CSC305: Software Engineering

Syllabus – Spring 2003

January 15, 2003

**Time:** MWF 11-12, Location: CHAF 219

**Lab:** T 9-10:15, Envision Lab

**Webpage:** http://homepage.cs.uri.edu/courses/csc305

**Prerequisites:** CSC301

**Instructor**

Dr. Lutz Hamel  
email: hamel@cs.uri.edu  
office: Tyler Hall, Rm 128  
phone: 874-2701

**Texts**

**Required**


**Suggested**


**Course Goals**

This course covers programming environments and methodologies for the design, development, testing, and maintenance of large software systems. Student teams will develop a substantial software product from requirements to delivery using disciplined techniques.
Grading

40% Milestones
- Specification Document
- System Design Documents (object, scenario, class forms)
- Software integration and Testing Plan
- User interface Guide
- Software manual, other documentation, etc.

25% Final project and implementation.

10% Quizzes, Homework

25% Midterm, Final

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.

NOTES ON GRADING

Since you will be working in groups, but each of you will be getting an individual grade for the project, the following policy will be used. Each group will be given a (fictitious) dollar amount, at the end of the semester for the finished project. Each member of each group will provide the instructor with a confidential estimation of how the dollar amount should be divided among the participants and an explanation of the rationale for the monetary assignment. The instructor will use the semester project grade and the above dollar estimates to derive a suitable grade for each student on the project.

Tentative Schedule

Software Engineering –
- Introduction to Software Engineering (Chapters 1-3)
- Teams & Tools (Chapters 4-5)
- Project Milestone: Project Proposal

Requirements Analysis –
- Requirements Analysis (Chapter 10)
- Project Milestone: Requirements Documentation
Object Oriented Analysis –

- Object-Oriented Analysis (Chapter 12)
- Use Cases/Use Case Diagrams (Fowler Chapter 3)
- UML (Fowler Chapter 4)
- High-level Class Diagrams
- Project Milestone: High-level Design Documentation

Object-Oriented Analysis (contd.) –

- Object Diagrams (Fowler Chapter 6)
- Modules, Objects, Reusability, Portability, and Interoperability (Chapters 7-8)
- Project Milestone: Object Design Documentation

Object-Oriented Analysis (contd.) –

- Sequence Diagrams (Fowler Chapter 5)
- Design Phase (Chapter 13)
- Project Milestone: Sequence Diagram Document

Testing –

- Testing (Chapter 6, 14.7)
- Project Milestone: Test Plan Document

Implementation –

- Implementation Phase (Chapter 14)
- Implementation and Integration Phase (Chapter 15)
- Maintenance Phase (Chapter 16)

Formal Methods –

- OBJ3 Specification Techniques
- OBJ3 Proofs
- Project Milestone: Formal Verification Document (if time permits)

Final Project Milestone: Due Last Day of Class