|  |  |
| --- | --- |
| Logo Fast | **NATIONAL UNIVERSITY**  **of Computer & Emerging Sciences, Lahore** |

FAST School of Computing

**CS3004 – Software Design and Analysis**

**FALL 2023**

**Instructor Name:** Dr. Ali Afzal Malik **TA Name:** Hadeeqa Imran

**Email address:** ali.afzal@nu.edu.pk **Email address:** l201004@lhr.nu.edu.pk

**Office Location/Number:** C-145 (GF, CE Building) **Office Location:** GRL

**Office Hours:** Wednesdays 1130 - 1230 **Office Hours:** Tuesdays 1030 - 1130

**Course Information**

**Program:** BS (CS)

**Credit Hours:** 3

**Type:** Core

**Pre-requisite:** CS1004 Object-Oriented Programming, CL1004 Object-Oriented Programming Lab

**Course Website**: Google Classroom

**Class Meeting Days:** Wednesday and Friday

**Class Meeting Time:** 0830 – 0950 (BCS-5B); 1000 – 1120 (BCS-5A)

**Class Venue:** CS-16 (BCS-5B); E&M-3 (BCS-5A, Wednesday) and CS-15 (BCS-5A, Friday)

**Course Objectives**

The objective of this course is to provide a detailed introduction to the important upstream SDLC phases of software analysis and design. Important concepts of the Object-Oriented (OO) paradigm (e.g. abstraction, encapsulation, inheritance, polymorphism, etc.) are covered and students are given an overview of the most widely used modeling language i.e. Unified Modeling Language (UML). Students are also exposed to design principles (e.g. SOLID) and the advanced topic of design patterns (creational, structural, and behavioral).

After studying this course, the students will be able to:

* Appreciate the importance of analysis and design in SDLC.
* Understand the basic pillars of the OO paradigm.
* Construct different static and dynamic OO analysis and design models of real-world applications using UML 2.x
* Create flexible and elegant designs of software systems using proven design principles and common design patterns
* Translate OO design models to code using an OO programming language

**Course Learning Outcomes (CLOs)**

|  |  |  |  |
| --- | --- | --- | --- |
| CLO # | CLO description | BT Cognitive Level | PLO # |
| CLO 1 | **Explain** software development lifecycle | 2 (Understanding) | PLO 1 |
| CLO 2 | **Implement** object-oriented principles for software analysis and design | 3 (Applying) | PLO 3 |
| CLO 3 | **Use** different UML notations for software design | 3 (Applying) | PLO 3 |
| CLO 4 | **Develop** software design artifacts based on requirements specifications | 6 (Creating) | PLO 3 |
| CLO 5 | **Describe** software design guidelines and principles | 2 (Understanding) | PLO 1 |

**Relevant Program Learning Outcomes (PLOs)**

|  |  |  |
| --- | --- | --- |
| PLO # | PLO Name | PLO Description |
| PLO 1 | Computing Knowledge | Apply knowledge of mathematics, natural sciences, computing fundamentals, and a computing specialization to the solution of complex computing problems. |
| PLO 3 | Design/Develop Solutions | Design solutions for complex computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations. |

**Reference Material**

1. Object-Oriented Modeling and Design with UML, Michael R. Blaha and James R. Rumbaugh, 2nd Edition, Pearson, 2005.
2. Object-Oriented Analysis and Design with Applications, Grady Booch et al., 3rd Edition, Pearson, 2007.
3. Software Engineering: A Practitioner’s Approach, Roger S. Pressman, 6th Edition, McGraw-Hill, 2005.
4. Clean Architecture: A Craftsman's Guide to Software Structure and Design, 1st Edition, Robert C. Martin, Pearson, 2017.
5. Software Engineering: Theory and Practice, Shari Lawrence Pfleeger and Joanne M. Atlee, 4th Edition, Pearson, 2009.
6. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Pearson, 1995.

**Course Grading (Tentative)**

* Quizzes: 10%
* Project: 20%
* Midterm Exams: 30%
* Final Exam: 40%

Absolute grading scheme will be used for this course.

