



|  |   |
|--|---|
| <b>Title:</b>  | Front End Frameworks <b>APPROVED</b>  |
| <b>Long Title:</b>   | Front End Frameworks  |
| <b>Module Code:</b>  | SOFT8034  |
| <b>Duration:</b>   | 1 Semester  |
| <b>Credits:</b>  | 5   |
| <b>NFQ Level:</b>  | Advanced  |
| <b>Field of Study:</b>   | Computer Software   |
| <b>Valid From:</b>   | Semester 1 - 2017/18 ( September 2017 )   |
| <b>Module Delivered in</b>   | <a href="#">2 programme(s)</a>  |
| <b>Module Coordinator:</b>   | Sean McSweeney  |
| <b>Module Author:</b>  | GARY COUSE  |
| <b>Module Description:</b>   | The use of third party frameworks in conjunction with web-standards has been prevalent in the development of web applications' front-ends and interfaces. This module gives the student the skills necessary to select the most appropriate Web Standards frameworks for a specific front-end project, and to learn how best to integrate these frameworks properly into those projects. They will also learn the advanced software techniques used by many of the most popular frameworks so that they can derive maximum benefit from their use. They will also learn how to use frameworks and other reusable components to improve their development workflows. |
| <b>Learning Outcomes</b>   |   |
| <i>On successful completion of this module the learner will be able to:</i>  |   |
| LO1  | Assess how front end frameworks can improve the development workflow for front end applications.  |
| LO2  | Identify appropriate criteria in evaluating web frameworks and select the most appropriate front end framework based on an applications requirements.   |
| LO3  | Apply an appropriate front end framework technology to enhance layouts and functionality within an application.   |
| LO4  | Employ software techniques and practices commonly used by reusable software so that they can maximise the benefits of widely used frameworks.   |
| <b>Pre-requisite learning</b>  |   |
| <b>Module Recommendations</b>  |   |
| <i>This is prior learning (or a practical skill) that is strongly recommended before enrolment in this module. You may enrol in this module if you have not acquired the recommended learning but you will have considerable difficulty in passing (i.e. achieving the learning outcomes of) the module. While the prior learning is expressed as named CIT module(s) it also allows for learning (in another module or modules) which is equivalent to the learning specified in the named module(s).</i> |   |
| <b>Incompatible Modules</b>  |   |
| <i>These are modules which have learning outcomes that are too similar to the learning outcomes of this module. You may not earn additional credit for the same learning and therefore you may not enrol in this module if you have successfully completed any modules in the incompatible list.</i>   |   |
| No incompatible modules listed   |   |
| <b>Co-requisite Modules</b>  |   |
| No Co-requisite modules listed   |   |
| <b>Requirements</b>  |   |
| <i>This is prior learning (or a practical skill) that is mandatory before enrolment in this module is allowed. You may not enrol on this module if you have not acquired the learning specified in this section.</i>   |   |
| No requirements listed   |   |

**Module Content & Assessment**

**Indicative Content**

**Development workflow**

Web framework overview. Advantages and disadvantages. Features - Version Control, CSS Preprocessing: variables, operators, interpolations, functions, mixins. Extensions. Case Studies: Sass, Less, Stylus.

**Layout**

HTML boiler plate, CSS Reset stylesheets, Grids, Responsive Design, Case Studies: Bootstrap, Boilerplate, Pure

**Software Practices/Techniques**

Design Patterns: Routing, Databinding, Templates, Data Access, Models, MV\*

**Selecting Frameworks**

Assessing Frameworks. Security, licensing, maintenance, performance, size, standards compliance, browser compatibility, obtrusiveness

**Researching and Using Frameworks**

Documentation, Components, packages, persistence, controllers, templates, scope. Case Studies: E.g. Ember, React, AngularJS, Backbone.JS

**Assessment Breakdown**

|             | %       |
|-------------|---------|
| Course Work | 100.00% |

**Course Work**

| Assessment Type | Assessment Description  | Outcome addressed | % of total | Assessment Date |
|-----------------|---|-------------------|------------|-----------------|
| Project         | Assess and select the most appropriate front end framework to meet the functional requirements of a web application interface.                    | 1,2               | 35.0       | Week 7          |
| Essay           | An example title of the essay would be to critically evaluate an appropriate front end framework for a dynamic web application.                   | 1,2,3,4           | 25.0       | Week 9          |
| Project         | Create a specified interactive web application using your chosen web standards front-end frameworks (applying more advanced software techniques). | 3,4               | 40.0       | Week 13         |

