DEREE COLLEGE SYLLABUS FOR:

ITC 4247 SECURE SOFTWARE DEVELOPMENT **3/0/3**
(Previously: ITC 4447 SECURE SOFTWARE DEVELOPMENT) **UK LEVEL: 6**
(Updated Fall 2023) **UK CREDITS: 15**

PREREQUISITES:	ITC 2088 Introduction to Programming ITC 3160 Fundamentals of RDBMS
COREQUISITES:	None.
CATALOG DESCRIPTION:	Best practices for developing secure software; coding techniques for data validation, session management, exception handling, data encryption; configuration techniques. Mitigating security risk from external and internal sources.
RATIONALE:	The course focuses on the design and implementation of secure software. Students will explore secure coding and testing techniques.
LEARNING OUTCOMES:	As a result of taking this course, the student should be able to: 1. Explain the role of security throughout the Software Development Life Cycle process. 2. Determine software application security vulnerabilities and analyze attack consequences. 3. Apply secure design principles for developing attack resistant software. 4. Analyze insecure software, utilizing automated code review tools with static analysis and symbolic execution. 5. Compare tools and techniques for testing software resilience.
METHOD OF TEACHING AND LEARNING:	In congruence with the teaching and learning strategy of the college, the following tools are used: • Lectures and laboratory sessions. • Office hours held by the instructor to provide further assistance to students. • Use of the online content management system (Blackboard CMS) to further facilitate communication.

<table>
<tr>
<td>ASSESSMENT:</td>
<td>
Summative:
<table>
<tr><td>1st assessment: Midterm exam
Short answers and case problems</td><td>20%</td></tr>
<tr><td>2nd assessment: Project defence and presentation</td><td>10%</td></tr>
<tr><td>Final assessment: Group project
Design and assessment of secure SW policy for a given set of SW application requirements, including a programming implementation.</td><td>70%</td></tr>
</table>
Formative:
<table>
<tr><td>Take-home short problems</td><td>0%</td></tr>
</table>
The formative assessments aim to prepare students for the summative assessments.
The 1st summative assessment tests the LOs 1, 5.
The 2nd summative assessment tests the LOs 2-5.
The final summative assessment tests the LOs 2-5.

Students are required to resit failed assessments in this module.
</td>
</tr>
<tr>
<td>INDICATIVE READING:</td>
<td>
REQUIRED READING:

James Ransome & Anmol Misra. Core Software Security (Security at the Source), CRC Press, 2013, ISBN-13: 978-1466560956
Instructor notes.

RECOMMENDED READING:

Jason Grembi. Secure Software Development: A Security Programmer’s Guide, Cengage, 2006
Gray McGraw: Software Security – Building Security In, Addison Wesley, 2008

</td>
</tr>
<tr>
<td>INDICATIVE MATERIAL:
(e.g. audiovisual, digital material, etc.)</td>
<td>REQUIRED MATERIAL: N/A

RECOMMENDED MATERIAL: N/A</td>
</tr>
<tr>
<td>COMMUNICATION REQUIREMENTS:</td>
<td>Daily access to the course’s site on the College’s Blackboard CMS and the acg email.
Effective communication using proper written and oral English.
Use of word processing and/or presentations software for documentation and presentation of deliverables and the final project.</td>
</tr>
<tr>
<td>SOFTWARE REQUIREMENTS:</td>
<td>MS-Office
VMWare
Kali Linux
C, C++, Python, Java</td>
</tr>
</table>

WWW RESOURCES:	• https://www.sans.org/security-resources/policies/application-security/doc/web-application-security-policy • http://www.securitydevelopmentconference.com/ • https://distrinet.cs.kuleuven.be/events/essos/2013/ • http://paris.utdallas.edu/sere12/ • http://ce.sharif.edu/courses/91-92/2/ce384- • http://www.ares-conference.eu/conf/
INDICATIVE CONTENT:	1. Software Security principles and importance 2. Software assessment methods and techniques 3. Vulnerability classification and management 4. Assessment reporting 5. Software attack surface 6. Threat actors 7. Common attack patterns 8. Security controls