

**COMP305: Software Engineering
Spring 2017**

SYLLABUS

COURSE DESCRIPTION

Analysis, design, implementation, and testing of a medium-scale software system as a member of a project team. Significant real-world group projects covering all the phases of software of development life cycle using high-level automated analysis and design tools. Experience with other important skills such as fact-finding, communications, and project management

CREDITS : 3

Course Objectives

This course fulfills suggested curriculum requirements of ACM, IEEE, and the Computer Science Accreditation Board for Computer Science departments. The course will provide the student with an overview of the tools and techniques of software engineering.

This course is designed to prepare the student to be able to join a team involved in software development projects and quickly become productive in that environment.

The course will provide the student with experience in system documentation by incorporating a major project that is to be completed by the students throughout the course of the semester. This project will involve significant writing, programming, and presentation/speaking as the students design, implement, and test their solutions. Each team will be expected to present its solution to the class providing the students an opportunity to prepare and deliver a technical presentation.

Object-Oriented development has become a hot topic in software development. This course explains object-oriented development by presenting the basic principles. It then introduces object-oriented analysis as a way of modeling user requirements and possible software solutions, and as a means of implementing the results, of offering possible software solutions, and as a means of implementing the results of object-oriented analysis. Transition from analysis to design is discussed along specific design techniques. Prototypical object-oriented programming languages are introduced and compared.

Upon completion of this course the student should be:

- describe a phased approach to information systems development and describe cross-life cycle activities that overlap the entire cycle.
- to present a technical material to a group.
- to create precise and informative documents for each stage of software development.
- to apply several approaches to software design such as data flow oriented, data structure oriented, and object oriented techniques.
- understand the importance of quality assurance and reliability of software systems.
- understand the process of graphic user interface(GUI) and dialogues design using Visual C#
- access relational data using ADO.NET
- apply object-oriented techniques to Visual C# or VB.NET programs

Educational Objectives

- Broaden existing software design and programming skills using current methodologies.
- Enhance existing software document skills
- Incorporate design-for-maintainability considerations through appropriate project work

Professional Objectives

- Exposure to design and programming technology that is widely-used in industrial and research environments.
- Enhance planning and time management skills by making students work through a significant project with multiple, staged deliverables.
- To provide a practical software development techniques that will prepare graduates to lead large software projects in industry.
- Define the systems analyst role and responsibilities in a typical organization.
- Prepare and use various information gathering techniques for eliciting user information requirements and system expectations.
- Develop a better plan for becoming a better systems professional or user/manager of systems, by understanding personal strengths and weaknesses and matching those with the ethical success factors of a modern business manager.
- Write a formal technical or professional report that organizes a systems documentation

COURSE OVERVIEW

How to specify requirements using Use Cases

How systems are specified using Object, Interaction and Dynamic Models

How to build an object oriented solution based on a suitable target system architecture

How to specify, design, build and deliver a two-tier client/server and a three-tier application systems.

Lectures & Labs : Mondays, Wednesdays, Fridays; 10.00 a.m. – 10.50 a.m.

Instructor : Martin Dwomoh-Tweneboah

Office : Renshaw 209

Office Phone : 2426

E-mail : mdwomoh@linfield.edu

Office Hours and Lab Consultation: Daily 3.00 – 5.00 PM or by appointment.

Text(Required) : Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition

ISBN : 9781118804674

Alan Dennis, Barbara Haley Wixom, David Tegarden

Software: Visio, Microsoft Project, Visual Studio.NET, Microsoft SQL, Rational Rose Enterprise or Visible Analyst.

Copies of all the software are provided free of charge. Contact the System Administrator for copies of the software for your personal machines.

Note: The textbook serves as reference material. You should read the assigned chapters. We will not go through the text chapter by chapter as done in some disciplines. You are strongly advised to read the assigned chapter as the quizzes will be drawn from the textbook.

GRADING

Grading for this course will be based on quizzes, assignments, exams and the deliverables for a course project. Because this course is an upper-level course, the standards by which you are assessed and the grading scale used are more rigorous than those in lower division courses.

