

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет Программной Инженерии и Компьютерной Техники

Экономика программной инженерии

Лабораторная работа № 1

Выполнили студенты

Стеберг Артём Алексеевич

Кирик Михаил Андреевич

Группа № Р34232

Преподаватель: Машина Екатерина Алексеевна

г. Санкт-Петербург

2024

## Оглавление

Задание.....	3
Функциональные требования .....	3
Оценка трудоемкости работ наивным методом .....	5
Оценить трудоемкость разработки проекта методом PERT .....	5
Сетевая диаграмма и критический путь .....	6
Финальная оценка .....	6
Метод функциональных точек .....	7
Определение типа оценки.....	7
Определение области оценки и границ продукта .....	7
Подсчет функциональных точек, связанных с данными.....	7
Подсчет функциональных точек, связанных с транзакциями.....	7
Определение суммарного количества не выровненных функциональных точек (UFP) .....	7
Определение значения фактора выравнивания (FAV).....	8
Расчет количества выровненных функциональных точек (AFP).....	8
COCOMO II .....	9
Факторы масштаба .....	9
Множители трудоемкости.....	9
Оценка трудоемкости продукта .....	9
Use Case Points.....	10
Оценка веса акторов .....	10
Оценка веса прецедентов .....	10
Оценка веса технических факторов.....	10
Оценка веса факторов окружения .....	12
Подсчет UCP .....	12
Подсчет фактора продуктивности на основе прошлого проекта.....	13
<b>Основные функциональные возможности:</b> .....	13
Актеры:.....	14
Варианты использования: .....	14
Оценка веса акторов .....	14
Оценка веса прецедентов .....	14
Оценка веса технических факторов .....	15
Оценка веса факторов окружения .....	15
Подсчет UCP .....	15
Итоги UCP .....	15
Результаты.....	16

## Задание

1. Сформировать набор функциональных требований для разработки проекта.
2. Оценить трудоемкость разработки проекта наивным методом.
3. Оценить трудоемкость разработки проекта методом PERT (Project Evaluation and Review Technique). Нарисовать сетевую диаграмму взаимосвязи работ и методом критического пути рассчитать минимальную продолжительность разработки. Предложить оптимальное количество разработчиков и оценить срок выполнения проекта.
4. Оценить размер проекта методом функциональных точек, затем, исходя из предположения, что собранной статистики по завершенным проектам нет, рассчитать трудоемкость методом COSOMO II ([Обновленная таблица количества строк на точку для разных языков программирования](#))
5. Оценить размер проекта методом оценки вариантов использования (Use Case Points). Для расчета фактора продуктивности PF использовать любой свой завершенный проект с известными временными трудозатратами, оценив его размер методом UCP.
6. Сравнить полученные результаты и сделать выводы.

## Функциональные требования

1. Система должна предоставлять редактор аватара
  - 1.1. Редактор предоставляет выбор гендера аватара
  - 1.2. Редактор предоставляет выбор цвета кожи аватара
  - 1.3. Редактор предоставляет выбор одежды для аватара
  - 1.4. Редактор предоставляет выбор типа глаз для аватара
  - 1.5. Редактор предоставляет выбор прически для аватара
  - 1.6. Редактор предоставляет выбор тип рта для аватара
  - 1.7. Редактор предоставляет выбор заднего фона для аватара
  - 1.8. Редактор реализует случайную генерацию аватара
  - 1.9. Редактор предоставляет полную информацию по составленному аватар
  - 1.10. Редактор показывает процентное соотношение выбранных элементов ко всем пользователям сайта
  - 1.11. Редактор позволяет делиться ссылкой на составленного аватара
  - 1.12. Редактор позволяет оплатить созданного аватара
2. Система должна реализовывать аутентификацию через кошельки
  - 2.1. Аутентификация может проходит через кошелек Rainbow
  - 2.2. Аутентификация может проходит через кошелек Coinbase Wallet
  - 2.3. Аутентификация может проходит через кошелек MetaMask
  - 2.4. Аутентификация может проходит через кошелек WalletConnect
3. Система должна реализовывать получения кошелька для новых пользователей

