Course Syllabus

CS 325 - Software Engineering Concepts

Credits: 3 Contact hours: 3

Instructor’s or course coordinator’s name: Dr. Bahman Khosravi-Sichani

Required Textbook and Other Materials:

Stephens, Rod. *Beginning Software Engineering*, John Wiley & Sons, 2015

Course Description:

Overview of software engineering concepts, analysis/design techniques, software documentation, and group development of software. Requires collaboration and teamwork on a significant software project moving through its entire development lifecycle. Some portions of the work will be done within the groups and other portions will be done individually. This is a writing intensive course.

Prerequisite(s): Computer Science 205 passed with a grade of "C" or higher and EN 101 and 102

Required or selected elective: Required

Specific outcomes of instruction:

When students complete this course they should be able to:

* Demonstrate that they have a thorough understanding of the various phases of the Software Development Life Cycle.
* Move their own projects from phase to phase.
* Produce deliverable documentation to be reviewed and baselined by the project’s stakeholders.
* Assume the role of the stakeholders to critically review documentation and presentations delivered by other project teams.

Relationship of course to student outcomes listed in criterion 3:

In this course students are given an opportunity to:

* Apply knowledge of computing and mathematics appropriate to the discipline.
* Analyze a problem, identify and define its computing requirements.
* Design, implement and evaluate a system, process, component or program to meet desired needs.
* Function effectively on teams to accomplish a goal.
* Communicate effectively with a range of audiences.
* Use current techniques, skills and tools necessary for practice.
* Apply design and development principles in the construction of software systems of varying complexity.

Topics Covered:

Business Needs Analysis

Feasibility Studies

Market Analysis

Business Requirements Development

Project Development Approaches

Project Architecture and Scope Analysis

Identifying Stakeholders

Building Development Teams – Size, Skills, etc.

Software Architecture – UML Modeling

Design Methods - Object Design, Class and Object Diagrams

Software Requirements – Use Cases, Time Lines, User Interface

System Analysis – Database Design, Modules, Languages, Tools, Libraries, Environment

System Design – Development, Code Review, Unit Testing

Testing – Test Plans, Integration Testing, System Testing User Acceptance Testing

Documentation – Operations Guides, User Manuals, Release Features