



# CS251: Intro to Software Engineering

## Homework 2 (15)

Due Dates 26 April, 2 May and 10 May 2024 @ 11:59 pm

### Objectives

By doing this assignment, you will:

- 1- Learn about software design and how to move from requirements (what) to design (how) to implementation (real software).
- 2- Learn how to develop architecture, class, state and sequence diagrams to describe a system.
- 3- Learn how to enhance designs / code with SOLID principles and design patterns.
- 4- Learn how to transform design models to code.
- 5- Learn how to document code by Java doc and to use GitHub as a repository for the project.

### Setup

- 1- This assignment will be solved in **groups of 3 students from the same lab**. They do not have to be the same group who worked on Homework 1. تخلص من الأندال و ابحث عن الرجال ، Teams of less than 3 people are still required to do the **whole assignment**.
- 2- Each team will design and implement part of a system for “Investment Management” or “Budget Management”
- 3- If all team members were doing “Investment” or “Budget”, they will continue doing the same project with the **new SRS posted**.
- 4- If some were doing this and others were doing the that, **they will do the one that the majority of team were doing**. (2 members were budget, 1 was investment => team does budget)
- 5- This is a big assignment that will require each team to work together closely and independently and will need self-learning and self-discipline. هذه المسألة هي مسك الختام للمقرر وتطلب قدرا كبيرا من التعلم الذاتي والبحث وتنظيم الوقت والتعاون الجيد مع أعضاء الفريق.
- 6- The group will submit together **one combined** solution.
- 7- The entire group is responsible of helping any weak member to be able to do his/her task **by his/herself**, by providing the necessary support, knowledge, hands-on demos, etc.
- 8- **Only submit original work. Any copied (or AI-generated) work will be severely penalized.**
- 9- مسؤولية الفريق تضامنية عن عمله و أى غش من أى فرد سيكون مسؤولية الجميع و يخصم منهم مثل الدرجة
- 10- **Please read the marking criterion very carefully to understand how you will be marked.**

### Task1: System Design (7)

- 1- **Read carefully the attached SRS** given for your project. **If anything is unclear, ask the authors of the SRS sample by email**. These are not perfect but they give very good description of how the program should behave and look like and the user tasks performed through the program. **(We got 20 excellent SRS. I had to select only 2 🍀).**
- 2- **Read carefully the Software Design Specifications template** given with this assignment. **Read the instructions in red very carefully**. Again, read red instructions to know how to fill the form.
- 3- **Write the SDS document of your project**, including all the required details. **Use the template attached with this homework.**

# CS251: Intro to Software Engineering

## Homework 2 (15)



Due Dates 26 April, 2 May and 10 May 2024 @ 11:59 pm

### Task 1.1: System Architecture

- 1- **Identify** all the main components or subsystems of the system and describe their role.
- 2- **Decide** the suitable architectural design for your application.
- 3- **Draw** an architectural diagram showing the different parts of the system using a suitable notation like C4 or arrow and box.

### Task 2.1: Class Diagram

- 1- **Identify** all the important classes in the system in each component.
- 2- **Divide** these classes into packages or subsystems, each one includes the related classes.
- 3- **Decide** the responsibility of each class and what role it plays.
- 4- **Design the attributes and operations** of each class. For each attribute decide the type and visibility. For operations, decide the name, parameters and return type.
- 5- **Design the relations** between classes (inheritance, association, aggregation, composition). For each relation, decide the direction and multiplicity and give it a label.
- 6- **Read again**, the Software Design Specifications template given with this assignment.
- 7- **Finally, draw a complete class diagram** for the system, divided into packages or systems. **Use a tool for drawing.**

### Task 1.3: Sequence Diagrams

Sequence diagrams help developers understand how a system works and how a use case is implemented using the classes and methods in the class diagram.

- 1- **Select** the most important **six** user stories from your SRS or the sample SRS included
- 2- **Implement** a detailed sequence diagram for each one of them.
- 3- **Read again**, the Software Design Specifications template given with this assignment.

### Task 1.4: State Diagrams

State diagrams help developers understand the different possible states for an important object and the events that cause it to move it from a state to another.

For the **one important** object, draw a state diagram to show the developer the different states it can be in.

### Task 1.5: SOLID and Design Patterns

Demonstrate how your design applies 3 of SOLID principles and uses 3 Design Patterns.

## Task2: Implementation (5)

Implement **6 user stories in yellow background in SRS** of the system in Java / C# or Kotlin. Implement as a desktop / mobile app. The purpose is to learn transferring model to code, good quality code. **Your code must work. Connect with database or files if needed.** Your task will be:

- 1- **Read** the Corona Store example in the course page.

