

**ITT Technical Institute**  
**CS340**  
**Software Engineering**  
**Onsite Course**

**SYLLABUS**

---

**Credit hours:** 4

**Contact/Instructional hours:** 50 (30 Theory Hours, 20 Lab Hours)

**Prerequisite(s) and/or Corequisite(s):**

Prerequisite: CS300 Application Design or equivalent

**Course Description:**

This course discusses the requirements for designing and managing the software development process covering design, implementation and support. Various commonly accepted methodologies are used throughout the course to provide students a broad background in the required activities.

# Syllabus: Software Engineering

---

Instructor: \_\_\_\_\_

Office hours: \_\_\_\_\_

Class hours: \_\_\_\_\_

## Major Instructional Areas

1. Software engineering approaches
2. Software process models
3. Requirements analysis
4. Object-oriented design
5. Testing
6. Configuration management
7. Documentation
8. Cost estimation and schedules
9. Project metrics
10. Work breakdown structures
11. Risk management
12. Quality management
13. Deployment planning

## Course Objectives

1. Describe various approaches to software engineering.
2. Evaluate the benefits and drawbacks of the various software process models.
3. Perform project management tasks, including estimation, project scheduling, and risk management.
4. Apply requirement analysis methods to create a requirements document.
5. Describe various application architectures and select the best architecture to meet requirements.
6. Identify the primary concerns when designing and developing a critical system.
7. Prepare a test plan that includes all the phases of the testing process: planning, design, building tests, executing tests, and reporting results.
8. Describe the importance of and create a configuration management plan.
9. Create a quality plan.
10. Evaluate project performance and plan for improvement.

## SCANS Objectives

SCANS is an acronym for Secretary's Commission on Achieving Necessary Skills. The committee, created by the National Secretary of Labor in the early 1990s, created a list of skills and competencies that the committee feels are necessary for employees to function in a high-tech job market.

1. Explain how technological systems work and operate effectively.
2. Identify the need for data and evaluate its relevance and accuracy.
3. Explain trends in technological change in systems and deduce how the change has impacted the system operations.
4. Identify the related problems in computers and other technologies.
5. Create written or computerized records and other forms of information in a systematic fashion.
6. Analyze information using oral, written, graphic, pictorial, or multimedia methods.
7. Process information using computers.

## Course Outline

Note: All graded activities, except the Project and Final Exam, are listed below in the pattern of <Unit Number>.<Assignment Number>. For example, Lab 1.1 refers to the 1<sup>st</sup> lab activity in Unit 1.

Unit	Activities
1— Introduction to Software Engineering	<ul style="list-style-type: none"> <li>• Content Covered:               <ul style="list-style-type: none"> <li><i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 1, "Introduction"</li> <li>○ Chapter 2, "Socio-technical Systems"</li> </ul> </li> </ul> </li> <li>• Labs: 1.1</li> <li>• Assignments: 1.1</li> <li>• Project: Part 1</li> </ul>
2— Software Process Models	<ul style="list-style-type: none"> <li>• Read from <i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 4, "Software Processes"</li> </ul> </li> <li>• Labs: 2.1</li> <li>• Assignments: 2.1</li> <li>• Project Part 2</li> </ul>
3— Project Management	<ul style="list-style-type: none"> <li>• Read from <i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 5, "Project Management"</li> <li>○ Chapter 25, "Managing People"</li> </ul> </li> <li>• Labs: 3.1</li> <li>• Assignments: 3.1</li> <li>• Project Part 3</li> </ul>
4— Requirements Gathering	<ul style="list-style-type: none"> <li>• Read from <i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 6, "Software Requirements"</li> <li>○ Chapter 7, "Requirements Engineering Processes"</li> <li>○ Chapter 8, "System Models"</li> </ul> </li> <li>• Labs: 4.1</li> <li>• Assignments: 4.1</li> <li>• Project Part 4</li> <li>• Quizzes: 4.1</li> </ul>
5— System Design	<ul style="list-style-type: none"> <li>• Read from <i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 11, "Architectural Design"</li> <li>○ Chapter 12, "Distributed Systems Architectures"</li> <li>○ Chapter 13, "Application Architectures"</li> </ul> </li> <li>• Labs: 5.1</li> <li>• Assignments: 5.1</li> <li>• Project Part 5</li> </ul>
6— Critical Systems	<ul style="list-style-type: none"> <li>• Read from <i>Software Engineering:</i> <ul style="list-style-type: none"> <li>○ Chapter 3, "Critical Systems"</li> </ul> </li> </ul>

