

**SABANCI UNIVERSITY**  
**CS 442 / CS 542 - Software Design Patterns**  
**Spring 2022-2023**  
**Updated Syllabus<sup>1</sup>**

---

## **Instructors**

Anil Koyuncu

- FENS G013
- E-mail: anil.koyuncu
- Lectures
  - Tuesday 11:40-12:30
  - Wednesday 14:40-16:30
- Office Hours
  - Wednesday 16:30-17:30
- <https://sabanciuniv.zoom.us/j/99542656193>

## **CONTENT**

This course introduces the use of software design patterns. Topics include design patterns (creational, structural, and behavioral patterns), architectural patterns, anti-patterns, object-oriented design principles, and programming idioms will be discussed.

This course will use term-long group projects to give students real-life practical experience with patterns by building software systems.

## **OBJECTIVE**

To introduce the principles of object-oriented (OO) analysis and design, as well as software architecture, through OO principles and design/architectural patterns.

To practice the application of object-oriented software development principles through a team project.

To develop teamwork and communication skills through a team project.

## **LEARNING OUTCOMES**

Describe the basics of object-oriented design principles and OOP mechanism

Define and recognize design patterns

Identify suitable design patterns for design problems

Compare the design patterns and explain which design pattern should be used

Design object-oriented software systems that meet requirements by applying object-oriented design principles and design/architectural patterns

## **Course outline**

OOP mechanism

Object-oriented design principles

Creational design patterns

Behavioral design patterns

Structural design patterns

Architectural design patterns

## **Textbook (Optional)**

The textbook is just for reference, all the necessary materials will be provided with lecture notes.

1- Gamma, Helm, Johnson, and Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994

2- Head First Design Patterns, 2nd Edition by Eric Freeman, Elisabeth Robson

(<https://www.oreilly.com/library/view/head-first-design/9781492077992/>)

## **Tentative Grading (subject to change)**

Final Exam	50%
Course Project*	50%
Quiz & Homework & Participation**	5%

\*Receiving a project score of less than 50/100 automatically results in a failing grade.

\*Receiving a score of less than 50/100 on the exam automatically results in a failing grade.

\*\*Bonus extra grade

## **Course project**

The course project will provide hands-on software engineering experience involving 3 students.

You will design and develop a framework to integrate automated software engineering (ASE) tools. These ASE tools “range across AI toolkits, theorem provers, and model checkers; requirements, design, coding, and testing support tools; various configuration management, process enactment, and project management support tools; and code generators, code analysis, visualisation, refactoring and reverse engineering tools”<sup>2</sup>.

Each team first must select a software engineering task to automate. Some of the tasks are listed below:

- code clones
- code summarization
- commit message generation
- bug localization
- bug classification
- code classification
- program repair
- code autocompletion
- code translation
- code search
- code/API recommendation

After selecting the task, you will first perform a literature survey; look at papers in software engineering conferences ICSE, FSE, ESEC/FSE, ASE

---

<sup>2</sup> Grundy, John, and John Hosking. "Guest editors introduction: special issue on innovative automated software engineering tools." *Automated Software Engineering* 20 (2013): 137-139

and ISSTA and software engineering journals IEEE TSE and ACM TOSEM for state-of-the-art tools (ASE tools) for your task.

In the project, team members select an ASE tool using the outcome of the survey. You need to install the tool and find a software system or a set of programs that the tool can be used for. Eventually, as a group you will design and develop a framework that integrates all the ASE tools for the selected task and conduct a case study or an experimental evaluation.

The teaching team serves as both customer and manager. You must submit a project proposal and present it to show your **plan** and convince the teaching team.

Just like in real software development projects, you will have milestones, and each week you will submit reports and attend meetings, in addition to developing your product:

- I. hold weekly team meetings (before weekly status report submission),
- II. submit a weekly status report (every Tuesday at 11:40),
- III. hold weekly project meetings (TBD).

In terms of programming languages and technologies to use you are free to select any, however, your project must follow certain constraints that will be announced in the lectures.

**Constraints:**

- 3-person teams
- SCRUM-like methodology
- Issue & Project Tracking
- Version Control
- Feature branches
- Pull requests
- Project milestones

## Course Grading Catalog

A	100-95
A-	94-90
B+	89-85
B	84-80
B-	79-75
C+	74-70
C	69-65
C-	64-60
D+	59-55
D	54-50
F	<50

## Exam Policies

Unless otherwise noted, exams are in the classroom, closed books, and closed notes; no computing or communication devices are allowed during the exams.

In the case of online exams :

- During the exam, the TAs will be proctoring online over Zoom. You'll be assigned to the Zoom session of one of the TAs.
- Each student will also be assigned a personal Google Meet link. When the exam begins, the students are supposed to click the provided link and press 'Record Screen'. The TA will join this meeting arbitrarily in order to check whether the recording has started appropriately and the entire screen is shared.
- Note that during the exam, each student must be connected to both Zoom and Google Meet links. Once connected, Zoom will be used to share your computer's camera and microphone, and GoogleMeet for screen recording.
- Make sure to check that your setup (i.e., webcam, microphone, etc.) functions correctly before the exam.

In the case of non-compliance with this and other declared exam procedures, your exam will be void.

## Expectations from Students

- Students are expected to attend all classes. You must attend the Zoom lectures, recitations, and real-time online exams with your SU email account.
- Students are responsible for the material covered in class even if it is not part of the lecture notes published on the website (Sometimes we explain some stuff on the board). That is why attendance is important.
- Students are responsible to check their emails (Sabanci University accounts), SUCourse, and course website daily for any announcements related to this course.
- **PLAGIARISM WILL NOT BE TOLERATED.**
  - Any act of plagiarism may result in a direct failure (F) of the course.

## Make-up Policy

No make-up is allowed for **quizzes, homework, or project milestones**. Students automatically get 0 (zero) from the respective grade item if any of them is missed.

Make-up is only allowed for the midterm and final examination to those with an official report from the University Health Center and to those with an official permission notice from the university for participating in a university event on the date of the exam in question. Students must submit their reports/notices to one of the instructors **before the exam in question**. The ones having other excuses should contact the instructors within the day of the exam to be missed and then the instructors will decide whether these students are allowed to take the make-up exam. Any excuses to be brought to the attention of the instructors after the exam will **not** be considered. **No exceptions to these rules!**