Computer Science CSC 311
Software Engineering II

Credits and Contact Hours: 3 Credits, Contact Hours: 42 (1 contact hour = 50 minutes)

Instructor: Dr. Vinitha Hannah Subburaj

Textbook:

Course description:
a. Catalog description: This course is offered in the second semester of a two semester software engineering course sequence. This course introduces students to advanced topics in software requirements specification, design, testing, and maintenance of a software system. The courses will emphasis on writing skills and oral presentation skills which forms the basic platform for students entering into software industries. Software engineering II is a course that involves practical implications along with team work. The major part of the course will involve students to work in teams to design and develop software systems. Students will be supervised, but are expected to be organized well while working with team members and developing their presentation and management skills.
b. Prerequisites: CSC-310
c. Required/Elective:
   a. Computer Science – Not Required
   b. Software Engineering – Required

Specific Goals of Course:
a. Specific outcomes of instruction

After completion of the course, students will be able to
i. Appreciate software engineering techniques and understand software development is just not “coding” and far beyond it.
ii. Understand the concepts of software life cycle models and the process
iii. Develop thorough understanding in the requirements gathering, design, implementation, and testing phases of the software life cycle.
iv. Develop reliable software systems for different platform such as PC, web, and mobile following the software life-cycle process that meets customers’ requirements
v. Translate software requirements to design, design to code, and then test the software system based on appropriate software engineering methodologies
vi. Choose appropriate models, design patterns, architecture, and programming language for real-world software projects based on the particular domain
vii. Employ team work – that includes project management skills, interpersonal and communication skills.
b. CAC Criterion 3 outcomes addressed by the course:

c. An ability to design, implements, and evaluate a computer-based system, process, component, or program to meet desired needs

d. An ability to function effectively on teams to accomplish a common goal

e. An understanding of professional, ethical, legal, security and social issues and responsibilities

f. An ability to communicate effectively with a range of audiences

g. An ability to analyze the local and global impact of computing on individuals, organizations, and society

i. An ability to use current techniques, skills, and tools necessary for computing practice.

k. An ability to apply design and development principles in the construction of software systems of varying complexity.

c. EAC Criterion 3 outcomes addressed by the course:

a. An ability to apply knowledge of mathematics, science, and engineering

b. An ability to design and conduct experiments, as well as to analyze and interpret data

c. An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability

d. An ability to function on multidisciplinary teams

e. An ability to identify, formulate, and solve engineering problems

f. An understanding of professional and ethical responsibility

g. An ability to communicate effectively

h. The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context

i. A recognition of the need for, and an ability to engage in life-long learning

j. A knowledge of contemporary issues

k. An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice

Brief list of topics to be covered:

a. Managing software projects – project management, project scheduling

b. Requirements modeling - Understanding requirements, Scenario based modeling, Flow, Behavior, Patterns and Webapps

c. High level Design concepts – Design concepts, Architecture design, Component level design, User Interface design, Web app design, and Pattern based design

d. Testing strategies – Verification and validation, Testing techniques, Testing web applications

e. Software Configuration management

f. Software maintenance and reengineering