

Федеральное государственное автономное образовательное
учреждение высшего образования

Университет ИТМО

Дисциплина: Экономика программной инженерии

Лабораторная работа 1

Вариант <https://www.khanacademy.org>

Выполнили:

Брудар.А

Артемкин.А

Группа: Р34131

Предводитель:

Клименков Сергей Викторович

2023 г.

Санкт-Петербург

Оглавление

Задание	4
Вариант	4
Выполнение	4
Наивный метод	4
PERT метод	8
Метод критического пути	10
Метод функциональных точек	12
Определение типа оценки	12
Определение области оценки и границ продукта	12
Подсчет функциональных точек, связанных с данными	12
Подсчет функциональных точек, связанных с транзакциями	13
Определение суммарного количества не выровненных функциональных точек (UFP)	15
Определение значения фактора выравнивания (FAV)	15
Расчет количества выровненных функциональных точек (AFP)	16
COCOMO II	17
Оценка размера программного продукта в KSLOC	17
Оценка уровней факторов масштаба	17
Оценка уровней множителей трудоемкости	17
Оценка трудоемкости проекта	18
Use Case Points	19
Use Case диаграмма для пользователя:	19
Use Case диаграмма для суперадмина:	20
Оценка веса прецедентов	20
Оценка веса акторов	21
Оценка веса технических факторов	21
Оценка веса факторов окружения	22
Подсчет UCP	22
Подсчёт фактора продуктивности (PF) на основе прошлого проекта	22
Список UseCase-ов:	23
Оценка веса прецедентов	23
Оценка веса акторов	24
Оценка веса технических факторов	24
Оценка веса факторов окружения	25
Подсчет UCP	25
Подсчет трудоемкости проекта:	25
Анализ результатов	26

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.

Вариант

<https://www.khanacademy.org>

Выполнение

Функциональные требования

1. Система должна предоставлять возможность просмотра раздела “courses”, включающего:
 - a. Курсы по математике от дошкольного уровня до 8 класса.
 - b. Подготовительные курсы от дошкольного уровня до 8 класса, а также к экзаменам по алгебре и геометрии.
 - c. Курсы по математике для средней школы и колледжа, включающего:
 - i. Курсы по математике от дошкольного уровня до 8 класса.
 - ii. Подготовительные курсы от дошкольного уровня до 8 класса, а также к экзаменам по алгебре и геометрии.
 - iii. Курсы по математике для средней школы и колледжа, включающего:

1. 2 части Алгебре
2. Геометрию
3. 3 части интегрированной математики
4. Основы алгебры
5. Тригонометрию
6. Статистика и вероятность
7. Многомерные исчисления
8. Дифференциальные уравнения
9. Линейную алгебру

d. Подготовку к тестированию с разделами:

- i. Digital SAT
- ii. LSAT
- iii. Praxis Core
- iv. MCAT

e. Раздел “Наука”, включающая:

- i. Биологию средней школы
- ii. Среднюю школу наук о Земле и космосе
- iii. Физику средней школы
- iv. Биологию старшей школы
- v. Биологию старшей школы - NGSS
- vi. Физику старшей школы
- vii. Физику старшей школы - NGSS
- viii. AP/College Biology
- ix. AP/College Chemistry
- x. AP/College Environmental Science
- xi. Health and Medicine
- xii. See all Science

f. Раздел “Вычислительная техника”, включающая:

- i. Компьютерное программирование
- ii. Принципы компьютерных наук в колледже
- iii. Pixar in a Box
- iv. See all Computing

g. Раздел “Искусство и гуманитарные науки”:

- i. История США
- ii. Колледж истории США
- iii. Правительство и гражданское право США

- iv. Колледж правительства и политики США
- v. Проект по всемирной истории - от истоков до настоящего времени
- vi. Проект по всемирной истории - с 1750 года по настоящее время
- vii. Всемирная история колледжа
- viii. Большая история
- ix. История искусств
- x. История искусств колледжа
- h. Раздел "Экономика":
 - i. Макроэкономика
 - ii. Микроэкономика
- i. Раздел "Чтение и языковое искусство с дошкольного уровня до 9 класса.
- j. Раздел "Жизненные навыки":
 - i. Социальное и эмоциональное обучение
 - ii. Искусственный интеллект для образования
 - iii. Финансовая грамотность
 - iv. Безопасность в интернете.
 - v. Грамотность в социальных сетях
 - vi. Growth mindset
 - vii. Прием в колледж
 - viii. Карьера
 - ix. Личные финансы
- k. Раздел "Партнерские курсы":
 - i. Биоразнообразие
 - ii. Музыка
 - iii. NASA
 - iv. NOVA Labs
 - v. Философия