- 3.1. Получить кошелек Rainbow
  - 3.1.1. Добавить расширение для Chrom
  - 3.1.2. Установить приложение на телефон
    - 3.1.2.1. Генерация QR-кода
- 3.2. Получить кошелек Coinbase Wallet
  - 3.2.1. Добавить расширение для Chrom
  - 3.2.2. Установить приложение на телефон
    - 3.2.2.1. Генерация QR-кода
- 3.3. Получить кошелек MetaMask
  - 3.3.1. Добавить расширение для Chrom
  - 3.3.2. Установить приложение на телефон
    - 3.3.2.1. Генерация QR-кода
- 4. Система должна содержать страницу с информацией о реферальной системе
- 5. Система должна содержать ссылки на социальные медиа проекта
  - 5.1. Ссылка на Discord
  - 5.2. Ссылка на Twitter
  - 5.3. Ссылка на Facebook
  - 5.4. Ссылка на Telegram
- 6. Система должна предоставлять информацию о привилегиях при покупке аватара
- 7. Система должна предоставлять Roadmap проекта
- 8. Система должна содержать информативный footer
  - 8.1. Раздел Information
    - 8.1.1. NFT Art
    - 8.1.2. NFT Artwork
    - 8.1.3. NFT Avatars
  - 8.2. Раздел About
    - 8.2.1. Minting guide
  - 8.3. Старая версия сайта
  - 8.4. Политика конфиденциальности
  - 8.5. Условия эксплуатации
  - 8.6. Контактная почта

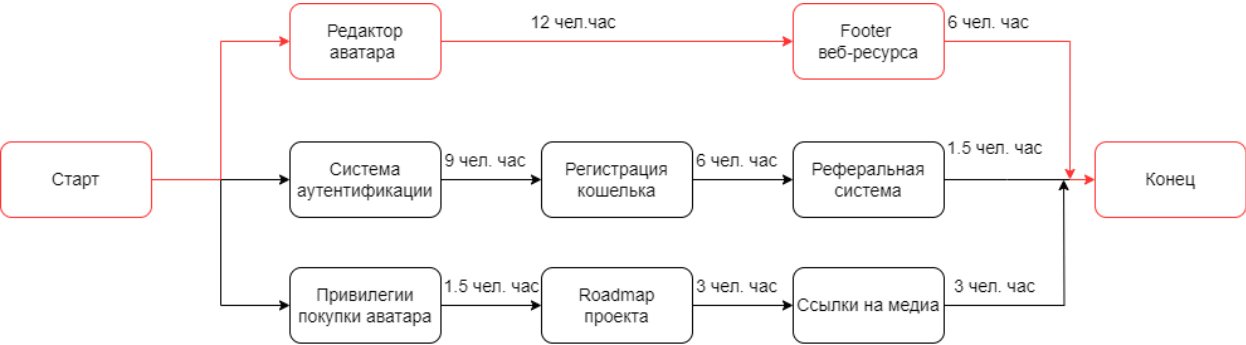
# Оценка трудоемкости работ наивным методом

№	Функции	Оценка, мин. /чел. час	Оценка, макс. /чел. час
1	Редактор аватара	8	16
2	Аутентификация	6	12
3	Получения кошелька	4	8
4	Информация о реферально системе	1	2
5	Ссылки на медиа	2	4
6	Привилегии покупки аватара	1	2
7	Roadmap проекта	2	4
8	Footer	4	8
Сумма		28	56

# Оценить трудоемкость разработки проекта методом PERT

№	Функции	Оценка, мин. /чел. час	Оценка, макс. /чел. час	Ожидаемое время (PERT) /чел. час
1	Редактор аватара	8	16	12
2	Аутентификация	6	12	9
3	Получения кошелька	4	8	6
4	Информация о реферально системе	1	2	1.5
5	Ссылки на медиа	2	4	3
6	Привилегии покупки аватара	1	2	1.5
7	Roadmap проекта	2	4	3
8	Footer	4	8	6
Сумма		28	56	42

## Сетевая диаграмма и критический путь



## Финальная оценка

Общая оценка независимых работ	Суммарная СКО	Суммарная трудоемкость	Количество человек
42	≈ 1,99	≈ 45,97213	3

