CSC 450: Introduction to Software Engineering

Catalog Description:
Prerequisite: CSC 325 and CSC 335 and CSC 365. Principles, techniques and tools used to effect the orderly production of medium and large scale computer programs will be studied. These techniques will be applied to programming projects with students working in teams and managing all phases of a programming project. 3(3-0) F

This is a terminal course. It is not a prerequisite for anything.

Required Texts:
Essential Scrum, Kenneth S. Rubin, Addison-Wesley, 2012
Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin, Prentice Hall, 2009

Major Topics (including information for course sequence or transition)
1. Software life cycle and the waterfall model
2. Agile methods
3. Scrum
   a. Roles
   b. Activities and artifacts
   c. Sprint planning and execution
   d. Daily scrum
   e. Sprint review
   f. Sprint retrospective
4. Envisioning (product planning)
5. Requirements and user stories
6. Estimation and velocity
7. Technical debt
8. Tools
   a. Version control and workflow
   b. Issue tracking
   c. Task management
   d. Team communication
9. Testing
10. Code quality and coding standards
11. Design patterns and anti-patterns

Student Outcomes Assessed in CSC 450
B. Students will attain an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
C. Students will attain an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
D. Students will attain an ability to function effectively on teams to accomplish a common goal
E. Students will attain an understanding of professional, ethical, legal, security and social issues and responsibilities
F. Students will attain an ability to communicate effectively with a range of audiences
I. Students will attain an ability to use current techniques, skills, and tools necessary for computing practice
K. Students will attain an ability to apply design and development principles in the construction of software systems of varying complexity

CAC Characteristics Enabled But Not Assessed in CSC 450

A. Students will attain an ability to apply knowledge of computing and mathematics appropriate to the discipline

Table 1. Student Outcomes assessed by CSC 450

CSC 450 Student Outcomes	CSC 450 Performance Indicators	CSC 450 Assessment Goals
CSC 450 contributes to SO B: Students will attain an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution	**PI 450-1:** Write an envisioning report for a product that includes an initial product backlog (feature list) and architectural overview	**PI 450-1:** all reports score at least 70%
CSC 450 contributes to SO C: Students will attain an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs	**PI 450-2a:** Design and implement a product (a software system) to meet client expectations that may change over time **PI 450-2b:** Write a final report that includes a product completion percentage and a summary of missing features	**PI 450-2a:** all projects score at least 70% **PI 450-2b:** all reports meet the requirement and are consistent with the product backlog
CSC 450 contributes to SO D: Students will attain an ability to function effectively on teams to accomplish a common goal	**PI 450-3a:** Participate in daily scrum meetings throughout six two-week sprints [teamwork] **PI 450-3b:** Peer evaluation of team contribution based on metrics and observation	**PI 450-3a:** integrated into course; see below **PI 450-3b:** standard deviation of individual percentages is <= 5 for all teams
CSC 450 contributes to SO E: Students will attain an understanding of professional, ethical, legal, security and social issues and responsibilities	**PI 450-4:** Select a license for your product acceptable to both you and your client and consistent with the licenses of your development tools and third-party components	**PI 450-4:** 100% completion
CSC 450 contributes to SO F: Students will attain an ability to communicate effectively with a range of audiences	**PI 450-5a:** Participate in daily scrum meetings throughout six two-week sprints [communication] **PI 450-5b:** Present a sprint review: explain features completed during the sprint and get feedback from stakeholders	**PI 450-5a:** integrated into course; see below **PI 450-5b:** 95% participation
CSC 450 contributes to SO I: Students will attain an ability to use current techniques, skills, and tools necessary for computing practice	**PI 450-6a:** Use Scrum to develop a software product with a team [use of current technique] **PI 450-6b:** Use a distributed version control system **PI 450-6c:** Use an online issue tracker **PI 450-6d:** Use an online system for managing the product and sprint backlogs	**PI 450-6a,b,c,d:** integrated into course; see below
CSC 450 contributes to SO K:	**PI 450-7a:** Design and	**PI 450-7a,b:** integrated

Students will attain an ability to apply design and development principles in the construction of software systems of varying complexity	implement a product (a software system) to meet client expectations that may change over time [system of high complexity] **PI 450-7b:** Use Scrum to develop a software product with a team [apply design and development principles]	into course; see below

Table 2. CAC Characteristics enabled by CSC 450

CSC 450 CAC Characteristics	CSC 450 Characteristic Enablers
CSC 450 enables Characteristic A: Students will attain an ability to apply knowledge of computing and mathematics appropriate to the discipline	**CE 450-1:** Design and implement a product (a software system) to meet client expectations that may change over time