2. Система должна поддерживать интерфейс поиска по всему сайту
3. Система должна поддерживать форму регистрации и авторизации
4. Система должна иметь возможность электронной оплаты и форму для донатов
5. Система должна иметь три роли зарегистрированных пользователей:

- a. Обучающийся
 - b. Учитель
 - c. Родитель
6. Система должна поддерживать возможность прохождения тестов, а также записывать все результаты в личный профиль обучающегося.
7. Система должна поддерживать возможность смены языка интерфейса и набора курсов в зависимости от местоположения входа в систему.
8. Система должна поддерживать раздел “О проекте”, включающий:
- a. Новости
 - b. Влияние
 - c. Наша команда
 - d. Наши стажеры
 - e. Наши эксперты
 - f. Наше руководство
 - g. Нас поддерживают
 - h. Наши участники
 - i. Наши финансы
 - j. Вакансии
 - k. Практика
9. Система должна поддерживать раздел “Контакты”, включающий:
- a. Центр поддержки
 - b. Помощь сообществу
 - c. Поделиться своей историей
 - d. Для прессы
10. Система должна поддерживать раздел “Курсы”, включающий:
- a. Математика
 - b. Естественные науки
 - c. Экономика и финансы
 - d. Информатика
 - e. Искусство и гуманитарные науки
11. Система должна поддерживать в роли Учителя добавление учеников через:
- a. Google Classroom
 - b. Ссылкой на электронную почту
12. Система должна поддерживать в роли Учителя добавление классов через:

- a. Создание нового класса путем ввода имени класса
 - b. Импортирования класса из Google Classroom
13. Система должна поддерживать в роли Учителя добавление материалов к курсу
14. Система должна поддерживать в роли Учителя панель управления классом:
- a. Назначение дедлайна по заданиям.
 - b. Создание заданий.
 - c. Отслеживание прогресса по курсу
 - d. Отслеживание проектов по курсы, которыми поделились ученики
 - e. Изменение имени класса
 - f. Изменение кода класса
 - g. Изменение курса
 - h. Удаление класса
15. Система должна поддерживать в роли Ученика:
- a. Выбор своего текущего уровня знаний(grade)
 - b. Выбор желаемых курсов
 - c. Вход в класс
 - d. Редактирование профиля
 - e. Отслеживание прогресса
 - f. Просмотр текущий учителей
16. Система должна поддерживать в роли Родителя:
- a. Добавление детей
 - b. Отслеживание прогресса обучения своих детей
 - c. Удаление детей
 - d. Изменение своего профиля
 - e. Добавление курсов себе и детям

Наивный метод

#	Название	Optimistic (h-h)	Pessimistic (h-h)	Optimal (h-h)
1	Просмотр курсов	100	250	170

2	Интерфейс поиска информации по всему сайту	8	32	16
3	Форма авторизации и регистрации	15	45	30
4	Электронная оплаты и форму для донатов	30	90	60
5	Написание трех ролей пользователей сайта	15	45	30
6	Прохождение тестов	50	150	80
7	Мультиязычность и разные курсы для разных регионов	30	50	40

8	О проекте	80	200	140
9	Контакты	50	100	80
10	Курсы	500	1000	800
11	Добавление учеников	50	100	80
12	Добавление классов	50	100	70
13	Добавление материалов к курсу	20	40	30
14	Панель управления классом	80	120	100
15	Роль ученика	100	150	120
16	Роль родителя	60	100	80

ИТОГО	1926
-------	------

PERT метод

#	Название	Optimistic (h-h)	Pessimisti с (h-h)	Optimal (h-h)	$E_i = \frac{(P+O+4M)}{6}$	$CKO = \frac{(P-O)}{6}$
1	Просмотр курсов	100	250	170	171,67	25,00
2	Интерфейс поиска информации по всему сайту	8	32	16	17,33	4,00
3	Форма авторизации и регистрации	15	45	30	30,00	5,00
4	Электронная оплата и форму для донатов	30	90	60	60,00	10,00
5	Написание трех ролей пользователей сайта	15	45	30	30,00	5,00
6	Прохождение тестов	50	150	80	86,67	16,67
7	Мультиязычность и разные курсы для разных регионов	30	50	40	40,00	3,33
8	О проекте	80	200	140	840	20,00
9	Контакты	50	100	80	78,33	8,33
10	Курсы	250	500	300	325,00	41,6 7
11	Добавление учеников	50	100	80	78,33	8,33
12	Добавление классов	50	100	70	71,67	8,33
13	Добавление материалов к	20	40	30	30,00	3,33

