Design Patterns

Table of contents

1 Design Patterns

This lab focuses on improving the Quackstagram codebase through the identification and implementation of a suitable design pattern. Your task is to analyze the newly integrated features and the overall system to determine how a design pattern can enhance the application's structure, maintainability, or functionality.

Selecting and implementing the right pattern is important in preparing the application for future expansions and ensuring a robust architecture.

Lab Duration: Approximately 2 hours

Learning Objectives:

- Recognize the importance of design patterns in software development for solving common problems and improving code quality.
- Identify a suitable design pattern that addresses a specific need within the Quackstagram application.
- Implement the chosen design pattern, integrating it thoughtfully into the existing codebase.
- Assess the impact of the design pattern on the application's architecture and future development.

Preparation:

1. Review various design patterns and understand their purposes, benefits, and typical use cases from our lectures.

2. Analyze the current state of the Quackstagram application, focusing on the new features added and areas where a design pattern might be beneficial.

Lab Tasks:

1. **Pattern Identification**:
    - Evaluate different areas of the Quackstagram codebase to identify where a design pattern might solve existing issues, improve design, or enhance scalability.
    - Select a design pattern that is well-suited to the application's current needs and future direction. Consider patterns like Singleton, Observer, Strategy, Factory, and Decorator.
2. **Design and Planning**:
    - Plan how to integrate the chosen design pattern into the Quackstagram codebase. Outline any necessary changes to the classes, interfaces, or data flow.
    - Ensure that the integration plan maintains the application's functionality and aligns with overall project objectives.
3. **Pattern Implementation**:
    - Implement the chosen design pattern. Write clean, well-documented code to ensure that the pattern integrates seamlessly with the existing application.
    - Continuously test the application during implementation to ensure that the new design doesn't introduce any bugs or issues.
4. **Review and Refinement**:
    - Review the implementation with your team or instructors. Discuss how the design pattern has impacted the codebase and whether it has achieved the desired improvements.
    - Make any necessary refinements to optimize the integration of the design pattern.

End of Lab:

- Finalize all changes, ensuring that the codebase is stable, well-documented, and fully tested.

Preparation for Next Steps:

- Reflect on any remaining features or improvements that could be added to the Quackstagram application.
- Plan for a final review and testing phase to ensure the application is ready for submission.