1. **Title of the module**

COMP5390 (CO539) - Web Development

1. **Division or partner institution which will be responsible for management of the module**

Division of Computing, Engineering, Mathematical Sciences (CEMS)

1. **The level of the module (Level 4, Level 5, Level 6 or Level 7)**

Level 5

1. **The number of credits and the ECTS value which the module represents**

15 credits (7.5 ECTS)

1. **Which term(s) the module is to be taught in (or other teaching pattern)**

Autumn or Spring

1. **Prerequisite and co-requisite modules**

Pre-requisite (Year in Computing):

COMP5830: An Introduction to Programming and Web Technologies

Co-requisite (2nd year direct entry):

COMP5230: Fundamentals of Programming and Logic

Pre-requisite (all other Canterbury courses):

COMP3200: Introduction to Object-Oriented Programming

COMP3230: Databases and the Web

1. **The course(s) of study to which the module contributes**

BSc Computer Science, including all variants, both with and without Year in Industry.

BSc Business Information Technology, BSc Computing, both with and without Year in Industry.

BSc Artificial Intelligence, BSc Data Science, BSc Software Engineering, both with and without Year in Industry.

1. **The intended subject specific learning outcomes.  
   On successfully completing the module students will be able to:**

8.1 Build and deploy highly interactive, scalable and maintainable web-based systems using various tools, platforms and frameworks.

8.2 Understand the technologies, and the usability and performance tradeoffs, involved in creating highly interactive web-based applications.

8.3 Implement simple web services and understand the relationship between web sites and web services

8.4 Build responsive systems for mobile devices, using the web and as applications.

1. **The intended generic learning outcomes.  
   On successfully completing the module students will be able to:**

9.1 Demonstrate comprehension of the trade-offs involved in design-choices.

9.2 Make effective use of IT facilities and information sources for solving problems.

9.3 Be able to manage their own learning and development, through self-directed study and working on continuous assessment.

9.4 Make effective use of a range of tools, such as a web browser and database query browser.

1. **A synopsis of the curriculum**

Building scaleable web sites using client-side and and server-side frameworks (e.g. JQuery, CodeIgniter). Data transfer technologies, e.g. XML and JSON. Building highly interactive web sites using e.g. AJAX. Web services. Deploying applications and services to the web: servers, infrastructure services, and traffic and performance analysis. Web and application development for mobile devices.

1. **Reading list (Indicative list, current at time of publication. Reading lists will be published annually)**

Robin Nixon Learning PHP, MySQL, and JavaScript: A Step-by-Step Guide to Creating Dynamic Websites, O’Reilly, 2009

Adam Tracy, Robert Hamson, Jason Essington and Anna Tokke, GWT in Action, Manning, 2nd Edition, 2013.

Wei-Meng Lee, Beginning Android 4 Development, Wrox (Wiley), 2012

1. **Learning and teaching methods**

Total contact hours: 26

Private study hours: 124

Total study hours: 150

1. **Assessment methods**
   1. Main assessment methods

CodeIgniter software development (25%)

AJAX software development (25%)

2-hour unseen written examination (50%)

13.2 Reassessment methods

Like for like.

1. **Map of module learning outcomes (sections 8 & 9) to learning and teaching methods (section12) and methods of assessment (section 13)**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Module learning outcome** |  | *8.1* | *8.2* | *8.3* | *8.4* | *9.1* | *9.2* | *9.3* | *9.4* |
| **Learning/ teaching method** | **Hours allocated** |  |  |  |  |  |  |  |  |
| **Private Study** | 74 | **X** | **X** | **X** | **X** | **X** | **X** | **X** | **X** |
| **Lectures** | 22 |  | **X** | **X** |  | **X** |  |  |  |
| Practical Classes | 4 | **X** |  | **X** | **X** |  | **X** | **X** | **X** |
| **Assessment method** |  |  |  |  |  |  |  |  |  |
| **Practical Assessments** | 50 | **X** |  | **X** | **X** |  | **X** | **X** | **X** |
| **Examination** |  |  | **X** | **X** |  | **X** |  |  |  |

1. **Inclusive module design**

The Division recognises and has embedded the expectations of current equality legislation, by ensuring that the module is as accessible as possible by design. Additional alternative arrangements for students with Inclusive Learning Plans (ILPs)/declared disabilities will be made on an individual basis, in consultation with the relevant policies and support services.

The inclusive practices in the guidance (see Annex B Appendix A) have been considered in order to support all students in the following areas:

a) Accessible resources and curriculum

b) Learning, teaching and assessment methods

1. **Campus(es) or centre(s) where module will be delivered**

Canterbury

1. **Internationalisation**

The topics addressed by this module relate to a field which is of international importance, given the global role of computers in today's technological innovation. The topics covered by this module are international in nature, being identical worldwide and independent of traditional spoken language.

**DIVISIONAL USE ONLY**

**Revision record – all revisions must be recorded in the grid and full details of the change retained in the appropriate committee records.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date approved | Major/minor revision | Start date of delivery of revised version | Section revised | Impacts PLOs (Q6&7 cover sheet) |
| 10/11/2020 | Minor |  | 1, 6, 7, 16 | No |
|  |  |  |  |  |