CS 3540: INTRODUCTION TO SOFTWARE ENGINEERING

Semester Hours: 3.0  Contact Hours: 3
Coordinator: Yan Wu
Text: Head First Software Development
Author(s): DAN PILONE AND RUSS MILES
Year: 2007

SPECIFIC COURSE INFORMATION

Catalog Description:
Overview of software engineering as a discipline. Software life-cycle models and phases of the software development process. Introduction to Human Computer Interaction (HCI), user-centered development, teams and project management. Prerequisite: Grade of C or better in CS 2020.

Course type: REQUIRED

SPECIFIC COURSE GOALS

- I can understand key terms used when analyzing human interaction with software.
- I can analyze, specify and document software requirements for a software system.
- I can understand user interface design standards.
- I can express and understand the importance of professional ethics, etiquette, and communication in a software development environment.
- I can develop alternative design solutions to a given problem and recommend the best one within limitations of cost, time, knowledge, existing systems, and organizations.
- I can apply the process of project initiation, planning, execution, and management.
- I can analyze and compare various software development lifecycle methods that include requirements analysis, design, implementation, testing and maintenance.
STUDENT OUTCOMES ADDRESSED BY THIS COURSE

- B.1 Analyze a given problem, and identify and define the computing requirements appropriate to its solution
- B.2 Use current techniques, skills, and tools in computing practice
- B.4 Understand the impact of professional, ethical, and social issues in computing
- B.5 Apply design and development principles in the construction of software systems of varying complexity
- B.7 Contribute effectively to professional teams in order to accomplish a common goal

LIST OF TOPICS COVERED

- Software Processes and Models
  - Software engineering concepts
  - SDLC, Process model
  - Agile Software Development
- Planning and Requirements Analysis
  - User and/or system requirements
  - Effort estimation
- Design and Development Methodologies
  - Human and Computer Interaction and Design
  - Team design
  - Architecture and design patterns
  - Internal and external design factors
  - Coding methods and guidelines
- Documentation, Testing and Evaluation
  - Standards and best practices
  - Testing methods
  - Assurance and acceptance criteria
  - Reliability and performance
- Project Management
  - Resources and configuration control
- Risk analysis
- Product integration
- Best practices
- Release management and source control