DEREE COLLEGE SYLLABUS FOR:

ITC 2205 SOFTWARE ENGINEERING PRACTICES **3/1/3**
(Fall 2021) **UK LEVEL: 4**
UK CREDITS: 15

PREREQUISITES:	ITC 2088 Introduction to Programming ITC 2197 Object Oriented Programming Techniques
COREQUISITES:	None.
CATALOG DESCRIPTION:	Software engineering (SE) major methodologies; sw qualities; sw lifecycle; CI/CD; essential tools; GitHub; concepts in project planning; code of ethics and professional practice.
RATIONALE:	The course introduces students to practical aspects of software engineering and professional practice with a focus on the development phase. In this context, students have the opportunity to develop skills for high and low-level software design and essential SE process tasks, particularly those for requirements identification, reliability, maintainability, system validation and testing, and become acquainted with contemporary practices and tools.
LEARNING OUTCOMES:	As a result of taking this course, the student should be able to: 1. Demonstrate understanding of the major SE methodologies and the SW development lifecycle. 2. Use essential SE tools and techniques. 3. Utilize contemporary code development practices. 4. Demonstrate understanding of the attributes of good software and the SE code of ethics in professional practice.
METHOD OF TEACHING AND LEARNING:	In congruence with the teaching and learning strategy of the college, the following tools are used: • Lectures, class discussions, laboratory practical sessions and problem solving. • Office hours: Students are encouraged to make full use of the office hours of their instructor, where they can ask questions and go over lecture material. • Use of the Blackboard Learning platform, where instructors post lecture notes, assignment instructions, timely announcements, as well as additional resources.
ASSESSMENT:	**Summative:** 1st assessment: Midterm Exam Short answers to problems and/or case studies — **30%** 2nd assessment: Portfolio of student work and oral assessment — **10%** Final assessment: Team project SE process and code development — **60%** **Formative:** case studies and coding problems — **0%** The formative assessments aim to prepare students for the summative assessments and expose them to teamwork.

	The 1st summative assessment tests LO 2, 3, 4. The 2nd summative assessment tests LO 1,5. The final summative assessment tests LOs 1-5. *The final grade for this module will be determined by averaging all summative assessment grades, based on predetermined weights for each assessment. If students pass the **final summative assessment,** which tests all Learning Outcomes for this module, and the average grade for the module is 40 or above, students are not required to resit any failed assessments.*
INDICATIVE READING:	**REQUIRED READING:** 1. Thomas D., Hunt A. (2019). *The Pragmatic Programmer: your journey to mastery. 20th anniversary*, 2nd ed. Addison-Wesley, Kindle edition available. 2. Instructor's notes. **RECOMMENDED READING:** 1. Loubser N. (2021). *Software Engineering for Absolute Beginners: Your Guide to Creating Software Products*, Apress, eBook available. 2. Richards M., Ford N. (2020). *Fundamentals of Software Architecture: An Engineering Approach*, 1st ed., O'Reilly Media, Kindle edition available. 3. Jacobson I. et al. (2019). *The Essentials of Modern Software Engineering*. ACM Books 4. Martin R.C. (2017). *Clean Architecture: A Craftman's Guide to Software Structure and Design*, 1st ed., Pearson, Kindle edition available 5. IEEE-CS/ACM Joint Task Force on SE Ethics and Professional Practices (2001). *Software Engineering Code of Ethics and Professional Practice*. Science and Engineering Ethics 7, pp 231-238 https://www.researchgate.net/publication/278417404_Software_Engineering_Code_of_Ethics_and_Professional_Practice
INDICATIVE MATERIAL: *(e.g. audiovisual, digital material, etc.)*	**REQUIRED MATERIAL:** N/A **RECOMMENDED MATERIAL:** Software Engineering – Quick Guide https://www.tutorialspoint.com/software_engineering/software_engineering_quick_guide.htm Fundamental Pragmatics for Successful Software Developers https://hackernoon.com/fundamental-pragmatics-for-successful-software-developers-79a9d327f430
COMMUNICATION REQUIREMENTS:	Daily access to the course's site on the College's Blackboard CMS. Effective presentation skills using proper written and oral English. Communicate and coordinate during team activities.
SOFTWARE REQUIREMENTS:	Ms Office MS Visio GitHub access Scrum, Kanban, Jira, Trello, Adobe XD SEMAT Essence platform Python, Java
WWW RESOURCES:	• http://sematacc.meteor.com • http://www.software-engineering-essentialized.com/book

	• https://www.adobe.com/products/xd.html# • https://kanbanize.com/kanban-resources/getting-started/what-is-kanban • https://www.atlassian.com/agile • https://www.atlassian.com/software/jira/free • https://trello.com/
INDICATIVE CONTENT:	1. Introduction to Software Engineering 2. SE process a. Major methodologies b. Tasks and tools c. Managing the process 3. Software lifecycle 4. Code development a. Modular architectures, reusable code, and version control (git) b. Continuous integration, continuous development (CI/CD) c. Test-driven development (TDD) 5. Unified Modelling Language (UML) 6. Securing the code 7. Validation and verification 8. SW quality 9. Code of ethics and professional practice