

НИУ ИТМО
Факультет программной инженерии и
компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА № 1
по дисциплине
‘Экономика программной инженерии’

Выполнили:
Студенты группы
Р34111
Привалов Ярослав
Соколов Иван
Денисович

Преподаватель:
Машина Екатерина Алексеевна



Санкт-Петербург, 2024

| | |
|--|-----------|
| Вариант | 3 |
| Функциональные требования | 3 |
| Оценка трудоёмкости наивным методом | 6 |
| Оценка трудоёмкости методом Pert | 9 |
| Метод функциональных точек | 13 |
| Тип оценки | 13 |
| Границы продукта | 13 |
| Подсчет функциональных точек, связанных с данными | 14 |
| Подсчет функциональных точек, связанных с транзакциями | 14 |
| Суммарное количество невыровненных точек | 16 |
| Определение значения фактора выравнивания (VAF) | 16 |
| Расчет количества выровненных функциональных точек (AFP) | 17 |
| Метод COCOMO II | 17 |
| Оценка размера продукта в KSLOC | 17 |
| Оценка уровней множителей трудоемкости | 17 |
| Оценка трудоемкости | 18 |
| Use Case Points | 18 |
| Оценка веса прецедентов | 19 |
| Оценка веса акторов | 19 |
| Оценка веса технических факторов | 19 |
| Оценка веса факторов окружения | 20 |
| Расчёт UCP | 21 |
| Расчёт PF | 21 |
| Прецеденты | 21 |
| Оценка веса прецедентов | 22 |
| Оценка веса акторов | 22 |
| Оценка веса технических факторов | 22 |
| Оценка веса факторов окружения | 23 |
| Расчёт UCP | 24 |
| Расчёт трудоёмкости | 24 |
| Итог трудоёмкости для проекта | 24 |
| Сравнение методов | 24 |
| Вывод | 25 |

Вариант

<https://www.codewars.com>

Функциональные требования

1. Регистрация и аутентификация
 - 1.1. Система должна предоставлять возможность создать аккаунт с уникальным именем пользователя, паролем и email или зарегистрировать свой GitHub аккаунт.
 - 1.2. Система должна предоставлять возможность пользователям входить в аккаунт через email и пароль или через GitHub.
2. Ведение блога
 - 2.1. Система должна предоставлять возможность пользователям читать блоги.
 - 2.2. Система должна предоставлять возможность искать блоги по маркерам, которые отражают наполнение этого блога.
 - 2.3. Система должна предоставлять возможность пользователям перейти на сторонние сервисы, где будет размещена дополнительная информация о блоге.
 - 2.4. Система должна предоставлять возможность подписаться на рассылку уведомлений о предстоящих мероприятиях.
3. Сотрудничество
 - 3.1. Система должна предоставлять возможность пользователям читать информацию о сотрудничестве.
 - 3.2. Система должна предоставлять возможность пользователям перейти на сторонние сервисы, где будет размещена дополнительная информация о сотрудничестве.
 - 3.3. Система должна предоставлять возможность оставить свои контактные данные.
4. Определение уровня знаний пользователя
 - 4.1. Система должна предоставить возможность пользователю выбрать свой уровень знаний, а также выбрать языки программирования, на которых он хочет обучаться.

- 4.2. Система должна предоставить возможность пользователю подписаться на рассылку уведомлений о различных подготовительных курсах.

5. Тренировка

- 5.1. Система должна подобрать оптимальные задачи, в соответствии с заявленным уровнем знаний пользователя.
- 5.2. Система должна предоставить возможность пользователю читать текст заданий и выбирать задания, которые он хочет делать.
- 5.3. Система должна предоставить возможность искать задания по маркерам, которые отражают тип этой задачи.
- 5.4. Система должна предоставить возможность написать код для выбранной задачи прямо на сайте, а также уметь правильно форматировать введенный пользователем код.
- 5.5. Система должна уметь правильно засчитывать прохождение задачи пользователем, а также сообщать об ошибках со стороны пользователя, если таковые есть.
- 5.6. Система должна предоставить пользователю посмотреть другие решения данной задачи от других пользователей после прохождения данной задачи, а также открывать доступ к форуму, в котором ведутся дискуссии по данной задаче.
- 5.7. Система должна предоставить возможность пользователю перейти к следующей задаче вне зависимости от того, успешно ли он завершил предыдущую или нет.

