

Федеральное государственное автономное образовательное учреждение
высшего образования
Университет ИТМО

Дисциплина: Экономика программной инженерии

Лабораторная работа 1

Вариант <https://selectel.ru/>

Выполнили:

Гасюк Александр Андреевич

Чжоу Хунсян

Группа: Р34131

Преподаватель:

Гаврилов Антон Валерьевич

2024 г.

Санкт-Петербург

Оглавление

1. Задание:.....	3
2. Функциональные требования	4
1. Функциональные требования к управлению пользователями.....	4
2. Выбор товаров и услуг	5
3. Функциональные требования к управлению инфраструктурой.....	5
4. Функциональные требования к поддержке разработчиков	6
5. Функциональные требования к технической поддержке.....	6
1. Наивый метод	7
2. PERT метод	12
3. Метод критического пути	14
4. Метод функциональных точек	16
Определение типа оценки	16
Определение области оценки и границ продукта	16
Подсчет функциональных точек, связанных с данными.....	16
Подсчет функциональных точек, связанных с транзакциями	18
Определение суммарного количества не выровненных функциональных точек (UFP).....	20
Определение значения фактора выравнивания (VAF).....	20
Расчет количества выровненных функциональных точек (AFP)	21
5. COSOMO II.....	22
Оценка размера программного продукта в KSLOC.....	22
Оценка уровней факторов масштаба.....	22
Оценка уровней множителей трудоемкости.....	22
Оценка трудоемкости проекта	23
6. Use Case Points.....	24
Technical Complexity Factors:	25
Environmental Complexity Factors	26
Unadjusted Use Case Points (UUCP).....	26
Unadjusted Use Case Weight (UUCW):.....	26
Unadjusted Actor Weight (UAW):	27
Productivity Factor:	27
Анализ результатов	28
Вывод	29

1. Задание:

Для выданного веб-проекта:

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.

2. Функциональные требования

1. Функциональные требования к управлению пользователями

Регистрация пользователя и вход в систему

- Разрешить пользователям регистрировать новые учетные записи и входить в существующие учетные записи.
- Поддержка регистрации электронной почты и номера мобильного телефона.
- Многофакторная аутентификация (2FA).

Управление профилями пользователей

- Пользователи могут управлять информацией об учетной записи.
- Обновить информацию об учетной записи (имя, контактную информацию и т. д.).
- Изменение паролей и настроек безопасности.

Управление доступом пользователей

- Разрешить пользователям управлять доступом других пользователей к ресурсам.
- Разрешить пользователям создавать группы пользователей для массового управления правами доступа.
- Разрешить ограничивать доступ к учетным записям по IP-адресу или подсети

Биллинг и управление счетами

- Пользователи могут просматривать записи потребления, пополнять счета, выставять счета и расходы.
- Поддержка автоматического пополнения баланса
- Отображение баланса счета в режиме реального времени.
- Поддерживает несколько способов оплаты (кредитная карта, банковский перевод, электронный кошелек).
- Скачать детали счета.

2. Выбор товаров и услуг

Каталог услуг и цены

- Подробное динамическое отображение подробных конфигураций и цен всех категорий товаров.
- Предоставляет калькулятор цен для оценки затрат в режиме реального времени.
- Интегрируйте часто задаваемые вопросы и обращения клиентов, чтобы повысить доверие клиентов.

3. Функциональные требования к управлению инфраструктурой

Управление ресурсами облачных вычислений

- Пользователи могут отслеживать и управлять приобретенными облачными ресурсами в режиме реального времени.
- Создание, запуск и остановка виртуальных машин.
- Просмотр использования ресурсов (ЦП, память, сеть) в режиме реального времени.
- Автоматическое масштабирование ресурсов вверх и вниз.

Управление хранилищем и резервным копированием

- Разрешить пользователям управлять пространством хранения и задачами резервного копирования.
- Создавайте контейнеры для хранения и устанавливайте права доступа.
- Настройте регулярные задачи резервного копирования.
- Управляйте хранилищем файлов через API или интерфейс управления.

Управление сетевыми услугами

- Управление сетевыми ресурсами.
- Настройте виртуальную сеть (VPC).
- Настройка правил брандмауэра.
- Управление доменными именами и сертификатами SSL.

