BSE4201 Software Design Patterns (4CU)

Course Objectives; Design patterns are standard solutions to common software design problems. Instead of focusing on how individual components work, design patterns are a systematic approach that focus and describe abstract systems of interaction between classes, objects, and communication flow. This course explores advanced principles of object-oriented design by studying key software design patterns. The patterns are drawn from a variety of sources and illustrated through examples and case studies .Examples are presented in either Java, C++ or C sharp. Students will also have an opportunity to apply these patterns through a series of hands-on exercises.

CourseContent

•Basic Patterns: Structural patterns Creational patterns

•Behavioral Grasp and Cooper, V lissides Pattern Hatching

•Relationships between Patterns: Pree’s Met patterns Zimmers Relations. Tichys classification, Classification in

•Automation of Patterns: Automation of Design Patterns Together, Open Java and Design

Patterns, Compost/Recoder Refactoring Copliens Symmetries

•Historic Roots: The timeless way of building

•Advanced Patterns: Parallelism Patterns Exclusion, State Dependence, PLOP2, Coordination, Reactive patterns Analysis Patterns Reengineering Patterns Automation of Design Patterns Process Patterns Organizational Patterns

•Applications of Design Patterns: Extreme Programming Cope: Multi-Paradigm Design

References

•Erich Gammaetal. Design Patterns

•Wolfgang Pree. Design Patterns for Object-Oriented Software Development

•Frank Buschmann, Kevlin Henney, Douglas C. Schmidt” On Patterns and Pattern Languages”

.