College of Science · Computer Science

Object-Oriented Design Section 06

CS 151

Spring 2023   3 Unit(s)   01/25/2023 to 05/15/2023   Modified 01/24/2023

👤 Contact Information

Abhishek Gaikwad

Email: abhishekprakash.gaikwad@sjsu.edu
Phone: (669) 247-8573

Classroom: MH222

Office Hours

Before class
Tuesday, Thursday, 3:00 PM to 4:00 PM, via. Zoom https://sjsu.zoom.us/j/6306253552

🖥 Course Description and Requisites

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.

Prerequisite(s): MATH 42, CS 46B, and CS 49J (or equivalent knowledge of Java) (with a grade of "C-" or better in each); Allowed Declared Majors: Computer Science, Applied and Computational Math, Software Engineering, or Data Science; or instructor consent.

Letter Graded

📋 Program Information

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

◎ Course Goals

- Understand the fundamentals of object-oriented design and programming in Java.
- Be aware of various methodologies and principles in software design and development.
- Have the ability to design, implement, and document an application using best practices.

📊 Course Learning Outcomes (CLOs)

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

- **Java Language:**

1. Implement Program

2. Write code
3. Perform unit testing
4. Integrate subsystems
5. Resolve defects and revise and adapt existing code

- **OO Design and principles:**

1. To explore & understand the principles of Object Oriented Programming (OOP)
2. Improve the quality and productivity of system analysis and design by making it more usable.
3. Interpret and produce UML diagrams.

- **GUI Programming:**

1. Create a GUI that enables users to communicate with a computer through the use of symbols, visual metaphors, and pointing devices.

📘 Course Materials

I will not be following any specific textbook for this course. Here are some of the links which might be helpful if you are stuck on assignments:

- [Java Documentation. (https://docs.oracle.com/javase/8/docs/)](https://docs.oracle.com/javase/8/docs/)
- [Stack Overflow. (https://stackoverflow.com/)](https://stackoverflow.com/)
- The references at the end of each presentation.

Most of my course content is based on:

1. Cay Horstmann, "Object-Oriented Design & Patterns," 3rd edition. ISBN: 9780471744870. The resources can be found at: http://horstmann.com/oodp3/
2. Eric Freeman, Elisabeth Freeman, with Kathy Sierra and Bert Bates, "Head First design patterns," 1st ed. ISBN: 9780596007126

📋 Course Requirements and Assignments

Programming Assignments (25%)

There will be several programming assignments involving OO programming, OO design, and UML diagrams. All assignments are individual. Late assignments will not be accepted.

Quizzes (10%)

There will be short quizzes after each module. The purpose of these quizzes is to encourage students to read and understand the concept through the course material. The objective of these quizzes is to help students understand the concepts better and apply them in the assignments and in preparation for the midterm and final.

Project (20%)

A group project with 3 members per group in the last month of the semester involves the OO design and GUI programming. Even if its a group project the grade will be individual based on your contributions to the project.

Midterm and Final (20% each)

There will be one midterm and one final consisting of MCQs and written answers. Questions can come from quizzes, class notes, slides, assignments, and from discussions in class.

Class Participation (5%)

To encourage participation 5% of your final grade will come from your participation. Points are given on the basis of participation in exercises, in-class/discord discussions, helping teammates/peers, etc.

✔ Grading Information

Grading Scale

Grade	Percentage
A plus	96 to 100%
A	93 to 95%
A minus	90 to 92%
B plus	86 to 89 %
B	83 to 85%
B minus	80 to 82%
C plus	76 to 79%
C	73 to 75%
C minus	70 to 72%
D plus	66 to 69%
D	63 to 65%
D minus	60 to 62%
F	59% and below

Criteria

Type	Weight	Topic	Notes
Assignments	25%		
Quizzes	10%		
Project	20%		
Midterm	20%		
Final	20%		
Participation	5%		

🏛 University Policies

Per University Policy S16-9 (http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on Syllabus Information web page (https://www.sjsu.edu/curriculum/courses/syllabus-info.php) (https://www.sjsu.edu/curriculum/courses/syllabus-info.php). Make sure to visit this page to review and be aware of these university policies and resources.

📅 Course Schedule

When	Topic	Notes
Week 1 & 2	Great Software Begins Here	Introduction to Course, Syllabus Discussion, Java Recap, Attributes, Methods, Modifiers, Encapsulation, Introduction to OO Design
Week 3	Modelling the world	Functions in Java, Packages, Classes and Objects, Constructors and Destructors, Memory Models, JVM vs JRE vs JDK
Week 4	Object-oriented way of becoming #wealthy	Inheritance
Week 5	"BIG BRAIN ENGINEERING"	Polymorphism, Abstraction, Inner Class, Interfaces.
Week 6	Bringing Order to Chaos	UML Diagrams: Class, Sequence, and State. Design Documentation.
Week 7	Revision and Midterm	Midterm Date: March 09, 2023 Thursday (4:30PM to 5:45PM in MH222)
Week 8	Midterm Discussion and Project Inception	
Week 9	"It works on my machine"	Exception Handling.
Week 10	SPRING BREAK	
Week 11	Taking your code into the real world	GUI
Week 12, 13	Originality is Overrated	Design Patterns
Week 14	"All code is guilty until proven innocent"	Iterating and Testing. Generics and Reflection.
Week 15	With great performance comes great memory management responsibility	Concurrent Programming
Week 16	Project Presentation	
	The Great Battle For Our Semester	Final Exam. Thursday, May 18 2:45-5:00 PM in MH222