4. Функциональные требования к поддержке разработчиков

Интеграция API и поддержка документации

- Разработчики могут взаимодействовать с платформой через API.
- Предоставьте полную документацию по API и примеры использования.
- Предоставление инструментов SDK и CLI для поддержки многоязычной разработки.
- Настройте управление ключами API и контроль доступа.

5. Функциональные требования к технической поддержке

Онлайн-система обслуживания клиентов и поддержки

- Пользователи могут обратиться в техническую поддержку в любое время.
- Круглосуточная онлайн-поддержка в чате.
- Часто задаваемые вопросы и база знаний: охватывают общие вопросы и решения.

Центр сообщества и документации

- Пользователи имеют доступ к техническим сообществам и центрам документации.
- Взаимодействие с пользователем и ответы на вопросы.
- Регулярно публиковать туториалы и технические кейсы.
- Видеообучающие ресурсы.

1. Наивый метод

№	Название	Описание	Оптимистичное время (часы)	Пессимистичное время (часы)	Лучшее время (часы)
1	Подготовка		200	450	300
1.1	Прототип сайта	Разработка начального прототипа сайта, проектирование пользовательского опыта и анимаций взаимодействия для главной страницы, страницы услуг, страницы продуктов и других разделов.	50	100	70
1.2	Выбор технологий	Определение технологического стека: использование ReactJS для динамического взаимодействия на фронтенде, Spring для быстрого обработки данных на бэкенде, выбор PostgreSQL в качестве базы данных.	10	40	20
1.3	Разрешение на обработку данных	Ввиду работы с регистрацией пользователей и хранением данных необходимо получить разрешение на обработку данных и обеспечить соблюдение	100	250	150

		законодательных норм.			
1.4	Настройка хостинга и домена	Определение оптимального хостинга и конфигурации домена с учетом производительности и стоимости.	40	60	50
2	Фронтенд		300	500	400
2.1	Главная страница	Адаптивный дизайн, отображение основных услуг, динамическая загрузка контента, оптимизация анимаций, акцент на ключевой информации.	40	80	60
2.2	Страница входа/регистрации	Динамическая валидация форм, адаптивный макет, поддержка двухфакторной аутентификации (2FA).	30	50	40
2.3	Страница услуг	Отображение различных облачных услуг Selectel, поддержка динамической фильтрации и отображения подробной информации.	50	90	70

2.4	Страница продуктов	Представление подробной информации о продуктах, включая технические параметры и калькуляцию стоимости.	50	100	75
2.5	Центр поддержки	Включает раздел часто задаваемых вопросов, форму отправки запросов и поддержку в реальном времени через чат.	40	80	60
2.6	Страница документации	Отображение технической документации с возможностью постраничного чтения и полнотекстового поиска.	30	60	45
2.7	Панель управления пользователем	Включает управление аккаунтом, историю заказов, мониторинг состояния услуг и устройств, поддержку пользовательской настройки.	60	120	90
3	Бэкенд		250	450	350
3.1	Проектирование и настройка базы данных	Создание структуры базы данных, включающей данные пользователей, заказы, состояние услуг и устройств, оптимизация индексов.	60	120	90

3.2	Аутентификация и управление правами доступа	Реализация логики входа, регистрации и управления правами доступа, поддержка различных ролей (обычные пользователи, администраторы).	40	80	60
3.3	Проектирование и разработка API	Проектирование и разработка RESTful API для взаимодействия фронтенда и бэкенда.	80	150	100
3.4	Безопасность и шифрование данных	Реализация шифрования данных для хранения и обеспечение защищенной связи (например, через SSL/TLS).	40	80	60
4	Тестирование		150	300	200
4.1	Модульное тестирование	Проверка надежности функциональности каждого модуля, обеспечение корректности логики фронтенда и бэкенда.	50	100	75
4.2	Интеграционное тестирование	Проверка корректности взаимодействия между фронтендом, бэкендом и внешними сервисами.	40	80	60

4.3	Функциональное тестирование	Тестирование полных пользовательских сценариев, включая регистрацию, заказ услуг, просмотр документации и т. д.	60	120	90
5	Релиз		100	200	150
5.1	Alpha и Beta тестирование	Проведение внутреннего тестирования и открытого тестирования основных функций, сбор обратной связи и оптимизация.	50	100	75
5.2	Настройка SSL-сертификата	Установка и настройка SSL-сертификата для работы сайта по протоколу HTTPS.	10	20	15
5.3	Развертывание сервера	Размещение кода на облачном сервере и выполнение необходимой конфигурации окружения.	30	60	45
Z	Итого		1000	1950	1400

Таблица 1.