No End of Module Formal Examination

**Reassessment Requirement**

**Coursework Only**

*This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination.*

**The institute reserves the right to alter the nature and timings of assessment**

**Module Workload**

| <b>Workload: Full Time</b>                    |  |              |                  |  |
|---|--|--------------|------------------|--|
| <i>Workload Type</i>                          | <i>Workload Description</i>                | <i>Hours</i> | <i>Frequency</i> | <i>Average Weekly Learner Workload</i> |
| Lecture                                       | Lecture underpinning learning outcomes.    | 2.0          | Every Week       | 2.00                                   |
| Lab   | Lab supporting content delivered in class. | 2.0          | Every Week       | 2.00                                   |
| Independent & Directed Learning (Non-contact) | Independent Study.                         | 3.0          | Every Week       | 3.00                                   |
| Total Hours                                   |  |              |                  | 7.00                                   |
| Total Weekly Learner Workload                 |  |              |                  | 7.00                                   |
| Total Weekly Contact Hours                    |  |              |                  | 4.00                                   |

| <b>Workload: Part Time</b>                    |  |              |                  |  |
|---|--|--------------|------------------|--|
| <i>Workload Type</i>                          | <i>Workload Description</i>                | <i>Hours</i> | <i>Frequency</i> | <i>Average Weekly Learner Workload</i> |
| Lecture                                       | Lecture underpinning learning outcomes.    | 2.0          | Every Week       | 2.00                                   |
| Lab   | Lab supporting content delivered in class. | 2.0          | Every Week       | 2.00                                   |
| Independent & Directed Learning (Non-contact) | Independent Study.                         | 3.0          | Every Week       | 3.00                                   |
| Total Hours                                   |  |              |                  | 7.00                                   |
| Total Weekly Learner Workload                 |  |              |                  | 7.00                                   |
| Total Weekly Contact Hours                    |  |              |                  | 4.00                                   |

## Module Resources

### Recommended Book Resources

- Pam Selle, Tim Ruffles, Christopher Hiller, Jamie White, *Choosing a JavaScript Framework: A comparison of pros, cons, and trade-offs*, 2014 Ed., Bleeding Edge Press [ISBN: 9781939902085]
- Addy Osmani 2012, *Learning JavaScript Design Patterns*, O'Reilly [ISBN: 9781449331818]

### Supplementary Book Resources

- Addy Osmani 2014, *Developing Backbone.js Applications*, O'Reilly [ISBN: 9781449328252]
- Stoyan Stefanov 2016, *React: Up & Running: Building Web Applications*, O'Reilly [ISBN: 9781491931820]
- Shyam Seshadri, Brad Green 2014, *AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps*, O'Reilly [ISBN: 9781491901946]
- Stoyan Stefanov 2010, *JavaScript Patterns: Build Better Applications with Coding and Design Patterns*, O'Reilly 9780596806750 [ISBN: 9780596806750]

This module does not have any article/paper resources

### Other Resources

- Website: *jQuery*  
<https://jquery.com>
- Website: *Boilerplate*  
<https://html5boilerplate.com>
- Website: *Bootstrap*  
<http://getbootstrap.com>
- Website: *Pure.CSS*  
<http://purecss.io>
- Website: *AngularJS*  
<https://angularjs.org>
- Website: *Backbone.js*  
<http://backbonejs.org>
- Website: *Ember*  
<http://emberjs.com>
- Website: *React*  
<https://facebook.github.io/react/>
- Website: *Sass*  
<http://sass-lang.com>
- Website: *Less*  
<http://lesscss.org>

**Module Delivered in**

| <b>Programme Code</b> | <b>Programme</b>   | <b>Semester</b> | <b>Delivery</b> |
|-----------------------|--|-----------------|-----------------|
| CR_KSDEV_8            | <a href="#"><u>Bachelor of Science (Honours) in Software Development</u></a> | 7               | Elective        |
| CR_KDNET_8            | <a href="#"><u>Bachelor of Science (Honours) in Computer Systems</u></a>     | 7               | Elective        |