	курсу					
14	Панель управления классом	80	120	100	100,00	6,67
15	Роль ученика	100	150	120	121,67	8,33
16	Роль родителя	60	100	80	80,00	6,67

$$E = \sum_i E = 2160.00$$

$$CKO = \text{КОРЕНЬ}(\sum CKO)^2 = 59.95$$

$$E_{95\%} = E + 2 * CKO = 2280.56$$

E_i - оценка средней трудоемкости

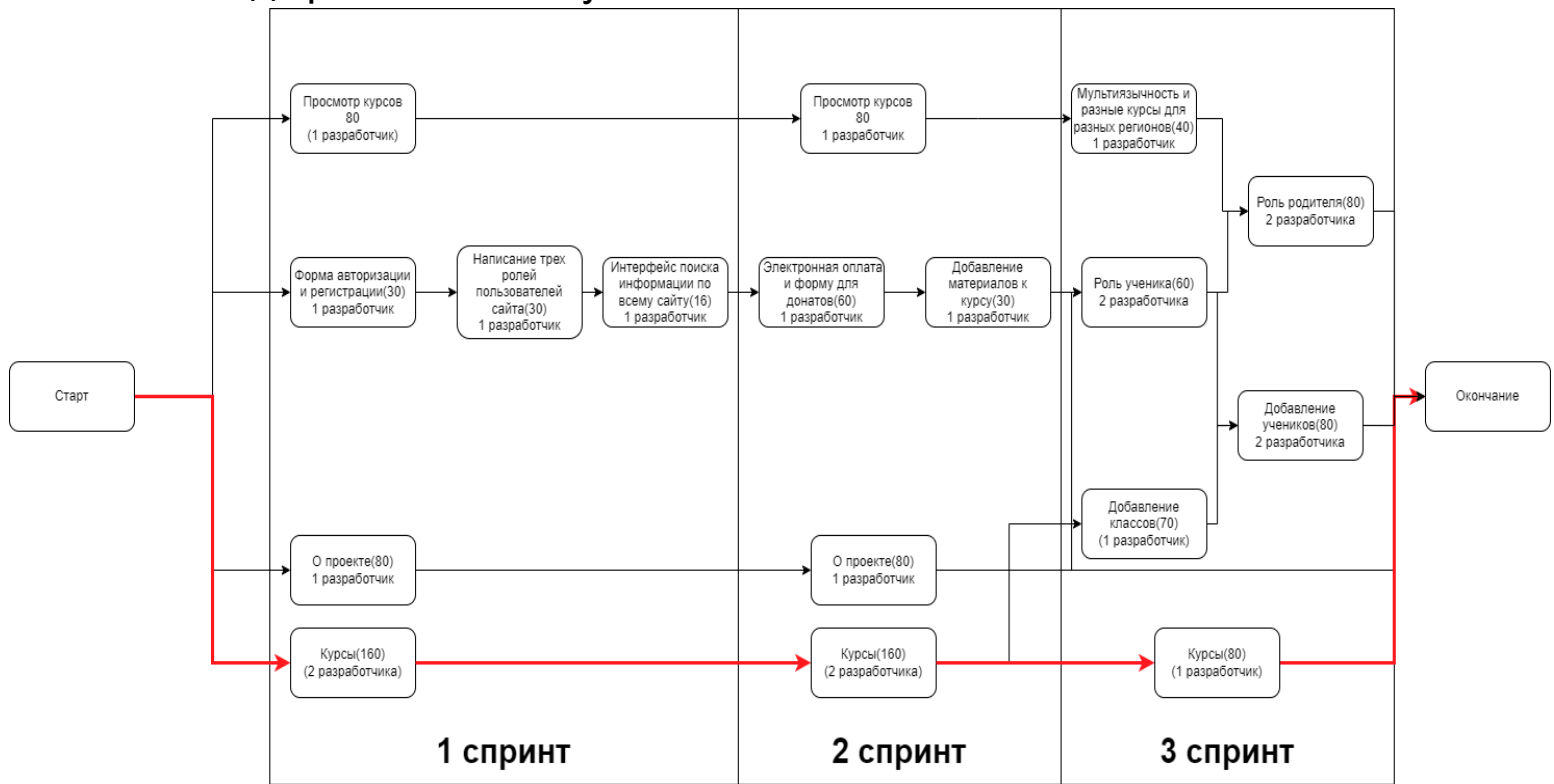
E - общая оценка статически независимых работ

