CpS 420 Software Engineering 01 Fall 2023

College of Arts and Science

Fall Semester 2023

Instructor: Sarah Gothard, Ph.D.
Email: sgothard@bju.edu
Office: Alumni 84

Telephone: (864) 242-5100 ext. 8152
                (937) 321-5167 for urgent texts

Office Hours: TTh   9:25 — 9:50 am
                MW   8:10 — 8:50 am
                MW   10:00—10:50 am
                F   8:10—10:50 am
                MWF 12:00—12:50 pm (by appointment)

Required Materials

- *Modern Software Engineering* by David Farley. Boston, MA: Pearson Addison-Wesley 2022. ISBN-13: 978-0-13-731491-1.
- Lab fee: $30 (required, used to purchase access to project support tools)

Course Description

An introduction to methodologies and tools for efficient design, development and testing of object-oriented software. Emphasizes current software engineering best practices for developing medium-to-large scale business systems, including requirements analysis and project management. Students apply the techniques in a semester-long team project. Credits 3. Prerequisite: CPS 301

Course Context

This course addresses the following learning outcomes of the computer science major:
CS1. Design and implement efficient solutions to problems in various domains.
CS3. Communicate technical information effectively, including software design and requirements documents.
CS4. Evaluate and assess software technologies for use in solving specific problems.

Course Goals

The primary goals of this course are to introduce students to the following concepts:

- The standard Software Development Lifecycle
- Asset Management Strategies
- Requirements Engineering
- Software Design
- Software Development Management Strategies

- Test Driven Development Strategies
- Continuous Integration / Continuous Deployment Strategies
- Developing Safety / Life - Critical Software Systems

Assignments

Quizzes will cover both lecture and reading material to gauge student understanding and ensure readiness for future material.

Exams are used to gauge both broad and deep understanding for course material and principles. discussed.

Project is a phased, semester-long project requiring teamwork, planning, coordination of responsibilities, documentation, and time management.

Grading

Qty	Item	Points	Total	Scale:	
7	Quizzes	10-20	100	A	90-100%
2	Exams	150	300	B	80-89%
1	Written Project	50	50	C	70-79%
1	Project	600	600	D	60-69%
Total Points:		1,050		F	<60%

Course Policies

We will be following the "continuous delivery" model. Thus, you should have something in runnable form before the deadline, even if not all the features work. For this reason, I do not accept late work. Work is due at the deadline. **Late work receives a 0**. Notify me immediately if a situation arises necessitating an **extension**.

Do not share class notes with anyone who is not enrolled in the same class section as you are during the same semester.

Professionalism

University classes are a place to sharpen your professional habits. Arrive on time. Dress appropriately. Engage with the material. Take pride in your work. Build relationships. Encourage growth in others.

University Policies

Handbook Policies

Compliance with student handbook policies is expected during class.

Attendance Policy

You are expected to attend class and be on time: https://home.bju.edu/bju-policies/. A partial attendance will be recorded when you miss the beginning or end of a class. If you miss more than 15 minutes of class, you will be marked absent. Students who exceed the 3 allowed absences may be withdrawn from class.

If you need to miss class any reason, please contact me as soon as possible. Assignments and tests should be completed before planned absences.

Accommodations for Students with Disabilities

Students are required under Section 504 to communicate the need for accommodations and provide documentation to the Academic Resource Center Accommodations Office in AL 213. Visit https://success.bju.edu/ for more information. Students are encouraged to seek an appointment in the first week, as accommodations are not provided retroactively.

Academic Honesty and Integrity Policy

See the Computer Science Department's Academic Integrity Policy:

https://cs.bju.edu/academics/policies/academic-integrity-policy/

Taking credit for someone else's work is unethical in any setting. In a university setting, it undermines the ability of faculty to accurately evaluate your competence, harming you and the reputation of the department. For these reasons, the penalties for academic dishonesty may be severe.

