

## Лабораторна робота №6 (максимально - 30)

### Тема: Фінальний проєкт

**Мета роботи:** навчитися слідувати принципам чистого коду та реалізовувати потрібні патерни програмування в програмному коді

### Завдання на лабораторну роботу

#### Завдання 0: Підготовка до виконання завдання

1. Створіть окремий ПУБЛІЧНИЙ репозиторій на GitHub для проєкту.
2. Репозиторій має бути публічним, для того, щоб будь-хто з Ваших колег міг переглянути Ваш проєкт і захопитися красою і чистотою Вашого коду.
3. Перейти до Завдання 1 😊

#### Завдання 1: Оберіть тему проєкту

1. Теми проєктів можете знайти [тутечки](#).
2. Продумайте функціональність Вашого проєкту. Додайте опис функціональності в **README.md** з посиланням на файли, де це доречно. Приклад опису функціональності можете знайти [отут](#).  
Головні вимоги до функціональності:
  - а. середній розмір проєкту (**>=2000** рядків коду в файлах з **програмним кодом**; конфіги, розмітка і стилі сюди не входять). Кількість рядків коду можете знайти за допомогою наступної команди `git ls-files '*.cs' '*.js' | xargs wc -l`, де `'*.cs' '*.js'`, це blob файлів, в яких було написано код;
  - б. наявна UI частина;
  - с. наявне збереження даних у файлі або БД.
3. Перед тим як обрати тему перевірте, чи в Вашій групі вже не обрали таку тему двічі. Однакова тема не може повторюватися в одній і тій же групі більше ДВОХ разів.
4. Після того, як Ви обрали свою тему, обов'язково запишіть її в [журналі](#). Для цього знайдіть лист зі своєю групою і в своєму рядку в останній колонці "Проєкт для ЛР №6" залиште КОМЕНТАР з назвою теми і посиланням на ПУБЛІЧНИЙ репозиторій, де буде

виконуватися проєкт. Викладач потім перенесе Ваш коментар у клітинку.

5. **ЗВЕРНІТЬ УВАГУ:** в кожній з груп можливі максимум **ДВА** проєкти на однакову тему.

### Завдання 2: Опишіть функціонал і красу коду

1. Обов'язково додайте README.md з описом функціоналу програми та процесом запуску локально. Синтаксис для .md файлів можна знайти [тут](#) і [тут](#).
2. Ви можете додавати функціонал як на початку, так і вкінці створення проєкту. Але для себе бажано все продумати наперед. Планування – найважливіший етап написання коду.
3. Додайте в README.md окремі секції для опису дотриманих принципів програмування, патернів проєктування та технік рефакторингу: “Programming Principles”, “Refactoring Techniques”, “Design Patterns”.
4. “Programming Principles” – тут опис буде схожим на опис з ЛР №1. Очікується мінімум **5 принципів**.
5. “Design Patterns” – очікується, що у Вашому коді буде застосовано патерни проєктування (**мінімум 3**). Додайте в README.md посилання на відповідні файли і пояснення, для чого було використано той чи інший паттерн.
6. “Refactoring Techniques” – тут Ви наводите [список технік](#), які Ви використали під час написання програми (**мінімум 5**).

### Завдання 3: Напишіть класну програму

1. Зважаючи на специфіку нашого предмету, в Вашому проєкті в першу чергу будуть оцінюватися чистота і структурованість коду, відсутність code smells, дотримання поширених принципів проєктування, використані патерни проєктування.
2. Також важливо створити зрозумілу структуру файлів та директорій проєкту. Намагайтеся слідувати правилу: одна сутність - один файл.
3. Зверніть увагу, що викладач також буде оцінювати Вашу вмілість користування Git. Від Вас очікується, що Ви будете розробляти

окремі фічі проєкту в окремих feature/\* гілках. Гілки повинні мерджитися через пулл-ріквести. Ваші коміти повинні бути максимально дрібними та зі зрозумілим описом.

### Процес здачі

Після того, як Ви завершили роботу над власним проєктом, Ви маєте засабмітити посилання на нього, як зазвичай черз портал, **а також додати його до [журналу](#) “Рейтинг”**. Посилання додаєте у вигляді коментаря до Вашої клітинки в останній колонці “Проект для ЛР №6”. Ми це робимо для того, щоб Ваші ревьюери не чекали поки викладач долізе до перевірки лабораторних і перенесе посилання в журнал.