

# **CSC305: Software Engineering**

Syllabus -- Fall 2004

**Time:** TR 3:30-4:45, Location: Ranger Hall 103

**Lab:** W 12:30-1:45, Envision Lab

**Webpage:** <http://homepage.cs.uri.edu/faculty/hamel/courses/fall2004/csc305>

**Prerequisites:** CSC301

## ***Instructor***

Dr. Lutz Hamel

email: [hamel@cs.uri.edu](mailto:hamel@cs.uri.edu)

office: Tyler Hall, Rm 251

## ***Course Description***

Arguably, software systems are some of the most complex artifacts humanity has ever produced. They are typically too large for a single human being to construct let alone comprehend. In this course we will look at tools, techniques, and methodologies that allow teams of developers to deal with the complexity of modern software systems and effectively manage the software construction process from inception, design and implementation all the way to product testing and delivery.

This course offers a first hand view of a typical software lifecycle. Teams will develop a substantial software product from requirements to delivery. Development techniques and processes are unique to every situation and institution, but this course will expose you to a disciplined approach to software development and will prepare you for software development in the industrial setting.

## ***Texts***

### **Required**

**UML Distilled**, Martin Fowler, 3rd Edition, Addison-Wesley, 2004.

### **Suggested**

**Object-Oriented and Classical Software Engineering**, Stephen R. Schach, Fifth Edition, McGraw-Hill, 2002.

## ***Grading***

- Milestones – 40%
  - Specification Document
  - System Design Documents (object, scenario, class forms)

- Software integration and Testing Plan
- User interface Guide
- Software manual, other documentation, etc.
- Final project and implementation – 20%
- Midterm, Final – 20%
- Quizzes, Homework – 20%

## **Attendance**

Attendance to the weekly group meetings is required. Once you have agreed to a group project, the group must schedule one, 1 1/2 hour meeting once a week during lab time.

## **Tentative Schedule**

### **Software Engineering**

- Introduction to Software Engineering
- Teams & Tools
- *Project Milestone:* Project Proposal

### **Requirements Analysis**

- Requirements Analysis
- *Project Milestone:* Requirements Documentation

### **Object Oriented Analysis**

- Object-Oriented Analysis
- Use Cases/Use Case Diagrams (Fowler Chapter 3)
- UML (Fowler Chapter 4)
- High-level Class Diagrams (Fowler Chapter 3)
- *Project Milestone:* High-level Design Documentation

### **Object-Oriented Analysis (contd.)**

- Object Diagrams (Fowler Chapter 6)
- Modules, Objects, Reusability, Portability, and Interoperability
- *Project Milestone:* Object Design Documentation

### **Object-Oriented Analysis (contd.)**

- Sequence Diagrams (Fowler Chapter 5)
- Design Phase
- *Project Milestone:* Sequence Diagram Document

### **Testing**

- Testing
- *Project Milestone:* Test Plan Document

**Implementation**

- Implementation Phase
- Implementation and Integration Phase
- Maintenance Phase

**Formal Methods**

- OBJ3 Specification Techniques
- OBJ3 Proofs

**Final Project Milestone:** Due Last Day of Class