Software Engineering, ESOF 322, Fall 2019
Review for exam 1, Sept. 27

You may bring notes on one side of a sheet of paper into the exam. These notes must have been written by you. They cannot be images from the text or Internet or notes gotten from another student.

"Programming in the Small" Chapter 1 – Writing a Program

- Be able to define what is meant by software engineering
- Know the difference between functional requirements, non-functional requirements, project requirements, design constraints, and quality attributes
- Know the difference between verification and validation
- Know the difference between white box and black box testing

"Building a System" Chapter 2

- Know that developing software is a coordination effort between process, product and people
- Know that programs can be complex in terms of many functions, features, interfaces to external systems, simultaneous users, number of data types and data structures, transfer of control, sharing data
- Know what is meant by loose coupling and high cohesion
- Know what is meant by a software development process
- Know the importance of estimation
- Be able to discuss the article "Velocity in Software Engineering" by Tom Killalea

"Engineering of Software" Chapter 3

- Know what was meant by the "Software crisis" and when and where the term "software engineering" was coined. (First used at 1968 North Atlantic Treaty Organization (NATO) conference held in Germany.)
- Know what is needed for startups and next-generation technology companies

"Software Process Models" Chapter *4*

- Be able to describe the waterfall, incremental, spiral, and RUP process models, and to discuss pros and cons of each
- Know what CMM stands for and its goals

"New and Emerging Process Methodologies" Chapter 5

- Know the four key principles of Agile development
- Be able to describe XP, Crystal, UP, Scrum, and the Kanban agile process models and to discuss the pros and cons of each

"Design: Architecture and Methodology" Section 7.2

- Know that architectural knowledge has been codified in architectural styles or patterns, architectural tactics and reference architectures, be able to give examples of each, and to categorize architectural knowledge in one of these
- Know what is meant by pipes-and-filters, event-driven, client-server, model-view-controller, layered, database centric, model centric and 3-tiered architectures are and be able to give examples
- Know what REST stands for and its 6 principles

"Implementation," Section 9.4

- Know the difference between infrastructure as a service, platform as a service and cloud application services, be able to give examples, to categorize a given service, and to discuss pros and cons of each

Testing and Quality Assurance, Section 10.5

- Know the inspection process that we have used in class and the GitLab inspection process we will be using during Sprint 1

Configuration Management, Integration, and Builds, Chapter 11

- Be able to discuss what configuration management is, what might be tracked and the complication of intra-artifact relations
- Know the 3 tiers of configuration management support