Instructor:	Paul Nguyen
Office Location:	ENG 281
Email:	paul.nguyen@sjsu.edu (mailto:paul.nguyen@sjsu.edu)
Office Hours:	Saturdays, 3:00 pm – 4:00 pm (by appointment only)
Class Days/Time:	Saturdays, 12:00 pm – 2:45 pm (Section 3)
Classroom:	ENG 189
Prerequisites:	Computer Engineering and Software Engineering Majors only. Instructor Consent Required. Not available to Open University students. *Students who do not provide documentation of having satisfied the class prerequisite and co-requisite requirements (if any) by the second class meeting will be dropped from the class.*

Week	Date	Topics, Readings, Assignments, Deadlines
		CMPE 202 - Books, Resources & Tools (https://sjsu.instructure.com/courses/1244332/pages/cmpe-202-books-resources-and-tools)

1	8/26	**Topics:** Class Introduction, Software Engineering Overview, OOP Foundations
		
		Readings:
		
		• ***Kanban and Scrum*** (https://www.infoq.com/minibooks/kanban-scrum-minibook)
		  ◦ *Part 1 - Comparison*
		  ◦ *Note:* ***Only read "Part 1"***
		
		Assignments:
		
		• **CMPE 202 - Sign Honesty Pledge & Evidence of Classified Standing** (https://sjsu.instructure.com/courses/1244332/assignments/4522722)
		• **CMPE 202 - CANVAS TEST** (https://sjsu.instructure.com/courses/1244332/assignments/4522721)
		• **CMPE 202 - LAB #0 - OOP Concepts** (https://sjsu.instructure.com/courses/1244332/assignments/4524882)
		
		Handouts:
		
		• **1708 - CMPE-202-F17-Nguyen (v1.0).pdf** (https://sjsu.instructure.com/courses/1244332/files/47654260/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47654260/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47654260/download?wrap=1)
		• **honestyPledge 20160202.pdf** (https://sjsu.instructure.com/courses/1244332/files/47643252/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47643252/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47643252/download?wrap=1)
		• **Hello Software Engineering - 2017.PDF** (https://sjsu.instructure.com/courses/1244332/files/47669899/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669899/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669899/download?wrap=1)
		• **Classical vs. Object-Oriented - 2017.PDF** (https://sjsu.instructure.com/courses/1244332/files/47669895/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669895/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669895/download?wrap=1)
		• **Foundations of OOP Paradigm.png** (https://sjsu.instructure.com/courses/1244332/files/47669896/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669896/download?wrap=1)
		• **Object-Oriented Domain Modeling.png** (https://sjsu.instructure.com/courses/1244332/files/47669897/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47669897/download?wrap=1)

2	**9/2**	**Topics:** Agile Methodologies, CRC Cards, JUnit, Beyond OOP
		
		Readings:
		
		• ***Do Better Scrum*** (https://www.infoq.com/minibooks/do-better-scrum)
		  ◦ *What is Scrum?*
		  ◦ *Understanding Scrum*
		  ◦ *Adopting Scrum*
		  ◦ *More on Agile and Lean*
		• ***UML Distilled*** (https://martinfowler.com/books/uml.html)
		  ◦ Chapter 1 - Introduction
		  ◦ Chapter 2 - Development Process
		  ◦ Chapter 12 - Communication Diagrams
		
		Assignments:
		
		• Review Scrum Guide & Take Scrum Developer Open Assessment:
		  ◦ **https://www.scrum.org/open-assessments** (https://www.scrum.org/open-assessments)
		  ◦ **http://www.scrumguides.org** (http://www.scrumguides.org)
		• **CMPE 202 - LAB #1 - Code Maintainability** (https://sjsu.instructure.com/courses/1244332/quizzes/1223117)
		• **CMPE 202 - LAB #2 - Agile Practices** (https://sjsu.instructure.com/courses/1244332/quizzes/1223118)
		
		Handouts:
		
