



## **Patterns in Software Engineering (40-787)**

**Dr. Raman Ramsin**

*Graduate Course, 3 Units, Elective (for MS in Systems and Software Engineering)*

*Prerequisites: None*

### **Overview**

The aim of this course is to familiarize graduate students with software patterns, their relevant concepts and principles, and their applications in software engineering. In addition to gaining knowledge and insight on common patterns of software systems analysis, design, architecture, reengineering, and development process engineering, students will also be introduced to antipatterns and refactoring techniques. Widely-used patterns will be described in detail; however, due to the large number of the patterns introduced, the main focus is on introducing the basic pattern structures and their underlying principles and philosophies. The course thus strives to provide a basis for pattern analysis and pattern-based software engineering.

### **Topics and Schedule**

- 1) Introduction to the History and Basic Concepts of Software Patterns (1 session – each session is 90 minutes in duration)
- 2) Forerunners: Coad Patterns (1 session)
- 3) GoF Design Patterns (5 sessions)
- 4) Object-Oriented Rules and Principles as Patterns – GRASP (2 sessions)
- 5) GoV Architectural Patterns (4 sessions)
- 6) GoV Design Patterns (2 sessions)
- 7) Refactoring Patterns (4 sessions)
- 8) Reengineering Patterns (4 sessions)
- 9) Software Process Patterns (1 session)
- 10) Antipatterns (3 sessions)
- 11) Analysis Patterns (2 sessions)
- 12) Methods of Pattern Classification, Complexity Management, and Analysis (1 session)

### **Exams, Assignments, and Research Project**

- Two exams (Midterm and Final) – Comprising %60 of the total grade
- Three study assignments – Comprising %25 of the total grade
- Research Project: Based on a topic of the student's choice, the research project is conducted under the guidance and supervision of the instructor – Comprising %15 of the total grade

## **Main References**

- BUSCHMANN, F., MEUNIER, R., ROHNERT, H., SOMMERLAD, P., AND STAL, M. 1996. *Pattern-Oriented Software Architecture: A System of Patterns*, Vol. 1. Wiley, New York, NY.
- BUSCHMANN, F., HENNEY, K., AND SCHMIDT, D. C. 2007. *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, Vol. 5. Wiley, New York, NY.
- FOWLER, M. 1996. *Analysis Patterns: Reusable Object Models*. Addison-Wesley, Reading, MA.
- GAMMA, E., HELM, R., JOHNSON, R., AND VLISSIDES, J. 1995. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison Wesley, Reading, MA.
- KERIEVSKY, J. 2004. *Refactoring to Patterns*. Addison-Wesley, Reading, MA.
- MANOLESCU, D., VOELTER, M., AND NOBLE, J. 2006. *Pattern Languages of Program Design*, Vol. 5. Addison-Wesley, Reading, MA.
- SHALLOWAY, A., TROTT, J. 2005. *Design Patterns Explained: A New Perspective on Object-Oriented Design*, 2<sup>nd</sup> ed. Addison-Wesley, Reading, MA.