Department: Mathematics and Computer Sciences
Division: Applied Mathematics
Level and Major: Undergraduate

Course Title: Principles of Software Design
Number of Credits: 3
Prerequisite: Data Structures and Algorithms
Lecturer:

Course Description: Software Development Lifecycle, Process, Planning, Estimation, Agile methodologies, Scrum, Opportunity Canvas, User story mapping, MVP design, Requirement management, Introduction to Architecture, Design (GRASP, GoF), Data Flow Diagram, Coding practices, Quality Control and Assurance

Course Goals and Objectives: Introducing the concepts of process, requirements, design, construction, testing and maintenance of commercial software. In this course, software engineering methods and object-oriented design are taught to students.

Course Topics:

- Software Development Life Cycle-SDLC
- Agile methodologies
- Rational Unified Process-RUP
- Requirement Management
- User Stories and Use Cases
- User Story Mapping for Building MVP
- Requirement Modeling in UML
- Object-Oriented Design in UML
- Design Patterns
- Software Architecture
- Data-flow Diagram
- Quality Assurance
- Software Testing
- CASE Tools
- Version Control and Team Work

Reading Resources:

- Roger S. Pressman, Software Engineering, A Practitioner's Approach, 7th edition, 2010
- Martin Fowler, UML Distilled, 2003
- Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, 1994
- Ash Maurya, Running Lean: Iterate from Plan A to a Plan That Works, 2012
- Jeff Sutherland and J.J. Sutherland, Scrum: The Art of Doing Twice the Work in Half the Time, 2014

Evaluation: