**TEXAS TECH UNIVERSITY**

**DEPARTMENT OF COMPUTER SCIENCE**

CS 4365 - SOFTWARE ENGINEERING II

Spring 2020

**Course Description**

This course introduces some advanced topics in software engineering for senior undergraduate students. The course covers the mainstream engineering activities often involved in software development process. Advanced topics such as fundamentals of software project management, basics of software quality assurance and software testing, processes involved in configuration management, component-based and reusable software engineering, and the practice of security engineering are the essential concepts that any software engineer needs to know in order to accomplish software projects and deliver high quality products. The materials offered in this course play critical roles for many stakeholders. A project manager would need to know the basics of project management. A system analyst would need to know dependencies in the system. A security expert would need to know whether the software under development complies with security standards. And finally a system designer would need to know whether the system under development is engineered properly. The topics delivered in this course not only will be needed for the capstone and senior project course but also for those who would choose software engineering as their career.

**Prerequisites**

* Software engineering I (CS 3365)

**Instructor**

* Name: Akbar S. Namin
* Office: 306F
* Office Hours: Mondays 2 - 4
* Email: [akbar.namin@ttu.edu](mailto:akbar.namin@ttu.edu)
* TAs: TBA

**Textbook, Lecture Notes, and Course Website**

The following book is the primary textbook of this course (all exam questions will be based on the selected chapters of the following two books).

* Instructors’ notes on software testing and programming analysis [**INST**]
* Software Engineering, Ian Sommerville, 10th Edition (main reference) [**SE-IS**] (Chapters 13, 18, 25, 22, 23)
  + <https://iansommerville.com/software-engineering-book/>

**Student's Evaluation**

The student's evaluation will be based on the following categories and their respective weights:

* Individual Assignments (5%each) (3 individual assignments = 15%)
* Team-based project (a mini project) (1 project = 20%)
* Quizzes (5% each) (3 quizzes = 15%)
* Mid-term exams (20%) (1 mid terms = 20%)
* Final exam (30%)

**Ethical Conduct**

Although discussion about ideas and problems is one of the major learning methods, students must write their own assignments and essays in their own words. When taking ideas or sentences from another author, students must acknowledge their debt by citation. Plagiarism is the most serious academic offence and there will be zero-tolerance for academic dishonesty. Scholastic dishonesty includes, but is not limited to, cheating, plagiarism, collusion, and falsifying academic record. For a detailed list of offenses, refer to Texas

Tech University on-line resource for students, found in Part IX, pp. 21-30, available on line at:

*http://www.depts.ttu.edu/studenta\_airs/publications/2008 2009 Handbook and Code.pdf*

All assignments are individual works. You may discuss approaches to problems among yourselves; however, the actual details of the work (assignment coding, answers to concept questions, etc.) must be an individual effort. The standard departmental penalty for assignments that are judged to be the result of academic dishonesty is, for the student's first offence, a mark of zero for the assignment, and possible consequences, possibly suspension from the university.

***Additional Ethical Conduct:***

*Students are required to present proficiency and maturity. In particular, the following cases are considered as unprofessional behavior and the students demonstrating similar behavior are asked to leave the classroom:*

* *Sleeping in the class*
* *Talking to the classmates during the lectures continuously.*
* *Using electronic devices of any kinds, including: laptops, cell phones, etc. (you* ***MUST*** *turn them off before entering the classroom)*
* *Screaming, whistling, and any action that is considered to be as rude behavior.*

**Cheating:**

Cheating will not be tolerated and the students who have conducted cheating will receive zero in their submissions, or the exam.

**Students with Special Needs**

Any student who, because of a disability, may require special arrangements in order to meet the course requirements should contact the instructor as soon as possible to make any necessary arrangements. Students should present appropriate verification from Student Disability Services during the instructor's office hours. Please note instructors are not allowed to provide classroom accommodations to a student until appropriate verification from Student Disability Services has been provided. For additional information, you may contact the Student Disability Services office in 335 West Hall or 806-742-2405.