# CS251: Intro to Software Engineering

## Homework 2 (15)



Due Dates 26 April, 2 May and 10 May 2024 @ 11:59 pm

- 2- **Implement** the parts of your class diagram related to the functions shown with **yellow background in the SRS** document given.
- 3- **Implement sequence diagrams** related to these functions.
- 4- **Create a private repo on GitHub and use it to develop the code by doing multiple commits. Add snapshots of your GitHub Repo for every team member in a pdf file with the code.**
- 5- **Zip your code** with the final SDS as in the instructions below.

### Task 3: Documentation and Clean Code (1)

- 1- **Document** all classes and functions by JavaDoc (or similar tools) & **generate documentation**
- 2- **Review code quality and readability & apply** Java Coding Style manual attached with A2.

### Task 4: Presentation (1)

- 1- Develop a **10 minutes presentation** (10 - 15 slides) to present your work to the TA.
- 2- Include a **pdf or ppt of the slides in your submission.**

### Task 5: Individual Task - TechRadar (1)

- 1- Each team member individually goes to TechRadar [www.thoughtworks.com/radar](http://www.thoughtworks.com/radar)
- 2- Review the tools, languages and technologies mentioned
- 3- Choose any technology or tool you like - each team members chooses a different one.
- 4- Try this technology and write a 1-page review (font size 12, single-spaced) about it.

### Task 6: More Implementation (...)

- 1- Implement more functionality and **negotiate with TA a bonus.**

### Deliver and Assessment

- 4- Upload the **draft version by 26 April 2023 as a pdf file.** Draft version should include the **architecture diagram, the class diagram and class responsibility table and 3 sequence diagrams.** File name must be *CS251-2025-SectionNumber-TAFirstName-Topic-IDs-DraftSDS.pdf* for example *CS251-2025-S1-Ahmed-20230001-20230002-20230003-Budget-DraftSDS.pdf*
- 5- **On 2 May 2025 load the final SDS version. Only one member of the group** should upload the final version named *CS251-2025-SectionNumber-TAFirstName-Topic-IDs-SDS.pdf*. Report should include all items needed in the Software Design Specification document and a **cover page with team names, IDs, emails, course name and number and document topic and title.**
- 6- **On 10 May 2025 load the final submission including SDS final version + implementation and documentaiton + presentation + pdf of technology review (task 5). You will upload 1 zip file with:**
  - A pdf file containing your design (SDS) document.
  - A directory with your implementation and documentation. Add a Readme.txt file explaining the files included and the tools used to develop the program. (no exe or jar)

# CS251: Intro to Software Engineering

## Homework 2 (15)



Due Dates 26 April, 2 May and 10 May 2024 @ 11:59 pm

- A pdf with tech radar review + pdf or ppt of your presentation + screenshots of GitHub
- All pdf files should have a cover page with your team names, **emails**, **phone of leader**, document name, FCAI logo, course name, etc.



# CS251: Intro to Software Engineering

## Homework 2 (15)

Due Dates 26 April, 2 May and 10 May 2024 @ 11:59 pm

### Marking criteria

#### Software Design Specifications (7)

1 mark	For excellent well-designed and explained architecture diagram.
2.0 marks	For an excellent class diagram and class description that correctly captures most of the classes, their details and their relations
2 marks	For excellent 6 sequence diagrams and class-sequence table.
0.5 mark	For an excellent state diagram for order states
1.5 mark	For an excellent use of SOLID and DPs and explaining that
-7.0	For copied or non-original report or giving yours to another group.
-1.0	Disorganized - badly written report.

#### Implementation (5)

4.5 marks	For excellent working code that does the required functions.
0.5 mark	For organizing the code in a nice way.
-5	For copied or non-original code or giving yours to another group.
-1	Failing to use GitHub

#### Documentation and Clean Code (1)

1 mark	JavaDoc pages created for all classes and methods with useful info Java Coding Style is accurately followed and code is readable.
--------	--------------------------------------------------------------------------------------------------------------------------------------

#### Presentation (1)

1 mark	For providing power point / pdf presentation of your work
--------	-----------------------------------------------------------

#### Tech Radar Review (1)

1 mark	Good review for each student of one technology in Tech Radar
--------	--------------------------------------------------------------

### Policy Regarding Plagiarism:

1. تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهرها لعملية تعليمية سليمة
2. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم في الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
3. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
4. قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
5. استعمال أدوات الذكاء الاصطناعي لعمل الحل في هذه المسألة يعتبر غشا (تدرب على هذه الأدوات لمستقبلك لكن ليس في هذه المسألة)
6. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
7. في حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و في حالة تكرار الغش سيرسب الطالب في المقرر.