is not allowed. **(No marks will be awarded if this happens)**
- The ASP.NET MVC application is due for feedback **Week 7 onwards**.

Basic Credit level requirements:

- The website must have customised pages. (The pages must be informative. **All default generated placeholders should be removed**). You must include at least **one image** on the front page. This image must be "responsive" in nature.
- It must use a **responsive web design (RWD)**. (This can be achieved using Bootstrap, Materialize CSS or Foundation). Please note that only Bootstrap was shown in the tutorials. If you have used a different front-end framework, please mention it in your report.
- Log in & Log out (Using Visual Studio Default Individual Identity)
- There must be at least 2 tables present in the database. (You must either use Code First, Model First or the Database First Approach in which these tables are generated)
- You will then introduce the Views & Controllers from that. (Scaffolding feature). Customise these Views and Controllers to fit your application design.
- There must be a "Date" present as one of the values stored.

Upon the completion of the Basic C level requirements. A modular approach will be used to assess your assignment. These modules should be use case driven. **In other words, there must be a user story that relates to the module in which you would like to create.**

Depending on your target grade, you can choose to complete up to 3 modules at various levels. However, you will need to take a **stepped approach during this process**. So, before you move on to complete a module at a higher level, you will need to show your tutor your work for feedback first. Each module at a given level should be about the same difficulty.

Your grade for this section will depend on the number and level of modules you have completed up to the satisfaction of your tutor. Please remember that you are only allowed to proceed, once you have obtained permission from your tutor.

Number of Modules Completed	Grade
Basic Application (1 Preliminary Module development)	C → C++
2 Basic Modules (extending Preliminary Module)	D → D++
1 Basic Module (from above) + 1 Advanced Module (extending above)	HD → HD+

* Please do not attempt to complete more than 2 modules (except for interest and fun). If you wish to do so, you must obtain permission for your tutor. However, the maximum marks obtained is capped at HD+.

The completion of each module requires extensive time and effort, it is suggested to start early. **At no point during demos should your application crash. If your demos crash (before final demo), take the feedback from your tutor and improve your work and demo again the following week.**

Please refer to the following modules and the level descriptors (for Basic, medium and advanced level modules). Your tutor will determine the completeness of the modules based on its features and functionality and ease of use and more importantly if it meets the use case specified.

C → C+ level:

To go beyond the C (60 mark), start working on a basic module from the D level.

D->D+ Level

Complete two basic level components from the given component below, proper validation and user experience should be considered:

Component ID	Module Name	Basic Level Module Component Description
1.1	Reservation Module	Customer: Ability to make a reservation Keyword: Date Time selection, Reservation status
2.1	Map Module	Map user: Ability to manipulate maps Keyword: place of interest, custom map marker
3.1	Customer engagement Module	Customer: Ability to contribute the website Keyword: editor, like a post
4.1	Social check-in module	Customer: Ability to check-in Keyword: Add/Invite friends, propose/specify check-in location (Updated 20/08/2018)
5.1	Your own module (as an option)	Your Own D level Component Description (signed off by the tutor)
6.1	Your own second module (as an option)	Your Own (second/substantially different) D level Component Description (signed off by the tutor) (Updated 20/08/2018)

HD → HD+ level:

Complete one basic module (from D+ level) and extend one module to the advanced level from the given list below. You must consider validation and user experience if you want to achieve HD level.

Component ID	Module Name	Advanced Level Module Component Description
1.3	Reservation Module	**Customer:** Ability to make a reservation Keyword: Conflict prevention, email notification **Website Administrator:** Ability to manage a reservation Keyword: Calendar integration, reservation confirming, reservation reschedule
2.3	Map Module	Map user: Ability to manipulate maps+ Keyword: Navigation between two places, location service, link places of interest together to form a trip, save the trip & route info, dynamic update route with places of interest
3.3	Customer engagement Module	**Customer:** Ability to contribute the website Keyword: WYSIWYG editor, File upload, reCAPTCHA **Administrator:** Ability to moderate customer information Keyword: Approve/ disapprove post, archive inactive users, newsletter template & newsletter batch sending
4.3	Social check-in module	**Customer: Ability to check-in** Keyword: identify the current location **Customer: Ability to interact** Keyword: 3rd party integration (Zomato, 4 square, Flickr), like/ favourite a check-in, Restful API
5.3	Your own module (as an option)	Your Own HD Component Description (signed off by the tutor)

For each feature used in your application, you are required to match it with a related use case and include it in your Design Report.