**Learning Outcomes:** Students who have completed this course should have

1. Understanding of project management concepts and teamwork. (Learning Outcomes d, e, i)
2. Ability to develop secure software. (Learning Outcomes e)
3. Ability to manage the quality of software systems. (Learning Outcome c)
4. Understanding of high productivity techniques, such as software reuse and process improvement. (Learning Outcomes c, i)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Topics** | **Text** | **Source** | **Homework due** |
| Jan 15 | Risk Management | Chapter 22 | SE-IS |  |
| Jan 17 | Risk Management | Chapter 22 | SE-IS |  |
| **Jan 20** | **No Classes (MLK Day)** |  |  |  |
| Jan 22 | Plan-driven Development | Chapter 23 | SE-IS |  |
| Jan 24 | Project Scheduling | Chapter 23 | SE-IS |  |
| Jan 27 | Estimation Techniques | Chapter 23 | SE-IS |  |
| Jan 29 | Functional Testing |  | INST |  |
| Jan 31 | Functional Testing |  | INST | Assignment 1 |
| Feb 3 | Functional Testing |  | INST |  |
| **Feb 5** | **Quiz #1** |  |  |  |
| Feb 7 | Structural Testing - CFG |  | INST |  |
| Feb 10 | Structural Testing - CFG |  | INST |  |
| Feb 12 | Structural Testing - CFG |  | INST |  |
| Feb 14 | Structural Testing - DFG |  | INST |  |
| Feb 17 | Structural Testing - DFG |  | INST |  |
| Feb 19 | Structural Testing - DFG |  | INST |  |
| Feb 21 | Version Management | Chapter 25 | SE-IS | Assignment 2 |
| Feb 24 | System Building | Chapter 25 | SE-IS |  |
| Feb 26 | Change Management | Chapter 25 | SE-IS |  |
| **Feb 28** | **Quiz #2** |  |  |  |
| Mar 2 | Release Management | Chapter 25 | SE-IS |  |
| Mar 4 | Security and Dependability | Chapter 13 | SE-IS |  |
| Mar 6 | Security and organizations | Chapter 13 | SE-IS |  |
| Mar 9 | Security requirements | Chapter 13 | SE-IS |  |
| **Mar 11** | **Hands-on Experiences (Q/A)** |  |  |  |
| **Mar 13** | **Mid Term** |  |  |  |
| **Mar 16** | **No classes (March Break)** |  |  |  |
| **Mar 18** | **No classes (March Break)** |  |  |  |
| **Mar 20** | **No classes (March Break)** |  |  |  |
| Mar 23 | Secure Systems Design | Chapter 13 | SE-IS |  |
| Mar 25 | Secure Systems Design | Chapter 13 | SE-IS |  |
| Mar 27 | Secure Systems Design | Chapter 13 | SE-IS |  |
| Mar 30 | Secure Systems Design | Chapter 13 | SE-IS |  |
| Apr 1 | Security Testing and Assurance | Chapter 13 | SE-IS |  |
| Apr 3 | Security Testing and Assurance | Chapter 13 | SE-IS | Assignment 3 |
| Apr 6 | Security Testing and Assurance | Chapter 13 | SE-IS |  |
| Apr 8 | Security Testing and Penetration Testing | Chapter 13 | SE-IS |  |
| **Apr 10** | **Quiz #3** |  |  |  |
| **Apr 13** | **No Classes** |  |  |  |
| Apr 15 | Service-oriented Architecture | Chapter 18 | SE-IS |  |
| Apr 17 | Service-oriented Architecture | Chapter 18 | SE-IS |  |
| Apr 20 | Service-oriented Architecture | Chapter 18 | SE-IS |  |
| Apr 22 | RESTful Services | Chapter 18 | SE-IS |  |
| Apr 24 | RESTful Services | Chapter 18 | SE-IS |  |
| Apr 27 | RESTful Services | Chapter 18 | SE-IS |  |
| Apr 29 | Service Engineering | Chapter 18 | SE-IS |  |
| May 1 | Service Engineering | Chapter 18 | SE-IS |  |
| **May 4** | **Hands-on Experiences (Q/A)** |  |  |  |
|  |  |  |  |  |

\* Tentative Schedule