Midterm Exam Study Guide

CSC 131

Software Engineering - fall semester 2015

This exam will develop, exercise and measure your mastery of three very important skills that every software engineer needs:

- The ability to reuse what you know and to gather information independently
- The ability to gain enough understanding of new information to be able to make practical use of it
- The ability to combine this old and new information to propose solutions to design problems

Preparation checklist:

- Review the material we covered in class, any notes, class quizzes (2 of them), etc.
- Review the material linked from our class web site (SacCT).
- Study with one or more classmates/project teammates, quiz each other on the list of topics.

Format of exam:

You will be given four questions. One question is multiple choice. The remained three are problem based questions where you will be addressing using the information you learned from your checklist.

You will need to justify your plan thoroughly using what you know about the topics below and any other background you have in software engineering and computer science.

Grading:

Your answers will be graded on your demonstration of how well you understand the software engineering topics we have covered in class, and also how well you have been able to understand and incorporate topics from among the list on the following page:

Topics to understand for midterm exam (no particular ordering)

Software engineer

Software engineering

Software Engineering Proposal and its key points

Software process model

Four primary software engineering activities: Specification , Development, Validation, Evolution

Waterfall process model

Spiral process model

Incremental model

Spiral model

Agile Manifesto

Agile software development

Extreme Programming (activities)

Pair Programming

Requirement Engineering

Software Requirement Specification (SRS)

Functional requirements

Non-Functional requirements

Use case diagram

Reuse-oriented software engineering

Types of software components for reuse

Modeling Diagrams:

- Activity Diagram
- Sequence Diagram
- State Diagram
- Data Flow Diagram

Structure Analysis

Object Oriented Analysis

Requirements Process: Elicitation, Analysis, Specification, Validation

Scenario vs Use Case

User Story

Refactoring

Requirement Specification (natural language vs. structured specification)

Requirement validation

Business Requirements - User Requirements – System Requirements

Characteristics of excellent requirements

Advantages of agile processes

Risks/disadvantages of agile processes

Use case text (template)

Basic Flow of events

Alternative flow