# Метод функциональных точек

## Определение типа оценки

**Продукт.** Оценивается объем уже существующего и установленного продукта

## Определение области оценки и границ продукта

Только функции, реально используемые, или все функции

## Подсчет функциональных точек, связанных с данными.

№	Название	RET	DET	Сложность	UFP
1	Вход в крипто кошелек	Личная информация	QR-code	LOW	7
2	Получить крипто кошелек	Личная информация	QR-code	LOW	7

## Подсчет функциональных точек, связанных с транзакциями

№	Название	Тип	FTR	DET	Сложность	UFP
1	Вход в крипто кошелек	EI	1	4	LOW	3
2	Получить крипто кошелек	EI	1	4	LOW	3
3	Конструктор	EO	2	60	High	7
4	Реферальна программа	EO	1	1	LOW	4
5	Социальные медиа	EO	1	1	LOW	4

## Определение суммарного количества не выровненных функциональных точек (UFP)

$UFP = 14 + 21 = 35$

## Определение значения фактора выравнивания (FAV)

№	Параметр	DI
1	Обмен данными	2
2	Распределенная обработка данных	1
3	Производительность	0
4	Ограничения по аппаратным ресурсам	0
5	Транзакционная нагрузка	0
6	Интенсивность взаимодействия с пользователем	1
7	Эргономика	3
8	Интенсивность изменения данных	2
9	Сложность обработки	1
10	Повторное использование	1
11	Удобство инсталляции	0
12	Удобство администрирования	1
13	Портируемость	2
14	Гибкость	0

$$TDI = 14$$

$$VAF = (TDI * 0.01) + 0.65 = 0.79$$

## Расчет количества выровненных функциональных точек (AFP)

$$AFP = UFP * VAF = 35 * 0.79 = 27.65$$



# COCOMO II

## Факторы масштаба

Факторы масштаба	Уровень фактора	Значение фактора
PREC	Low	4.96
FLEX	High	2.03
RESL	High	2.83
TEAM	High	2.19
PMAT	Low	6.24

## Множители трудоемкости

Факторы масштаба	Уровень фактора	Значение фактора
PERS	Nominal	1.00
RCPX	Low	0.83
RUSE	Nominal	1.00
PDIF	Low	0.87
PREX	Nominal	1.00
FCIL	Low	1.10
SCED	Nominal	1.00

## Оценка трудоемкости продукта

$$KSLOC = UFP * size = 35 * \left(\frac{47}{1000}\right) = \mathbf{1.645}$$

$$E = B + 0.01 * \sum_{j=1}^5 SF_j = 0.91 + 0.01 * (4.96 + 2.03 + 7.07 + 2.19 + 7.80) = \mathbf{1.0925}$$

$$PM = A * size^E * \prod_{i=1}^n EM_i = 2.94 * 1.645^{1.0925} * (1.00 * 0.83 * 1.00 * 0.87 * 1.00 * 1.10 * 1.00) \approx \mathbf{4.02}$$

$$4.02 \text{ ч/мес} \approx \mathbf{643.6 \text{ ч/ч}}$$

# Use Case Points



## Оценка веса акторов

Сложность	Вес (Awi)	Ni	Затраты
Low	1	1	1
Medium	2	1	2
High	3	2	6
UAW			9

## Оценка веса прецедентов

Сложность	Вес (Awi)	Ni	Затраты
Low	5	14	70
Medium	10	7	70
High	15	0	0
UUCW			140

## Оценка веса технических факторов

TF	Описание	Вес Wi	Сложность Fi	Затраты
----	----------	--------	--------------	---------

T1	Распределённость	2	0	0
T2	Производительность	1	3	3
T3	Эффективность для пользователя	1	1	1
T4	Сложность внутренней обработки	1	3	3
T5	Повторное использование кода	1	1	1
T6	Простота установки	0.5	0	0
T7	Простота использования	0.5	0	0
T8	Переносимость	2	2	4
T9	Простота изменений	1	1	1
T10	Многопоточность	1	0	0
T11	Дополнительные возможности безопасности	1	1	1
T12	Доступ к другим системам	1	1	1
T13	Необходимы тренажеры	1	0	0
Сумма TF				15

$$TCF = 0.6 + 0.01 \sum_{i=1}^{13} W_i * F_i = 0.75$$

## Оценка веса факторов окружения

EF	Описание	Вес Wi	Сложность Fi	Затраты
E1	Знакомство с использованным процессом разработки	1,5	2	3
E2	Опыт применения	0,5	1	0.5
E3	Объектно-ориентированный опыт команды	1,0	4	4
E4	Возможности ведущего аналитика	0,5	3	1.5
E5	Мотивация команды	1,0	5	5
E6	Стабильность требований	2,0	2	4
E7	Персонал, занятый неполный рабочий день	-1,0	3	-3
E8	Сложный язык программирования	-1,0	5	-5
Сумма EF				10

