Note:

Course content may be changed, term to term, without notice. The information below is provided as a guide for course selection and is not binding in any form, and should <u>not</u> be used to purchase course materials.

COURSE SYLLABUS

CSIS 643

SOFTWARE DESIGN

COURSE DESCRIPTION

In this course, the student will be introduced to the architectural design of complex software systems and explore how to successfully design complex software systems. Software system design requires an individual to be able to describe, evaluate, and create systems at an architectural level of abstraction. Therefore, this course considers commonly-used software system structures, techniques for designing and implementing these structures, models, and formal notations for characterizing and reasoning about architectures, tools for generating specific instances of an architecture, and case studies of actual system architectures. Students will learn the skills and background needed to properly evaluate the architectures of existing systems and to design new systems in principled ways using well-founded architectural paradigms.

RATIONALE

The purpose of CSIS 643, Software Design, is to teach the student the concepts of architectural design and to apply these concepts to successfully design complex software systems. This course is important for the student because the quality and longevity of a software system is determined by its architecture, thus making these skills valued by employers.

I. PREREQUISITE

For information regarding prerequisites for this course, please refer to the Academic Course Catalog.

II. REQUIRED RESOURCE PURCHASE

Click on the following link to view the required resource(s) for the term in which you are registered: http://bookstore.mbsdirect.net/liberty.htm

III. ADDITIONAL MATERIALS FOR LEARNING

A. Computer with basic audio/video output equipment

B. Internet access (broadband recommended)

C. Blackboard recommended browsers

D. Microsoft Office

IV. Measurable Learning Outcomes

Upon successful completion of this course, the student will be able to:

A. Discuss the relevance of course material and the use of software design to a biblical worldview. (PLO-1)

B. Recognize major architectural styles in existing software systems. (PLO-3)

C. Assess software design alternatives for a problem. (PLO-2)

D. Use existing definitions and development tools to expedite such tasks. (PLO-2)

E. Apply domain knowledge to specialize an architecture for a particular family of applications. (PLO-3)

F. Recognize how to manage software projects by being able to plan, manage, and control a software project. (PLO-2)

V. Course Requirements and Assignments

A. Textbook readings and lecture presentations

B. Course Requirements Checklist

After reading the Course Syllabus and Student Expectations, the student will complete the related checklist found in Module/Week 1.

C. Discussion Board Forums (4)

Discussion boards are collaborative learning experiences. Therefore, the student is required to create a thread in response to the provided prompt for each forum. Each thread must be at least 400 words and demonstrate course-related knowledge. In addition to the thread, the student is required to reply to the threads of at least 2 classmates. Each reply must be at least 200 words. Each thread and reply must include at least 1 citation in current APA format.

D. Project

Throughout the course, the student will architect and program a software application utilizing object-oriented techniques and graphical user interfaces using Java. This project will be completed in the following steps:

Architecture Part 1

The student will write a 10–12-page (approximately 300 words per page) paper in current APA format that describes the business case and architecturally significant requirements for the project. The paper must include at least 10 scholarly articles as references in addition to the course textbook and the Bible.

Architecture Part 2

The student will write a 6–8-page (approximately 300 words per page) paper (not including models) in current APA format that describes and justifies the complete architecture and design decisions for the project. Architecture Part 2 must include at least 15 fully developed models of at least 8 different types. The paper must include at least 6 scholarly articles as references in addition to the course textbook and the Bible.

The student will add Architecture Part 1 to Architecture Part 2 to create the Architecture Part 2 submission. The final submission will be 16–20 pages (not including models) and contain at least 15 models and at least 16 scholarly articles as references.

Implementation

This step will display the student's completed software application, the culmination of the software design and architecture process. The student will submit his or her Java source code and the architecture.

E. Utility Tree Analysis

The student will complete an interactive activity in which he or she will be given a set of architecturally significant requirements (ASR) to evaluate in terms of business value and quality attributes.

F. Peer Evaluation

The student will provide a reflection thread of at least 400 words and an attachment of his or her final project. In addition to the thread, the student will conduct a peer review of the projects of least 2 classmates. Each peer review must be at least 200 words.

VI. Course Grading and Policies

A. Points

Course Requirements Checklist		10
Discussion Board Forums (4 at 40 pts ea)		160
Project		
Architecture Part 1		200
Architecture Part 2		250
Implementation		250
Utility Tree Analysis		60
Peer Evaluation		80
	Total	1010

B. Scale

A = 940–1010 A- = 920–939 B+ = 900–919 B = 860–899 B- = 840–859

C+ = 820–839 C = 780–819 C- = 760–779 F = 0–759

C. Quizzes/Tests/Exams

For timed quizzes/tests/exams, the student is required to complete the quiz/test/exam within the assigned time. For the student who exceeds this time limit, a penalty of 1 point will be deducted for each minute, or part thereof, he/she exceeds the assigned time limit.

D. Disability Assistance

Students with a documented disability may contact Liberty University Online's Office of Disability Accommodation Support (ODAS) at LUOODAS@liberty.edu to make arrangements for academic accommodations. Further information can be found at www.liberty.edu/disabilitysupport.

If you have a complaint related to disability discrimination or an accommodation that was not provided, you may contact ODAS or the Office of Equity and Compliance by phone at (434) 592-4999 or by email at equityandcompliance@liberty.edu. Click to see a full copy of Liberty's Discrimination, Harassment, and Sexual Misconduct Policy or the Student Disability Grievance Policy and Procedures.

Course Schedule

CSIS 643

Textbook: Bass et al., *Software Architecture in Practice* (2013).

Module/ Week	**Reading & Study**	**Assignments**	**Points**
1	Bass et al.: chs. 1–3 2 presentations	Course Requirements Checklist Class Introductions DB Forum 1	10 0 40
2	Bass et al.: chs. 4–7 1 presentation	Project – Architecture Part 1	200
3	Bass et al.: chs. 8–11 1 presentation	DB Forum 2	40
4	Bass et al.: chs. 12–14 1 presentation	Project – Architecture Part 2	250
5	Bass et al.: chs. 15–18 1 presentation	DB Forum 3 Utility Tree Analysis	40 60
6	Bass et al.: ch. 19 1 presentation	DB Forum 4	40
7	Bass et al.: ch. 20 1 presentation	Project – Implementation	250
8	Bass et al.: chs. 21–22 1 presentation	Peer Evaluation	80
		Total	**1010**

DB = Discussion Board

NOTE: Each course module/week begins on Monday morning at 12:00am. (ET) and ends on Sunday night at 11:59 p.m. (ET). Module/Week 8 ends on at 11:59 p.m. (ET) on **Friday.**