6 - Module Review and Futures

## Learning Outcomes

On completion of this lab you will have:

* Reviewed the important theoretical material of the module
* Been introduced to one possible future in Web App development through Elm

## Organisation

Please complete the exercises individually.

## Grading

This worksheet is worth up to 10% of your overall module grade. You must attend and sign in at the labs in order to obtain the credit for the associated worksheets. You may work on this worksheet during lab 12 and lab 13 with instructor assistance.

## Submission

The deadline for submission is Wednesday Dec 14, 2016 @23:59 through Webcourses.

## Demonstration

There will be no demonstration required for this lab

## Requirements

For this lab you will need to

* Have reviewed the module lecture material (part 1)
* Watched the lecture and tutorial videos on Elm programming (part 2) - these will be published in the coming days

## Resources

You are free to research whatever you need to solve the problems in this lab. Some recommended resources include:

* <http://elm-lang.org/>
* <http://elm-lang.org/docs>
* <https://guide.elm-lang.org/>

## Problem Sets (Part 1)

Provide brief (1-2 paragraphs/bullets, etc) answers for the following. It is important that you have reviewed the module material in advance

|  |  |  |
| --- | --- | --- |
| 1 | Explain what is meant by Rich Web Application Development. Distinguish it from traditional web development  *Rich web applications use Javascript on the client side and employ lean data transfer with servers. This creates web applications that are powerful and provide a better user experience. This is all done at the presentation layer. This is the newest form of web application presenting new protocols and technologies.* | 6 Marks |
| 2 | What is the Document Object Model? Explain, giving a couple of examples, how to interact with the DOM in Javascript  *The DOM is the final output of your code. It is how it is structured and built by your browser. The DOM is the working together/ collaboration and output of the HTML, CSS and Javascript code written.The DOM can be manipulated even after the browser has rendered it and this give rise to concurrency and dynamic content. The DOM works with the structured layout of web apps and the parent and child aspects of this structure. It does this by treating them as a tree like structure of varying nodes.* | 6 Marks |
| 3 | What does it mean for a data structure to be described as a functor? Give a code example in Javascript in your explanation  *A functor is a function which takes in a function and/or a value/ set of values and applies the function to these values. It removes the need to unwrap arrays and applies a function to all elements. The map function in JS is a functor.*  *[1, 2, 3].map(n => n+1) // ==>> [2, 3, 4]*  *['123', '456', '789']*  *.map(s => parseInt(s))*  *.map(n => n / 10) // ==>> [12.3, 45.6, 78.9]* | 6 Marks |
| 4 | What is the de facto standard for data serialisation in the web app world? Give an example  *Creating objects for serialization is a standard and JSON is now used for structuring this data for simpler serialization for transport and storage.*  *JavaScript Object Notation (JSON) is used to structure object as seen below:*  *{*  *"employees":[*  *{"firstName":"John", "lastName":"Doe"},*  *{"firstName":"Anna", "lastName":"Smith"},*  *{"firstName":"Peter", "lastName":"Jones"}*  *]*  *}* | 6 Marks |
| 5 | Describe how the flexbox model works in CSS  *Flexbox or Flexible Layout is a way to cater for dynamic content, varied screen sizes and other unknown variables to best fill the screen and fit its content. This is done using a flex container which holds flex items. The items are arranged based on the flex properties assigned to it and its container. Flex containers can have flex directions to allow for rows or inline column style layouts. There are many alignment and positioning elements for flex containers and flex items.* | 6 Marks |
| 6 | Explain how you can make a network request to a server-side resource in a web app using Javascript  *AJAX is used to make network requests and supply dynamic requests without the need to reload the entire web page. This is achieved with a callback function.*  *XMLHTTPRequest is the most basic network request API for these dynamic requests. They can request a resource and return it in an object.*  *Cors or cross-origin-request policy allow for requests to multiple different origins and allow for GET and POST requests.*  *The Fetch API is the newer way to complete these requests and replaces the XMLHTTPRequest API.* | 6 Marks |
| 7 | CSS allows the reuse of code for styling DOM elements. Javascript functions can all be used for element styling and support code reuse. Compare the two approaches  *CSS styling should be used wherever possible to make your web app standardized and cross platform as CSS is the styling standard. CSS can be cached also saving load time. CSS is generally maintainable.*  *Javascript styling is great for dynamic content and classes. They can be written once and applied to many different elements. It is best for elements which are not predictable or uniform. Javascript may not always have browser compatibility.* | 6 Marks |
| 8 | In asynchronous programming, we have three approaches to handling data which may or may not arrive at some point in the future, namely callbacks, promises and streams. Describe each of these approaches. Are there any significant drawbacks of each in your opinion?  *Callbacks are blocks of code executed when a request has been fulfilled. Callbacks are used to read data from the network or storage and can take a long time but allows the other processes to continue and completes in the background. The problem with callbacks is if callbacks are nested they become difficult to debug and handle potential errors.*  *Promises provide a deliverable in the future.This deliverable is either a success or failure. The promise has a constructor and return a resolve or reject. Promises are easily chainable using the Then operator. Debugging and error handling is a lot easier using this structure of resolve and rejects and then chaining.*  *Streams are used in all types of asynchronous programming from data handling to mouse input, these are all streams. Reactive programming allows you to turn anything into a stream for processing. This allows you to listen to anything and react to various events. Streams can be utilized for large data sets in which you don't know when it will be completed. Streams are time separated into chunks for processing. This creates ongoing events ordered in time. The listening to of these streams is called subscribing to it. Functions are created when the stream is executed, errors occur or the stream has finished. These functions are called observers, taken from the Observer design pattern.* | 6 Marks |
| 9 | Complete the Q6A survey for the module through webcourses  *Done.* | 10 Marks |

## Problem Sets (Part 2)

Coming soon…