

Университет ИТМО

Факультет программной инженерии и компьютерной техники
Направление подготовки 09.03.04 Программная инженерия
Дисциплина «Экономика программной инженерии»

Отчет

По лабораторной работе №1

Выполнили:

Шабалов Антон

Крисанов Роман

Преподаватель:

Машина Е.А.

Санкт-Петербург, 2023 г.

Содержание

Задание.....	3
Функциональные требования.....	3
Оценка трудоемкости работ.....	4
Сетевая диаграмма взаимосвязи работ и критический путь.....	7
Оценка размера проекта методом функциональных точек.....	8
Расчет трудоемкости методом COSOMO II.....	11
Оценка с помощью User Case Points.....	13
Оценка исходного проекта.....	13
Неадаптированный вес участника (UAW).....	13
Нескорректированный вес варианта использования (UUCW)....	13
Фактор технической сложности (TCF).....	14
Фактор сложности окружающей среды (ECF).....	14
Точки варианта использования (UCP).....	15
Оценка стороннего завершенного проекта.....	16
Список прецедентов проекта.....	16
Неадаптированный вес участника (UAW).....	16
Нескорректированный вес варианта использования (UUCW)....	16
Фактор технической сложности (TCF).....	17
Фактор сложности окружающей среды (ECF).....	17
Точки варианта использования (UCP).....	17
Расчет фактора продуктивности и итогового UCP.....	18
Сравнение использованных методов.....	18
Вывод.....	19

Задание

Вариант задания: <https://varlamov.ru/>

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

Функциональные требования

1. Система должна предоставлять возможность авторизации на сайте
2. Система должна предоставлять возможность комментировать/оценивать следующий контент на сайте
 - 2.1. Новости
 - 2.2. Комментарии как свои так и других пользователей
3. Система должна иметь возможность подать жалобу на любой контент сайта
4. Система должна предоставлять возможность поиска контента по части содержимого
 - 4.1. Название статьи или его часть
 - 4.2. Часть статьи
 - 4.3. Хештеги
5. Система должна предоставлять различные подборки новостей по разной тематике
6. Система должна предоставлять возможность перехода на другие информационные ресурсы автора
7. Система должна предоставлять возможность пожертвовать материальные средства автору
8. Система должна собирать информацию о поведенческих факторах пользователя для каждой новости и предоставлять часть этой информации читателю а часть использовать для генерации подборок.
9. Система должна иметь возможность публикации каких либо новостей в других информационных ресурсах

9.1. фейсбук

9.2. вконтакте

9.3. твитер

9.4. телеграмм

9.5. одноклассники

9.6. Вотсап

10. Возможность отправки писем автору

Оценка трудоемкости работ наивным методом

№	Функционал	Оценка, мин. /чел. час	Оценка, вероятн. /чел. час	Оценка, макс. /чел. час
1	Авторизация на сайте	20	24	28
2	Возможность комментировать/оценивать ресурсы	24	30	36
3	Возможность подавать жалобы на ресурсы+ модерация	20	24	28
4	Возможность поиска по записям	40	50	60
5	Предоставление различных подборок новостей	20	25	30
6	Возможность делать пожертвования автору	16	20	24
7	Возможность делиться записью	8	12	14
8	Возможность отправлять письма автору	4	4	8
9	Разработка главной страницы	16	20	24
10	Разработка страницы с новостью	8	12	16
11	Разработка страницы новости	8	12	16
12	Разработка всех второстепенных страниц	16	20	24
13	Тестирование	30	36	42
14	SSL certificate	4	6	8

