|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Module Code** | | CS2062 | **Module Title** | Object Oriented Software Design | | | |
| **Credits** | | 3 | **Hours/ Week** | **Lectures** | 2 | **Pre/Co – requisites** | CS2012 |
| **GPA/NGPA** | | GPA | **Lab/Assignments** | 3 |
| **Learning Outcomes** | | | | | | | |
| At the end of the module the student will be able to:   * Analyze an existing system and develop clear, concise, and sufficiently formal requirements for extensions to the system based on true needs of users and other stakeholders. * Apply design principles and design tools to develop a system based on object oriented design and analysis methodologies. * Create design documents using UML to model aspects of the domain and the software architecture. * Use the design documents to implement a proof-of-concept with tests for quality assurance. | | | | | | | |
| **Topics** | | | | | | | |
| LO 1. | **Process of Object Oriented Software Design (4 hours)**: Role and skills of a system analyst. SDLC models with predictive, adaptive and unified processes. Iterative UP model and disciplines. Elements of project management and related activities. | | | | | | |
| LO 2. | **Model of Object Oriented Software Design (10 hours)**: Elicitation of functional and non-functional requirements. Techniques for validation of requirements. Transforming of system events to use cases and problem domain classes. Creating domain model class diagrams and design class diagrams using UML. Use of SSD and state charts to add details to a software design. | | | | | | |
| LO 3. | **Software Design Realization (10 hours)**: Identifying and designing structural components in a solution. Developing dynamic interactions between components. Developing design class diagrams by extending the domain model. Develop interaction diagrams by extending SSD based on the principles of object responsibility and use case controllers. Use of statecharts to model system behavior and method logic. Use of design patterns in OOD. Design of data access layer. Principles of user-centered design and the design of user interface layer. | | | | | | |
| LO 4. | **Design Implementation and Testing (4 hours)**: Creating a test plan and a test suite. Preparing a deployment strategy and validating system operation. Conduct of UAT for system handover. | | | | | | |
|  | | | | | | | |
| **Practical Work:** 15 laboratory practicals (3 hours duration)  1 Client Identification and Requirements Description  2 Inception Phase Activities for the Unified Process  3Work Breakdown Structure  4Work Flow Diagrams  5 Domain Class Modeling  6 Use Case Modeling  7-9 Use case realization  10 Semester project mid-evaluation  11-14 System implementation and testing  15 Evaluation of the semester project | | | | | | | |
| **Assessment**  a. Mid-semester evaluation of the semester project (group report and individual viva) on LO 1, LO 2 and first part of LO 3 in 10th week 10%  b. Mid-semester examination on LO 1 and LO 2 in 7th week 5%  c. Individual demonstration of semester project for LO 3and LO 4 in 15th week 20%  d. 2 in-class quizzes 5%  e. End Semester written examination: 2 hours test 60% | | | | | | | |
| **Rec. Bks.** | | **Module Textbook**:  Object-Oriented Analysis &Design - with the Unified Process, John W Satzinger, Robert B Jackson, and Stephen D. Burd, Thomson, 2005  **Recommended Textbooks:**  Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Addison-Wesley, 1994  Domain-Driven Design: Tackling Complexity in the Heart of Software, Eric Evans, Addison-Wesley, 2003  Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Craig Larman, Prentice-Hall, 2004 | | | | | |

Last edited: 11 November 2015

**Lesson Plan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Week** | **Topic** | **Chapter\*** | **Lecture Hours** | **Learning Outcome** | **Lab Hours** | **Lab Activity** |
| W01 | OOD | 1 | 2 | LO1 | 3 |  |
| W02 | Unified Process | 2 | 2 | LO1 | 3 |  |
| W03 | PM and Inception phase | 3 | 2 | LO1 | 3 |  |
| W04 | Requirements | 4 | 2 | LO2 | 3 |  |
| W05 | UCD and Domain class diagrams | 5 | 2 | LO2 | 3 |  |
| W06 | SSD | 6 | 2 | LO2 | 3 |  |
| W07 | State-chart | 6 | 2 | LO2 | 3 |  |
| W08 | design discipline with UP iterations | 8 | 2 | LO2 | 3 |  |
| W09 | design discipline with UP iterations | 8 | 2 | LO3 | 3 |  |
| W10 | System Statecharts | 9 | 2 | LO3 | 3 | Evaluations |
| W11 | Design patterns | 10 | 2 | LO4 | 3 |  |
| W12 | Design patterns | 10 | 2 | LO4 | 3 |  |
| W13 | Design patterns | 10 | 2 | LO4 | 3 |  |
| W14 | Testing and Deployment | 13 | 2 | LO4 | 3 |  |
| W15 | Revision and feedback | - | 2 | - | 3 | Evaluations |

\*Chapters are from the main module textbook

**Mapping of Learning Outcomes to Program Outcomes**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Module Learning**  **Outcomes** | **Program Outcomes based on Washington Accord** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO  10 | PO  11 | PO  12 |
| LO1 | **H** |  | **L** |  |  |  | **L** |  |  |  |  |  |
| LO2 |  |  | **L** |  | **H** |  |  |  |  |  |  |  |
| LO3 |  |  |  |  | **H** |  |  |  |  |  | **L** |  |
| LO4 |  |  |  |  | **H** |  |  |  |  |  | **L** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Overall Contribution to POs** | **L** |  | **L** |  | **H** |  | **L** |  |  |  | **L** |  |

**Program Outcomes according to Washington Accord** (with key words)

* Apply mathematics, science, engineering fundamentals and an engineering specialization to the conceptualization of engineering models
* Identify, formulate, research literature and solve complex engineering problems reaching substantiated conclusions using first principles of mathematics and engineering sciences
* Design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations
* Conduct investigations of complex problems including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions
* Create, select and apply appropriate techniques, resources, and modern engineering tools, including prediction and modeling, to complex engineering activities, with an understanding of the limitations
* Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings
* Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions
* Demonstrate understanding of the societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to engineering practice
* Understand and commit to professional ethics and responsibilities and norms of engineering practice
* Understand the impact of engineering solutions in a societal context and demonstrate knowledge of and need for sustainable development
* Demonstrate a knowledge and understanding of management and business practices, such as risk and change management, and understand their limitations
* Recognize the need for, and have the ability to engage in independent and life-long learning