**Лабораторная №7**

**Экономические аспекты разработки ПО**

Задания:

**1. Провести оценку размера ПО для вашего проекта в соответствии с методом функциональных точек.**

Анализ функциональных точек — стандартный метод измерения размера программного продукта с точки зрения пользователей системы.

Метод предназначен для оценки на основе логической модели объема программного продукта количеством функционала, востребованного заказчиком и поставляемого разработчиком. Несомненным достоинством метода является то, что измерения не зависят от технологической платформы, на которой будет разрабатываться продукт, и он обеспечивает единообразный подход к оценке всех проектов в компании.

1. Определение типа оценки.

Тип выполняемой оценки нашего проекта - Проект разработки. В этом типе оценивается количество функциональности поставляемой пользователям в первом релизе продукта.

2. Определение области оценки и границ продукта.

Область оценки - Все разрабатываемые функции.  
Границы: все данные поддерживаются системой, программа обрабатывает события ввода с клавиатуры пользователем.

3. Подсчет функциональных точек, связанных с данными.

Провести оценку количества функциональных точек, связанных с данными, в нашем проекте невозможно, поскольку программа не содержит ни информации о пользователей, ни каких-либо других данных, которые могут быть оценены.

4. Подсчет функциональных точек, связанных с транзакциями.

Типы транзакций, присутствующие в нашем проекте:

* EI (external inputs) — внешние входные транзакции, элементарная операция по обработке данных или управляющей информации, поступающих в систему извне.
* EO (external outputs) — внешние выходные транзакции, элементарная операция по генерации данных или управляющей информации, которые выходят за пределы системы. Предполагает определенную логику обработки или вычислений информации из одного или более ILF.

Оценку сложности транзакции можем провести на основе только одной ее характеристики - DET (data element type). А FTR (file type referenced) примем за 0.

1. Транзакция передвижения:

EI: кнопки управления - 4 DET

Сложность: Low

Оценка транзакций в не выровненных функциональных точках (UFP): 4

1. Транзакция готовки:

EI : кнопки управления - 4 DET.

EO: сообщения об ошибке - 3 DET.

Сложность для EI: 0 FTR + 4 DET = Low

Сложность для EO: 0 FTR + 3 DET = Low

Оценка транзакций в не выровненных функциональных точках (UFP): 7

1. Транзакция выдачи заказа:

EI : кнопки управления - 1 DET.

EO: сообщения об ошибке - 1 DET, сообщение о начисление баллов - 1 DET

Сложность для EI: 0 FTR + 1 DET = Low

Сложность для EO: 0 FTR + 2 DET = Low

Оценка транзакций в не выровненных функциональных точках (UFP): 7

5. Определение суммарного количества не выровненных функциональных точек (UFP).

UFP = 7 + 7 + 4 = 18

6. Определение значения фактора выравнивания (FAV).

1. *Требуется ли резервное копирование данных? 0*
2. *Требуется обмен данными? 0*
3. *Используются распределенные вычисления? 0*
4. *Важна ли производительность? 0*
5. *Программа выполняется на сильно загруженном оборудовании? 4*
6. *Требуется ли оперативный ввод данных. 0*
7. *Используется много форм для ввода данных? 0*
8. *Поля базы данных обновляются оперативно? 0*
9. *Ввод, вывод, запросы являются сложными? 0*
10. *Внутренние вычисления сложны? 1*
11. *Код предназначен для повторного использования? 2*
12. *Требуется преобразование данных и установка программы? 0*
13. *Требуется много установок в различных организациях? 0*
14. *Требуется поддерживать возможность настройки и простоту использования? 0*

Расчет суммарного эффекта 14 системных характеристик (total degree of influence, TDI) осуществляется простым суммированием: TDI = 7

Расчет значения фактора выравнивания производится по формуле

*VAF = (TDI \*0.01) + 0.65*

VAF = (7 \* 0.01) + 0.65 = 0,72

7. Расчет количества выровненных функциональных точек (AFP).

Проект разработки продукта оценивается в DFP (development functional point) по формуле:

*DFP = (UFP + CFP) \* VAF,*

*где CFP (conversion functional point) — функциональные точки, подсчитанные для дополнительной функциональности, которая потребуется при установке продукта.*

DEF = (18 + 0) \* 0.72 = 12,96

**2. Сравнить полученный результат с реальным размером вашего ПО**

Вопросы.

1. Перечислите известные вам способы оценки затрат на разработку ПО.

2. Когда целесообразно применение модели COCOMO?

3. Насколько точно, по вашему мнению, можно оценить стоимость разработки при использовании метода функциональных точек?