Course Syllabus

✎ Edit

San Jose State University
College of Science
Department of Computer Science
CS151, Object-Oriented Design, Section 5, Fall 2025

Course and Contact Information

- Instructor: Dr. Suneuy Kim
- Instruction: In Person
- Office Location: MacQuarrie Hall 217 (MH217)
- Telephone: 408-924-5122
- E-mail: suneuy.kim@sjsu.edu (Preferred mode of contact is via email.)
  - When you send me an e-mail to ask a question, use [Q] in a subject line to get a reply from me within a reasonable response time. Here is an example subject line to ask a question.

```
[Q] Your subject goes here
```

- Class Days/Time/Classroom: TuTh 9:00 AM - 10:15 AM, MacQuarrie Hall 324
- Office Hours: TuTh 10:20 AM - 11:20 PM at Zoom, Meeting ID: 838 0090 4104
- Course Prerequisites (each listed course should be with a grade of C- or better)
  MATH 42, and CS46B or an equivalent course to CS46B in Java
  or
  MATH42, and CS49J and an equivalent course to CS46B in non-Java
  or
  Instructor consent.

Course Description

Design of classes and interfaces. Object-oriented design methodologies and notations. Design patterns. Generics and reflection. Exception handling. Concurrent programming. Graphical user interface programming. Software engineering concepts and tools. Required team-based programming assignment.

Course Objectives

- OO Design:

    - Introduce core UML concepts
    - Introduce a simplified OO analysis and design methodology
    - Present the concept of design pattern
    - Present the concept of a software framework
- Java Language:
    - Make students proficient in the use and creation of interfaces and inheritance hierarchies
    - Make students proficient in the Java type system
    - Introduce threads and thread safety
- Software Engineering:
    - Introduce a GUI toolkit, including basic widgets and the event handling mechanism
    - Introduce basic software engineering concepts and tools

Course Learning Outcomes

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

- OO Design
    - Interpret and produce UML class diagrams and UML sequence diagrams
    - Develop simple use cases, perform noun-verb analysis, interpret and produce CRC cards
    - Appropriately select and apply key design patterns in the construction of a software application
    - Be able to follow a systematic OO design methodology
- Java language
    - Create a class hierarchy involving existing and new interfaces and classes, including inner classes.
    - Design, implement, test, and debug programs in an object-oriented language, involving the creation of at least 10 classes and interfaces
    - Use generic types, reflection, and lambda expressions
    - Implement concurrent programs and use thread-safe data structures
- Software Engineering
    - Use a GUI toolkit to create a graphical user interface involving frames, buttons, text components, panels, menus, and simple geometric shapes
    - Be able to document use cases for a simple team project
    - Be able to plan and track a simple team project
    - Be able to use a version control system and an automated build system

BS in Computer Science Program Outcomes Supported

These are the BSCS Program Outcomes supported by this course:

- (A) An ability to apply knowledge of computing and mathematics to solve problems
- (B) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

- (C) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
- (D) An ability to function effectively on teams to accomplish a common goal
- (I) An ability to use current techniques, skills, and tools necessary for computing practice
- (J) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the trade offs involved in design choices
- (K) An ability to apply design and development principles in the construction of software systems of varying complexity

Course Topics

Topics	Weeks
Object-Oriented Design Process	2
Interface Types, Polymorphism, Lambda Expressions, and GUI programming	2.5
Design Patterns	2.5
Inheritance and Abstract Classes	2
The Java Object Model - Java Type Systems, java.lang.Object class, Serialization, Reflection, and Generics	3
Concurrent Programming	2
Total	14

Required Texts/Readings

- Textbook: Object-Oriented Design & Patterns, 3rd edition, by Cay Horstmann

Course Requirements and Assignments

- Programming Assignments and Project
  - Four individual programming assignments involving design and implementation. (All submitted programs should be original. For some assignments, you are allowed to borrow a code from the textbook. If this is the case, it will be clearly specified in the assignment description and you have to acknowledge it in the corresponding source code.)
  - A group project with 3 members per group in the last month of the semester involves OO design and GUI programming. A project description and guidelines will be posted later.
  - Unless I specifically ask for hard copies, all assignments will be submitted through my course web page. Find the homework submission link corresponding to your section at the left side of the course web page.
- Submission/Late Policy

- Any assignments/project turned in past the deadline will get a penalty: For each late day, a 20% of the maximum obtainable score of the work will be taken out of what you earned. (a late day is one 24 hour period beyond the due date). For example, suppose the maximum score of an assignment is 100 and you earned 80 points. If the submission is late by two days, the final score of the assignme nt would be 80 - 2 * 20 = 40.
- Any submission turned in more than 48 hours past the deadline will result i n a grade of zero for that assignment.
- On-line submission: You can submit your work multiple times. If then, the latest one will be considered as the final submission. If the final submission is late, the late policy will be applied.
- E-mail submissions will not be accepted for grading.

