Module Descriptor

Title	Secure Programming		
Session	2025/26	**Status**	Published
Code	COMP10068	**SCQF Level**	10
Credit Points	20	**ECTS (European Credit Transfer Scheme)**	10
School	**Computing, Engineering and Physical Sciences**		
Module Co-ordinator	Paul Keir		

Summary of Module

Security in software begins with an initial design and engineering effort, conscious of classic and contemporary security vulnerabilities; as well as corresponding remedial actions and protocols. In this hands-on module we will first explore the nature of secure programming before introducing a taxonomy of established coding errors, as well as information sources such as the MITRE reference system for Common Vulnerabilities and Exposures (CVE). Conventional programming languages including assembly language, C, C++ and Java, along with related compiler tools, form a foundation for the module, while the benefits of contemporary languages such as Mozilla's Rust and Apple's Swift are also thoroughly analysed. The relevance of strong, static typing; functional programming; advanced type systems; and theorem provers for secure software development will also be introduced.

This module will work to develop a number of the key 'I am UWS' Graduate Attributes to make those who complete this module: Universal (Analytical, Critical Thinker & Socially Responsible), Work Ready (Digitally Literate, Problem-Solver& Ambitious), and Successful (Incisive, Creative & Autonomous).

Module Delivery Method	**On-Campus[1]** ☒	**Hybrid[2]** ☐	**Online[3]** ☐	**Work -Based Learning[4]** ☐
Campuses for Module Delivery	☐ Ayr ☐ Dumfries	☒ Lanarkshire ☐ London	☒ Online / Distance Learning ☐ Other (specify)	

[1] Where contact hours are synchronous/ live and take place fully on campus. Campus-based learning is focused on providing an interactive learning experience supported by a range of digitally-enabled asynchronous learning opportunities including learning materials, resources, and opportunities provided via the virtual learning environment. On-campus contact hours will be clearly articulated to students.

[2] The module includes a combination of synchronous/ live on-campus and online learning events. These will be supported by a range of digitally-enabled asynchronous learning opportunities including learning materials, resources, and opportunities provided via the virtual learning environment. On-campus and online contact hours will be clearly articulated to students.

[3] Where all learning is solely delivered by web-based or internet-based technologies and the participants can engage in all learning activities through these means. All required contact hours will be clearly articulated to students.

[4] Learning activities where the main location for the learning experience is in the workplace. All required contact hours, whether online or on campus, will be clearly articulated to students

			☐ Paisley		Online Delivery / Distance Learning applies to delivery in the BSc (Hons) Data, AI and Software Engineering programme only	
Terms for Module Delivery	Term 1	☐	Term 2	☒	Term 3	☐
Long-thin Delivery over more than one Term	Term 1 – Term 2	☐	Term 2 – Term 3	☐	Term 3 – Term 1	☐

Learning Outcomes	
L1	Demonstrate knowledge that covers and integrates most of the principal areas, features, boundaries, terminology and conventions of cyber security and secure programming.
L2	Critically identify, define, conceptualise and analyse both public and private programmatic security hazards.
L3	Use a range of tools and formal methods to audit and support the development of secure software.
L4	Apply knowledge, skills and understanding of the security features offered by a range of programming languages and libraries.
L5	N/A

Employability Skills and Personal Development Planning (PDP) Skills	
SCQF Headings	**During completion of this module, there will be an opportunity to achieve core skills in:**
Knowledge and Understanding (K and U)	**SCQF 10** Recognise CVE ID numbers, and prepare a response appropriate to the associated threat level. Comprehend the relationship between a programming language and the underlying computer hardware; the abstract machine.
Practice: Applied Knowledge and Understanding	**SCQF 10** Apply standard secure coding guidelines to avoid common security loopholes. Demonstrate the utility of tools such as compilers; debuggers; profilers; model checkers; and virtual machines for secure programming.
Generic Cognitive skills	**SCQF 10** Understand the advantages and limitations of programming within an advanced type system. Appreciate the feature set of libraries for authentication and encryption.
Communication, ICT and Numeracy Skills	**SCQF 10** Apply secure software development principles to a range of application domains.

Autonomy, Accountability and Working with Others	**SCQF 10** Manage complex professional issues in accordance with current professional codes or practices: in particular, following published coding standards for specific programming languages.