$СКО_i$ - среднеквадратичное отклонение

$СКО$ - среднеквадратичное отклонение для оценки суммарной трудоемкости

$E_{95\%}$ - суммарная трудоемкость проекта (с вероятностью 95%)

Метод критического пути



[Full](#)

Критический путь: 400 ч./ч.

Выполнение проекта: при ориентире на минимальное время разработки (критический путь) получаем, что для выполнения нам необходимо 400 ч./ч.

Команда:

- 5x Frontend-разработчик
- 5x Backend-разработчик
- 5x Аналитики-разработчик

Рабочий день считаем: 8 часов

Время для тестирования: 80 часов

Время для отладки и фикса багов: 40 часов

Время для деплоя и подготовки к показу: 40 часов

Время разработки: $400 + 80 + 40 + 40 = 560$ часов

Общее время: $560/8 = 70$ рабочих дней

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
2-5 RET	Low	Average	High
6+ RET	Average	High	High

№	Название	RET	DET	Сложность	UFP
1	Личный кабинет	Личная информация (1)	имя, фамилия, пол, дата рождения, город, телефон (7)	Low	7
2	Регистрация	Данные входа, личная информация (2)	Email, пароль, имя, фамилия, пол, дата рождения, город, телефон (9)	Low	7
3	вход	данные для авторизации(1)	email, пароль(2)	Low	7
4	Электронная оплата	Информация по карте(3)	Номер карты, владелец, дата, CVC (4)	Low	7

Подсчет функциональных точек, связанных с транзакциями

№	Название	Тип	FTR	DET	Сложность	UFP
1	Форма обратной связи	EI	1	4	Low	3
2	Форма регистрации	EI	1	10	Low	3
3	Форма регистрации	EI	1	10	Low	3
4	Просмотр профиля	EQ	1	8	Low	3
5	Поиск	EQ	3	1	Low	3
6	Просмотр курсов	EO	0	1	Low	4
7	Просмотр материалов	EQ	0	1	Low	3
8	Просмотр статических страниц	EI	1	1	Low	3
9	Информация о классе	EO	3	1	Low	4

Определение значения фактора выравнивания (VAF)

№	Параметр	Вес (DI)
1	Обмен данными	2
2	Распределенная обработка данных	0
3	Производительность	0
4	Ограничения по аппаратным ресурсам	0
5	Транзакционная нагрузка	0
6	Интенсивность взаимодействия с пользователем	3
7	Эргономика	1
8	Интенсивность изменения данных	1
9	Сложность обработки	0
10	Повторное использование	1
11	Удобство инсталляции	0
12	Удобство администрирования	2
13	Портируемость	1
14	Гибкость	0
$TDI = \sum DI = 12$ $VAF = (TDI * 0.01) + 0.65 = 0.77$		

DI (degree of influence)

TDI (total degree of influence)

Расчет количества выровненных функциональных точек (AFP)

$$AFP = UPF \times VAF = 73 * 0.77 = 56.21$$

Метод анализа функциональных точек (FPA) не предоставляет информации о трудоемкости разработки оцененного продукта. Этот вопрос может быть легко решен, если у разработчика есть своя статистика трудозатрат на реализацию функциональных точек. В случае отсутствия такой статистики для оценки трудоемкости и сроков проекта часто используется метод COSOMO II.