- Teamwork Policy
  - Once a team is formed, it will last through out the semester. If you dissolve your team, a significant amount of penalty will be determined by the instructor and given to both parties.
  - For the project, students are expected to report their own results as well as their collaborators. The task responsibility and contribution of every team member must be precisely documented in a report. Team members will be graded individually based on the report and peer evaluation.
- Software
  - Programming Language: JDK 21 is the latest long-term support release of Java SE Platform
    - **Download (https://www.oracle.com/java/technologies/downloads/)** at https://www.oracle.com/java/technologies/downloads/
  - UML Tools
    - **Violet (http://horstmann.com/violet)** at http://horstmann.com/violet
    - You may use a UML tool of your choice.
  - IDE
    - **IntelliJ IDEA (http://jetbrains.com/idea)** at **http://jetbrains.com/idea (http://jetbrains.com/idea)**
    - **Eclipse (http://eclipse.org/)** at http://eclipse.org/

Evaluation (Exams)

- There will be two midterm exams and one comprehensive final exam. All exams will be given online via Canvas. The dates of midterm exams will be announced with fair notice. The final exam date is scheduled by the university and cannot be changed. See the semester schedule at the end of the syllabus for the final exam schedule.
- Makeup Exam Policy
  Absolutely no make-up exams will be offered under any circumstances. For those who couldn't take the exam or worked hard but had a bad day on the exam day ending up with a low score, I offer the following opportunity to possibly replace your worst midterm score with the final score. If your final exam (percentage) grade is higher than your worst midterm (percentage) grade, then I will replace the worst midterm grade with your final exam grade. For example, if you have a 60% on your worst

midterm and you receive an 80% on the final exam, I will replace the 60% by 80% in the computation of your course grade.

Grading Information

You will receive the final grade based on the weighted average score on your performance. The grading weights are as follows.

- Exam I: 20 %
- Exam II: 20 %
- Final Exam: 35 %
- Programming Assignments: 15 %
- Project: 8 %
- Participation: 2 % (poll in class)

I first try scores of 90, 80, and 70 to cut off letter grades of A-, B-, and C-, respectively. If overall class performance is too low to use these cut offs, I set a cut off of C- to a lower score than the class total average but a higher score than 60 (this number may change), and divide the students' group above the cut off of C- into A+, A, A-, B+, B, B-, C+, C, C-. The rest of students will be given by a grade of D+, D, D-, F or WU depending on their class performance.

Technology Requirements

Students are required to have an electronic device (laptop, desktop or tablet) with a camera and built in microphone. SJSU has a free **equipment loan (https://www.sjsu.edu/learnanywhere/equipment/index.php)** (https://www.sjsu.edu/learnanywhere/equipment/index.php) program available for students.

Students are responsible for ensuring that they have access to reliable Wi-Fi during tests. If students are unable to have reliable Wi-Fi, they must inform the instructor at the latest one week before the test date. See **Learn Anywhere (https://www.sjsu.edu/learnanywhere/equipment/index.php)** website (https://www.sjsu.edu/learnanywhere/equipment/index.php) for current Wi-Fi options on campus.

Online Exams

Proctoring Software and Exams

Exams will be proctored in this course through Respondus Monitor and LockDown Browser. Please note it is the instructor’s discretion to determine the method of proctoring. If cheating is suspected the proctored videos may be used for further inspection and may become part of the student’s disciplinary record. Note that the proctoring software does not determine whether academic misconduct occurred, but does determine whether something irregular occurred that may require further investigation.

Students are encouraged to contact the instructor if unexpected interruptions (from a parent or roommate, for example) occur during an exam.

