Module Template for New and Revised Modules[1]

Module Code	EEMT05
Module Name	CREATIVE CODING
ECTS Weighting[2]	5 ECTS
Semester taught	Semester 1
Module Coordinator/s	
Module Learning Outcomes with reference to the Graduate Attributes and how they are developed in discipline	On successful completion of this module, students should be able to: To think independently. To communicate effectively To write effective computer code to realize graphical and interactive applications.
Module Content	Creative Coding is a one-semester module. It covers introductory concepts for students without a programming background or those at the level of novice. The course is intended to enable students to develop programs in the Processing language - a streamlined version of Java originally designed to teach coding fundamentals within a visual context. The focus is on writing code for creative and/ or artistic applications; interactive sketches with graphics, animation and sound. The module is intended to enable students to develop programs in the Processing language - a streamlined version of Java originally designed to teach programming fundamentals within a visual context. On successful completion of this module, students will be able to: • Use the Processing Development Environment to run and write programs. • Design and plan application implementations (with pseudocode). • Write code at an intermediate level, producing working programs with a focus on interactive music and multimedia applications.

[1] An Introduction to Module Design from AISHE provides a great deal of information on designing and re-designing modules.
[2] TEP Glossary

Teaching and Learning Methods	The teaching strategy is a mixture of lectures and problem-solving tutorials. The format of lectures uses informal interaction as well as formal delivery of theory. There is a significant practical aspect - material delivered in lectures is put into practice using exercises and coding tasks.

Assessment Details[3]
Please include the following:

- **Assessment Component**
- **Assessment description**
- **Learning Outcome(s) addressed**
- **% of total**
- **Assessment due date**

Assessment Component	Assessment Description	LO Addressed	% of total	Week due
2 Assignments	Coding exercises exploiting various software strategies	Independent Coding Implementation	40% and 60%	Week 4 and Week 12

Reassessment Requirements

Contact Hours and Indicative Student Workload[3]

Contact hours: 11 Lecture hours and 11 tutorial hours
Independent Study (preparation for course and review of materials): 30 hours
Independent Study (preparation for assessment, incl. completion of assessment): 20 hours

Recommended Reading List

Ira Greenberg: Processing: Creative Coding and Computational Art
Daniel Shiffman: Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction
Casey Reas and Ben Fry: Getting Started with Processing
On Line Resources: https://processing.org

Module Pre-requisite

None

Module Co-requisite

None

Module Website

None

Are other Schools/Departments involved in the delivery of this module? If yes, please provide details.

No

Module Approval Date

Approved by

[3] TEP Guidelines on Workload and Assessment

Academic Start Year
Academic Year of Date