



Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

## **Практична робота №4**

### **Економіка ІТ-індустрії**

**Тема:** Додаткові види точок для оцінки розміру програмного забезпечення

Виконав

студент групи ІП-11:

Панченко С. В.

Перевірив:

Родіонов П. В.

Київ 2024

## ЗМІСТ

1 Мета практичної роботи.....	6
2 Завдання.....	7
3 Виконання.....	8
3.1 Feature points.....	8
3.2 Object points.....	9
3.3 Use-case points.....	10
Висновок.....	11

## 1 МЕТА ПРАКТИЧНОЇ РОБОТИ

Навчитися оцінювати розмір програмного забезпечення шляхом застосування додаткових видів точок, а саме Feature points, Object points та Use-case points.

## 2 ЗАВДАННЯ

1. Використовуючи власні або запропоновані викладачем програмні застосунки обчислити наступне:

- кількість Feature points;
- кількість Object Points;
- кількість Use-case points за першим і другим методами.

2. Визначити економічний вплив отриманих результатів на реалізацію програмного застосунку.

## 3 ВИКОНАННЯ

### 3.1 Feature points

Для розрахунку feature points (FP) використано кілька параметрів із раніше проведеного аналізу. Зовнішні входи (EI) були визначені як три форми (логін, управління працівниками та відділами), з вагою 4 для середньої складності, що дає 12 FP. Зовнішні виходи (EO) включають таблиці працівників і відділів, оцінені у 5 балів, що дає 10 FP. Внутрішні логічні файли (ILF), як користувачі, працівники та відділи, мають вагу 7 для середньої складності, з сумарним результатом у 21 FP. Зовнішні запити (EQ) стосуються пошуку працівників і відділів, із вагою 4 для середньої складності, що додає 8 FP. Таким чином, загальна кількість feature points для проєкту становить 51 FP.

Для цього проєкту можна визначити основні характеристики системи наступним чином:

1. Обмін даними: 0 – продукт працює автономно, немає зовнішнього обміну даними.
2. Розподілена обробка даних: 0 – продукт не потребує розподіленої обробки.
3. Продуктивність: 0 – немає спеціальних вимог щодо продуктивності.
4. Обмеження по апаратних ресурсах: 0 – немає обмежень.
5. Транзакційна навантаженість: 1 – є кілька транзакцій при роботі з базою даних.
6. Інтенсивність взаємодії з користувачем: 3 – середній рівень взаємодії (логін, управління даними).
7. Ергономіка: 0 – немає спеціальних вимог.
8. Інтенсивність змін даних: 1 – редагування даних є.
9. Складність обробки даних: 1 – обробка незначна, але є операції з базою.
10. Повторне використання: 0 – не використовується повторно.
11. Зручність інсталяції: 0 – немає особливих потреб.
12. Зручність адміністрування: 0 – не вимагається.

13.Можливість портування: 2 – можлива установка на різні платформи.

14.Гнучкість: 0 – не вимагається.

$$TDI = 1 + 3 + 1 + 1 + 2 = 8$$

$$VAF = (TDI * 0.01) + 0.65 = (8 * 0.01) + 0.65 = 0.73$$

Нормована кількість функціональних точок:

$$AFPC = UFC * VAF = 51 * 0.73 = 37.23$$

### 3.2 Object points

Щоб оцінити проєкт за методологією Object Points (OP), підрахуємо кількість екранів, звітів і модулів.

#### 3.2.1 Оцінка об'єктних одиниць:

1. Екрани: У проєкті є форми логіну, додавання/редагування працівників і відділів, що дає приблизно 3 екрани. Кожен екран є середньої складності (3 таблиці даних).

- $3 \text{ екрани} \times 3 = 9 \text{ OP}$

2. Звіти: У проєкті генерується таблиця з працівниками та відділами, що відповідає 2 звітам, кожен середньої складності.

- $2 \text{ звіти} \times 5 = 10 \text{ OP}$

3. Модулі 3GL: Проєкт використовує Flask для веб-додатку, але немає складних обчислювальних або графічних компонентів.

- Модулі 3GL не вимагаються.

Підрахунок OP:

$$\text{Загальна кількість OP} = 9 \text{ (екрани)} + 10 \text{ (звіти)} = 19 \text{ OP.}$$

#### 3.2.2 Оцінка нових об'єктних одиниць:

Якщо у проєкті використовується 20% повторного коду, то кількість нових об'єктних одиниць буде:

$$OP_{\text{new}} = OP \times (100 - 20) / 100 = 19 \times 0.8 = 15.2$$

### 3.2.3 Оцінка трудовитрат:

Оцінимо продуктивність розробки як номінальну (13 ОР на людину-місяць):

$$E = OP_{new} / PROD = 15.2 / 13 = 1.17 \text{ людино-місяців.}$$

Таким чином, трудовитрати для цього проекту оцінюються приблизно в 1.17 людино-місяців.

### 3.3 Use-case points

#### 1. Актори (UAW):

У проєкті є кінцевий користувач, який взаємодіє з веб-інтерфейсом, що класифікується як складний актор. Складний актор = 1 актор  $\times$  3 = 3 UAW.

#### 2. Use-case (UUCW):

У проєкті можна виділити три Use-case: логін, управління працівниками, управління відділами. Кожен Use-case має кілька транзакцій, що робить їх середньої складності. Середній Use-case = 3 Use-case  $\times$  10 = 30 UUCW.

#### 3. Підрахунок Unadjusted Use-case points (UUCP):

$$UUCP = UAW + UUCW = 3 + 30 = 33$$

#### 4. Adjustment Factors (VAF):

Припустимо, що фактори технічної складності (TCF) та фактори середовища (EF) мають середні значення, що дає  $VAF = 1.0$ .

#### 5. Adjusted Use-case points (UCP):

$$UCP = UUCP \times VAF = 33 \times 1.0 = 33$$

Таким чином, оцінка Use-case points для проєкту дорівнює 33 UCP.

## ВИСНОВОК

Використання трьох методів оцінки (Feature Points, Object Points та Use Case Points) дозволяє більш точно оцінити час та ресурси, необхідні для розробки. Для проєкту було визначено 51 Feature Points, 15.2 нових Object Points та 33 Use Case Points. Згідно з цими даними, приблизна трудомісткість проєкту становить 1.17 людино-місяців. Ці оцінки дають змогу краще планувати бюджет, розподіл робочої сили та час для реалізації, що дозволяє зменшити ризики перевищення бюджету та термінів.