6. Документация

- 6.1. Система должна предоставить возможность пользователю читать документацию, связанную с использованием системы.
- 6.2. Система должна предоставить возможность пользователю перейти на сторонний сервис для предложения своих корректировок в документацию.

7. Ранг пользователя

- 7.1. Система должна рассчитывать количество и ранг пройденных задач, и, в соответствии с этим, назначать пользователю определенный ранг.

8. Профиль пользователя

- 8.1. Система должна предоставить возможность пользователю просматривать свой профиль.
- 8.2. Система должна предоставить возможность пользователю выйти из своего профиля.

- 8.3. Система должна предоставить возможность пользователю дополнить свой профиль данными.
- 8.4. Система должна предоставить возможность пользователю просматривать свой ранг, список решенных задач, а также их решения.
- 9. Свободный спарринг
 - 9.1. Система должна предоставить возможность пользователю просматривать список текущих “спаррингов”.
 - 9.2. Система должна предоставить возможность пользователю создавать свой “спарринг”.
 - 9.3. Система должна предоставить возможность пользователю присоединиться к “спаррингу”.
 - 9.4. Система должна предоставить возможность пользователю выбирать желаемый язык для поиска или создания “спаррингов”.
 - 9.5. Система должна предоставить возможность пользователю просматривать результаты своих “спаррингов”.
- 10. Обсуждения
 - 10.1. Система должна предоставить возможность пользователю перейти на сторонний сервис, который будет представлять из себя форум.
- 11. Таблица лидеров
 - 11.1. Система должна предоставить возможность пользователю просматривать таблицу лидеров.
 - 11.2. Система должна предоставить возможность пользователю использовать различные фильтры для поиска лидеров по тем или иным показателям.
- 12. Подписка “Red”
 - 12.1. Система должна предоставить возможность пользователю купить подписку и выбрать желаемый срок.
 - 12.2. Система должна предоставить пользователю доступ к расширенному инструменту сравнения возможных решений задачи.
 - 12.3. Система должна предоставить возможность пользователю отключить рекламу.
 - 12.4. Система должна предоставить доступ пользователю к использованию функционала, находящегося в стадии бета-тестирования.
 - 12.5. Система должна разместить особый значок в профиле пользователя, указывающий на наличие у него подписки.

Оценка трудоёмкости наивным методом

| № | Функциональное требование | Подзадача | Оценка мин./чел.-час | Оценка макс./чел.-час |
|---|--|---|----------------------|-----------------------|
| 1 | Регистрация и аутентификация | | 120 | 160 |
| | | Создание аккаунта (email/пароль, GitHub) | 60 | 80 |
| | | Вход в аккаунт (email/пароль, GitHub) | 60 | 80 |
| 2 | Ведение блога | | 60 | 100 |
| | | Интерфейс для чтения блогов | 20 | 30 |
| | | Реализация поиска по маркерам | 20 | 30 |
| | | Переход на сторонние ресурсы | 10 | 20 |
| | | Подписка на рассылку уведомлений | 10 | 20 |
| 3 | Сотрудничество | | 30 | 60 |
| | | Интерфейс для информации о сотрудничестве | 10 | 20 |
| | | Переходы на сторонние ресурсы | 10 | 20 |
| | | Форма для ввода контактных данных | 10 | 20 |
| 4 | Определение уровня знаний пользователя | | 20 | 40 |

| | | | | |
|---|----------------------|---|-----|-----|
| | | Выбор уровня знаний и языков программирования | 10 | 20 |
| | | Подписка на уведомления о курсах | 10 | 20 |
| 5 | Тренировка | | 200 | 320 |
| | | Подбор задач по уровню знаний | 50 | 80 |
| | | Интерфейс для выбора и чтения задач | 30 | 40 |
| | | Поиск задач по маркерам | 20 | 30 |
| | | Редактор для написания кода | 50 | 80 |
| | | Проверка прохождения задачи | 30 | 50 |
| | | Доступ к форуму и другим решениям | 10 | 20 |
| | | Переход к следующей задаче | 10 | 20 |
| 6 | Документация | | 30 | 50 |
| | | Интерфейс для чтения документации | 20 | 30 |
| | | Переход на сторонний сервис для корректировок | 10 | 20 |
| 7 | Ранг пользователя | | 20 | 30 |
| | | Механизм подсчета задач и назначения ранга | 20 | 30 |
| 8 | Профиль пользователя | | 60 | 100 |