2. PERT метод

№	Название	Optimistic (h)	Pessimistic (h)	Optimal (h)	$E_i = \frac{P_i + O_i + 4M_i}{6}$	$CKO_i = \frac{P_i - O_i}{6}$
1.1	Прототип сайта	50	100	70	95.00	25.00
1.2	Выбор технологий	10	40	20	28.33	8.33
1.3	Разрешение на обработку данных	100	250	150	208.33	58.33
1.4	Настройка хостинга и домена	40	60	50	66.67	16.67
2.1	Главная страница	40	80	60	80.00	20.00
2.2	Страница входа/регистрации	30	50	40	53.33	13.33
2.3	Страница услуг	50	90	70	93.33	23.33
2.4	Страница продуктов	50	100	75	100.00	25.00
2.5	Центр поддержки	40	80	60	80.00	20.00
2.6	Страница документации	30	60	45	60.00	15.00
2.7	Панель управления пользователем	60	120	90	120.00	30.00
3.1	Проектирование и настройка базы данных	60	120	90	120.00	30.00
3.2	Аутентификация и управление правами доступа	40	80	60	80.00	20.00
3.3	Проектирование и разработка API	80	150	100	138.33	38.33
3.4	Безопасность и шифрование данных	40	80	60	80.00	20.00
4.1	Модульное тестирование	50	100	75	100.00	25.00
4.2	Интеграционное тестирование	40	80	60	80.00	20.00
4.3	Функциональное тестирование	60	120	90	120.00	30.00
5.1	Alpha и Beta тестирование	50	100	75	100.00	25.00

5.2	Настройка SSL-сертификата	10	20	15	20.00	5.00
5.3	Развертывание сервера	30	60	45	60.00	15.00

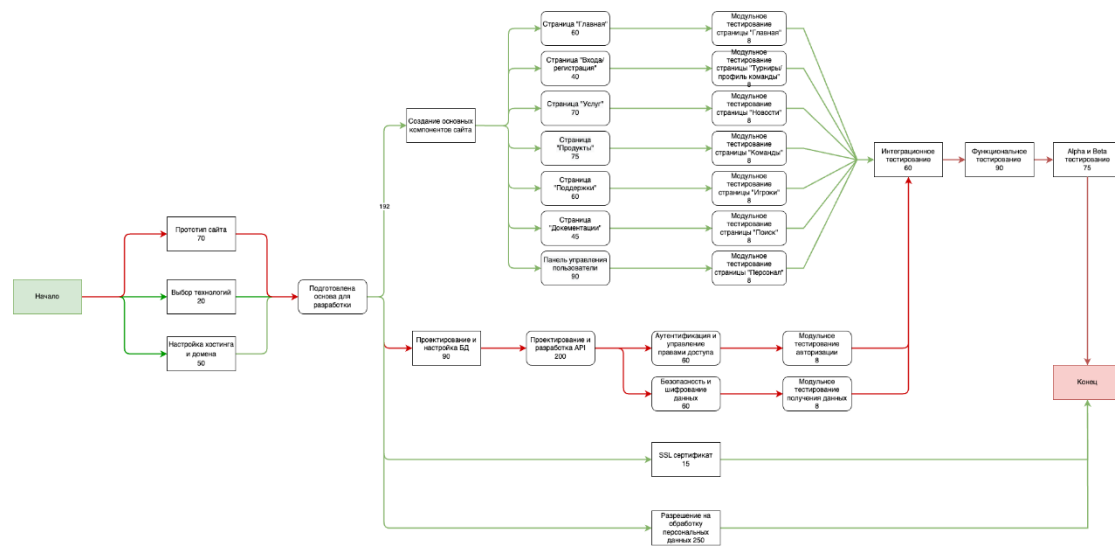
1883.33

Таблица 2

$$E = \sum E_i \approx 1883.33$$

$$CKO = \sqrt{\sum CKO_i^2} \approx 116.69$$

3. Метод критического пути



Длинный путь: 653 ч./ч.