The completion of the Keyword is not a requirement, however it would depend on your Design Report and your tutor.

Different tutor may have different requirement regarding how to complete each of these keywords and should be mentioned in your design report. During the interviews, you are expected to explain this to the tutors who will be doing your interview.

Features that can be used in the modules for C to HD+ Requirements [You can add your own features as well]

Map Module (Week 5)

- Displaying features on the map. (MapBox or similar mapping API).
- There should be a "Search" feature on the map.
- Using "Geocoding" or "Reverse" Geocoding Features (Mapify or Mapbox) **(Not shown)**
- Routing between two points. **(Not shown)**

Calendar Module (Week 5 & 7)

- Calendar display and integration (FullCalendar.io)
- Users are able to create "Events" on this calendar.
- The constraint of dates should be taken into consideration during the creation of an event. For example, can an event take place in the past and how far into the future can an event take place?

DataTables Module (Week 5)

- DataTables is implemented.
- DataTables now support pagination features. (Pagination is the concept of breaking a large data set into smaller chunks of data) **(Not shown)**
- Please demonstrate this with at least 100 rows inserted into the database. In order to generate this mock data, you will use Mockaroo. **(Not shown)**

Advanced Features in the User Module (Week 7 & 8)

- An "Admin" user is present and this user can send email to all users using SendGrid API.
- The administrator is able to manage the user posted data including deleting files user uploaded directly from the website.
- Allow users to upload an image of themselves and display in their details page.

Real-Time Features (Week 8)

- SignalR is used to implement a real-time feature. (You will need to implement a different one other than the one shown in the tutorials)

Web API Features (Week 10)

- The application is now extended to also have a Web API according to best practices. You can justify your choice of best practices in your design report and documentation. **(It must be within the same project and not a different project which is part of the solution - Not shown)**
- The WebAPI is now extended to have "Filter" & "Pagination" features. **(Not shown)**
- This WebAPI is described using Apiary or Swagger. **(Not shown)**

Your Own features (not shown)

- Design your own features after doing some research, testing and investigation

Portfolio Demo

Starting in **Week 7,** you will be given the opportunity to show your work to your tutor during the tutorials or during the consultation times.

The final portfolio due date for portfolio demo is on Week 12. All demos should be completed in this week or before this week.

If your target grade is C or higher, your tutor will need to mark that task as "Completed" on Doubtfire. This grade can only be obtained if you have demoed your application to your tutor.

During this demo, you have two options

- You will be the "pilot" of your system
- Your tutor will be the "pilot" of your system

**Pilot here means the person using the system*

If you are the pilot of your system, you can determine **the flow** of how you would like to present your project to your tutor. (You can imagine your tutor being the client)

If you are planning to let your tutor be the pilot of your system, you should have the confidence that everything it is working up to expectations.

During this demo, you are required to tell your tutor the grade you are trying to achieve and the components you have completed in order to achieve the grade. Your tutor will also be asking you various questions regarding what you did and how you accomplished the task.

Keep in mind, that it is quite common for your tutor to easily find mistakes or errors. **If this happens, it is highly suggested to approach the issue in a calm and collected manner.** The objective, of the portfolio demo, is to give your tutor the opportunity to advise on what you can do to improve your project. The earlier you demo your work, the more time you will have to correct and improve your work.

Remember that, since we are striving for continuous improvement, you are given the opportunity to fix the issues in which your tutor found (assuming it is before the final demo). After you have fixed the issue, you can show your tutor your work once again. Hence, it is advised to constantly ask for feedback.

Your work will be signed off at a given level, once your tutor is satisfied with your work. The final grade will be based on the work demoed and signed off, including the design reports.

Research Report Requirements

The research report is due for feedback before the end of the semester. **(Week 12)**

Turnitin will be enabled for this submission (on Moodle).