Средняя трудозатратность 295 ч/ч

Pert метод

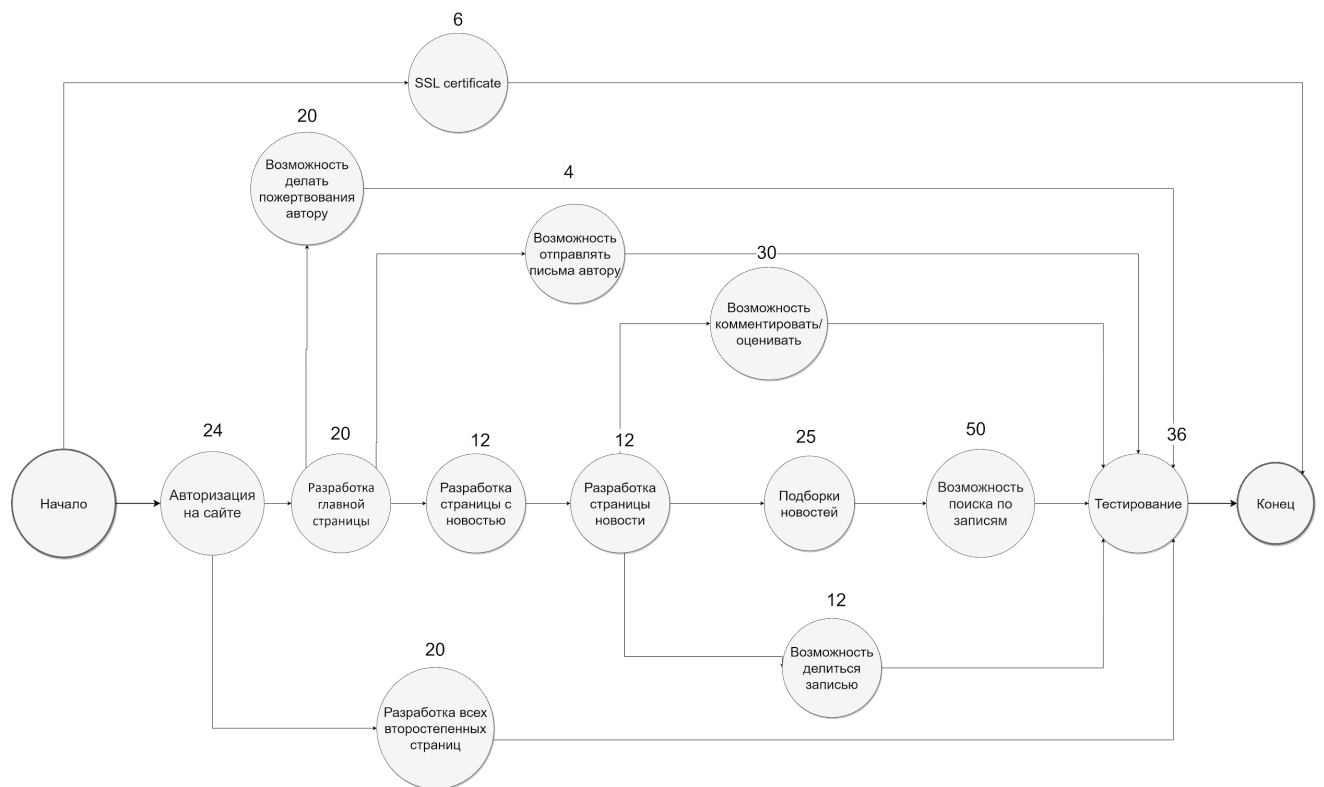
№	Функционал	Оценк а, мин. /чел. час	Оценк а, вероя тн. /чел. час	Оценк а, макс. /чел. час	Средн я трудо емкос ть	СКО
1	Авторизация на сайте	20	24	28	24,00	1,33
2	Возможность комментировать/оценивать ресурсы	24	30	36	30,00	2,00
3	Возможность подавать жалобы на ресурсы+ модерация	20	24	28	24,00	1,33
4	Возможность поиска по записям	40	50	60	50,00	3,33
5	Предоставление различных подборок новостей	20	25	30	25,00	1,67
6	Возможность делать пожертвования автору	16	20	24	20,00	1,33
7	Возможность делиться записью	8	12	14	11,67	1,00
8	Возможность отправлять письма автору	4	4	8	4,67	0,67
9	Разработка главной страницы	16	20	24	20,00	1,33
10	Разработка страницы с новостью	8	12	16	12,00	1,33
11	Разработка страницы новости	8	12	16	12,00	1,33
12	Разработка всех второстепенных страниц	16	20	24	20,00	1,33
13	Тестирование	30	36	42	36,00	2,00
14	SSL certificate	4	6	8	6,00	0,67

E=295,33

СКО= 6,02

E(95)=307,37

Сетевая диаграмма взаимосвязи работ и критический путь



Критический путь- 179 ч/ч

Весь путь- 295 ч/ч

Для разработки данного сайта в достаточно 2 разработчика (не джуны)

Таким образом проект будет выполнен $(141/8) = 18$ дней на разработку критического пути одним разработчиком

Разработка распараллеливаемые частей $62/8 = 8$ дней

Тестирование $36/16=3$

Таким образом можно сказать что распараллеливающая часть и критический путь будет выполнена за 18 дней, один разработчик будет делать критический путь другой параллеливаемую часть.