Выполнение проекта: при ориентире на минимальное время разработки (критический путь) получаем, что для выполнения нам необходимо 653ч./ч.

Команда:

- 1x Аналитик
- 2x Frontend-разработчик
- 2x Backend-разработчик
- 2x Тестировщика

Рабочий день считаем: 8 часов (6 разработки + 1 обед + 1 тех. перерыв)

Таким образом, наша команда сможет выполнить проект за:

- Дизайн: 140 часов (24 раб. день)
- Фронтенд: 440 часов (74 раб. день)
- Бэкенд: 426 часов (71 раб. день)
- Тестирование: 222 часов (37 раб. день)
- Релиз: 135 часов (23 раб. день)

Время разработки: $140+440+426+222+135 = 1363$ часов

Общее время: $24+74+71+37+23 = 229$ рабочих дней

4. Метод функциональных точек

При анализе методом функциональных точек надо выполнить следующую последовательность шагов:

1. Определение типа оценки
2. Определение области оценки и границ продукта
3. Подсчет функциональных точек, связанных с данными
4. Подсчет функциональных точек, связанных с транзакциями
5. Определение суммарного количества не выровненных функциональных точек (UFP)
6. Определение значения фактора выравнивания (FAV)
7. Расчет количества выровненных функциональных точек (AFP)

Определение типа оценки

Продукт. Оценивается объем уже существующего и установленного продукта.

Определение области оценки и границ продукта

Все функции. Рассчитываем все необходимые (реально используемые), а не дополнительные или только основные функции. Границы системы определены на

UseCase диаграмме.

Подсчет функциональных точек, связанных с данными

DET (data element type) - неповторяемое уникальное поле данных.

- Имя человека (имя) - 1 DET
- Адрес человека (индекс, страна, город, улица, дом, корпус, квартира) - 7 DET

RET (record element type) - логическая группа данных

- адрес
- паспорт
- телефонный номер.

	1-10 DET	11-20 DET	20+ DET
--	----------	-----------	---------

1 RET	Low	Low	Average
2 RET	Low	Average	High
3+ RET	Average	High	High

Таблица 3. Матрица сложности данных

Сложность данных	Количество UFP (ILF)	Количество UFP (EIF)
Low	7	5
Average	10	7
High	15	10

Таблица 4. Оценка данных в не выровненных функциональных точках (UFP) для внутренних логических файлов (ILFs) и внешних интерфейсных файлов (EIFs)

№	Название	RET	DET	Сложность	UFP
1	Форма регистрации	Данные входа, личная информация (2)	Email, пароль, имя, фамилия, пол, дата рождения, город, телефон (9)	Low	7
2	Авторизация (логин)	Данные входа (1)	Email, пароль (2)	Low	7
3	Система оплаты	Заказ, информация о платеже (2)	Сумма, дата, метод оплаты (4)	Low	7
4	Просмотр услуг	Каталог услуг (1)	Название услуги, описание, цена, тип, доступность (5)	Low	7
5	Заказ услуги	Заказ, информация о платеже (2)	Услуга, дата заказа, сумма, метод оплаты, контактные данные (6)	Low	7
6	Личный кабинет	Данные пользователя (1)	Имя, фамилия, email, телефон, адрес, история заказов, предпочтения (8)	Low	7
7	Контакт с поддержкой	Обращение пользователя (1)	Тема запроса, описание, дата, статус, приоритет (5)	Low	7
8	Отчеты по заказам	История заказов (1)	Заказ, дата, статус, сумма, услуги (5)	Low	7
9	Регистрация оплаты	Данные о платеже (2)	Сумма, дата, метод оплаты, статус (4)	Low	7
10	Продуктовый каталог	Данные товаров/услуг (1)	Название, описание, категория, цена, доступность, дата обновления (6)	Low	7

Таблица 5. Список функциональных точек, связанных с данными

Подсчет функциональных точек, связанных с транзакциями

Транзакция — это элементарный неделимый замкнутый процесс, представляющий значение для пользователя и переводящий продукт из одного консистентного состояния в другое.