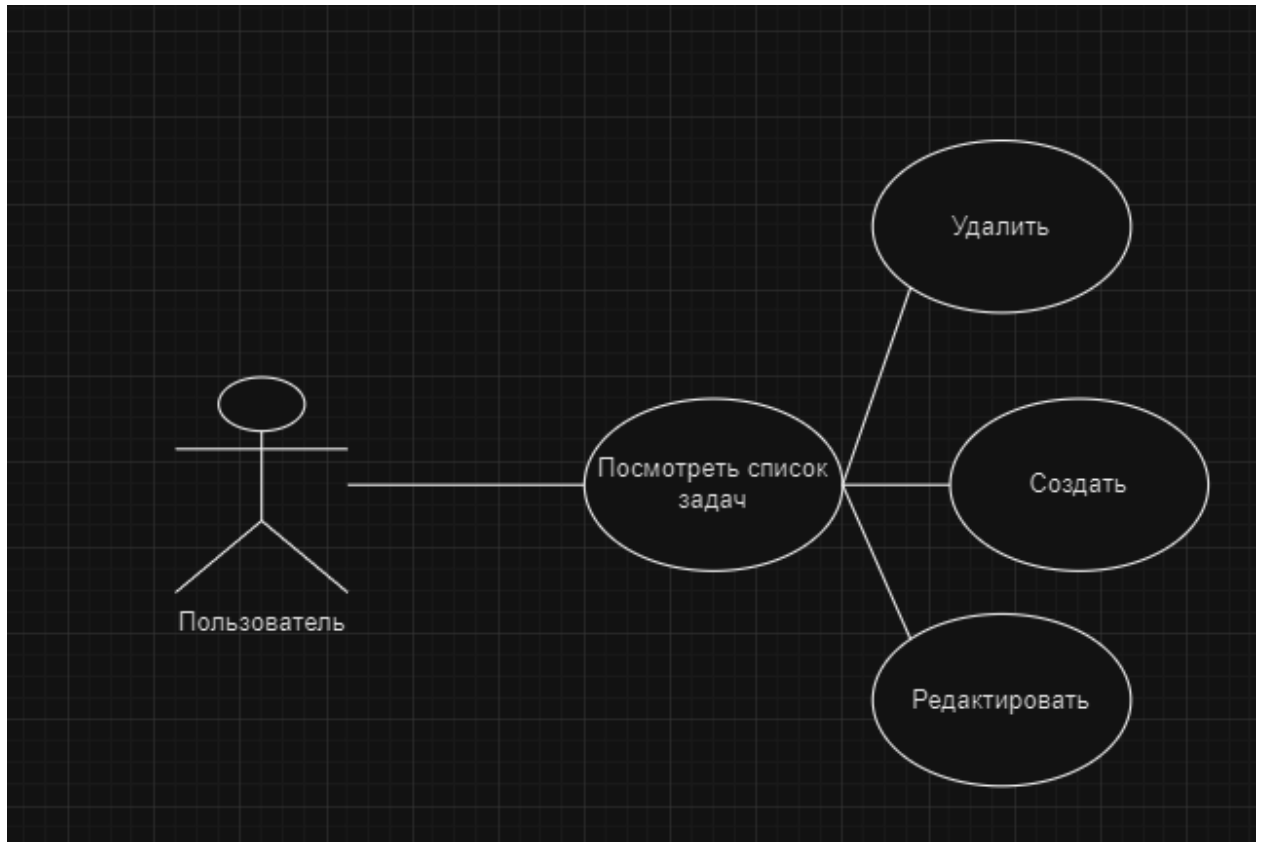
$$ECF = 1.4 - 0.03 \sum_{i=1}^8 Wi * Fi = 1.1$$

