

COS30043

Interface Design and Development



Lecture 12 – Summary

2022 – Semester 1



1

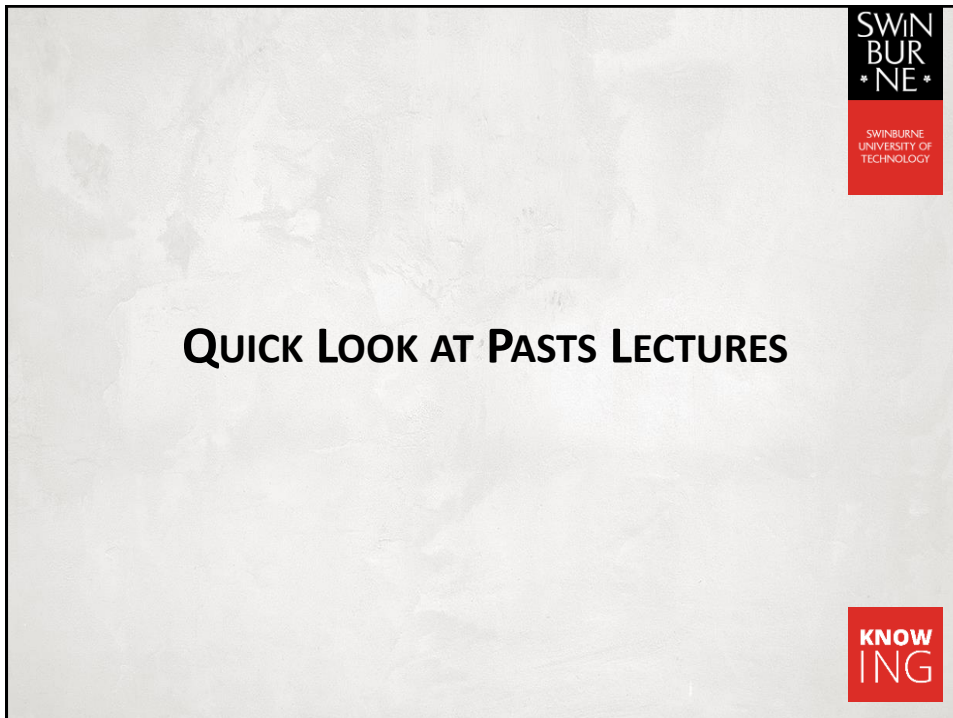
Contents

- Quick Look at Past Lectures
- Usability, Accessibility and Testing
- The Framework – Bootstrap and VueJS
- Single Page Application
- Asynchronous Data



2 - Interface Design and Development, © Swinburne

2



3

Web Development – Languages

- **HyperText Markup Language (HTML)** is a markup language designed to specify structure and content of a web page
 - HTML is NOT a “programming” language
 - HTML is NOT a “formatting” language
- **Cascading Style Sheets (CSS)** a simple markup language for adding style (e.g., fonts, colors, spacing) to Web documents.
- **Client-side scripting (JavaScript)** is a language that runs on a client’s browser (client tier) instead of on a Web server (processing tier)

4 - Interface Design and Development, © Swinburne



4

Web Development – Languages

- JavaScript is prone to errors
 - Many unexpected behaviours
 - hard to test (client-server)
 - hard to build tool support

5 - Interface Design and Development, © Swinburne



5

Web Development – Languages

- Server-side scripting refers to a scripting language that is executed at a Web server
- Hypertext Preprocessor (PHP) is a server-side embedded scripting language that is used to develop interactive Web sites
 - Includes object-oriented programming capabilities
 - Supports many types of databases (MySQL, Oracle, Sybase, ODBC-compliant)
- Others - Active Server Pages (ASP), ASP.NET, Cold Fusion, ... and more
- Server-side scripting is not that interactive

6 - Interface Design and Development, © Swinburne



6



7

Usability

- Usability is about the purpose or use that visitors have, and how they utilise a web site to achieve goals (tasks).
- Good Usability is *also* about ensuring good accessibility
 - websites should be accessible to all users, all devices
 - accessibility is so important it has it's own requirements!



8

Accessibility

- Web accessibility means that people with disabilities can use the Web
- More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web.
- Web accessibility also helps older people with changing abilities due to aging, and those who have temporary impairments.



9 - Interface Design and Development, © Swinburne

9

Testing

- increase in productivity
- decrease in user training requirements
- decrease in calls to the Help Desk and need for technical support
- decrease in user error rate
- decrease in programming costs associated with late design
- decrease in maintenance costs.



