San José State University
College of Science/Department of Computer Science
CS151, Object-Oriented Design, Section 5, Fall, 2018

Course and Contact Information

Instructor: Dr. Katarzyna Tarnowska
Office Location: MacQuarrie Hall (MH) 215
Telephone: (408) 924-5076
Email: Katarzyna.tarnowska@sjsu.edu
Office Hours: Monday & Wednesday, 10:30AM – 11:30AM
Class Days/Time: Monday & Wednesday, 9:00AM – 10:15AM
Classroom: Duncan Hall 450
Prerequisites: Math 42, CS46B, and CS 49J (or equivalent knowledge of Java) with a grade of C- or better in each or instructor consent.

Course Format

Technology Intensive, Hybrid, and Online Courses
Students are required to bring their own laptops to the class or borrow laptops from the library (check Student Technology Resources at http://www.sjsu.edu/gup/syllabusinfo/#StudentTechResources) for the designated “lab” sessions.

Faculty Web Page and MYSJSU Messaging
Course materials such as syllabus, handouts, notes, assignment instructions, etc. will be published on Canvas Learning Management System course login website at https://sjsu.instructure.com/courses/1263537. You are responsible for regularly checking with the messaging system through MySJSU at http://my.sjsu.edu to learn of any updates.

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

Course Goals

- **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

GUI Programming:
  Introduce a GUI toolkit, including basic widgets and the event handling mechanism

Course Learning Outcomes (CLO)

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

1. 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 the following design patterns in the construction of a software application: Composite, Decorator, Iterator, Strategy, Template method, and Observer
   - Be able to follow a systematic OO design methodology

2. 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
   - Implement correctly the equals, hashCode, clone, toString methods
   - Use serialization, reflection, and generics
   - Throw, propagate and catch exceptions

3. GUI Programming
   - Use a GUI toolkit to create a graphical user interface involving frames, buttons, text components, panels, menus, and simple geometric shapes

Required Texts/Readings

Textbook

C. Horstmann, Object-Oriented Design & Patterns, 3rd edition. Online copy will be made available on Canvas for personal use of students enrolled in this section of the course- you must not share your copy with others.

Other Readings

- S. J. Metsker, W. C. Wake, Design Patterns in Java, 2nd edition, Addison-Wesley
- M. Fowler, UML Distilled, 3rd Ed., Addison-Wesley.
- E. Gamma et al., Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley.

Other technology requirements / equipment / material

- Hardware: laptops in the classroom are required for “Lab” sessions.
- Software
  - Programming Language: Java Platform SE 8
  - IDE:
    - Eclipse at http://eclipse.org/
  - Version control (for group projects)
    - SVN at https://svnkit.com
      - SVN for Eclipse at https://www.eclipse.org/subversive/
    - Git at https://git-scm.com/
• Git for Eclipse at https://www.eclipse.org/egit/
  
  • UML design
  
  • Microsoft Visio
  
  o Microsoft Software for Students Owned Machines at http://its.sjsu.edu/services/software/microsoft-students/index.html
  
  o Microsoft’s Dreamspark program is currently available to all SJSU students allowing no-cost access to MS Visio – download instructions at https://sjsu.onthehub.com/WebStore/Welcome.aspx
  

Course Requirements and Assignments

• Readings with understanding new topic before each week are required. Lectures will summarize and clarify the reading content. Students are required to know the material in the listed chapters, even if not all of it is covered in the lecture.

• University Policy S16-9: “Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally three hours per unit per week) for instruction, preparation/studying, or course related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus.”

• There will be 6 lab sessions, deliverables due during or at the end of each session by showing the results to the instructor or uploading on Canvas. Some deliverables might be due later that day. Each “lab” will be graded up to 10 points. Two lab sessions will be reserved for the team project (start and end of the project). Each lab session requires you to bring your own laptop and work individually. You can install all the required software at the lab (notice will be given). The assignments might be partly required on paper and partly in code. Instructions for the lab will be delivered at the beginning of each session on Canvas or on paper. Problems to solve will be similar as Exercises at the end of Chapters required for the lectures preceding the lab. The lab exercises will be open book, but some parts might be closed book. The lab assignments will be done individually. No collaboration or sharing is allowed and can result in 0 points for the lab once discovered.

• Project (programming assignment) in teams of 3 will be partly done in 2 lab sessions: Lab 2 (start of the project) and Lab 6 (end of the project). Instructions for the project will be given no later than Lab 2.

• Extra points (up to 10 points) can be received by attending Oracle Code One (October 22-25, 2018 held within Oracle Open World) in San Francisco and writing a report (2-3 pages) about your discoveries from the event regarding Java. The event is free for students: https://www.oracle.com/code-one/register.html -> Students Registrations. The report will be due no later than a week after attending. You will be excused from attending the class on these days once you provide a proof of attendance.

Final Examination or Evaluation

Midterm and final exams will be closed book with test questions/ short answers and problems to solve.

Grading Information

Determination of Grades

• Total points for the course will be weighted by:
  
  o Midterm 20% 20 points
  
  o Final 20% 20 points
  
  o Programming Assignments 40% 40 points
• Letter grades will be assigned according to the following policy:
  100 -99-----A+
  93 – 98 ---- A
  89 -- 92 ---- A-
  87 -- 88---- B+
  83 -- 86 ---- B
  80 -- 82 ---- B-
  77 -- 79 ---- C+
  73 -- 76 ---- C
  70 -- 72 ---- C-
  67 -- 69 ---- D+
  63 -- 66 ---- D
  60 -- 62 ---- D-
  0 -- 59 ---- F
• Each assignment and exam will be scored (given points) but not assigned a letter grade.
• Final individual class letter grades might be assigned based on the class curve - your final class grade can be adjusted up or down depending on your level and quality of class / project performance.
• No late work accepted or make-ups.

