

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ»**

**Факультет программной инженерии и компьютерной техники
Направление «Нейротехнологии и программирование»**

**Дисциплина:
«Экономика программной инженерии»**

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

Выполнил:
Студент группы Р34202
Выблов Глеб

Проверил:
Машина Екатерина Алексеевна

Санкт-Петербург
2024г.

Для выданного веб-проекта: Райффайзен Банк (<https://www.raiffeisen.ru>)

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.

1. Сформировать набор функциональных требований для разработки проекта.

Основные разделы и функции

1. Главная страница
 - 1.1. Приветственный баннер с основными предложениями (вклады, кредиты, инвестиции, премиальное обслуживание)
 - 1.2. Интерактивное меню для перехода к основным разделам
2. Интернет-банк и мобильное приложение
 - 2.1.>Login/регистрация (с использованием двухфакторной аутентификации)
 - 2.2. Управление счетами и картами (просмотр баланса, операций и переводов)
 - 2.3. Оплата услуг (ЖКХ, штрафы, связь)
 - 2.4. Подключение дополнительных сервисов (вкладов, кредитов)
3. Кредитование
 - 3.1. Информация о кредитных продуктах (потребительские, ипотека, автокредиты, бизнес-кредиты)
 - 3.2. Онлайн-заявка на кредит с возможностью расчета через кредитный калькулятор
4. Вклады и счета
 - 4.1. Просмотр и открытие вкладов
 - 4.2. Расчет доходности с помощью онлайн-калькулятора
5. Карты
 - 5.1. Выбор и оформление дебетовых и кредитных карт
 - 5.2. Информация о кэшбэке и скидках
6. Инвестиции
 - 6.1. Услуги брокерского обслуживания

- 6.2. Покупка/продажа акций, облигаций
 - 6.3. Актуальная информация о курсах валют и стоимости активов
- 7. Обслуживание
 - 7.1. Информация о бизнес-кредитах, расчетных счетах
 - 7.2. Онлайн-заявка для бизнеса
- 8. Поддержка пользователей
 - 8.1. Раздел с часто задаваемыми вопросами (FAQ)
 - 8.2. Чат с поддержкой
 - 8.3. Телефон горячей линии

Функции безопасности

1. Использование SSL/TLS для шифрования данных
2. Защита от SQL-инъекций, XSS-атак и CSRF
3. Верификация пользователя через СМС, биометрию или токены
4. Лимитирование количества неудачных попыток входа

Адаптивность и доступность

1. Адаптивный дизайн для корректного отображения на мобильных устройствах, планшетах и десктопах
2. Соответствие стандартам доступности WCAG 2.1 (включая текстовые альтернативы, удобную навигацию, поддержку скринридеров)

Интеграции

1. Интеграция с платежными системами (Visa, MasterCard, Мир, СБП)
2. Взаимодействие с сервисами проверки данных
3. Интеграция с государственными сервисами (Госуслуги, налоговая служба)

Отчеты и уведомления

1. Генерация отчетов по операциям
2. Уведомления о транзакциях, изменениях тарифов, актуальных предложениях (через СМС, push-уведомления, email)

Панель администратора

1. Управление контентом сайта (баннеры, описания продуктов, акции)
2. Управление пользователями (разделение на роли, блокировка, восстановление)
3. Мониторинг активности пользователей

Другие функциональные требования

1. Многоязычная поддержка (например, русский, английский)
2. Возможность подачи заявки на продукты без обязательной регистрации

2. Оценить трудоемкость разработки проекта наивным методом.

№	Функционал	Модулей
1	Меню и ссылки на основные разделы	1
1.2	Логин/регистрация	2
	Управление счетами и картами	3
	Оплата услуг	2
	Подключение дополнительных сервисов	2
1.3	Информация о продуктах	1
	Кредитный калькулятор	1
	Онлайн-заявка	2
1.4	Просмотр и открытие вкладов	2
	Калькулятор доходности	1
1.5	Выбор и оформление карт	2
	Информация о кэшбэке и скидках	1
1.6	Брокерское обслуживание	2
	Курс валют и активов	2
1.7	Информация о продуктах	1
	Онлайн-заявка	1
1.8	FAQ	1
	Чат поддержки	2
	Горячая линия	1

