

San José State University
College of Science/Department of Computer Science
CS251B, Object-Oriented Design, Section 1
Spring Semester, 2018

Course and Contact Information

Instructor:	Jon Pearce
Office Location:	416 MacQuarrie Hall
Telephone:	(408) 924-5065
Email:	jon.pearce@sjsu.edu
Office Hours:	MW 10:30 – 12:00
Class Days/Time:	MW 12:00 – 13:15
Classroom:	225 MacQuarrie Hall
Prerequisites:	C- or better in CS 160 (Software Engineering) or instructor consent

Course Description:

Catalog Description:

Course covers important concepts, activities, and artifacts of the design phase of object-oriented software development. Topics include design metrics, design patterns, refactoring, frameworks, and testing. Prerequisite: CS 160 or instructor consent.

Section Description:

Topics in the approximate order of presentation include: modeling collaborations using UML, Design principles, design patterns, and architecture patterns. Architecture patterns are implemented as frameworks.

Course Learning Outcomes:

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

1. Create design models using UML and design patterns
2. Implement design models
3. Distinguish between good and bad designs

Required Texts/Readings

Textbook

Lecture note and other materials will be posted at CS251B Course Website:

<http://www.cs.sjsu.edu/faculty/pearce/modules/courses/Sp18/cs251B/index.htm>

Other Readings

My lectures draw substantially from the following books:

Pattern-Oriented Software Architecture, volume 1; Buschmann, et. al.; Wiley; 1996.

Design Patterns, Elements of Reusable Object-Oriented Software; Gamma, et. al.; Addison-Wesley; 1994.

Other equipment / material requirements

Students should bring laptops to class. The following software should be installed:

- *Star UML 2*; <http://staruml.io/>
- *Eclipse IDE for Java Developers*; <http://www.eclipse.org/downloads/>

Course Requirements and Assignments

There will be two types of assignments, labs and projects. Both will require students to create UML models using StarUML and, in some cases, to implement those models in Java or JavaScript. Labs are begun in class and completed at home and usually consist of multiple problems. Projects are single problems requiring detailed solutions. Assignments will be submitted through Canvas. Rubrics will be used to grade the assignments. Models will typically be judged on accuracy, completeness, and implementability.

There will be a midterm and a comprehensive final exam. Exams will be posted and submitted through Canvas. Students will use their laptops to create UML diagrams and write code. Access to notes and Internet is permitted, but all forms of communication are forbidden (except with the proctor).

SJSU classes are designed such that in order to be successful, it is expected that students will spend a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on. More details about student workload can be found in [University Policy S12-3](#) at <http://www.sjsu.edu/senate/docs/S12-3.pdf>.

Grading Policy

Grades will be determined by programming assignments (50%), one midterm (20%), and a final exam (30%).

Assuming a standard distribution of point totals I will use the following scale for assigning final grades:

A/A-	85% - 100%
B+/B/B-	70% - 84%
C+/C/C-	55% - 69%
D+/D/D-	40% - 54%
F	0% - 39%

Note: The grade totals and percentages posted on Canvas do not reflect the weights given above and should be ignored.

Classroom Protocol

Students should bring laptops to class and be prepared to work together on in-class labs.

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' [Syllabus Information web page](http://www.sjsu.edu/gup/syllabusinfo/) at <http://www.sjsu.edu/gup/syllabusinfo/>

Course Schedule

The tentative course schedule below is subject to change. Notification of changes will be made in class.

Date	Day	Lecture	Assignment
24-Jan	Wed	Overview SE, OOA, OOP	
29-Jan	Mon	Overview SE, OOA, OOP	1
31-Jan	Wed	Modeling collaborations in UML	
5-Feb	Mon	Modeling collaborations in UML	2
7-Feb	Wed	Modeling collaborations in UML	
12-Feb	Mon	Modeling collaborations in UML	3
14-Feb	Wed	Design principles, patterns, & metrics	
19-Feb	Mon	Design principles, patterns, & metrics	4
21-Feb	Wed	Design principles, patterns, & metrics	
26-Feb	Mon	Design principles, patterns, & metrics	
28-Feb	Wed	Pipeline architectures	5
5-Mar	Mon	Pipeline architectures	
7-Mar	Wed	Model-view-controller architectures	
12-Mar	Mon	Model-view-controller architectures	6
14-Mar	Wed	Model-view-controller architectures	
19-Mar	Mon	Midterm Review	
21-Mar	Wed	Midterm Exam	
26-Mar	Mon	Spring Recess	
28-Mar	Wed	Spring Recess	
2-Apr	Mon	Distributed architectures	
4-Apr	Wed	Distributed architectures	
9-Apr	Mon	Agent-based architectures	7
16-Apr	Wed	Agent-based architectures	
18-Apr	Mon	Agent-based architectures	
23-Apr	Wed	Agent-based architectures	
25-Apr	Mon	Open architectures	8

30-Apr	Wed	Open architectures
2-May	Mon	Open architectures
8-May	Wed	Open architectures
14-May	Mon	Final Review
16-May	Wed	Final Exam

Assignment details can be found at:

[. Assignments](#)

