Object-Oriented Design Patterns

CSCI/CSIS 603
Fall Semester 2013

Instructor

Deepti Joshi, Ph.D. Phone: 843-953-7895
Office: Thompson Hall 224 E-mail: djoshi@citadel.edu

Course Description

A course in software design using design patterns as a tool for communicating software design solutions and as an aid in software refactoring. Creational, structural and behavioral patterns are emphasized. Also covered are finding and documenting patterns, software development patterns. The Unified Modeling Language is used as the design tool for software patterns, and programming projects are in an object-oriented programming language.

Prerequisite: Experience designing and developing software using an object-oriented programming language such as Java, C++, or C#.

Learning Outcomes

Upon successful completion of this course, a student will be able to

- Explain the role of patterns in the design of object-oriented software.
- Describe numerous well-known design patterns (patterns catalog).
- Use UML to diagram the static structure and dynamic interactions of objects and classes that participate in the patterns.
- Use patterns to design object-oriented software that is more reusable and more easily modified to adapt to changing requirements.

Class Schedule

Thursday, 5:30-8:15 p.m., LCGC 140.

Textbook

Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison Wesley, 1995. ISBN: 978-0201633610.

Also: Course Notes and Handouts

Additional Course References – Books

1. [Alur 2003] Deepak Alur, John Crupi, and Dan Malks, *Core J2EE Patterns: Best Practices and Design Strategies* (Second Edition), Prentice-Hall, 2003. ISBN: 978-0131422469

2. [Blaha 2004] Michael Blaha and James Rumbaugh, *Object-Oriented Modeling and Design with UML* (Second Edition), Prentice-Hall, 2004. ISBN: 978-0130159205

3. [Bloch 2008] Joshua Bloch, *Effective Java Programming Language Guide* (Second Edition), Sun Microsystems Press/Prentice-Hall, 2008. ISBN: 978-0321356680

4. [Booch 2005] Grady Booch, Ivar Jacobson, and James Rumbaugh, *The Unified Modeling Language User Guide* (Second Edition), Addison-Wesley, 2005. ISBN: 978-0321267979

5. [Freeman 2004] Eric Freeman and Elisabeth Freeman, *Head First Design Patterns*, O'Reilly Media, 2004. ISBN: 978-0596007126

6. [Kerievsky 2004] Joshua Kerievsky, *Refactoring to Patterns*, Addison Wesley, 2004. ISBN: 978-0321213358

Additional Course References – Links

1. Hillside Group Design Patterns Library, http://hillside.net/patterns

2. Huston Design Patterns, http://www.vincehuston.org/dp/

3. J2EE Design Patterns Catalog, http://www.oracle.com/technetwork/java/catalog-137601.html

4. Net Objectives Design Patterns Repository, http://www.netobjectives.com/PatternRepository/

5. Object-Oriented Design – Design Patterns, http://www.oodesign.com/

6. Wikipedia: Design Pattern (Computer Science),http://en.wikipedia.org/wiki/Design_pattern_%28computer_science%29

Grading

The final grade for the course is based on 4 grades as follows:

- Weekly quizzes and minor assignments– collectively count as 1 grade.
- Programming/Design projects
- Mid-term and Final Exams

Office Hours

By appointment. (But please feel free to call me or to send me an email.)

Expectations

1. Do not miss an assigned test or the final exam without a valid excuse! When possible, students should notify the instructor in advance if they will be unable to take an assigned test. All make-up tests will be given outside of normal class time.

2. Show up for class on time and prepared. That means that you have read the appropriate sections from the book plus any handouts, and you have worked all assigned homework.

3. If you have missed a class, it is your responsibility to find out about what you missed in class, any assigned assignments/ homework, and other information given out in class.

4. There should be no personal conversations or moving around during class without explicit permission. Be courteous and respect the rights of others.

Daily Schedule

Dates	**Topics Covered**
Aug. 29	Overview of Object Technology, and OOAD
Sep. 5	UML, and Overview of Object-Oriented Design Patterns
Sep. 12	Overview of Creational Patterns; Quiz 1
Sep. 19	Overview of Structural Patterns; Quiz 2
Sep. 26	Overview of Behavioral Patterns; Quiz 3
Oct. 3	Creational Patterns Continued; Quiz 4
Oct. 10	Review
Oct. 17	**Mid-Term**
Oct. 24	Structural Patterns Continued
Oct. 31	Behavioral Patterns Continued; Quiz 5
Nov. 7	Behavioral Patterns Continued; Quiz 6
Nov. 14	Behavioral Patterns Continued; Quiz 7
Nov. 21	Behavioral Patterns Continued
Nov. 28	NO CLASS
Dec. 5	Review
Dec. 12	**Final Exam**