Unit	Activities
	<ul style="list-style-type: none"> <li>○ Chapter 9, "Critical Systems Specification"</li> <li>● Labs: 6.1</li> <li>● Assignments: 6.1</li> <li>● Project Part 6</li> </ul>
7— Development	<ul style="list-style-type: none"> <li>● Read from <i>Software Engineering</i>: <ul style="list-style-type: none"> <li>○ Chapter 17, "Rapid Software Development"</li> <li>○ Chapter 20, "Critical Systems Development"</li> </ul> </li> <li>● Labs: 7.1</li> <li>● Assignments: 7.1</li> <li>● Project Part 7</li> <li>● Quizzes: 7.1</li> </ul>
8— Verification and Validation	<ul style="list-style-type: none"> <li>● Read from <i>Software Engineering</i>: <ul style="list-style-type: none"> <li>○ Chapter 22, "Verification and Validation," pp. 515-532</li> <li>○ Chapter 23, "Software Testing"</li> <li>○ Chapter 24, "Critical systems Validation"</li> </ul> </li> <li>● Labs: 8.1</li> <li>● Assignments: 8.1</li> <li>● Project Part 8</li> </ul>
9— Cost and Quality	<ul style="list-style-type: none"> <li>● Read from <i>Software Engineering</i>: <ul style="list-style-type: none"> <li>○ Chapter 26, "Software Cost Estimation"</li> <li>○ Chapter 27, "Quality Management"</li> </ul> </li> <li>● Labs: 9.1, 9.2</li> <li>● Assignments: 9.1</li> <li>● Project Part 9</li> </ul>
10— Improvement and Change Management	<ul style="list-style-type: none"> <li>● Read from <i>Software Engineering</i>: <ul style="list-style-type: none"> <li>○ Chapter 28, "Process Improvement"</li> <li>○ Chapter 29, "Configuration Management"</li> </ul> </li> <li>● Labs: 10.1</li> <li>● Assignments: 10.1</li> <li>● Project Part 10</li> <li>● Quizzes: 10.1</li> </ul>
11—Course Review and Final Exam	<ul style="list-style-type: none"> <li>● Review</li> <li>● Final Exam</li> </ul>

## Instructional Methods

This course provides you with knowledge of software development processes and methodologies you will encounter in the workplace.

The key to a successful course is to encourage team dynamics. You will work with classmates on a course project designed to let you work as a team through each phase of the software development lifecycle.

Hands-on practice is an essential part of this course. Labs allow you to create various types of UML diagrams, project management documents, and other types of documents. Some labs also ask you to write code to practice good coding practices, such as declaring variables and refactoring, or to illustrate a software engineering concept.

Writing assignments give you a chance to explain concepts and work on their writing skills. When completing a writing assignment, concentrate on clearly presenting an idea. You will also have the opportunity to prepare and present PowerPoint presentations about various software engineering issues.

## Instructional Materials and References

### Student Textbook Package

Sommerville, I. (2011). *Software engineering* (Custom 8th ed.). Boston, MA: Pearson Custom.

### Other Required Resources

In addition to the student textbook package, the following is also required in this course:

- Internet access

### Equipment and Tools

- Standard lab computer with the following:
  - Internet access
  - Windows XP Professional Service Pack 2
  - Visual Studio 2005
  - Microsoft Office
  - Microsoft Visio
  - Microsoft Project 2003

### References

#### ITT Tech Virtual Library

Log on to the ITT Tech Virtual Library at <http://www.library.itt-tech.edu/> to access online books, journals, and other reference resources selected to support ITT Tech curricula.

#### Books

You may click “Books” or use the “Search” function on the home page to find the following books.

- Books 24x7
  - Andersson, Eve, Philip Greenspun, and Andrew Grumet. *Software Engineering for Internet Applications*. Cambridge, MA: The MIT Press, 2006.
  - Jones, Caspers. *Estimating Software Costs: Realism to Estimating*. 2<sup>nd</sup> ed. New York: McGraw-Hill, 2007.
  - Jalote, Pankaj. *An Integrated Approach to Software Engineering*. 3<sup>rd</sup> ed. New York: Springer, 2005.
  - Kasse, Tim. *Practical Insight into CMMI*. Boston: Artech House, 2004.
  - Mak, Ronald. *The Martian Principles for Successful Enterprise Systems: 20 Lessons Learned from NASA's Mars Exploration Rover Mission*. Indianapolis, IN: John Wiley & Sons, 2006.
  - McManus, John. *Risk Management in Software Development Projects*. Boston: Butterworth-Heinemann, 2004.
  - Munson, John C. *Software Engineering Measurement*. Boca Raton, FL: Auerbach Publications, 2003.
  - Page, Alan, Ken Johnston, and B.J. Rollison. *How We Test Software at Microsoft*. Redmond, WA: Microsoft Press, 2009.
  - Peckham, Joan, and Scott J. Lloyd. *Practicing Software Engineering in the 21st Century*. Hershey, PA: IGI Publishing, 2003.
  - Peters, Lawrence J. *Getting Results from Software Development Teams*. Redmond, WA: Microsoft Press, 2008.
  - Stover, Teresa S. *Microsoft Project 2003 Inside Out*. Redmond, WA: Microsoft Press, 2004.
  - Thayer, Richard H., and Mark J. Christensen. *Software Engineering, Volume 1: The Development Process*. 3<sup>rd</sup> ed. Hoboken, NJ: John Wiley & Sons, 2005.

- Tomayko, James E., and Orit Hazzan. *Human Aspects of Software Engineering*. Hingham, MA: Charles River Media, 2004.
- van Vliet, Hans. *Software Engineering: Principles and Practice*. 3<sup>rd</sup> ed. West Sussex, England: John Wiley & Sons, 2008.

### Periodicals

You may click “Periodicals” or use the “Search” function on the home page to find the following periodicals.

- Dr. Dobbs
- Java World
- MSDN Magazine

## Course Evaluation and Grading

### Evaluation Criteria Table

The final grades will be based on the following categories:

CATEGORY	WEIGHT
Assignments	15%
Quizzes	15%
Labs	25%
Project	25%
Final Exam	20%
<b>Total</b>	<b>100%</b>

Note: Students are responsible for abiding by the Plagiarism Policy.

### Grade Conversion Table

The final grades will be calculated from the percentages earned in the course, as follows:

A	90–100%	4.0
B+	85–89%	3.5
B	80–84%	3.0
C+	75–79%	2.5
C	70–74%	2.0
D+	65–69%	1.5
D	60–64%	1.0
F	<60%	0.0