Software Engineering 2 Project Description:

Implementing **Software Design Patterns** in **Software Architecture**

Project due: March 20th at 9 pm

Objective:

The main goal of this project is to give students hands on experience in implementing design patterns in software architecture. The students will be tasked with developing a software system creating Unified Modeling Language (UML) diagrams which makes use of design patterns in modeling, and writing the code for the system by incorporating design patterns to improve its modularity and scalability.

Team Formation and Registration:

- Students are required to form teams of three.
- Teams must be formed by the 23rd of January.
- Once teams are formed, each team must register by entering their names into the provided Excel sheet on the cloud. https://shorturl.at/zDFQ5

Project Requirements:

1**. System Development**

- Create a software system. You can choose a real world problem or simulate a process.
- Ensure that the architecture of the system is sufficiently intricate to accommodate design patterns.
- Analyze the requirements and architecture of your system. Identify parts of your system where a design pattern can simplify the design, improve flexibility, or enhance maintainability.

2. **UML Modeling**

- Create UML diagrams for the system, including class diagrams, sequence diagrams and any other relevant UML representations.
- Make sure that the UML models accurately portray the structure and behavior of the system to effectively reflect the design choices, including the use of design patterns, in the system's architecture.

3. **Code Implementation**

- Implement the system using a programming language of your preference.
- The code should be documented organized and follow coding conventions.

4. **Design Patterns Integration:**

- Explicitly incorporate design patterns, into your software architecture including system model and code
- Provide descriptions of each design pattern used in your system.

5. **Documentation and format of the report**

It is important to compile the following information into a single comprehensive **report**.

-Start the report with a description of the system, including its purpose, functionalities and important features.

-Make sure to provide a list of all the design patterns used in the project. Feel free to utilize the design patterns we've discussed during class sessions or any other patterns that haven't been addressed yet. This approach enables you to delve into a spectrum of patterns and choose the ones that align effectively with your projects unique requirements and challenges.

-Explain why each design pattern was chosen and how it fits within the context of your system.

-Establish connections, between parts of your code and the components of the design patterns. This can be done through comments in your code and additional explanations in your documentation.

- Discuss how using design patterns has improved reusability, scalability and maintainability in your system. Support this with examples from your implementation.

6. **Presentation**

Prepare a presentation that provides an overview of your work highlighting aspects such, as integrating design patterns challenges faced and what you have learned, and your system model and code.

Submission Guidelines

- Please submit the source code, UML models along with a comprehensive report and the presentation slides.
- Make sure all files are well organized and named for navigation.
- Include a README file that provides instructions on running the system and navigating through the documentation and report.

This project is an exercise well as a conceptual exploration. It involves not building a system but also understanding and expressing the significance of design patterns, in software architecture. Your critical analysis skills and ability to justify your choices are aspects of this project.