

Course Number: CSC 668/868

Course Title: Advanced Object Oriented Software Design and Development

Number of Credits: 3

Schedule: Three hours of lecture/discussion per week.

Prerequisite: Senior or graduate standing, and at least a C grade in CSC 413, or consent of instructor

Catalog Description

Basic principles of object oriented analysis and design utilizing UML, advanced object oriented programming principles, design patterns, frameworks and toolkits; Agile software design processes. Development of a mid-size programming project working in teams. Paired with CSC 868. Students completing this course may not take CSC 868 later for credit. Extra fee required.

Expanded Description

1. An Introduction to Object-Oriented Analysis and Design
2. Some Notes on Java
3. Architectural Design
4. Java's Remote Method Invocation – RMI
5. Object Oriented Programming Principles
6. OOP and Software Engineering
7. Software Engineering Concepts
8. Unit Testing
9. Squeak/Smalltalk – 80
10. Reflection and Persistence
11. Programming and Design Principles
12. Event Notification
13. Design Patterns
14. Refactoring
15. Frameworks. Toolkits, and Polymorphism
16. Delegation
17. Presentation and Control
18. Active and Distributed Objects
19. *SCRUM*: An Empirically-Based Process for Software Project Management
20. Documentation and Coding Standards

Course Objectives and Role in Program

The prime objective of the course is to teach the student to analyze, design and implement object-oriented software systems by means of a mid-sized project. Students will learn the application of software architectures in various settings, including the application of design patterns, frameworks and toolkits. The ability to architect software systems is basic to all subsequent courses in which software development is an integral part. In addition, many of the concepts facing the prospective developer in modern software technology, such as Refactoring and team-oriented Agile software development processes are examined in detail and integrated into general software practice.

CSC 668 meets the undergraduate Group Project Requirement.

Learning Outcomes

At the end of the course students will be able to

- Utilize processes and artifacts to work effectively in a team-oriented development environment
- Apply various software architectures, including frameworks and design patterns, when developing software projects
- Develop Smalltalk applications
- Program distributed applications in a Java environment
- Effectively construct medium-sized object-oriented programs.

Project

Student teams will propose and develop a medium-sized project of their own.

Method of Evaluation

Student learning will be evaluated on the basis of

- Completeness and quality of mid-sized programming assignments.
- Completeness and quality of term programming project
- Class participation
- Oral presentation of research article (required only for CSC 868 students)

The weight assigned to each element of evaluation will be determined by the instructor of the course on the first day of the class.

Required Textbook

Squeak: Object-Oriented Design with Multimedia Applications, Mark Guzdial, Prentice Hall, 2000

Recommended References

- *Design Patterns, Elements of Reusable Object-Oriented Software*, Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Addison Wesley, Reading, MA. 1995.
- "Software Patterns", *Communications of the ACM*, October, 1996.
- *Refactoring to Patterns*, Joshua Kertevsky, Addison-Wesley 2005
- *Design Patterns Explained: A New Perspective on Object-Oriented Design*, A. Shalloway & J. Trott, Addison-Wesley, 2002
- *Pattern Hatching Design Patterns Applied*, J. Vlissides, Addison-Wesley, 1998
- *Agile Software Development with Scrum*, Ken Schwaber & Mike Beedle, Prentice-Hall, 2001
- *IEEE Computer, Special Issue on Agile Software Development*, June 2003
- *Refactoring: Improving the Design of Existing Code*, Martin Fowler, Addison-Wesley, 2000
- *Framework-Based Software Development in C++*, Gregory F. Rogers, Prentice Hall, 1997

Modified by: B. Levine
Last Revision Approved: March 8, 2008