This task can only be done if you have completed all the Pass Tasks, Credit Tasks and Distinction Task **moving towards the High Distinction Task**. This task is mainly for students who are targeting the HD+ grade. **This grade will be subject to benchmarking from the teaching team.** The total number of pages for this report should not exceed 10 pages in length, the work should be of high quality, rather than of large volume.

It is highly recommended to use proper sources when obtaining reference materials. (Google scholar search engine is a good starting point). You will need to use this link to gain access to the Monash search engine for Google Scholar.

Summary

You will create an **"intellectually stimulating"** Research Report based on this unit.

Your Research Report must

- Aim to answer a question related to the unit's learning outcomes that have been accepted by the tutor/lecturer.
- Propose a method by which the question can be answered.
- Demonstrate that the method was carried out, and show appropriate summaries of the data collected.
- Present an analysis of the data showing insights into the underlying concepts
- Discuss findings and their relevance to the internet applications development.

Types of Research Projects (choose one or some other accepted type of project)

- Survey of Literature (Pure review of literature), this is not normally very interesting for the person reading the report, the other types of research are recommended
- Experimentation, Testing and Reviewing Technology
- Application Development, applying some technology and reviewing it.
- Knowledge Discovery, discovering new knowledge relevant to the field

Here are some examples

- In the recent years, it is increasingly becoming important to learn newer front-end technologies such as ReactJS and VueJS. Please explain why people consider this so. You can compare these with ASP.NET MVC. Include code examples you have tried and tested.
- Microservices is becoming one of the buzzwords in the recent years, please explain why this is so. Please refer to link as an example. Can you implement a microservice as a test case and discuss?

Please note that these topics here are just examples, it is highly recommended to discuss your topic with your tutor. The research topic is subject to approval by either your lecturer or tutor.

If your research topic could be submitted to another unit, then it is likely to be not suitable for this unit’s learning outcomes.

Learning Summary

- It is a **compulsory** requirement for all students to submit a learning summary.
- This task is due at the end of **Week 12.**
- A template for the Learning Summary is available on Moodle.

Summary

This learning summary report gives you a chance to reflect on everything you have learned in this unit and make a case for being awarded a particular grade based on the learning outcomes that you have demonstrated you have achieved and the assessment criteria you have met.

Purpose

For you to express your understanding of Internet Application Developments and demonstrated your achievement of this unit's learning outcome.

Task

Write a short learning summary report document (based on the template provided), identifying the evidence you have created in your portfolio that shows that you have achieved the learning outcomes for this unit. In addition identify the grade that you believe you have achieved, by identifying the pieces of work that have met the assessment requirements for this unit. In addition add your reflections on the unit, including how your learning progressed and how it might be improved in the future.

Time: This task should be completed in your last two weeks of study.

Resources:

Learning Summary Report Template (Word version), Submitted and Completed tasks on Doubtfire, Tutor in Studio Session, Study partners in class Helpdesk (from week 3)

Pass Task Submission Details.

You must submit the following file to Doubtfire

- A PDF version of your Learning Summary Report. Make sure your file includes your targeted grade as well as the justifications for being awarded this grade. This document is a summary, your selected weekly submitted task (from Doubtfire) will be automatically added to your learning summary when you submit your portfolio, so you don’t need to repeat a lot of detail in your learning summary report that is already in your portfolio.

Summary of Submission Requirements & Due Dates

Item	**Feedback Due Date**	**Final Due Date**
All PASS Task	The end of each week itself	Week 12
Design Report	Week 5 onwards	Week 12
MVC Application	Week 7 onwards	Week 12
Research Report	After already meet HD requirements	Week 12
Learning Summary Report	Week 10	Week 12

On Friday Week 12, all these tasks should be zipped and uploaded onto Moodle.

So, you will not only need to submit everything on Doubtfire but on Moodle as well. The moodle submission of your work is the official archive that can be reviewed independently (if something happens to the tutor/lecturer/chief examiner)

Lastly, your assignment will be subjected to a similarity check against all current and past assignment submissions. Should your submission be flagged as highly similar to that of another student, both submission grades will be withheld and considered as per the university's policy on plagiarism and collusion. All work submitted MUST be your own work.

All codes borrowed must be cited and referenced accordingly.