Cheating on assignments and tests is forbidden. All work is to be done individually unless group work is explicitly permitted. No collaboration is allowed on tests. For regular individual assignments, we expect that the submitted work represents the student's own intellectual effort, defined as follows:

1. The program was written primarily by the student. This means that most of the code (aside from starting code provided by the instructor) must have been crafted, not copied, by the student.
2. External resources used, whether electronic or from another human, must be documented as follows:
   - Code snippets copied from online resources must be documented by a comment just above the copied snippet giving the URL of the page containing the source.
   - Explanatory help or advice regarding the design or implementation of the solution received from people other than the instructor must be documented in a report accompanying the assignment submission. This report must detail:
     - Source of information (e.g., name/email of the person who helped)
     - Relevance (i.e., how this resource helped and/or what it provided)
   - Note that students must not consult a solution to the assignment as a resource in crafting their own solution, nor share their own solution with another student. Doing so constitutes cheating.
3. The student must be able to explain, on demand, the entirety of the program on both the syntactic and semantic level.

Not all kinds of programming assignments require the same demonstration of personal intellectual effort. In the absence of any specific instructions, students should assume that at a minimum:

- For individual lab assignments, requirements 1 and 3 apply.

- For individual programming assignments, all three requirements apply.
- For group programming projects, only requirement 3 applies.

Failure to comply with any relevant integrity requirement constitutes cheating. Such incidents will be reported to the academic integrity committee. To avoid trouble:

- Do not look at another student's program code when seeking assistance. On the other hand, if another student is seeking help from you, never use your own program code as an example. The only acceptable reason another student may look at your code is to help you find a problem in your program.
- Do not write program code while another student (or lab assistant) is sitting with you. You may work out designs in pseudocode on paper with another student, but you must write program code by yourself.
- When seeking assistance from another person on a program assignment, always get his/her name so you can fulfill the documentation requirements.

Generative AI

Since the goal of the assignments in this course is to learn to develop the skills covered NOT complete the tasks assigned, and since the use of AI to complete or jumpstart tasks defeats the goal of the assignments, you may not use generative AI tools (i.e. Chat GPT, Bing Chat, Google Bard, etc.) in this course for any assignment without the professors express permission. Should an AI tool be used with permission, its use must be documented.

Tentative Schedule

Week	Topic	Readings	Asgs
8/23–26	Introduction, Software Engineering	MSE: forward, preface Download forward, preface, chapter 1 Video: Git	Sat: Team Setup
8/28–9/2	Working Iteratively, TDD	MSE: chapter 4 Video: Docker	Quiz 1
9/4–9	XP, Agile	MSE: chapter 5 Video: Terraform	Sat: First Sprint
9/11–16	SOLID	MSE: chapter 2 Video: GitHub Actions	Quiz 2
9/17–23	Design Patterns	MSE: chapter 3 Video: Backend	Sat: Second Sprint
9/25–30	Design Patterns	MSE: chap. 6-7 Video: Vue/Nuxt	Quiz 3

10/2–7	CICD	MSE: chapter 8 Video: [Digital Ocean](#)	Sat: [Third Sprint](#) Quiz 4
10/9–14	Requirements	MSE: chapter 9 Video: [Ansible/Kubernetes](#)	Midterm
10/16–21	*Fall Break*	Modeling	Quiz 5
10/23–28	Modularity	MSE: chapter 10 Video: Deployment	Tue: [Fourth Sprint](#)
10/30–11/4	Cohesion, Separation of Concerns	MSE: chapter 11 Video: Backend Security	Sat: Requirements Project
11/6–11/11	Information Hiding and Abstraction	MSE: chapter 12 Video: Testing	Tue: [Fifth Sprint](#)
11/13–11/18	Managing Coupling	MSE: chapter 13	Quiz 6
11/20–11/25	*Thanksgiving Break*		
11/27–12/2	Dependability, Availability Engineering	MSE: chapter 14	Tue: [Sixth Sprint](#) Quiz 7
12/4–12/9	Safety, Security Engineering, Software Product Lines	MSE: chapter 15	Sat: [Audit and Analysis](#)
12/12	8–9:10 a.m. **Final Exam**		

Copyright Policy