CS 4311 Course Outcomes (revised October 14, 2016)

Level 1: Knowledge and Comprehension
Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. The material has been presented only at a superficial level. Upon successful completion of this course, students will be able to:

a) Articulate design principles, including cohesion and coupling, encapsulation, and information hiding.
b) Describe software design concerns related to maintenance.
c) Describe different software architectural styles, such as blackboard, event systems, layered system, and pipe and filters.

Level 2: Application and Analysis
Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details. Upon successful completion of this course, students will be able to:

a) Apply different diagramming techniques for an architectural design.
b) Apply different textual and diagramming techniques for producing a detailed design of a system.
c) Relate general strategies to identify and implement appropriate software architecture styles (including distributed and cloud) for the system under development.
d) Relate general strategies for creating a design of a system.
e) Distinguish between the different levels of cohesion and coupling.
f) Use software development and maintenance tools, such as software documents creation and editing tools, GUI generators, comprehension and analysis tools, supporting activities tools (configuration management tools), verification and validation tools, and security vulnerability analysis tools.
g) Describe differences between unit, integration, system, and acceptance testing.
h) Apply black testing techniques to develop test cases for a variety of test coverages.
i) Apply white-box testing techniques to develop test cases for a variety of test coverages.
j) Apply static and dynamic techniques to analyze non-functional properties, including common security vulnerabilities such as password weakness, over/underflows, and race conditions.
k) Engage in self-directed study to learn new techniques and tools for software design, implementation, and/or testing.

Level 3: Synthesis and Evaluation
Level 3 outcomes are those in which the students can apply the material in new situations. This is the highest level of mastery. Upon successful completion of this course, students will be able to:

a) Conduct a technical review of software design, implementation, and V&V.
b) Create and implement a software configuration management plan.
c) Create an architecture design and a detailed design for a software system.
d) Construct software from a detailed design.
e) Develop a test plan for a software system.
f) Demonstrate an ability to orally present a software design and implementation.
g) Compose software design-related documents that are grammatically correct and technically sound.
h) Apply effective techniques for collaboration and problem-solving within a team.