10 - Interface Design and Development, © Swinburne

10



11

Bootstrap

- Supports a responsive mobile first fluid system
- Scales up to 12 columns as the device or viewport size increases
- Provides predefined layout (grid) classes
- Uses rows to create horizontal groups of columns
- Version 2 is "responsive"
- Version 3 is "mobile first"

- Bootstrap 5 was officially released on 5 May 2021

12 - Interface Design and Development, © Swinburne



12

Bootstrap: Row-Column Format

- Always observe the row-column format
- Remember to use "col-xs-12" for all single column row

```
<div class="row">  
  <div class="col-xs-12">  
    <h1>Hello World</h1>  
  </div>  
</div>
```

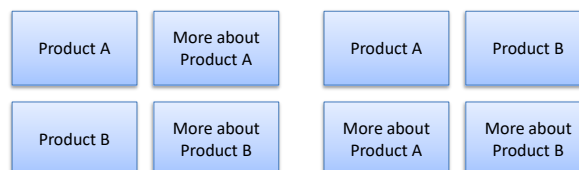
13 - Interface Design and Development, © Swinburne



13

Bootstrap: Context Grouping

- Visually they look the same
- But contextually they are not



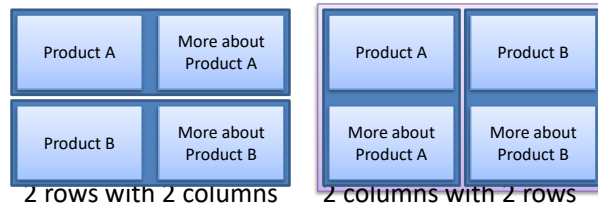
14 - Interface Design and Development, © Swinburne



14

Bootstrap: Context Grouping

- Design must be based on context not visual
- For example

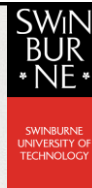


15 - Interface Design and Development, © Swinburne



15

THE FRAMEWORK – VUE JS



16

Vue Js Versions

Versions [edit]

Version	Release date	Title
3.0	September 18, 2020	One Piece ^[17]
2.6	February 4, 2019	Macross ^[18]
2.5	October 13, 2017	Level E ^[19]
2.4	July 13, 2017	Kill la Kill ^[20]
2.3	April 27, 2017	JoJo's Bizarre Adventure ^[21]
2.2	February 26, 2017	Initial D ^[22]
2.1	November 22, 2016	Hunter X Hunter ^[23]
2.0	September 30, 2016	Ghost in the Shell ^[24]
1.0	October 27, 2015	Evangelion ^[25]
0.12	June 12, 2015	Dragon Ball ^[26]
0.11	November 7, 2014	Cowboy Bebop ^[27]
0.10	March 23, 2014	Blade Runner ^[28]
0.9	February 25, 2014	Animatrix ^[29]
0.8	January 27, 2014	N/A ^[30]
0.7	December 24, 2013	N/A ^[31]
0.6	December 8, 2013	VueJS ^[32]

<https://en.wikipedia.org/wiki/Vue.js#Versions>

17 - Interface Design and Development, © Swinburne



17

MVVM

MVVM is an architectural pattern that separates an application into three main logical components: the model, the view and the view model

- Model: It holds the data/information of the app which is to be presented to the user for manipulation or interaction.
- View: It is used to render the information to the user.
 - the View doesn't know about the Model and vice-versa.
 - The View passes user input to the VM for processing.
 - The View presents 'state' defined by the VM to user.
- VieModel: The VM is the link between the Model and the View. It defines all business logics, such as
 - manipulate the data contained in the model
 - handle user interaction
 - Format data in the model and pass it to view

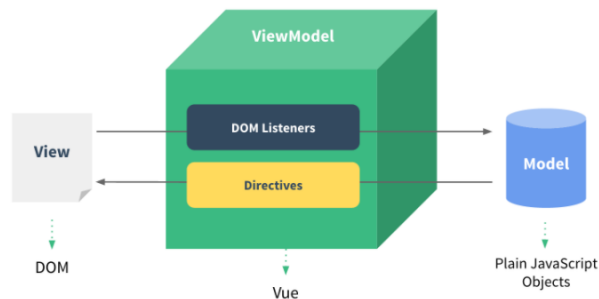


18 - Interface Design and Development,

18

VueJS and MVVM

- Vue.js is a framework for building interactive web interfaces.
- Vue.js is focused on the ViewModel layer of the MVVM pattern. It connects the View and the Model via two way data bindings.