		• **Agile Software Engineering.PDF** (https://sjsu.instructure.com/courses/1244332/files/47749720/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47749720/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47749720/download?wrap=1)
		• **CRC Cards & Object Collaborations.PDF** (https://sjsu.instructure.com/courses/1244332/files/47682750/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682750/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682750/download?wrap=1)
		• **UML Sketching - On the Cloud.PDF** (https://sjsu.instructure.com/courses/1244332/files/47682749/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682749/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682749/download?wrap=1)
		• **XP, jUnit, BDD & TDD.PDF** (https://sjsu.instructure.com/courses/1244332/files/47682748/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682748/download?wrap=1) (https://sjsu.instructure.com/courses/1244332/files/47682748/download?wrap=1)

3	**9/9**	**Topics:** UML Diagrams – Class & Sequence Diagrams
		
		Readings:
		
		• ***UML Distilled*** (https://martinfowler.com/books/uml.html)
		  ◦ *Chapter 3 - Class Diagrams*
		  ◦ *Chapter 4 - Sequence Diagrams*
		  ◦ *Chapter 5 - Class Diagrams (Advanced)*

4	**9/16**	**Topics**: UML Diagrams – Use Case, State & Activity Diagrams
		
		Readings:
		
		• ***UML Distilled*** (https://martinfowler.com/books/uml.html)
		  ◦ Chapter 9 - Use Cases
		  ◦ Chapter 10 - State Machines
		  ◦ Chapter 11 - Activity Diagrams

5	**9/23**	**Topics**: Design Patterns – State, Singleton, Adapter, Observer **Readings:** • ***JavaScript Design Patterns*** (http://proquest.safaribooksonline.com.libaccess.sjlibrary.org/book/programming/javascript/9781785882166) ◦ Chapter 2. Organizing Code ◦ Chapter 3. Creational Patterns (Singleton) ◦ Chapter 4. Structural Patterns (Adapter) ◦ Chapter 5. Behavioral Patterns (State, Observer)
6	**9/30**	**Topics**: Design Patterns – Proxy, Decorator, Factory Method, Chain of Responsibility **Readings:** • ***JavaScript Design Patterns*** (http://proquest.safaribooksonline.com.libaccess.sjlibrary.org/book/programming/javascript/9781785882166) ◦ Chapter 3. Creational Patterns (Factory Method) ◦ Chapter 4. Structural Patterns (Proxy, Decorator)
7	**10/7**	***Silicon Valley Code Camp Day (Extra Credit Event)*** (https://www.siliconvalley-codecamp.com/)
8	**10/14**	***Midterm Exam***
9	**10/21**	**Topics**: Design Patterns – Command, Composite, Iterator, Strategy **Readings:** • ***JavaScript Design Patterns*** (http://proquest.safaribooksonline.com.libaccess.sjlibrary.org/book/programming/javascript/9781785882166) ◦ Chapter 3. Creational Patterns (Composite) ◦ Chapter 5. Behavioral Patterns (Command, Iterator, Strategy)
10	**10/28**	**Topics:** Design Patterns Labs (Case Studies)
11	**11/4**	**Topics**: JavaScript Functional & Reactive Programming **Readings:** • ***JavaScript Design Patterns*** (http://proquest.safaribooksonline.com.libaccess.sjlibrary.org/book/programming/javascript/9781785882166) ◦ Chapter 6. Functional Programming ◦ Chapter 7. Reactive Programming
12	**11/11**	**Topics**: Advanced Design Patterns **Readings:** • ***JavaScript Design Patterns*** (http://proquest.safaribooksonline.com.libaccess.sjlibrary.org/book/programming/javascript/9781785882166) ◦ Chapter 8. Application Patterns (MVC, MVP, MVVM) ◦ Chapter 13. Advanced Patterns (AOP, Mix-ins, Etc..)
13	**11/18**	**Team Game Day**
14	**11/14**	***Thanksgiving Break (No Class)***

15	**12/2**	**Project Demo (Day 1)**
16	**12/9**	**Project Demo (Day 2)**
17	**12/16**	***Final Exam (Location and Times TBD)***