Testing Environment: Setup

- No earbuds, headphones, or headsets
- The environment is free of other people besides the student taking the test.
- No other browser or windows besides Canvas opened.
- A workplace that is clear of clutter (i.e., reference materials, notes, textbooks, cellphone, tablets, smart watches, monitors, keyboards, gaming consoles, etc.)
- Well-lit environment. Can see the students' eyes and their whole face. Avoid having backlight from a window or other light source opposite the camera.

Students must:

- Remain in the testing environment throughout the duration of the test.
- Keep full face in full view of the webcam

Technical difficulties

Internet connection issues: Canvas autosaves responses a few times per minute as long as there is an internet connection. If your internet connection is lost, Canvas will warn you but allow you to continue working on your exam. A brief loss of internet connection is unlikely to cause you to lose your work. However, a longer loss of connectivity or weak/unstable connection may jeopardize your exam.
Other technical difficulties: Immediately email the instructor a current copy of the state of your exam and explain the problem you are facing. Your instructor may not be able to respond immediately or provide technical support. However, the copy of your exam and email will provide a record of the situation.

Contact the SJSU technical support for Canvas:

```
Center for Faculty Excellence and Teaching Innovation:
cfeti@sjsu.edu 408.924.2337
```

Classroom Protocol

- Policy on Academic Integrity
  - Any cheating on an exam will result in a grade of F in the class.
  - The submitted programming assignments will be checked for plagiarism using Stanford MOSS against the submissions from the current and previous semesters. If duplicate programs are found, both the provider and the copier will receive 0 point on the assignment. A second offense results in a grade of F in the class.
  - Any incident of academic dishonesty will be reported to University for disciplinary action.
- Attendance: **University policy F15-12 (http://www.sjsu.edu/senate/docs/F15-12.pdf)** at http://www.sjsu.edu/senate/docs/F15-12.pdf states that "Students should attend all meetings of their classes, not only because they are responsible for material discussed therein, but because active

participation is frequently essential to insure maximum benefit for all members of the class. Attendance per se shall not be used as a criterion for grading."

- Consent for Recording of Class and Public Sharing of Instructor Material : **University Policy S12-7 (http://www.sjsu.edu/senate/docs/S12-7.pdf)** , http://www.sjsu.edu/senate/docs/S12-7.pdf, requires students to obtain instructor's permission to record the course:
  - "Common courtesy and professional behavior dictate that you notify someone when you are recording him/her. You must obtain the instructor's permission to make audio or video recordings in this class. Such permission allows the recordings to be used for your private, study purposes only. The recordings are the intellectual property of the instructor; you have not been given any rights to reproduce or distribute the material."
  - "Course material cannot be shared publicly without his/her approval. You may not publicly share or upload instructor generated material for this course such as exam questions, lecture notes, or homework solutions without instructor consent."

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 (http://www.sjsu.edu/gup/syllabusinfo/)** web page at http://www.sjsu.edu/gup/syllabusinfo/"
- Reference to deadlines pertaining to the current semester, please refer to **https://www.sjsu.edu/registrar/calendar/ (https://www.sjsu.edu/registrar/calendar/spring-2025.php)**

CS151 Object-Oriented Design, Fall 2025: Semester Schedule

Subject to change with fair notice at least one class period in advance. Students will be notified in class and/or via Canvas should any changes occur.

Week	Topics	Assignments
1	Introduction to CS151	
1	Object-Oriented Design Process	
2	Object-Oriented Design Process	
2	Object-Oriented Design Process	
3	Object-Oriented Design Process	
3	Object-Oriented Design Process	
4	Interface Types and Polymorphism	
4	Interface Types and Polymorphism	
5	Interface Types and Polymorphism	

5	Interface Types and Polymorphism	
6	Interface Types and Polymorphism	
6	Interface Types and Polymorphism	
7	Lambda Expression	
7	Patterns and GUI Programming	
8	Midterm I	
8	Patterns and GUI Programming	
9	Patterns and GUI Programming	
9	Patterns and GUI Programming	
10	Patterns and GUI Programming	
10	Inheritance and Abstract Classes	
11	Inheritance and Abstract Classes	
11	Inheritance and Abstract Classes	
12	Inheritance and Abstract Classes	
12	Midterm II	
13	The Java Object Model	
13	The Java Object Model	
14	The Java Object Model	
14	Concurrent Programming	
15	Concurrent Programming (Last Day of Instruction)	
Final Exam	Thu, December 11, 8:30-10:30 AM	