## Подсчет UCP

$$UCP' = (UAW + UUCW) * ECF * TCF \approx 122.9$$

# Подсчет фактора продуктивности на основе прошлого проекта

Название проекта: Task Tracker



## Основные функциональные возможности:

1. **Создание задачи:** Пользователь может создать задачу, задав название, описание, срок выполнения.
2. **Редактирование задачи:** Пользователь может редактировать уже созданную задачу.
3. **Удаление задачи:** Пользователь может удалять задачу.
4. **Просмотр задач:** Пользователь может видеть все свои задачи, фильтровать их по статусу (выполнена/невыполнена).
5. **Уведомления:** Пользователь получает уведомления о приближающемся дедлайне.

Акторы:

- 1. **Пользователь (User):** Взаимодействует с системой через пользовательский интерфейс для создания, редактирования, удаления и просмотра задач. (Средний актер)
- 2. **Система уведомлений (Notification System):** Генерирует и отправляет уведомления пользователю. (Простой актер)
- 3. **Администратор (Admin):** Может управлять пользователями и их задачами (сложный актер).

Варианты использования:

- 1. **Создать задачу.** (Средний)
- 2. **Редактировать задачу.** (Средний)
- 3. **Удалить задачу.** (Средний)
- 4. **Просмотреть список задач.** (Простой)
- 5. **Получить уведомление о задаче.** (Простой)
- 6. **Управление пользователями.** (Сложный)

Оценка веса акторов

Сложность	Вес (Awi)	Ni	Затраты
Low	1	1	1
Medium	2	1	2
High	3	1	3
UAW			6

Оценка веса прецедентов

Сложность	Вес (Awi)	Ni	Затраты
Low	5	2	10
Medium	10	3	30
High	15	1	15
UUCW			55

## Оценка веса технических факторов

TF	Описание	Вес Wi	Сложность Fi	Затраты
T1	Требования к производительности	1	2	2
T2	Дистрибуция системы	1	2	2
T3	Легкость установки	1	3	3
T4	Легкость использования	1	3	3
T5	Поддержка многопользовательской работы	1	2	2
T6	Интеграция с внешними системами	1	4	4
T7	Портативность системы	1	2	2
T8	Обучаемость пользователей	1	2	2
Сумма TF				20

$$TCF = 0.6 + 0.01 \sum_{i=1}^8 Wi * Fi = 0.8$$

## Оценка веса факторов окружения

EF	Описание	Вес Wi	Сложность Fi	Затраты
E1	Опыт работы с языком программирования	1	3	3
E2	Опыт работы с аналогичными системами	1	4	4
E3	Мотивация команды	1	3	3
E4	Стабильность требований	1	2	2
E5	Трудности в разработке	1	3	3
Сумма EF				15

$$ECF = 1.4 - 0.03 \sum_{i=1}^5 Wi * Fi = 0.95$$

## Подсчет UCP

$$UCP' = (UAW + UUCW) * ECF * TCF \approx 46.36$$

Однако, работы была выполнена суммарно за 43 часа

## Итоги UCP

$$PF = \frac{43}{46.36} \approx 0.93$$

$$UCP = 0.93 * 122.9 \approx 114.3$$

## Результаты

Метод	Затраты
Наивный	56
PERT	42
Функциональные точки	$27.65 * 4 = 124.4$
COCOMO II	643.6
UCP	114.3

Подводя итог, мы получили наглядные отличительные результаты. В методах с личностной оценкой трудозатрат и определения факторов (**наивный** и **PERT**) прослеживается логика оценки, так как опыт аналитика не может здраво оценить реализуемый проект. Что порождает недооценку возможно ключевых факторов.

В методах, где присутствует перечень факторов, а также веса этих факторов, аналитик способен на рассмотрения проекта под другими углами, что порождает добавления решаемых задач. Однако, метод COCOMO II выделяется на фоне других. Такой результат связан с принципом подсчета, ведь метод COCOMO II не делает отдельную оценку для транзакций и акторов, в итоге получает оценку как для более-менее среднего (по сложности) проекта, что, по субъективному мнению, таковым не является.

Стоит отметить, что каждый метод, позволяет оценить разрабатываемый продукт, причем замер учитываемых факторов варьируется, что и порождает вариативность в реализации оценки продукта.