Classroom Protocol

• Attendance: students should attend all meetings of their classes (University Attendance and Participation Policy F15-12).
• Arrival: students are expected to arrive on time.
• Behavior: eating, personal loud discussions, cell phones are not allowed in the classroom.
• Policy on Academic Integrity
  “Your commitment, as a student, to learning is evidenced by your enrollment at San Jose State University. The University Academic Integrity Policy F15-7 requires you to be honest in all your academic course work. Faculty members are required to report all infractions to the office of Student Conduct and Ethical Development. Visit the Student Conduct and Ethical Development website for more information.”
  o Any cheating on an exam will result in a grade of F in the class.
  o For the project, no collaboration outside own group is allowed
  o If duplicate programs/assignments are found, both the provider and the copier will receive 0 point on the assignment/project. A second offense results in a grade of F in the class.
  o Any incident of academic dishonesty can be reported to University for disciplinary action.

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 at http://www.sjsu.edu/gup/syllabusinfo/.
# Course Schedule

The schedule is subject to change with fair notice available to registered students through Canvas.

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Topics, Readings, Assignments, Deadlines</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8/22</td>
<td>Introduction to the course</td>
</tr>
<tr>
<td>2</td>
<td>8/27</td>
<td>A Crash Course in Java, Chapter 1</td>
</tr>
<tr>
<td>2</td>
<td>8/29</td>
<td>A Crash Course in Java, Chapter 1</td>
</tr>
<tr>
<td>3</td>
<td>9/3</td>
<td>Labor Day – Campus Closed</td>
</tr>
<tr>
<td>3</td>
<td>9/5</td>
<td>Lab 1</td>
</tr>
<tr>
<td>4</td>
<td>9/10</td>
<td>The Object-Oriented Design Process, Chapter 2</td>
</tr>
<tr>
<td>4</td>
<td>9/12</td>
<td>The Object-Oriented Design Process, Chapter 2</td>
</tr>
<tr>
<td>5</td>
<td>9/17</td>
<td>The Object-Oriented Design Process, Chapter 2</td>
</tr>
<tr>
<td>5</td>
<td>9/19</td>
<td>Guidelines for Class Design, Chapter 3</td>
</tr>
<tr>
<td>6</td>
<td>9/24</td>
<td>Lab 2 – start working on team-based projects</td>
</tr>
<tr>
<td>6</td>
<td>9/26</td>
<td>Guidelines for Class Design, Chapter 3</td>
</tr>
<tr>
<td>7</td>
<td>10/1</td>
<td>Interfaces and Polymorphism, Chapter 4</td>
</tr>
<tr>
<td>7</td>
<td>10/3</td>
<td>Interfaces and Polymorphism, Chapter 4</td>
</tr>
<tr>
<td>8</td>
<td>10/8</td>
<td>Patterns and GUI Programming, Chapter 5</td>
</tr>
<tr>
<td>8</td>
<td>10/10</td>
<td>Patterns and GUI Programming, Chapter 5</td>
</tr>
<tr>
<td>9</td>
<td>10/15</td>
<td><strong>Midterm</strong></td>
</tr>
<tr>
<td>9</td>
<td>10/17</td>
<td>Lab 3</td>
</tr>
<tr>
<td>10</td>
<td>10/22</td>
<td>Inheritance and Abstract Classes, Chapter 6</td>
</tr>
<tr>
<td>10</td>
<td>10/24</td>
<td>Inheritance and Abstract Classes, Chapter 6</td>
</tr>
<tr>
<td>11</td>
<td>10/29</td>
<td>The Java Object Model, Chapter 7</td>
</tr>
<tr>
<td>11</td>
<td>10/31</td>
<td>The Java Object Model, Chapter 7</td>
</tr>
<tr>
<td>12</td>
<td>11/5</td>
<td>The Java Object Model, Chapter 7</td>
</tr>
<tr>
<td>12</td>
<td>11/7</td>
<td>Lab 4</td>
</tr>
<tr>
<td>13</td>
<td>11/12</td>
<td>Veterans Day – Campus Closed</td>
</tr>
<tr>
<td>13</td>
<td>11/14</td>
<td>Concurrent Programming, Chapter 9</td>
</tr>
<tr>
<td>14</td>
<td>11/19</td>
<td>Frameworks, Chapter 8</td>
</tr>
<tr>
<td>Week</td>
<td>Date</td>
<td>Topics, Readings, Assignments, Deadlines</td>
</tr>
<tr>
<td>------</td>
<td>--------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>14</td>
<td>11/21</td>
<td>Frameworks, Chapter 8</td>
</tr>
<tr>
<td>15</td>
<td>11/26</td>
<td>More Design Patterns, Chapter 10</td>
</tr>
<tr>
<td>15</td>
<td>11/28</td>
<td>More Design Patterns, Chapter 10</td>
</tr>
<tr>
<td>16</td>
<td>12/3</td>
<td>Lab 5</td>
</tr>
<tr>
<td>16</td>
<td>12/5</td>
<td>Lab 6 – Team projects due</td>
</tr>
<tr>
<td>Final Exam</td>
<td>12/12</td>
<td>DH 450, 7:15AM-9:30AM</td>
</tr>
</tbody>
</table>