И в конце вместе будут тестировать по 3 дня

Итого 21 день.

Оценка размера проекта методом функциональных точек

Определение типа оценки - Продукт. Оценивается объем уже существующего и установленного продукта.

Определение области оценки и границ продукта-Только функции, реально используемые

Подсчет функциональных точек, связанных с данными

- DET (data element type) — неповторимое уникальное поле данных, например, Имя Клиента — 1 DET; Адрес Клиента (индекс, страна, область, район, город, улица, дом, корпус, квартира) — 9 DET's
- RET (record element type) — логическая группа данных, например, 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

№	Тип информации	RET	DET	Сложность	UFP
1	форма комментария	Личная информация, дата, комментарий(3)	Имя, фамилия, дата, комментарий(4)	low	7
2	личный кабинет	Личная информация, информация об аккаунте, информация о статье(3)	Имя, фамилия, подписчики, подписки, статьи, название статей, почта (6)	low	10
3	новость	Новость(1)	Название, содержание (2)	low	3

4	регистрация	Личная информация, данные для авторизации(2)	Фамилие, имя, тег, пароль (4)	low	6
5	вход	данные для авторизации(1)	тег, пароль(2)	low	3

Подсчет функциональных точек, связанных с транзакциями

№	Название	Сложность	тип	UFP
1	Добавление комментария	Средняя	EI	4
2	Открытие новости	Высокая	EO	6
3	Вход в систему	Низкая	EI	3
4	Регистрация	Низкая	EI	3
5	Жалоба на контент	Низкая	EI	3
6	Выдача подборки новостей	Высокая	EO	6
7	Поиск по записям	Высокая	EO	6
8	Вносить пожертвования	Низкая	EI	3
9	отправлять письма автору	Низкая	EI	3

Определение значения фактора выравнивания (FAV)

Обмен данными	2
Распределенная обработка данных	0
Производительность	0
Ограничения по аппаратным ресурсам	0

Транзакционная нагрузка	0
Интенсивность взаимодействия с пользователем	1
Эргономика	1
Интенсивность изменения данных (ILF) пользователями	1
Сложность обработки	0
Повторное использование	0
Удобство инсталляции	5
Удобство администрирования	0
Портируемость	2
Гибкость	0

TDI= 12

VAF=(TDI*0,01)+0.65=0,77

Расчет количества выровненных функциональных точек (AFP)

AFP=VAF*UFP=73*0,77 = 56,21

Данная информация нам ничего не дает, так как она нужна, когда у нас уже есть аналитика по работе нашей компании, в нашем случае ее нет, для расчета используем СОСОМО II

Расчет трудоемкости методом СОСОМО II

Оценка размера программного продукта в KSLOC

Стек технологий:

- React (JS)
- Backend (php)

Разделим функциональность между слоями: 1/2 - frontend и 1/2 - backend.

Подсчитаем размер по KSLOC:

$KSLOC=UFP*SIZE=(73 \times 1/2 \times 0.047)+(73 \times 1/2 \times 0.037)=3.066$

Факторы масштаба

Факторы масштаба	
PREC	2.48
FLEX	5.07
RESL	0.00
TEAM	4.38
PMAT	4.68

Оценка уровней множителей трудоемкости

Трудоемкость	
PERS	1
PREX	0,87
RCPX	0,6
RUSE	0,95
RDIF	1
FCIL	0,73
SCED	1

Оценка многокомпонентного продукта

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

$$A = 2,94$$

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

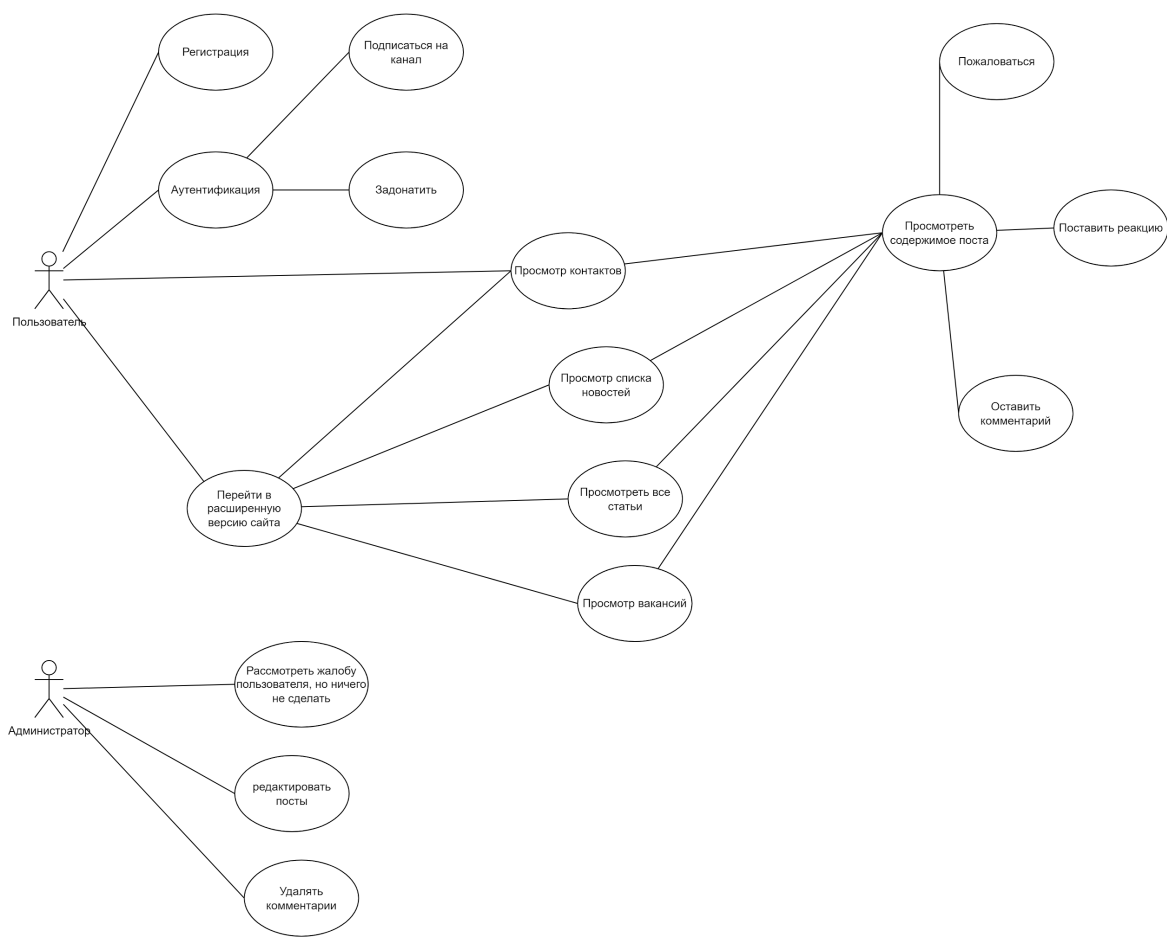
$$B = 0,91$$

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

$$E = 0,91 + 0,01 \times (2,48 + 5,07 + 4,38 + 4,68) = 1,0761$$

$$PM = 2,94 \times 3.066^{1,0761} \times (1 \times 0,87 \times 0,6 \times 0,95 \times 1 \times 0,73 \times 1) = 1.7035 \text{ ч/м} = 272.56 \text{ ч/ч}$$

Оценка с помощью User Case Point



Оценка исходного проекта

Точки варианта использования (UCP)

Точки варианта использования (UCP)	174,72
------------------------------------	--------

Оценка стороннего завершённого проекта
 Для оценки будет использована завершённая курсовая
 работа «Разработка информационной системы»

Список прецедентов проекта

№	Описание
1	Аутентификация пользователя
2	Регистрация пользователя
3	Просмотр полной версии сайта
4	Просмотр новостей
5	Донат
6	Просмотр контактов
7	Просмотр информации о рекламе
8	Просмотр других ресурсов с варламовым
9	Просмотр вакансий
10	Редактирование статей
11	Редактирование комментариев
12	Жалоба на статью
13	Комментирование статьи