Prerequisites	**Module Code**	**Module Title**
	Other	
Co-requisites	**Module Code**	**Module Title**

Learning and Teaching	
In line with current learning and teaching principles, a 20-credit module includes 200 learning hours, normally including a minimum of 36 contact hours and maximum of 48 contact hours.	
Learning Activities During completion of this module, the learning activities undertaken to achieve the module learning outcomes are stated below:	**Student Learning Hours** (Note: Learning hours include both contact hours and hours spent on other learning activities)
Lecture / Core Content Delivery	24
Tutorial / Synchronous Support Activity	12
Laboratory / Practical Demonstration / Workshop	12
Independent Study	152
Please select	
Please select	
TOTAL	200

Indicative Resources
The following materials form essential underpinning for the module content and ultimately for the learning outcomes: The following materials form essential underpinning for the module content and ultimately for the learning outcomes: Robert C. Seacord. Secure Coding in C and C++, Second Edition, Addison Wesley, 2013 Steve Klabnik and Carol Nichols. The Rust Programming Language (2023) Jim Blandy and Jason Orendorff. Programming Rust: Fast, Safe Systems Development, O'Reilly Media, 2017 Secure Programming HOWTO - Creating Secure Software by David Wheeler John Viega and Matt Messier. Secure Programming Cookbook for C and C++, O'Reilly Media, 2003 Brian Chess and Jacob West. Secure Programming with Static Analysis, Addison-Wesley Professional, 2007

SEI CERT C++ Coding Standard: Rules for Developing Safe, Reliable, and Secure Systems in C++ (2016 Edition) available online at https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88046682
(N.B. Although reading lists should include current publications, students are advised (particularly for material marked with an asterisk*) to wait until the start of session for confirmation of the most up-to-date material)

Attendance and Engagement Requirements
In line with the Student Attendance and Engagement Procedure, Students are academically engaged if they are regularly attending and participating in timetabled on-campus and online teaching sessions, asynchronous online learning activities, course-related learning resources, and complete assessments and submit these on time. **For the purposes of this module, academic engagement equates to the following:** The School of Computing, Engineering and Physical Sciences considers attendance and engagement to mean a commitment to attending, and engaging in, timetabled sessions. You will scan your attendance via the scanners each time you are on-campus and you will login to the VLE several times per week. Where you are unable to attend a timetabled learning session due to illness or other circumstance, you should notify the Programme Leader that you cannot attend. Across the School an 80% attendance threshold is set. If you fall below this, you will be referred to the Student Success Team to see how we can best support your studies.

Equality and Diversity
The University's Equality, Diversity and Human Rights Procedure can be accessed at the following link: UWS Equality, Diversity and Human Rights Code. Aligned with the University's commitment to equality and diversity, this module supports equality of opportunity for students from all backgrounds and learning needs. Using the VLE, material will be presented electronically in formats that allow flexible access and manipulation of content. This module complies with University regulations and guidance on inclusive learning and teaching practice. This module has lab-based teaching and as such you are advised to speak to the Module Co-ordinator to ensure that specialist assistive equipment, support provision and adjustment to assessment practice can be put in place, in accordance with the University's policies and regulations.
(N.B. Every effort will be made by the University to accommodate any equality and diversity issues brought to the attention of the School)

Supplemental Information

Divisional Programme Board	**Computing**
Overall Assessment Results	☐ **Pass / Fail** ☒ **Graded**
Module Eligible for Compensation	☐ **Yes** ☒ **No** **If this module is eligible for compensation, there may be cases where compensation is not permitted due to programme accreditation requirements. Please check the associated programme specification for details.**
School Assessment Board	Business and Applied Computing
Moderator	Graham Parsonage
External Examiner	Anish Jindal

Accreditation Details	
Module Appears in CPD catalogue	☐ **Yes** ☒ **No**
Changes / Version Number	1.11

Assessment (also refer to Assessment Outcomes Grids below)
Assessment 1
One coursework assignment worth 30% of the overall mark.
Assessment 2
One coursework assignment worth 40% of the overall mark.
Assessment 3
One class test worth 30% of the overall mark.
(N.B. (i) Assessment Outcomes Grids for the module (one for each component) can be found below which clearly demonstrate how the learning outcomes of the module will be assessed. (ii) An indicative schedule listing approximate times within the academic calendar when assessment is likely to feature will be provided within the Student Module Handbook.)

Component 1							
Assessment Type	**LO1**	**LO2**	**LO3**	**LO4**	**LO5**	**Weighting of Assessment Element (%)**	**Timetabled Contact Hours**
Laboratory/ Clinical/ Fieldnotebook	☐	☒	☒	☒	☐	30	

Component 2							
Assessment Type	**LO1**	**LO2**	**LO3**	**LO4**	**LO5**	**Weighting of Assessment Element (%)**	**Timetabled Contact Hours**
Laboratory/ Clinical/ Fieldnotebook	☐	☒	☒	☒	☐	40	

Component 3							
Assessment Type	**LO1**	**LO2**	**LO3**	**LO4**	**LO5**	**Weighting of Assessment Element (%)**	**Timetabled Contact Hours**
Class test (practical)	☒	☐	☐	☒	☐	30	
Combined total for all components						100%	hours

Change Control

What	**When**	**Who**

Attendance and Engagement Procedure and Equality and Diversity	17/1/25	F.Valentine