

Syllabus
SEN 201:(Introduction to Software Engineering :2 units C: LH 30)

Department of Software Engineering
Faculty of Computing
University of Delta, Agbor, Nigeria

Lecturer: Mr. Opone, Marshall Chime

Office Location: FOC Room 7

Email: marshall.opone@unidel.edu.ng

Phone: 08034260694

Office Hours: Monday, Tuesday, Wednesday, Thursday & Friday 8:00 am - 4:00 pm

You can access me in many ways. Face-to-face communication which often leads to more refined and focused questions resulting in your improved understanding is highly encouraged. Take advantage of my office hours as stated above. Questions during class or immediately after class are always welcomed. Email is an easy way to ask questions outside of class but is not productive as face-to-face communication.

Meeting Time and Place: Mondays, 8:00am to 10:00am, FOC LH 2

Attendance

You are expected to attend every class. If you must miss a class, it is your responsibility to make up for the work that you missed. If you are going to be absent from any class, you must please notify me through your class representative in advance.

Methods of Instruction

This syllabus contains an overview of what will be covered in class; for specific information, students are referred to the class web page maintained on the University website. Assignments will be posted on University of Delta LMS or given in the class and should be submitted through University of Delta LMS. Class attendance as well as participating in all your class and home works will help the borderline cases.

Overview

Software engineering is the branch of computer science that deals with the design, development, testing, and maintenance of software applications. Software engineers apply engineering principles and knowledge of programming languages to build software solutions for end users.

Objectives

At the end of this course, students should be able to:

1. describe the concept of the software life cycle;
2. explain the phases of requirements analysis, design, development, testing and maintenance in a typical software life cycle;
3. differentiate amongst the various software development models;
4. utilise UML for object-oriented analysis and design;
5. describe different design architectures;
6. explain the various tasks involved in software project management; and
7. describe the basic legal issues related to Software Engineering.

Learning outcomes

At the end of this course, students should be able to:

1. describe the concept of the software life cycle;
2. explain the phases of requirements analysis, design, development, testing and maintenance in a typical software life cycle;
3. differentiate amongst the various software development models;
4. utilise UML for object-oriented analysis and design;
5. describe different design architectures;
6. explain the various tasks involved in software project management; and
7. describe the basic legal issues related to Software Engineering.

Course Contents

Software Engineering concepts and principles. Design, development and testing of software systems. Software processes: software lifecycle and process models. Process assessment models. Software process metrics. Life cycle of software system. Software requirements and specifications. Software design. Software architecture. Software metrics. Software quality and testing. Software architecture. Software validation. Software evolution: software maintenance; characteristics of maintainable software; re-engineering; legacy systems; software reuse. Software Engineering and its place as a computing discipline. Software project management: team management; project scheduling; software measurement and estimation techniques; risk analysis; software quality assurance; software configuration management. Software Engineering and law.

Lecture Schedules

Week	Content	Lecture notes/slides
1.	Software Engineering concepts and principles.	
2.	Design, development and testing of software systems.	
3.	Software processes: software lifecycle and process models. Process assessment models.	
4.	Software process metrics. Life cycle of software system.	
5.	Software requirements and specifications.	
6.	Software design. Software architecture. Software metrics.	
7.	Software quality and testing. Software architecture. Software validation.	
8.	Software evolution: software maintenance; characteristics of maintainable software;	
9.	Re-engineering; legacy systems; software reuse.	
10.	Software Engineering and its place as a computing discipline.	
11.	Software project management: team management; project scheduling; software measurement and estimation techniques;	
12.	Risk analysis; software quality assurance;	
13.	Software configuration management. Software Engineering and law.	
14.	Revisions	
15.	Final Exam	

Examination schedule

- Attendance
- Homework
- Class Test
- End of Semester Exam

Grading

- Homework: 10% of grade
- Attendance and lecture material: 10% of grade
- Midterm Exam: 10% of grade
- Final Exam: 70% of grade

Text & References

All relevant resources as found in the University library as well as those in the faculty of computing library. Lecture notes will equally be used to support the libraries' materials.

Student Conduct

All students enrolled at the University shall follow the tenets of common decency and acceptable behaviour conducive to a positive learning environment. The code of student conduct is described in detail in the student handbook or University website.

Academic Honesty

"All students enrolled at the University shall follow the tenets of common decency and acceptable behaviour conducive for a positive learning environment." It is the policy of the University, that no form of plagiarism or cheating will be tolerated. Plagiarism is defined as the deliberate use of another's work and claiming it as one's own. This means ideas as well as text or code, whether paraphrased or presented verbatim (word-for-word). Cheating is defined as obtaining unauthorised assistance on any assignment. Proper citation of sources must always be utilised thoroughly and accurately. If you are caught sharing or using other people's work in this class, you will receive a 0 grade and a warning on the first instance. A subsequent instance will result in receiving an F grade for the course, and possible disciplinary proceedings. If you are unclear about what constitutes academic dishonesty, ask.

The screenshot shows the 'MANAGER COURSES' interface in the 'UNIDEL CMS' system. The page title is 'Courses'. The main content area contains a form for adding or editing course information. At the top, there is a 'Courseware Document' section with a 'Browse...' button, a 'No file selected.' status, and an 'UPLOAD PDF DOC' button. Below this are input fields for 'Course Code' and 'Course Title'. There are also dropdown menus for '1st' and '200 Level', and a 'FOC' button. Two large text areas are provided for 'Course Objective' and 'Course Synopsis'. At the bottom of the form, there are several input fields: 'Course Lecturer', 'Exam Mark', 'Test Mark', 'Assignment Mark', 'Lecturer Hour', 'Tutorial Hours', 'Practical Hours', and 'Course Unit'. At the very bottom, there are 'SAVE' and 'CANCEL' buttons.