COCOMO II

Оценка размера программного продукта в KSLOC

Стек технологий:

- React (TS / JS)
- Backend (Node.js)

Разделим функциональность между слоями:- $\frac{1}{2}$ frontend и $\frac{1}{3}$ backend. Подсчитаем

размер по KSLOC:

$$KSLOC = UFP * SIZE = (78 * \frac{1}{2} \times 0.060) + (78 * \frac{1}{3} \times 0.053) = 3.023$$

Оценка уровней факторов масштаба

- 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	High	2.48
FLEX	High	3.32
RESL	Low	4.24
TEAM	High	2.32
PMAT	Very Low	6.34

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
- CSED - требуемое выполнение графика работ

Название	Уровень	Значение
PERS	Nominal	1.00
RCPX	Very Low	0.83
RUSE	Low	0.91
PDIF	Low	0.81
PREX	High	1.00
FCIL	Nominal	0.92
CSED	Nominal	1.00

Оценка трудоемкости проекта

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

$$A = 2,94$$

$$E = B + 0,01 \times \sum_{j=1}^5 SF_j$$

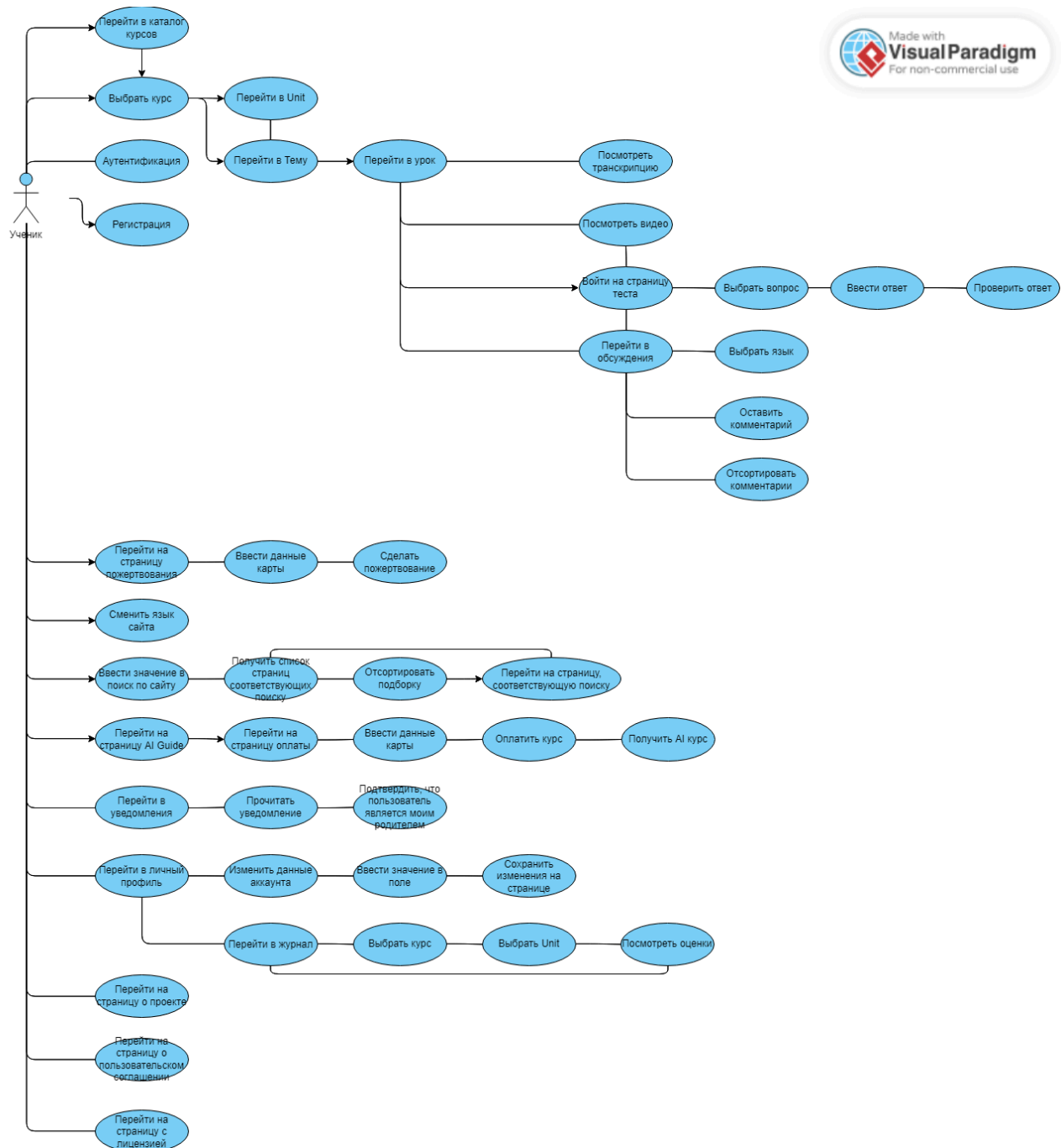
$$B = 0,91$$

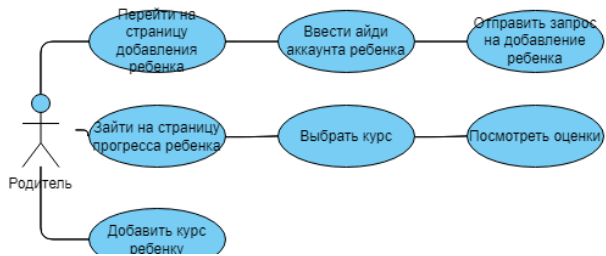
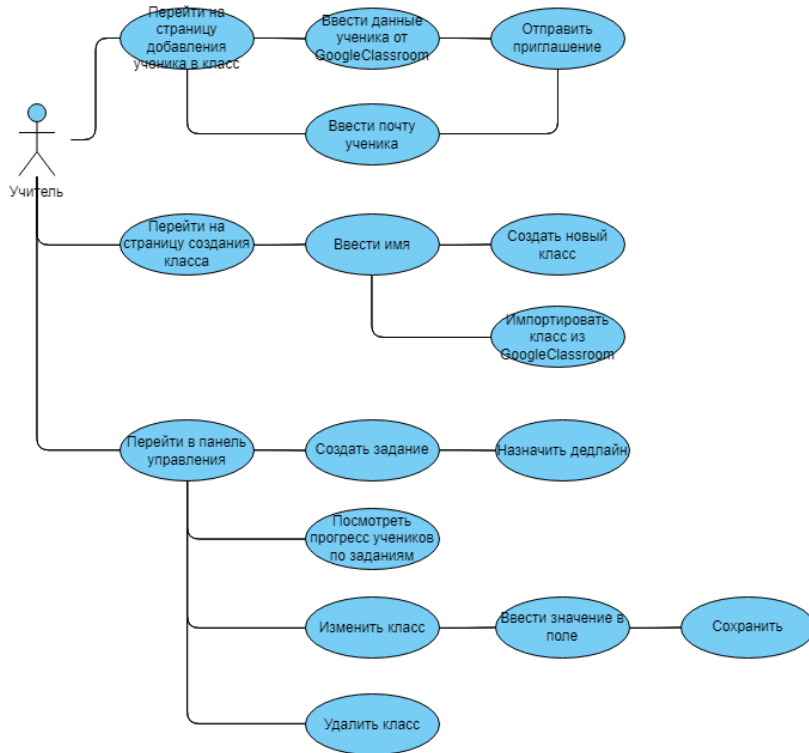
- SIZE — размер продукта в KSLOC
- EM_i — множители трудоемкости
