**Київський національний університет імені тараса Шевченка**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Факультет комп’ютерних наук та технологій

Кафедра інформаційних систем

Лабораторна робота № 1c

Peer Review (Architecture/Design)

**Підготував:**

студент групи ПІ-21

Пшоновський Є.І.

Київ – 2023

**Peer Review**

1. **Наскільки моделі є зрозумілими, наскільки вони описують предметну область, структуру та поведінку відповідної системи?**

Моделі є зрозумілими. Кожна діаграма відповідає предметній області в проєкті. Структура є чіткою та послідовною. Поведінка програмного продукту відображена повною мірою та легко сприймається. Використано професійне програмне запезпечення, що посприяло якісному створенню діаграм.

1. **Чи є якісь аспекти, які видаються важливими, але не відображені в моделі (на діаграмах)?**

На мій погляд, усі основні аспекти програмного продукту були висвітленими. Кожна діаграма детально описує всі важливі процеси та дає яскраве розуміння їх роботи.

1. **Чи є в моделі щось зайве, якісь аспекти описані занадто детально?**

На мою думку, зайві процеси в моделі не спостерігаються. Можливо, класова діаграма є трішки переобтяженою. Доцільно було б розбити її на декілька частин. Але тоді втратилась би повнота картини. Тому цей момент не є недоліком.

1. **Наскільки доцільно використані різні типи діаграм?**

Усі типи діаграм використано вдало: кожна з них описує чітко визначені компоненти програми, а також дає розуміння, навіщо використовувати даний вид діаграм.

1. **Наскільки коректно використана нотація UML, різні елементи та конектори?**

Оскільки було використано професійне програмне забезпечення, яке мало заготовлені шаблони діаграм, нотація UML є коректною. Дотримано правил застосування різних типів конекторів, а також кожен елемент відповідає певному типу діаграм.

1. **Наскільки вдалим є глосарій? Чи всі важливі поняття предметної області описано? Чи немає неоднозначностей?**

Глосарій є чітким та послідовним. Всі елементи моделей описані. Найважливіші компоненти виділені окремо для легшого сприйняття. Глосарій розділено на блоки, що забезпечує відсутність неоднозначностей.

1. **Чи всі важливі сценарії використання описано в моделі? Наскільки зрозумілі різні сценарії, зв’язки між ними?**

Всі діаграми поділено на окремі блоки. Кожен компонент відповідає за конкретний сценарій. Чіткий програмний інтерфейс забезпечив зрозумілий зв’язок між сценаріями використання. Наявний детальний опис найважливіших компонентів.

1. **Наскільки доцільним є поділ системи на частини/компоненти/модулі/…?**

Поділ системи на компоненти є чітким та послідовним. Кожна частина відповідає за певний функціонал, що дає змогу швидко зрозуміти всі особливості програми та полегшити пошук недоліків.

1. **Наскільки доцільними є зв’язки між компонентами/класами/об’єктами? Чи немає занадто тісно зв’язаних компонентів??**

Кожен компонент програми є автономним. Деякі з них є виокремленими в окремі бібліотеки, що забезпечує простий рефакторинг та оптимізацію якості коду. Існує чітке відокремлення інтерфейсу від логіки програми.

1. **Наскільки object-oriented design відповідає загальним принципам?**

* Застосовано принцип інкапсуляції.
* Наявне доцільне застосування наслідування.
* Використано поліморфізм.
* Код розбитий на багато автономних частин, що відповідає принципам ООП.