2	SSL/TLS	2
	Защита от атак (SQL, XSS, CSRF)	3
	Двухфакторная аутентификация	2

3	Адаптивный дизайн	2
	Стандарты доступности WCAG	2
4	Платёжные системы	2
	Сервисы проверки данных	2
	Государственные сервисы	2
5	Генерация отчетов	2
	Уведомления (СМС, email)	2
6	Управление контентом	2
	Управление пользователями	2
	Мониторинг активности	2
7	Поиск с фильтрацией	2
	Многоязычная поддержка	2
	Подача заявки без регистрации	1

Подсчет трудоемкости:

$$1+9+4+3+3+4+2+4+7+4+6+4+6+5=62$$

Предположим, что на реализацию одного модуля требуется 10 человеко-часов:

$$62 * 10 = 620 \text{ человеко-часов}$$

$$620 / (5 * 8) = 15.5 \text{ рабочих дней}$$

2. 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.

Метод PERT— это метод планирования и оценки проекта, который учитывает неопределенности и вероятности при оценке продолжительности задач

Разбиение проекта на задачи и взаимосвязи

Номер	Задача	Предшественники
A	Дизайн главной страницы	-
B	Разработка интернет-банка	A
C	Реализация кредитного калькулятора	A
D	Разработка раздела "Карты"	A
E	Вклады и счета	A
F	Интеграция с платёжными системами	B
G	Интеграция с государственными сервисами	B, E
H	Безопасность	B
I	Панель администратора	B, H
J	Поддержка пользователей	D, H
K	Тестирование и отладка	G, I, J

Оценка времени для задач

Задача	Оптимистичное время	Наиболее вероятное	Пессимистичное время	Среднее ожидаемое
A	2	3	5	3.33
B	10	12	15	12.33
C	2	3	5	3.33
D	3	4	6	4.33
E	2	3	5	3.33

F	3	4	6	4.33
G	4	5	7	5.33
H	5	6	8	6.33
I	4	5	7	5.33
J	4	5	7	5.33
K	7	8	10	8.33

Оценка времени по методу PERT:

Оценка оптимистическая (О) - минимальная возможная продолжительность

Оценка наиболее вероятная (М) - наиболее вероятная продолжительность

Оценка пессимистическая (Р) - максимальная возможная продолжительность

Формула для вычисления ожидаемого времени T_{exp} :

$$T_{exp} = \frac{O + 4M + P}{6}$$

Построение сетевой диаграммы

- A → B → H → I → K
- A → C → K
- A → D → J → K
- A → E → G → K
- B → F → K

Метод критического пути

- 1) A → B → H → I → K
3.33+12.33+6.33+5.33+8.33=35.65
- 2) A → C → K
3.33+3.33+8.33=15.99
- 3) A → D → J → K
3.33+4.33+5.33+8.33=21.32
- 4) A → E → G → K
3.33+3.33+5.33+8.33=20.32
- 5) B → F → K
12.33+4.33+8.33=24.99

Самый длинный путь: Путь 1 (A → B → H → I → K)

Минимальная продолжительность: 35.65 рабочих дней

Разобьем задачи критического пути на подзадачи:

Задача	Подзадачи
A	A1,A2
B	B1,B2
H	H1
I	I1,I2,I3
K	K1

Описание подзадач:

A1: Создание прототипа главной страницы

A2: Визуализация дизайна главной страницы

B1: Разработка пользовательского интерфейса

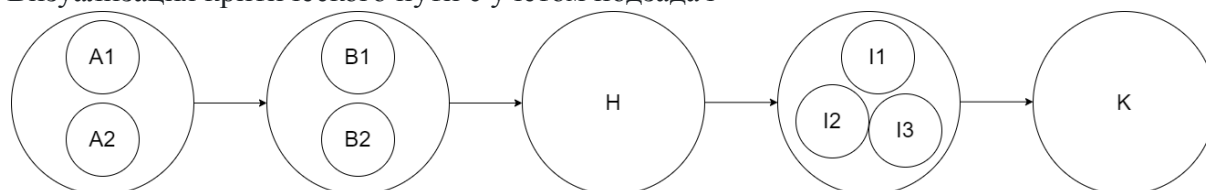
B2: Разработка функциональности

I1: Разработка интерфейса панели администратора

I2: Разработка функциональности панели администратора

I3: Тестирование и отладка

Визуализация критического пути с учетом подзадач



Вывод

1. Критический путь задает минимальный срок в 35.65 дней
2. 5 разработчиков достаточно для выполнения всех задач за это время
3. Увеличение числа разработчиков не сократит срок, но увеличит избыточные ресурсы и затраты