- SF_j — факторы масштаба
- $n=7$ — для предварительной оценки
- $n=17$ — для детальной оценки

$$E = 0.92 + 0.01 \times (2.28 + 2.32 + 5.32 + 2.19 + 7.80) = 1.1115$$

$$PM = 2.94 \times 4.5435^{1.1115} \times (1.00 * 0.83 * 0.91 * 0.81 * 1.00 * 0.92 * 1.00) \approx 2456.12 \text{ ч. /ч.}$$

Use Case Points





Оценка веса прецедентов

Сложность	Вес (UUCW)	Количество	Затраты
Low	5	50	250
Medium	10	21	210
High	15	1	15
Нескорректированный вес варианта использования (UUCW)			475

Оценка веса акторов

Сложность	Вес (AUW)	Количество	Затраты
Low	1	0	0
Medium	2	0	0
High	3	3	9
Масса актера без корректировки (UAW)			9

Оценка веса технических факторов

Фактор	Вес (W)	Номинальная стоимость(F)	Затраты
Распределенная система	2	3	6
Время отклика или целевые показатели производительности	1	4	4
Эффективность для конечного пользователя	1	4	4
Комплексная внутренняя обработка	1	2	2
Переиспользуемость кода	1	3	3
Простота установки	0.5	4	2
Простота использования	0.5	3	1.5
Портативность	2	2	4
Простота изменений	1	3	3
Многопоточность	1	4	4
Особые требования к безопасности	1	4	4
Доступ для других систем	1	3	3

