Overview
This course teaches students advanced skills in object-oriented design and programming through learning common design patterns and refactoring software source code. Students will work on projects based on real-world software to practice these skills.

Learning Outcomes
After taking this course, the students are expected to

- Understand and be able to apply incremental/iterative development
- Understand common design patterns
- Be able to identify appropriate patterns for design problems
- Be able to evaluate the quality software source code
- Be able to refactor badly designed program properly using patterns

Prerequisites
Prior knowledge of object-oriented programming is essential for this course. The students are expected to be proficient in Java. Because this is the first time that this course is being taught here, the prerequisite course is set to

- CS 3323 Principle of Programming Languages

A self-evaluation test in the first week of semester will be taken by the students to decide whether they should take this course.

Textbooks

Required


Recommended

Contents

- Review of object-oriented software development
- UML class, object and interaction diagrams
- Common design patterns
- Refactoring source code

Activities

The students need to perform the following learning activities in order to succeed in this course:

- Read textbook and other given reading materials before and after class
- Attend lectures and participate in class discussions and exercises
- Master one integrated development environment (IDE) such as Eclipse
- Finish assignments in timely manner, including
  - Identify the correct pattern in given design problems
  - Read and understand source code from selected software
  - Recognize the patterns from the model software
  - Criticize badly designed software
  - Refactor source code of the criticized software
- Take quizzes (online or in-class) and final exam

Assessment

The final grade is a combination of the following weighted components

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance</td>
<td>10%</td>
</tr>
<tr>
<td>Quizzes</td>
<td>20%</td>
</tr>
<tr>
<td>Projects</td>
<td>40%</td>
</tr>
<tr>
<td>Final exam</td>
<td>30%</td>
</tr>
</tbody>
</table>

The grading scale is $A = [90\%, \infty)$, $B = [75\%, 90\%)$, $C = [60\%, 75\%)$, $D = [50\%, 60\%)$, $F = [0\%, 50\%)$. The instructor reserves the right to justify the weight and grading scale.