BCIT 2214: Software Engineering-BY DR KENNEDY HADULLO

Credit Hours: 42
Pre-requisite: BCIT 2105 Object-Oriented Programming I

Course Purpose

This course is designed to introduce students to the fundamental principles and practices of software engineering. It explores methodologies, tools, and techniques required for the systematic development and management of software systems. The course prepares students to analyze, design, and develop software solutions using modern software engineering approaches, including object-oriented methods and industry-standard development tools.

Learning Outcomes

By the end of this course, students will be able to:

1. **Analyze** various software development process models and **evaluate** appropriate models for different types of software projects.
2. **Differentiate** between types of software requirements and **apply** engineering processes to gather and document them effectively.
3. **Design and develop** software system models using object-oriented techniques and **construct** appropriate UML diagrams.
4. **Apply** software project management principles to plan, organize, and manage software development projects effectively, including risk and resource management

---

10-Week Teaching Plan

Week	Topic	Summary
1	Introduction to Software Engineering	Overview of software engineering: its scope, importance, and challenges. Key differences between software engineering and programming.
2	Software Development Process Models	Examination of software development life cycle (SDLC) models: Waterfall, Agile, Spiral, and Incremental. Criteria for selecting appropriate models.
3	Software Project Management	Principles of project management in software engineering: planning, scheduling, cost estimation, and risk management.

4	Software Requirements Engineering	Introduction to requirements engineering processes. Elicitation, analysis, specification, and validation of functional and non-functional requirements.
5	Systems Analysis and Design	Key elements of analysis and design phases within SDLC. Introduction to use case analysis and object-oriented analysis.
6	UML and Object-Oriented Modelling	Use of Unified Modeling Language (UML) for system modelling: class diagrams, sequence diagrams, activity diagrams, and state diagrams.
7	Software Design Principles and Modularity	Design principles: cohesion, coupling, modularity, and abstraction. Design patterns and best practices in software design.
8	Software Verification, Validation, and Debugging Environments	Techniques for software testing, verification and validation (V&V), unit testing, integration testing, and debugging tools/environments.
9	Software Quality Assurance and Configuration Management	Quality assurance processes, software standards, metrics, version control, and configuration management tools (e.g., Git).
10	Software Maintenance and CASE Tools	Types of maintenance: perfective, adaptive, corrective. Role of CASE tools in software development. Certification and documentation best practices.

Instructional Methods

- Lectures and illustrations
- Practical lab demonstrations
- Discovery and problem-based learning
- Group projects and independent study
- eLearning and experiential learning

Assessment Methods

- Continuous Assessment Tests (CATs), Assignments, Presentations: **30%**
- Final Examination: **70%**
  Total: 100%

Instructional Materials

- Projector and computer
- Handouts and textbooks
- Whiteboards and appropriate software tools

Core Textbooks

1. Brooks, F. P. Jr. (2007). *The Mythical Man Month*. Addison-Wesley. ISBN: 0201835959
2. Sommerville, I. (2011). *Software Engineering* (10th ed.). Addison-Wesley. ISBN: 0133943038
3. Pfleeger, S. L. (2010). *Software Engineering Theory and Practice* (4th ed.). Prentice Hall. ISBN: 0136061699

Reference Textbooks

1. Booch, G. (2008). *Object-Oriented Analysis and Design with Applications* (3rd ed.). ISBN: 978-0201895513
2. Pooley, R., & Stevens, P. (2013). *Using UML: Software Engineering with Objects and Components*. ISBN: 978-0321269676
3. Bruegge, B., & Dutoit, A. H. (2004). *Object-Oriented Software Engineering*. ISBN: 978-0136061250

Course Journals

- *Journal of Software Engineering*, ISSN: 1819-4311
- *Journal of Software and System Modeling*, ISSN: 1619-1374
- *International Journal of Information Systems Modeling and Design*, ISSN: 1947-8186

Let me know if you'd like this in a downloadable format (PDF or Word) or if you need additions like **weekly activities or assessments**.