Неадаптированный вес участника (UAW)

Классификация	Количество участников	Вес
Простой	2	1
Среднее	0	2
Сложный	0	3
UAW	2	

Нескорректированный вес варианта использования (UUCW)

Классификация	Количество транзакций	Вес
Простые	9	5
Средние	2	10

Сложные	2	15
---------	---	----

UUCW 95

Фактор технической сложности (TCF)

Фактор	Описание	Вес	Сложность
T1	Распределенная система	2,0	4
T2	Время отклика / целевые показатели производительности	1,0	4
T3	Эффективность для конечного пользователя	1,0	5
T4	Сложность внутренней обработки	1,0	4
T5	Повторное использование кода	1,0	4
T6	Простота установки	0,5	4
T7	Простота использования	0,5	5
T8	Переносимость на другие платформы	2,0	4
T9	Обслуживание системы	1,0	3
T10	Параллельная обработка	1,0	5
T11	Функции безопасности	1,0	4
T12	Доступ для третьих лиц	1,0	3
T13	Обучение конечных пользователей	1,0	2
		TF	54,5
		TCF	1,145

Фактор сложности окружающей среды (ECF)

Фактор	Описание	Вес	Влияние
E1	Знакомство с использованным процессом разработки	1,5	4
E2	Опыт применения	0,5	4
E3	Объектно-ориентированный опыт команды	1,0	4
E4	Возможности ведущего аналитика	0,5	0

E5	Мотивация команды	1,0	5
E6	Стабильность требований	2,0	4
E7	Персонал, занятый неполный рабочий день	-1,0	1
E8	Сложный язык программирования	-1,0	1
		EF	23
		ECF	2,09

Точки варианта использования (UCP)

250,068

Оценка целевого проекта

Неадаптированный вес участника (UAW)

Классификация	Количество участников	Вес
Простой	2	1
Среднее	0	2
Сложный	0	3
UAW	2	

Нескорректированный вес варианта использования (UUCW)

Классификация	Количество транзакций	Вес
Простые	14	5
Средние	1	10
Сложные	2	15
UUCW	110	

Фактор технической сложности (TCF)

Фактор	Описание	Вес	Сложность
T1	Распределенная система	2,0	0
T2	Время отклика / целевые показатели производительности	1,0	1
T3	Эффективность для конечного пользователя	1,0	1
T4	Сложность внутренней обработки	1,0	1
T5	Повторное использование кода	1,0	1
T6	Простота установки	0,5	1

T7	Простота использования	0,5	1
T8	Переносимость на другие платформы	2,0	0
T9	Обслуживание системы	1,0	0
T10	Параллельная обработка	1,0	2
T11	Функции безопасности	1,0	0
T12	Доступ для третьих лиц	1,0	0
T13	Обучение конечных пользователей	1,0	0
		TF	7
		TCF	0,67

Фактор сложности окружающей среды (ECF)

Фактор	Описание	Вес	Влияние
E1	Знакомство с использованным процессом разработки	1,5	3
E2	Опыт применения	0,5	3
E3	Объектно-ориентированный опыт команды	1,0	1
E4	Возможности ведущего аналитика	0,5	0
E5	Мотивация команды	1,0	0
E6	Стабильность требований	2,0	2
E7	Персонал, занятый неполный рабочий день	-1,0	4
E8	Сложный язык программирования	-1,0	4
		EF	3
		ECF	1,49

Время разработки курсового проекта в часах	160
Фактор продуктивности (PF)	0,5169609503
Скореллированное УСР данного проекта	57,80119707

Сравнение использованных методов

Итоговая трудоемкость различными методами:

- Наивный метод: 295 человеко-часов
- РЕВТ: 295 человека-часов

- COSOMO-II: 272.56 человеко-часов
- ПСР: 271.62 человека-часов

Вывод

В процессе выполнения лабораторной работы мы примерили на себе роль менеджера, пытающегося адекватно выделить функции требуемого проекта и оценить время-затратность его выполнения. Ощутили, что оценивание одна из непростых задач, с которыми приходится сталкиваться в сфере программирования, как и реалистичные сроки выполнения.