Grading is based on the following:

Assignments	25%
Quizzes.	20%
Exams(2 Midterms)	30%
Final Project (Semester-long project)	25%

Grading Scale:

95 – 100	A
90 – 94	A-
85 – 89	B+
80 – 84	B
75 – 79	B-
70 – 74	C+
65 – 69	C
60 – 64	C-
50 – 59	D
Below 50	F

ASSIGNMENTS

General Information

Several short homework assignments will be given throughout the semester.

- Homework assignments will be posted on the course website.
- Except for occasional supplementary materials, hard copies of the assignments will not be provided.
- It is your responsibility to check the course web site for any materials relating to the course and to keep track of upcoming assignments, quizzes and midterms.
- Homework is due at the **beginning** of class on the due date.
 - You are expected to have completed the assignment before you arrive in class.
 - You will have a 30-minute window after the class during which you may turn in your assignment with no penalty.

Assignment Management

The Bachelor of Science Computer Science major focuses on decision-making skills, oral and written communications, the values and uses of information systems, project development and completion via teams, competency in a programming language and application development, familiarity with systems analysis and design, the completion of a systems development project and the use of the computer as a tool, hence a thorough understanding of the various platforms and operating systems.

The curriculum is designed to produce graduates ready to function in the computer industry with the competencies, skills, and attitudes necessary for success in the workplace or graduate school. It forms the basis for continued career growth, life-long learning as a computer professional or a future graduate program. Among other important skills for administrators and managers are time management and resource allocation. Specifically, appropriate attention to time management and resource allocation will aid you in meeting task deadlines with available resources. These skills will be important to your success in administration or management and to your success in this degree program. This syllabus describes course assignments and defines assignment due dates. Your effective use of time management and resource allocation will be key in meeting the assignment deadlines for both individual and study group assignments. From my own experience, I offer the suggestion that your time management plan include time for yourself and your academic work.

Late Assignments

In the general case, late assignments will not be accepted for grading. ALL ASSIGNMENTS MUST BE SUBMITTED IN CLASS ON THE DUE DATE. If you know you must be absent from a class session, you should take appropriate steps to ensure that your assignments are delivered on or before the scheduled due date and time. I will not accept any excuse for late delivery. In addition to submitting a hard copy, you must keep electronic copies of all assignments in a folder in your home directory on *nova*.

Assignment Format

Written assignments should be submitted, typed on 8-1/2 x 11 paper, one sided only, stapled in the upper left corner, and should be done neatly so that it is: (a) easy to grade, and (b) useful as a study aid. Assignments submissions that do not meet minimal standards for acceptability in the workplace (completeness, neatness, readability, etc.) will be returned ungraded. .

QUIZZES

Some class meetings will include a short quiz. These quizzes are conducted during the class meeting and at times as take-home. Quiz questions will often be selected from or derived from the questions found at the end of the assigned chapters and readings. I will not give make-up exams and quizzes.

COUSER PROJECT

The course is accompanied by a semester-long course project. It is recommended that you start working on the project as soon as the relevant materials are covered in class.

ATTENDANCE

Attendance is mandatory.

COURSE MATERIALS

All course materials, announcements, assignments, etc., can be found on Blackboard at <http://bblearn.linfield.edu>. It is your responsibility to check the web page on regular basis for course materials and due dates.

EXTRA CREDIT POLICY

There will not be any extra credit in this course. There are a lot of assessment exercises in this course for you to catch up with areas you don't perform well. Therefore, don't bank on extra credit to improve your grades.

MISSED CLASSES

It is your responsibility to make arrangements to obtain materials distributed on class days when you miss a class. This can be done through contacting a classmate who was present or by contacting me during my office hours or other times. Missed or late quizzes cannot be made up under any circumstances but with good cause and adequate notice, an early quiz may be given.

ACADEMIC HONESTY

Cheating and plagiarism will not be tolerated. Any student found to be engaging in either of these activities at any point in the course will receive a failing grade for the entire course and may be subject to further college sanctions.