В методе различаются следующие типы транзакций (Таблица 9):

- EI (external inputs) — внешние входные транзакции, элементарная операция по обработке данных или управляющей информации, поступающих в систему из вне.
- 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.
- EQ (external inquiries) — внешние запросы, элементарная операция, которая в ответ на внешний запрос извлекает данные или управляющую информацию из ILF или EIF.

Функция	Тип транзакции		
	EI	EO	EQ
Изменяет поведение системы	О	Д	NA
Поддержка одного или более ILF	О	Д	NA
Представление информации пользователю	Д	О	О

Таблица 6. Основные отличия между типами транзакций. Легенда: О — основная; Д — дополнительная; NA — не применима.

Оценка сложности транзакции основывается на следующих ее характеристиках:

- FTR (file type referenced) — позволяет подсчитать количество различных файлов (информационных объектов) типа ILF и/или EIF модифицируемых или считываемых в транзакции.
- DET (data element type) — неповторяемое уникальное поле данных.
Примеры. EI: поле ввода, кнопка. EO: поле данных отчета, сообщение об ошибке. EQ: поле ввода для поиска, поле вывода результата поиска.

EI	1-4 DET	5-15 DET	16+ DET
0-1 FTR	Low	Low	Average
2 FTR	Low	Average	High
3+ FTR	Average	High	High

Таблица 7. Матрица сложности внешних входных транзакций (EI)

EO & EQ	1-5 DET	6-19 DET	20+ DET
0-1 FTR	Low	Low	Average
2-3 FTR	Low	Average	High
4+ FTR	Average	High	High

Таблица 8. Матрица сложности внешних выходных транзакций и внешних запросов (EO & EQ)

Сложность транзакций	Количество UFP (EI & EQ)	Количество UFP (EO)
Low	3	4
Average	4	5
High	6	7

Таблица 9. Сложность транзакций в не выровненных функциональных точках (UFP)

№	Название	Тип	FTR	DET	Сложность	UFP
1	Форма регистрации	EI	1	10 (Email, пароль, имя, фамилия, пол, дата рождения, город, телефон, и т.д.)	Low	3
2	Авторизация (логин)	EI	1	2 (Email, пароль)	Low	3
3	Просмотр услуг	EO	2	5 (Название услуги, описание, цена, доступность, тип)	Low	4
4	Заказ услуги	EI	2	6 (Услуга, дата заказа, сумма, метод оплаты, контактные данные)	Average	4
5	Личный кабинет	EO	3	8 (Имя, фамилия, email, телефон, история заказов, предпочтения, и т.д.)	High	7
6	Система оплаты	EI	2	4 (Сумма, дата, метод оплаты)	Average	4
7	Контакт с поддержкой	EO	2	5 (Тема запроса, описание, дата, приоритет, статус)	Average	5
8	Отчеты по заказам	EQ	2	6 (Заказ, дата, статус, сумма, услуги)	Average	5
9	Регистрация оплаты	EI	2	4 (Сумма, дата, метод оплаты, статус)	Average	4
10	Продуктовый каталог	EO	3	6 (Название, описание, категория, цена, доступность, дата обновления)	High	7

Таблица 10. Список функциональных точек, связанных с транзакциями

Определение суммарного количества не выровненных

функциональных точек (UFP)

$$UFP = 70 + 46 = 116$$

Определение значения фактора выравнивания (VAF)

Помимо функциональных требований на продукт накладываются общесистемные требования, которые ограничивают разработчиков в выборе решения и увеличивают сложность разработки. Для учета этой сложности применяется фактор выравнивания (VAF). Значение фактора VAF зависит от 14 параметров, которые определяют системные характеристики продукта:

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

Таблица 11. Список значения DI

14 системных параметров (degree of influence, DI) оцениваются по шкале от 0 до 5. Расчет суммарного эффекта 14 системных характеристик (total degree of influence, TDI) осуществляется простым суммированием:

$$TDI = \sum DI = 27$$

Расчет значения фактора выравнивания производится по формуле

$$VAF = (27 \times 0.01) + 0.65 = 0.92$$

Расчет количества выровненных функциональных точек (AFP)

Дальнейшая оценка в выровненных функциональных точках зависит от типа оценки. Начальное оценка количества выровненных функциональных точек для программного приложения определяется по следующей формуле:

$$AFP = UFP \times VAF = 106.72$$

5. COCOMO II

Оценка размера программного продукта в KSLOC

Стек технологий:

- ReactJS (TS / JS)
- Backend (Java)

Разделим функциональность между слоями: - 3/4 frontend и – 1/4 backend.
Подсчитаем размер по KSLOC:

Таблица коэффициентов

$$\begin{aligned}KSLOC &= UFP \times SIZE \\&= (116 \times 3/4 \times 0.060) + (116 \times 1/4 \times 0.053) \\&= 5.22 + 1.537 = 6.757\end{aligned}$$

Оценка уровней факторов масштаба

В методике используются пять факторов масштаба SF_i, которые определяются следующими характеристиками проекта:

- PREC - прецедентность, наличие опыта аналогичных разработок
- FLEX - гибкость процесса разработки
- RESL - архитектура и разрешение рисков
- TEAM - сработанность команды
- PMAT - зрелость процессов

Название фактора	Уровень фактора	Значение уровня
PREC	Very High	1.24
FLEX	Nominal	3.04
RESL	High	2.83
TEAM	High	2.19
PMAT	High	3.12

Таблица 12. Значение фактора масштаба, в зависимости от оценки его уровня

Оценка уровней множителей трудоемкости

Для предварительной оценки проекта необходимо оценить уровень семи множителей трудоемкости M:

- PERS - квалификация персонала

- RCPX - сложность и надежность продукта
- RUSE - разработка для повторного использования
- PDIF - сложность платформы разработки
- PREX - опыт персонала
- FCIL - оборудование
- CSCE - требуемое выполнение графика работ

Название фактора	Уровень фактора	Значение уровня
PERS	Extra High	0.50
RCPX	Very High	1.91
RUSE	Nominal	1.00
PDIF	Nominal	1.00
PREX	Extra High	0.62
FCIL	Very High	0.73
SCED	Nominal	1.00

Таблица 13. Значения множителей трудоемкости, в зависимости от оценки их уровня

Оценка трудоемкости проекта

$$PM = A \times SIZE^E \times \prod_{i=1}^n EM_i \quad A = 2.94$$

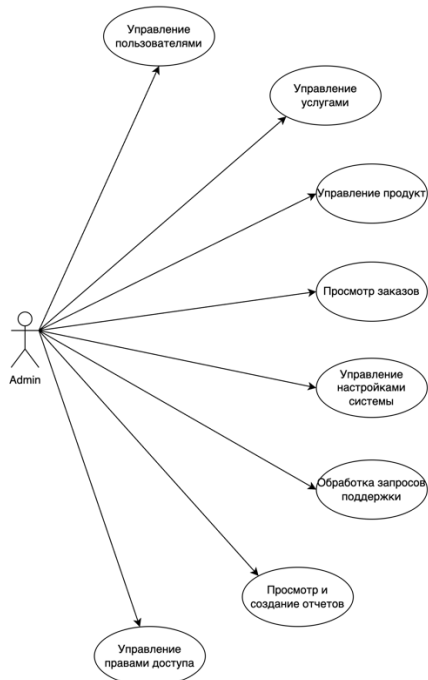
$$E = B + 0.01 \times \sum_{j=1}^n SF_j \quad B = 0.91$$

- $SIZE$ — размер продукта в KSLOC
- EM_i — множители трудоемкости
- SF_j — факторы масштаба
- $n = 7$ — для предварительной оценки
- $n = 17$ — для детальной оценки

$$E = 0.91 + 0.01 \times 12.42 = 0.91 + 0.1242 = 1.0342$$

$$PM = 2.94 \times 7.024 \times 0.5533 = 11.597 \text{ человеко-мес} = 1855.52$$

6. Use Case Points



$$UCP = TCP \cdot ECF \cdot UUCP \cdot PF$$

1. Technical Complexity Factor (TCF).
2. Environment Complexity Factor (ECF).

3. Unadjusted Use Case Points (UUCP).

4. Productivity Factor (PF).

Необходимые шаги для создания оценки на основе метода UCP:

- Определение и вычисление технических факторов.
- Определение и вычисление экологических факторов.
- Вычисление нескорректированных точек вариантов использования.
- Определение фактора производительности.
- Вычисление произведения переменных.