**Tentative Course Schedule**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S#** | **Topic** | **Week(s)** | **[[1]](#footnote-1)Readings** | **Project Deliverables** |
| 1 | INTRODUCTION AND OVERVIEW  SDLC Overview  Role of Analysis and Design  Development Paradigms  Basic Concepts of OO Paradigm  Benefits of OO Paradigm | 1 | [1] Chapters 1 and 2  [2] Chapters 1 and 2 |  |
| 2A | ANALYSIS MODELING USING UML  Use Case Diagrams and Descriptions  Activity Diagrams  Analysis Class Diagrams | 2 – 6 | [1] Chapters 3, 4, 7, 8  [2] Chapter 5 | Start of Week 3  Teams Formed  Start of Week 4  Proposal  End of Week 5  Phase 1: UCD, UCs, Analysis CD |
|  | MIDTERM EXAM 1 | 6 |  |  |
| 2B | ANALYSIS MODELING USING UML  System Sequence Diagrams  State Diagrams | 7 | [1] Chapters 5 – 8  [2] Chapter 5 |  |
| 3 | DESIGN MODELING USING UML  OOA to OOD Transition  Design Class Diagrams  Design Sequence Diagrams | 8, 9 | [1] Chapters 3, 4, 7, 8, 15  [2] Chapter 5 |  |
| 4 | DESIGN CONCEPTS & PRINCIPLES  Types of Cohesion & Coupling  SOLID Principles | 10 | [3] Chapter 11  [4] Chapters 7 – 11 |  |
| 5 | MEASURING OO DESIGN QUALITY  OO Metrics | 11, 12 | [5] Chapter 6 | End of Week 11  Phase 2: Design CD, Design SDs |
|  | MIDTERM EXAM 2 | 12 |  |  |
| 6A | OO DESIGN PATTERNS  Definition  Description Template  Classification and Selection  Benefits and Drawbacks | 13 | [6] Chapters 1, 2 |  |
| 6B | CREATIONAL DESIGN PATTERNS  Factory Method  Singleton | 13 | [6] Chapter 3 |  |
| 6C | STRUCTURAL DESIGN PATTERNS  Composite  Adapter | 14 | [6] Chapter 4 |  |
| 6D | BEHAVIORAL DESIGN PATTERNS  Observer  Template Method | 15 | [6] Chapter 5 | End of Week 15  Phase 3: Complete Executable Application |
|  | PROJECT DEMOS | 16 |  |  |
|  | FINAL EXAM |  |  |  |

**Course Policies**

1. Announcements related to different aspects of this course (e.g. lectures, quizzes, exams, etc.) may be posted on Google Classroom. Students are expected to view the Google Classroom Stream regularly.
2. All students are expected to attend all lectures from beginning to end. Partial or full absence from a lecture without a valid reason may hamper chances for securing good grades. University’s attendance requirements must be met in order to appear in the final exam.
3. Exams will be closed-book and closed-notes. Syllabus for the final exam will be comprehensive.
4. Students are encouraged to take full advantage of instructor’s office hours. Any doubts regarding concepts covered in class or any questions regarding quizzes, projects, etc. may be clarified during office hours. In case a student is not able to make it during office hours, he/she may schedule an appointment with the instructor for another time slot.
5. Quizzes may be announced or unannounced. A quiz will usually be about 5 – 15 minutes long and it may be given anytime during the lecture. Students missing a quiz will NOT be given a make-up quiz.
6. Students are encouraged to finish the assigned readings BEFORE the lecture. This is likely to improve lecture comprehension and class participation.
7. Students can contest their grades on quizzes and project deliverables ONLY within a week of the release of grades. Exams will be available for review according to university policies.
8. Students are expected to demonstrate the highest degree of moral and ethical conduct. Any student caught cheating, copying, plagiarizing, or using any other unfair means will be strictly dealt-with in accordance with university policies.

1. Numbers in square brackets correspond to books numbered in the “Reference Material” section. [↑](#footnote-ref-1)