School of Computing, Edinburgh Napier University

1. Module number	SET08702
2. Module title	Web Tech
3. Module leader	Simon Wells
4. Tutor with responsibility for this Assessment	Your first point of contact is your local QA tutor
5. Assessment	Please see attached descriptor for details
6. Weighting	**Part #1 (40%) | Part #2 (60%)**
7. Size and/or time limits for assessment	Please see attached for details. A short, approximately 5 minute, demonstration will be required.
8. Deadline of submission Your attention is drawn to the penalties for late submission	**Part#1 is due at 5:00PM on Tuesday 9th July 2019** **Part #2 is due at 5:00PM on Tuesday 23rd July 2019**
9. Arrangements for submission	Please see attached descriptor for details
10. Assessment Regulations	This assessment is subject to the University Regulations**.**
11. The requirements for the assessment	Please see attached descriptor for details
12. Special instructions	None
13. Return of work	We will aim to email marks to you within three **working** weeks.

14. Assessment criteria	Please see attached. With reference to the module descriptor, part #1 of this assessment covers aspect of learning outcomes **1, 2, & 3** and part #2 covers aspects of learning outcomes **1, 2, 3, & 4**. **L01** requires students to research the specific standards (technology, accessibility, privacy, and security related) that are applicable to their project and demonstrate understanding through their written report and translation of applicable standards to their implemented software. **L02** requires students to plan and implement their personal project using an appropriate methodological approach and to critically reflect on this in their written report. **L03** requires students to implement the user facing portion of their planned project by applying an appropriate mix of technologies studied in the lectures and practical sessions. **L04** requires the development of a cohesive integration of both server and client side components in their project applying technologies studied in lectures and practical sessions.

Coursework Assignment
Web Tech (SET08702)

Overview

The assignment for this module is a single project that is split into two parts. This document details both parts. The first part is worth 40% of the overall mark and the second part is worth 60% of the final mark. Because this is a project, there is an overall goal that you should be working towards, but you have considerable flexibility in how you achieve this and which features you choose to incorporate.

The overall aim of the project is to develop a simple note-taking application that uses a web interface for interaction with the user and that persists notes in an appropriate fashion using a suitable mix of client and server-side web technologies.

You must provide a web interface (HTML & CSS) where your user can type their notes, with buttons and supplementary JavaScript used to provide an engaging and satisfying user experience and reliable functionality. The user interface might be architected using a single-page, "web-app" design, or a more traditional, multi-page, web-site design, but either approach has merit and the decision is yours. You should also consider aspects such as online documentation pages or help to assist your user to effectively use your site as well as visual presentation of your features. The notes that your user writes must be stored on the server-side using a Node.JS web-app. This enables you to provide your users with functionality such as persisting & backing up notes and enabling notes to be accessed from different computers.

The first part of the assignment (due in week #2) includes a short report that will cover the features and overall design of your solution, alongside a prototype implementation of the Web interface. The second part of the assignment (due in week #4) is to implement the server side part of the project, to integrate the server and client as appropriate to your design, and to summarise, in a short report, how you performed and where your final project deviated from the initial plan. It is expected that what you design initially might well deviate from your final implementation and that techniques you discover later in the module might cause you to re-implement or re-approach things you have already done. This is perfectly fine and gives you an opportunity for reflection in your final report.

Part #1

For this part of the assignment you must:

- Plan your note-taking app by deciding on a list of features and deciding how each feature will be implemented and appear in your user interface. The absolute minimum functionality is a single text entry in which the notes can be written, and a button which will save the notes to the server once Part #2 is completed. Your may however prefer to design something that can support multiple notes or which stores meta-data, like the word count, or a title, or the date of creation

or last-update. You will likely also want to design for a satisfying user experience, and utilise a visually pleasing design, as this will make it easier to achieve a higher mark.

- Produce a prototype implementation of the client side (HTML & CSS) interface including any supplementary client-side JavaScript features as required (for example if you include any validation of user input). Your prototype should reflect your plan, however, as some of your features may require or rely upon server-side functionality these should be "mocked up" to show how a user might initiate the server-side feature. You will complete those aspects when you work on part #2 of the assignment.

Before you begin, it is worth doing a little research into the kinds of features that existing, real-world, note-taking apps support. Don't get too ambitious though as you have a limited amount of time and are looking at what teams of professional developers have achieved. As a rule it is worth having a simple, core plan that you can supplement with more elaborate functionality in the situation that you make better progress than you expected, but allowing you to fall-back to something more achievable if you experience challenges.

Your deliverables for this part of the assignment are the following:

1. A short (limited to no more than 4 pages) PDF report that covers the following:

    1.1. A list of features and some discussion of why each feature is included.

    1.2. A sketch of an initial user interface design for your system and some commentary on the motivation for your design, i.e. how does your design address the features you've listed. NB. Any designs can be hand-drawn and scanned/photographed for inclusion in your report.

    1.3. Screenshot(s) of your prototype implementations of your design accompanied by commentary on the process of moving from design to implementation, for example, drawing attention to the parts that you found to be more challenging.

2. A prototype implementation of your web interface that uses a set of HTML pages, supplemented by CSS, & JavaScript as appropriate. This prototype should be consistent with the range of features that you've described in your report. For those features that require server side implementation to fully implement you should mock-up those features (perhaps using a pop-up with suitable message to indicate when a server side feature has been initiated). Your prototype interface must run in the Chrome web browser. This should be archived into a single zip file containing the source code (HTML, CSS, & JS) for submission to Moodle

Both report and code archive must be uploaded to Moodle by 5PM on Tuesday 9th July, 2019.

Part #2

For this part of the assignment you must:

- Implement the server side functionality of your planned project using the Node.JS platform and JavaScript. This should fulfil the features that you mocked up in Part #1. Minimally, your Node app must provide some way of persisting your user's notes so that when they reopen their browser and navigate to the app, they can retrieve any previously saved notes. You may support any additional features that you feel will lead to an improved experience for your users within the remit of a note-taking app.

- Integrate your prototype interface with your Node app so that the client and server sides provide your designed functionality underpinned by a consistent and coherent user experience. Note that you may need to alter how your initial prototype worked in order for it to work smoothly with your Node app. In some cases this may require a complete rewrite or reimplementation of your initial prototype, for example, if you decide to use express.js and a templating system rather than raw HTML. This is perfectly fine.

Your deliverables for this part of the assignment are the following:

1. A short(limited to no more than 4 pages) PDF report that covers the following:

    1.1. Describes the differences between your initial plan and your final submission

    1.2. Reflects upon the challenges you faced and achievements you made during this assignment

2. Your final implementation of your site. This must include all server and client code required to rebuild and run your site.

Both report and code archive must be uploaded to Moodle by 5PM on Tuesday 23rd July, 2019.

Demonstrations

The main goal of the demos is to establish that the work you've submitted is your own. It is also a useful opportunity to provide some verbal feedback. Demos will be held during the period immediately after the deadline.

Due to the size of the class additional time on Thursday and Friday may be required but your local tutors will coordinate this. During your demo you will have the opportunity to show off your app and may be asked questions about your work.

Note that you should aim to be set up and ready to go **before** your tutor arrives to see you. It is your responsibility to ensure that you can demo the site that you have developed but **without a demonstration your submission will not be marked**.