Technical Complexity Factors:

Технический фактор	Описание	Масса	Воспринимаемая сложность	Рассчитанный фактор
T1	Распределенная система	2	3	6
T2	Производительность	1	4	4
T3	Эффективность для конечного пользователя	1	3	3
T4	Сложная внутренняя обработка	1	4	4
T5	Возможность повторного использования	2	1	2
T6	Легко установить	0.5	4	2
T7	Легко использовать	0.5	4	2
T8	Портативный	2	3	6
T9	Легко изменить	1	2	2
T10	Параллельный	1	3	3
T11	Специальные функции безопасности	1	3	3
T12	Предоставляет прямой доступ третьим лицам	1	4	4
T13	Требуются специальные условия для обучения пользователей.	1	1	1
			Общий фактор	42

$$TCF = 0.6 + (0.01 \times \text{Общий фактор}) = 1.02$$

Environmental Complexity Factors

Фактор	Описание	Масса	Воспринимаемая сложность	Рассчитанный фактор
E1	Знакомство с UML	1	4	4
E2	Опыт применения	1	3	3
E3	Объектно-ориентированный опыт	1	4	4
E4	Возможности ведущего аналитика	1	5	5
E5	Мотивация	0.5	3	1.5
E6	Требования к стабильности	1	5	5
E7	Неполный рабочий день	0.5	2	1
E8	Сложный язык программирования	1	3	3
			Общий фактор	26.5

$$ECF = 1.4 + (-0.03 \times \text{Общий фактор}) = 0.605$$

Unadjusted Use Case Points (UUCP)

Unadjusted Use Case Weight (UUCW):

Тип варианта использования	Описание	Масса	Количество вариантов использования	Результат
Простой	Простой пользовательский интерфейс, затрагивающий только один объект базы данных; его успешный сценарий состоит из 3 шагов или менее; его реализация включает менее 5 классов.	5	12	60
Средний	Больше дизайна интерфейса, затрагивающего 2 или более сущностей базы данных; от 4 до 7 шагов; его реализация включает от 5 до 10 классов.	10	5	50

Сложный	Включает в себя сложный пользовательский интерфейс или обработку и затрагивает 3 или более сущностей базы данных; более семи шагов; его реализация включает более 10 классов.	15	1	15
			Общий UUCW	125

Unadjusted Actor Weight (UAW):

Тип актера	Описание	Масса	Количество актеров	Результат
Простой	Актер представляет собой еще одну систему с определенным API.	1	1	1
Средний	Актер представляет собой еще 2 системы, взаимодействующие через протокол, например TCP/IP.	2	0	0
Сложный	Актер — это человек, взаимодействующий через интерфейс.	3	2	6
			Общий UAW	7

$$UUCP = UUCW + UAW = 125 + 7 = 132$$

Productivity Factor:

$$PF = 20$$

$$\begin{aligned}
 UCP &= TCP \times ECF \times UUCP \times PF \\
 &= 1.02 \times 0.605 \times 132 \times 20 \\
 &= 1629.144
 \end{aligned}$$

Анализ результатов

Метод	Затраты (h-h)
Наивный	1400
PERT	1884
Функциональных точек	1650
COCOMO II	1856
UCP	1630

Вывод

С помощью 5 методов расчета трудоемкости создания сайта kaggle.com, хотя 5 метода дают разные результаты, все они показывают, что проект средний, поэтому вопросы задавайте часто, время строительства большое, поэтому строительный блок. необходимо иметь оптимальную стратегию строительства, а также подготовить ресурсы для завершения строительства.