

# Course Syllabus

## QUICK MENU

<a href="#">Instructor</a>	<a href="#">Objectives</a>	<a href="#">Prerequisites</a>
<a href="#">Materials/Texts</a>	<a href="#">Schedule</a>	<a href="#">Assignment Outline</a>
<a href="#">Grading</a>	<a href="#">Policies</a>	

---

## Summary

<b>Course Title</b>	Software Engineering I
<b>Course No.</b>	CS 07321
<b>CRN</b>	41119
<b>Start/End Dates</b>	
<b>Term/Module</b>	

<b>Delivery Method</b>	<b>Number of Face-to-Face Meetings</b>
Face-to-Face	30

## Course Description

An introduction to the discipline of Software Engineering. Students will explore the major phases of the Software Lifecycle, including analysis, specification, design, implementation, and testing. Techniques for creating documentation and using software development tools will be presented. Students will gain experience in these areas by working in teams to develop a software system. Proficiency in programming is expected of the students entering this course.

This is a junior/senior level course where you are expected to do work on your own.

<p><b>I reserve the right to amend, alter or change the information in this course guide at my discretion. All terms and interpretations will be defined by me and are final.</b></p>
---

---

## Instructor

[Top Menu](#)

<b>Name</b>	Jack F. Myers
<b>Email</b>	myersjac@rowan.edu
<b>Phone</b>	856-256-4500 x3278
<b>Office Hours</b>	Maintained in Rowan Starfish Network. Please use Starfish to book time with me.

---

## Objectives

[Top Menu](#)

The main objective is to introduce students to software engineering and prepare them for future opportunities via immersion into a project lifecycle as a team combined with a academic appreciation of software engineering theory and principles.

### **Student Learning Outcomes** *(relevant outcomes associated with the BS degree)*

- **Language Proficiency**  
Students have demonstrated proficiency in one or more programming languages by designing and developing significant software projects.
- **Design Tradeoffs**  
Students have demonstrated comprehension of tradeoffs involved in design choices.
- **Analysis and Requirements Documents**  
Students have produced one or more documents that have analyzed a problem, identified and defined the computing requirements appropriate to its solution, developed a solution, evaluated multiple alternative solutions which may have conflicting constraints.
- **Well-Engineered Solutions**  
Students have designed and implemented a well-engineered solution that meets the desired needs.
- **Design and Development Principles**  
In the implementation of solutions of varying complexity, students have applied specific design and development principles.

- **Group Work**  
Students have participated in group projects.
- **Functioning Teams**  
Students have functioned and communicated effectively with their teammates to accomplish a common goal.
- **Good Documentation/Reports**  
Students have created effective written documentation, i.e., technical reports/papers and end-user/technical documentation.
- **Good Presentations**  
Students have delivered technically sound and logically organized presentations.
- **Engaging Presentations**  
Students have delivered engaging, audience-focused presentations.

## Course Objectives

After the successful completion of this course, students should:

- Be able to apply the Agile/Scrum methodology to a development project. This includes fully adhering to Scrum values, assuming Scrum roles, participating in Scrum events and producing Scrum artifacts.
- Know how and when to adapt or replace the Scrum model with the waterfall and hybrid software development lifecycle models.
- Understand the key elements of project initiation.
- Be able to collect and analyse user requirements using formalisms such as User Stories, UML, and business process modeling.
- Be able to translate end-user requirements into system and software requirements, in a structured software Requirements Document that combines user requirements and functional requirements..
- Be able to identify and apply appropriate software architectures and patterns to carry out high level design of a system. Students should be able to critically compare alternative choices. You should be able to structure this information in an architectural Design Document.
- Be able to work collaboratively in a team to implement and fully test detailed designs and code.
- Be able to produce comprehensive test and implementation plans.
- Be able to reflect on the outcome of a project and present your reflections in a verbal report.

---

## Prerequisites

[Top Menu](#)

- **Data Structures:** CS 04222 or CS 04225 with Minimum Grades of C- or D- respectively
- **Public Speaking:** CMS 06202 or CMS 04205 or ENGR 01202 with Minimum Grade of D-
- **Writing Arts:** COMP 01112 or ENGR 01201 or ENGL 01112 or HONR 01112 Minimum Grade of D-

---

## Materials and Texts

[Top Menu](#)

- Software Engineering [10th edition] -- Sommerville. 2016.
- Other readings as assigned

---

## Schedule

[Top Menu](#)

Class schedule is maintained at <http://jackmyers.info/sweng>

---

## Assignment Outline

Assignments are maintained at <http://jackmyers.info/sweng>

[Top Menu](#)

---

## Grading

[Top Menu](#)

