COMP 3350 - Software Engineering 1

Calendar Description: Introduction to software engineering. Software life cycle models, system and software requirements analysis, specifications, software design, testing, and maintenance, software quality.
Prerequisite: COMP 2150.
This course is a prerequisite for: COMP 4050, COMP 4350 and COMP 4560

Outline

1) Introduction to Software Engineering
    Software development and software engineering
2) Software Development Life-Cycle Models (1 week)
    SDL models: waterfall model; "code-like-hell" model; incremental, heavyweight versus lightweight models
3) Requirements Gathering (1 week)
    Purpose of requirements gathering, requirements elicitation, representation using use cases/user stories, how much detail is required, when is requirements gathering performed?
4) Requirements Analysis (1-2 weeks)
    Purpose of analysis, representation using UML diagrams, when is analysis necessary, representing objects, collaborations, and sequences of operations
5) Systems Design (3 weeks)
    Purpose of design, class and sequence UML diagrams, CRC cards, bundling analysis and design, design patterns, types of design patterns; specific design patterns, when are design patterns useful/not useful, how much design is necessary, when is design carried out, what happens when the design must be modified (see refactoring)
6) Implementation and Testing (4 weeks)
    Purpose of implementation and testing, testing methods: black-box, white-box, etc., test-first methodology (e.g. JUnit), refactoring code, iterating back to design, tools for version control etc.
7) Quality (1-2 weeks: covered throughout the course)
     Quality assurance; risk management; best practices
8) Project Presentations (1 week)

Text: none