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)

Метод функциональных точек оценивает размер проекта на основе количества и сложности различных функциональных компонентов системы. Это позволяет оценить объем работы и сложности до начала разработки

Оценка размера проекта методом функциональных точек (FP)

Е1 (Входные данные): формы ввода данных (форма входа в аккаунт, запросы на кредит) — 8 (средней сложности)

ЕО (Выходные данные): отчёты, обработанные данные (состояние счёта, выгрузки отчётов) — 12 (средней сложности)

EQ (Запросы): поисковые запросы или фильтрация данных (поиск отделений) — 5 (средней сложности)

ILF (Внутренние логические файлы): внутренние базы данных, связанные с клиентами, транзакциями — 6 (сложные)

EIF (Внешние интерфейсные файлы): интеграция с внешними системами (системы платёжных шлюзов) — 4 (сложные)

FP Unadjusted (Неоткорректированные функциональные точки)

Тип элемента	Количество	Сложность	Вклад FP
EI	8	4	32
EO	12	5	60
EQ	5	4	20
ILF	6	15	90
EIF	4	10	40
Итого:			242

CAF (Корректировка на фактор сложности)

$$CAF = 0.65 + 0.01 * 40 = 1.05$$

Итоговый FP (функциональные точки)

$$FP = 242 * 1.05 = 254.1 \sim 254$$

LOC (Перевод функциональных точек в количество строк кода)

Для веб-проекта предположим использование **Java**. Согласно таблице 1 FP для Java = **53 LOC**.

$$LOC = FP * 53 = 254 * 53 = 13462 \text{ строк кода}$$

Расчёт трудоёмкости методом COCOMO II

$$E = A * (Size)^B * \prod_{i=1}^n E_i * M_i$$

Где:

A — коэффициент, зависящий от типа проекта. Для средних коммерческих систем: A = 2.94

Size — размер проекта в KLOC (тысячах строк кода).

B — показатель степени, учитывающий сложность проекта. Для среднего уровня сложности: B = 1.12

E_i — факторы масштаба (значения из таблицы для конкретных параметров проекта).

Подставим данные:

$$\text{Размер проекта Size} = LOC / 1000 = 13,462 / 1000 = 13.462 \text{ KLOC}$$

Факторы масштаба (ЕМ) $\prod EM_i = 1.15 * 0.85 * 1.10 = 1.075$

Расчет трудоемкости $E = 2.94 * 13.462^{(1.12)} * 1.075 = 58.12$ человеко-месяцев

Оценка времени разработки

Время разработки TDEV определяется по формуле: $TDEV = C * E^{(D)}$

Где:

C — коэффициент, зависящий от типа проекта (для коммерческих систем $C = 3.67$).

D — показатель степени, зависящий от сложности проекта (обычно $D = 0.32$).

$TDEV = 13.47$ месяцев

Оптимальное количество разработчиков: $58.12 / 13.47 = 4.3 = 5$ человек

Итог

1. **Размер проекта:** 13,462 LOC (строк кода)
2. **Трудоемкость:** 58,12 человеко-месяцев
3. **Сроки выполнения:** 13,47 месяцев
4. **Оптимальный состав команды:** 5 разработчиков

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 UCR.