19 - Interface Design and Development, © Swinburne

19

Topics Include

- Data binding: v-bind v-model
- Directives: v-if v-for v-on
- Vue instance options: methods, filters, computed, watch...
- Component
- Form validation, Vuetify
- Routing
- Use external data, RESTful API
- Single page application and use of Vue CLI

20 - Interface Design and Development, © Swinburne

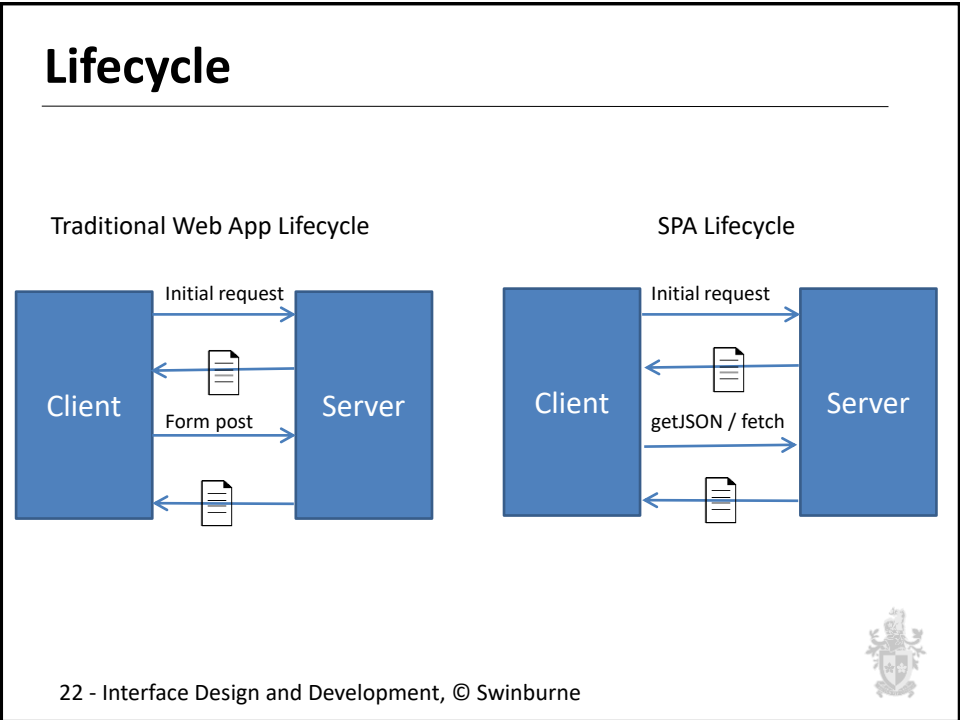
20

SWINBURNE
UNIVERSITY OF
TECHNOLOGY

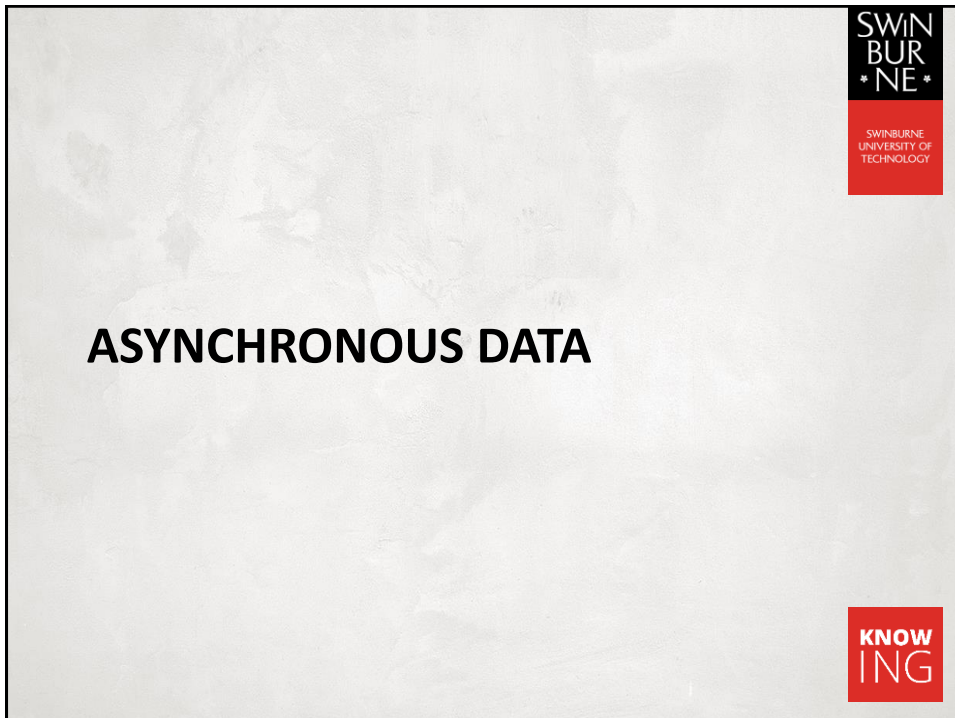
KNOW
ING

SINGLE PAGE APP

21



22



23

Fetch

```
fetch(url, requestOptions)
  .then( response =>{
    //turning the response into the usable data
    return response.json( );
  })
  .then( data =>{
    //This is the data you wanted to get from url
    self.msg=data;
  })
  .catch(error => {
    self.err=error
  });
```

24 - Interface Design and Development, © Swinburne



24