CLASSROOM AND GROUP DISCUSSIONS

Study groups are highly recommended for a course of this kind. However, copying someone's work for presentation will be treated as academic dishonesty. Active involvement of each student in class discussions and exercises are essential. Class attendance and active participation are expected and required in this course. Absences or lack of participation generally reduces a learner's aggregate point score and thus may affect a learner's final grade.

We must assume that we are all persons of intelligence and good will who are here to learn from each other in a team environment. Group discussions should not be a forum to impose our ideas on others. For the academic endeavor to succeed, we must treat each other with civility, courtesy and respect. Software development involves team work and all contributions by group members should be discussed and analyzed thoroughly.

PREPARATION FOR EACH CLASS

- Read the text chapter assigned per schedule.
- Visit the companion Web site on Blackboard and review the topic objectives and other materials, complete online study guide quizzes, and review other materials.
- Make note of any questions you may have to pose during class or via e-mail.
- Browse the PowerPoint presentation for the topic.
- Take notes in class.

DIFFICULTIES

If you find you are having problems with the class – the use of the software package, case tool, attendance, keeping up with the reading, fitting into a group, please let me know. I am always available to help you, but I have to know about the problem while it's going on and before the end of the course. The last few weeks to the end of the course is not the best time to ask for help.

STUDENTS WITH DISABILITIES

Students with disabilities are protected by the Americans with Disabilities Act and Section 504 of the Rehabilitation Act. If you are a student with a disability and feel you may require academic accommodations please contact Learning Support Services (LSS), as early as possible to request accommodation for your disability. The timeliness of your request will allow LSS to promptly arrange the details of your support. LSS is located in Melrose Hall 020 (503-883-2562).

Students with documented disabilities who may need accommodations for taking quizzes and tests, who have any emergency medical information I should know of, or who need special arrangements in the event of an evacuation, should make an appointment with me as early as possible, no later than the second week of the semester.

CELL PHONE USAGE, EMAIL AND WEB BROWSING

All cell phones should be turned off during lectures. Unless otherwise instructed, all applications, including browsers and emails must be closed during lectures.

COURSE OUTLINE

Course Expectations

Outcome : After this session, student should understand the scope and requirements of this course as well as the knowledge and skills need to be mastered to succeed in this course
Introduction to the software tools to be used.

Software Development Life Cycle and Software Management

Outcome After this session the student should be able to:

- identify the various phases of a development life cycle – requirements analysis, analysis phase, design phase, coding phase, testing phase.
- understand the techniques and concepts of project management.

Requirements Gathering

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

- Understand the techniques used in gathering information about the systems functionalities.

System Behavior – Use Cases

Analysis of the needed system behavior from a use case approach

Development of scenarios for use cases

Outcome : At the end of the session, students should be able to

- capture system requirements using Use Cases.

Finding Classes

Application of use case analysis to discover classes in the system

Definition of relationships needed for object interaction

Operations and Attributes

Definition of class structure and behavior

Outcome: At the end of the session, student should be able to :

- capture system requirements using object-oriented concepts, including classes, objects, attributes, operations, relationships and multiplicity

Sequence and collaboration diagrams

Development of state transition diagrams to graphically show the behavior of an object

Outcome : At the end of the session, students should be able to:

- create sequence and Collaboration diagrams with a Visual Modeling tool
- create state diagrams with a Visual Modeling tool

User interface design and Accessing the Database Using Visual C# or Visual Basic.NET

Outcome: At the end of the session, students should be able to:

- Apply appropriate user interface strategies to a system.
- Use the design models to plan and coordinate a user interface.

Select proper screen-based controls for input attributes that are to appear on a GUI input screen

Programming with Microsoft ADO.NET

Outcome: At the end of the session, students should be able to:

- Build data-centric applications with Microsoft® ADO.NET, Microsoft® SQL Server™ 2005, and the Microsoft® .NET Framework.

Testing and Deployment

Outcome: At the end of the session, students should be able to :

- understand the purpose and effectiveness of various testing strategies.