Метод оценки на основе анализа вариантов использования помогает оценить трудоемкость проекта, а также позволяет рассчитать объем работы, исходя из сложности системы

Список вариантов использования

На основе функциональных требований для веб-проекта "Райффайзен Банк" выделим основные варианты использования:

- Авторизация клиента
- Просмотр баланса и операций

- Переводы между счетами
- Платежи за услуги
- Оформление заявки на кредит

Предположим количество транзакций:

Use Case	Транзакции	Сложность
Авторизация клиента	3	Простая
Просмотр баланса и операций	5	Средняя
Переводы между счетами	7	Средняя
Платежи за услуги	10	Средняя
Оформление заявки на кредит	12	Средняя

Рассчитаем UUCW

Используем веса:

- Простая: 5
- Средняя: 10
- Сложная: 15

$$UUCW = (\text{Простые} * 5) + (\text{Средние} * 10) + (\text{Сложные} * 15) = 5 + 40 = 45$$

Рассчитаем UAW

Для акторов:

- Простые (API/скрипты): Вес = 1.
- Средние (GUI): Вес = 2.
- Сложные (внешние системы): Вес = 3.

Предположим:

Акторы	Тип	Вес	Количество	Итог
Клиент	Средний	2	1	2
Система оплаты услуг	Средний	3	1	3
Система авторизации банка	Средний	2	1	2

$$UAW = (1 * 2) + (1 * 3) + (1 * 2) = 7$$

Рассчитаем необработанный UCP

$$UCP = UUCW + UAW$$

$$UCP = 45 + 7 = 52$$

Уточняем UCP с учётом факторов

Технические факторы (TF):

Предположим оценки параметров, таких как распределенность системы, производительность и т.д.

$$TF = 0.6 + (0.01 * \text{сумма весов технических факторов})$$

Сумма весов: 30.

$$TF = 0.6 + (0.01 \times 30) = 0.9$$

Средовые факторы (EF):

Оценим уровень квалификации команды, опыта и т.п.

$$EF = 1.4 - (0.03 * \text{сумма весов средовых факторов})$$

Сумма весов: 20

$$EF = 1.4 - (0.03 * 20) = 0.8$$

Уточненный UCP:

$$UCP(\text{adjusted}) = UCP * TF * EF$$

$$UCP(\text{adjusted}) = 52 * 0.9 * 0.8 = 37.34$$

Фактор продуктивности (PF)

Используем данные завершённого проекта:

- Размер завершённого проекта (в UCP): 50
- Трудоемкость завершённого проекта: 1000 человеко-часов

$$PF = \text{трудоемкость (часы)} / UCP$$

$$PF = 1000 / 50 = 20 \text{ человеко-часов на UCP}$$

Трудоёмкость текущего проекта

$$\text{Трудоёмкость} = UCP(\text{adjusted}) * PF = 37.44 * 20 = 748.8 \text{ человеко-часов}$$

Итог

- **Размер проекта:** 37.44 UCP
- **Трудоемкость:** 748.8 человеко-часов
- **Вывод:** При 5 разработчиках проект завершится за $748.8 / 5 * 8 \approx 18.72$ рабочих дней (при 8-часовом рабочем дне)

Сравнение полученных результатов и выводы

Метод оценки	Трудоемкость (человеко-часов)	Срок выполнения (дни) (при 5 разработчиках и 8-часовом рабочем дне)
--------------	----------------------------------	--

Наивный метод	620	15.5
Метод PERT	1426	35.65
Метод функциональных точек (COCOMO II)	581.2	13.47
Метод вариантов использования (UCP)	748.8	18.72

Анализ полученных результатов

1. **Наивный метод:**

Этот метод дал усредненную оценку трудоемкости, но он не учитывает сложности отдельных задач и специфики разработки. Этот метод близок к другим методам, что говорит о корректности базовых предположений

2. **Метод PERT:**

Учитывает вероятности и неопределенности выполнения задач, предоставляя более точную оценку. Он также оказался близким к методу UCP, что подтверждает его надежность

3. **Метод функциональных точек (COCOMO II):**

Этот метод показывает низкую трудоемкость и короткий срок выполнения, что делает его привлекательным для проектов с ограниченными ресурсами. Однако, важно учитывать, что такая оценка может быть не всегда точной

4. **Метод вариантов использования (UCP):**

Результаты этого метода находятся между наивным методом и методом PERT. Он предлагает более сбалансированный подход, учитывая как трудоемкость, так и сроки

Выводы

- Метод PERT предоставляет наиболее полную картину, но требует больше ресурсов и времени, что может быть оправдано для крупных и сложных проектов
- Метод UCP является компромиссом между точностью и затратами, что делает его подходящим для многих типов проектов