### Final Grade Breakdown

Grading Criteria/Assignment	Percentage
<b>Team Component</b>	
<ul style="list-style-type: none"><li>Based on quality of project and individual increments as assessed by instructor and/or project sponsor. Team grade may be modified based on peer feedback.</li></ul>	<b>50%</b>
<b>Individual Component</b>	
<ul style="list-style-type: none"><li>Test average</li></ul>	<b>20%</b>
<ul style="list-style-type: none"><li>Presentation quality (each team member must conduct a Sprint Review)</li></ul>	<b>10%</b>
<ul style="list-style-type: none"><li>Major written deliverable</li></ul>	<b>20%</b>
<b>Total</b>	<b>100%</b>

### Grading Scale

<b>A</b>	93 and Up	<b>C</b>	73 – 76
<b>A-</b>	90 – 92	<b>C-</b>	70 – 72
<b>B+</b>	87 – 89	<b>D+</b>	67 – 69
<b>B</b>	83 – 86	<b>D</b>	63 – 66
<b>B-</b>	80 – 82	<b>D-</b>	60 – 62
<b>C+</b>	77 - 79	<b>F</b>	59 and Below

Assignments and tests are meant to reinforce lecture materials and may be graded according to one of two models:

- **Traditional:** A numeric grade based on the scale from 0 to 100. Assignments not turned in on time receive a grade of zero.
- **General Suitability:** Either “Complete”, “Incomplete”, or “Not Turned In”
  - **Complete** assignments will have no mathematical effect on the Assignment grade (will be set to the average of Traditional assignments);
  - **incomplete** assignments will receive a grade of 10% lower than the average of Traditional assignments;
  - assignments **Not Turned In** will receive a zero.

---

## Rowan Policies

### 1. Attendance

Attendance is mandatory. The attendance/class participation portion of the course grade will be computed based on the number of missed classes and student's contribution to class discussion. See the official [Rowan University Attendance policy](#). Please inform the instructor in advance, preferably by email, if you will be absent from a class or lab session. **As this is a team-based course, 5 points are removed from final grade from each unexcused absence.**

### 2. Academic Integrity

Plagiarism is a form of academic dishonesty which includes submitting someone else's work as your own. It is college policy that students who commit an act of academic dishonesty may be subject to failure in the course, suspension from the College, or both. See the official [Rowan University Academic Integrity policy](#)

If you use materials that you've obtained on the Internet, from a book, etc., for example as part of a programming assignment, you must include an appropriate reference. To use such materials without proper attribution is a form of plagiarism. Students who copy homework, cheat on tests, or plagiarize material for any test or assignment in this course will receive a **failing grade for the test or assignment.**

### 3. Late Assignment Submissions

Assignments not submitted on time will receive zero as a grade.

- Team assignments (i.e., product increments) **cannot be late.**

- Individual assignments **should not be late.** However, most professors are reasonable people. If for some reason, you believe you will not be able to turn in homework on time, let me KNOW AHEAD OF TIME and I MAY give you an extension.

### 4. Classroom Decorum

When meeting in a classroom, in order to show proper respect for the instructor and for your fellow students, please observe the following:

- a. Be on time! Class will begin promptly at the scheduled time. Allow yourself enough time to park and get to class, ready to learn, before the period begins. Quizzes will be given at the start of the class so if you are late you may miss a quiz.
- b. Do not eat in class.
- c. Do your best to remain in the room during the period. Exiting and entering during the period breaks the concentration of your fellow students, and makes it hard for you to get the full value of the class.
- d. Turn off all cell phones, pagers, and anything else that would cause a distraction to yourself or others around you.
- e. Students are permitted to use computers/laptops during class for note-taking and other class-related work only. Those using computers/laptops during class for work not related to that class (like e-mailing, instant messaging, game playing or internet surfing) will be asked to leave

the classroom for the remainder of the class period.

5. **Section 504 Accommodations**

Please be aware that Rowan University is committed to providing Section 504 accommodations for all persons with disabilities. If you have specific physical, emotional, or learning disabilities and require accommodations, please contact the Office of Equity and Diversity at 856-256-4294 as soon as possible to ensure that such accommodations can be implemented in a timely fashion.

6. **Academic Issue**

If at any time a student has an academic problem in any course with a grade or any other issue, the student's first course of action to resolve the matter should be to make an appointment with the instructor to discuss the issue.

7. **Dropping or Withdrawing from this Class**

Please visit the [Registrar page](#) for a list of key dates for the semester. These dates include the full academic calendar as well as the last days to withdraw from this class with and without a refund.

8. **Illness**

It is important to get a note from student health services, or your personal doctor, or other form of documentation if you miss a class or a lab meeting. If you are not feeling well on a given day, please email or call me ahead of time. In this case, if you miss a quiz, I may let you make up that quiz.. If you have to miss an exam (and I hope you will not), re-tests will be given only in cases of extreme hardship as defined by the rules of Rowan University, and I require documentation of the reasons for your absence.

9. **Time Commitment**

This is a 3 semester-hour class, which means that for success in this class you should expect to be spending at least 9 hours a week outside of class on project activities and material review.

[Top Menu](#)