Необходимы специальные средства для обучения пользователей	1	2	2
TFactor			42.5
$TCF = 0.6 + (TFactor/100)$			1,025

Оценка веса факторов окружения

Фактор	Вес (W)	Значение(F)	Взвешенное значение
Знаком с моделью проекта, которая используется	1.5	2	3
Опыт применения	0.5	1	0,5
Опыт в разработке онлайн платформ для обучения	1.0	0	0
Возможность ведущего аналитика	0.5	0	0
Мотивация	1.0	2	2
Стабильные требования	1.5	2	3
Персонал, занятый неполный рабочий день	-1.0	4	-4
Сложность языка программирования	-1.0	2	-2
Общий фактор окружающей среды (EFactor)			2,5
ECF = 1.4 + (-0.03 * EF)			1.325

Подсчет UCP

$$UCP' = (UCW + UAW) * TCF * ECF = (475 + 9) * 1.025 * 1.325 = 487$$

Подсчёт фактора продуктивности (PF) на основе прошлого проекта

В качестве примера мы выбрали проект по БЛПС, в которую входит выполнение всех лабораторных работ для двух человек.

Список UseCase-ов:

Оценка веса прецедентов

Сложность	Вес (UUCW)	Количество	Затраты
Low	5	8	40
Medium	10	0	0
High	15	0	0
Нескорректированный вес варианта (UUCW)			40

Оценка веса акторов

Сложность	Вес (AUW)	Количество	Затраты
Low	1	1	1
Medium	2	0	0
High	3	1	3
Масса актера без корректировки (UAW)			4

Оценка веса технических факторов

Фактор	Вес (W)	Номинальная стоимость(F)	Затраты
Распределённость	2.0	5	10
Производительность	1.0	1	1.0
Эффективность для пользователя	1.0	1	2.0
Сложная внутренняя обработка	1.0	0	1.0
Повторное использование кода	2.0	1	2.0
Простота установки	0.5	1	0.5
Простота использования	0.5	0	0
Переносимость	2.0	1	2.0
Простота изменений	2.0	3	6.0
Многopotочность	1.0	0	0
Дополнительные возможности безопасности	1.0	5	5

Доступ к другим системам	1.0	0	0
Необходимы тренажеры для пользователей	1.0	1	1.0
Общий технический фактор (TFactor)			29.5
TCF = 0.6 + (TF/100)			0.895

Оценка веса факторов окружения

Фактор	Вес (W)	Номинальная стоимость(F)	Затраты
Знаком с моделью проекта, которая используется	1.5	3	4.5
Опыт применения	0.5	2	1
Опыт в веб разработке	1.0	1	1
Возможность ведущего аналитика	0.5	1	0
Мотивация	1.0	2	2
Стабильные требования	1.5	1	1.5
Частичная занятость	-1.0	5	-5
Сложность языка программирования	-1.0	3	-3
Общий фактор окружающей среды (EFactor)			2
ECF = 1.4 + (-0.03 * EF)			1.34

Подсчет UCP

$$UCP' = (UCW + UAW) * TCF * ECF = 44 * 0.895 * 1.34 = 53$$

Подсчет трудоемкости проекта:

Предыдущая работа была выполнена за 125 часов на 2-ух человек

$$PF = UCP/UCP' = 2.35$$

$UCP' = 487$ - для нашего сайта

$UCP = PF * UCP' = 1\,144$ ч/ч + работа с авторскими правами + первоначальное наполнение сайта = $1\,144 + 225 + 200 = 1569$ ч/ч

Анализ результатов

Метод	Затраты (h-h)
Наивный	1926
PERT	2280
COCOMO II	2456
UCP	1569
Метод критического пути	2820

Как можно видеть, временные затраты разнятся между 1,5 и 2,8 тысячами человекочасов на выполнение проекта. Минимальная оценка - UCP обусловлена тем, что базовой единицей расчета берется use-case. Чтобы узнать их все, необходимо попробовать все возможные варианты использования, включая платные, на что у нас пока нет средств. Также не все use-case-ы, которые выглядят одинаково, потребуют одинаковое кол-во часов исполнения.

Вывод

В процессе выполнения лабораторной работы мы узнали о нескольких методах прогнозирования временных затрат на выполнение проекта и поупражнялись в их применении на реальном проекте.