| | | | | |
|----|--------------------|------------------------------------|----|-----|
| | | Интерфейс профиля | 20 | 30 |
| | | Выход из профиля | 10 | 20 |
| | | Дополнение профиля | 10 | 20 |
| | | Отображение ранга и решенных задач | 20 | 30 |
| 9 | Свободный спарринг | | 90 | 150 |
| | | Интерфейс для чтения спаррингов | 20 | 30 |
| | | Создание спарринга | 20 | 40 |
| | | Присоединение к спаррингу | 20 | 30 |
| | | Выбор языка для спарринга | 10 | 20 |
| | | Просмотр результатов | 20 | 30 |
| 10 | Обсуждения | | 10 | 20 |
| | | Переход на сторонний форум | 10 | 20 |
| 11 | Таблица лидеров | | 40 | 60 |
| | | Интерфейс для таблицы лидеров | 20 | 30 |
| | | Фильтрация по показателям | 20 | 30 |
| 12 | Подписка "Red" | | 90 | 150 |
| | | Покупка подписки и выбор срока | 30 | 50 |
| | | Доступ к расширенным инструментам | 20 | 40 |
| | | Отключение рекламы | 10 | 20 |
| | | Доступ к бета-функционал | 20 | 30 |

| | | | | |
|--|--|---------------------------|----|----|
| | | у | | |
| | | Значок подписки в профиле | 10 | 10 |

Оценка трудоёмкости методом Pert

| № | Функциональное требование | Подзадача | Оптимистичное (ч-ч) | Пессимистичное (ч-ч) | Ожидаемое (ч-ч) | Ei | СКОi |
|---|------------------------------|--|---------------------|----------------------|-----------------|-----|---------------------|
| 1 | Регистрация и аутентификация | | 120 | 160 | 140 | 140 | 6,666 6666 67 |
| | | Создание аккаунта (email/пароль, GitHub) | 60 | 80 | 70 | 70 | 3,333 3333 33 |
| | | Вход в аккаунт (email/пароль, GitHub) | 60 | 80 | 70 | 70 | 3,333 3333 33 |
| 2 | Ведение блога | | 60 | 100 | 80 | 80 | 6,666 6666 67 |
| | | Интерфейс для чтения блогов | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Реализация поиска по маркерам | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Переход на сторонние ресурсы | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Подписка на рассылку уведомлений | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| 3 | Сотрудничество | | 30 | 60 | 45 | 45 | 5 |
| | | Интерфейс для информации о | 10 | 20 | 15 | 15 | 1,666 6666 67 |

| | | | | | | | |
|---|---|---|-----|-----|-----|-----|---------------------|
| | | сотрудниче ст | | | | | |
| | | Переходы на сторонние ресурсы | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Форма для ввода контактных данных | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| 4 | Определение уровня знаний пользователя | | 20 | 40 | 30 | 30 | 3,333 3333 33 |
| | | Выбор уровня знаний и языков программиро вания | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Подписка на уведомления о курсах | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| 5 | Тренировка | | 200 | 320 | 260 | 260 | 20 |
| | | Подбор задач по уровню знаний | 50 | 80 | 65 | 65 | 5 |
| | | Интерфейс для выбора и чтения задач | 30 | 40 | 35 | 35 | 1,666 6666 67 |
| | | Поиск задач по маркерам | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Редактор для написания кода | 50 | 80 | 65 | 65 | 5 |
| | | Проверка прохождения задачи | 30 | 50 | 40 | 40 | 3,333 3333 33 |
| | | Доступ к форуму и другим решениям | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Переход к следующей задаче | 10 | 20 | 15 | 15 | 1,666 6666 67 |

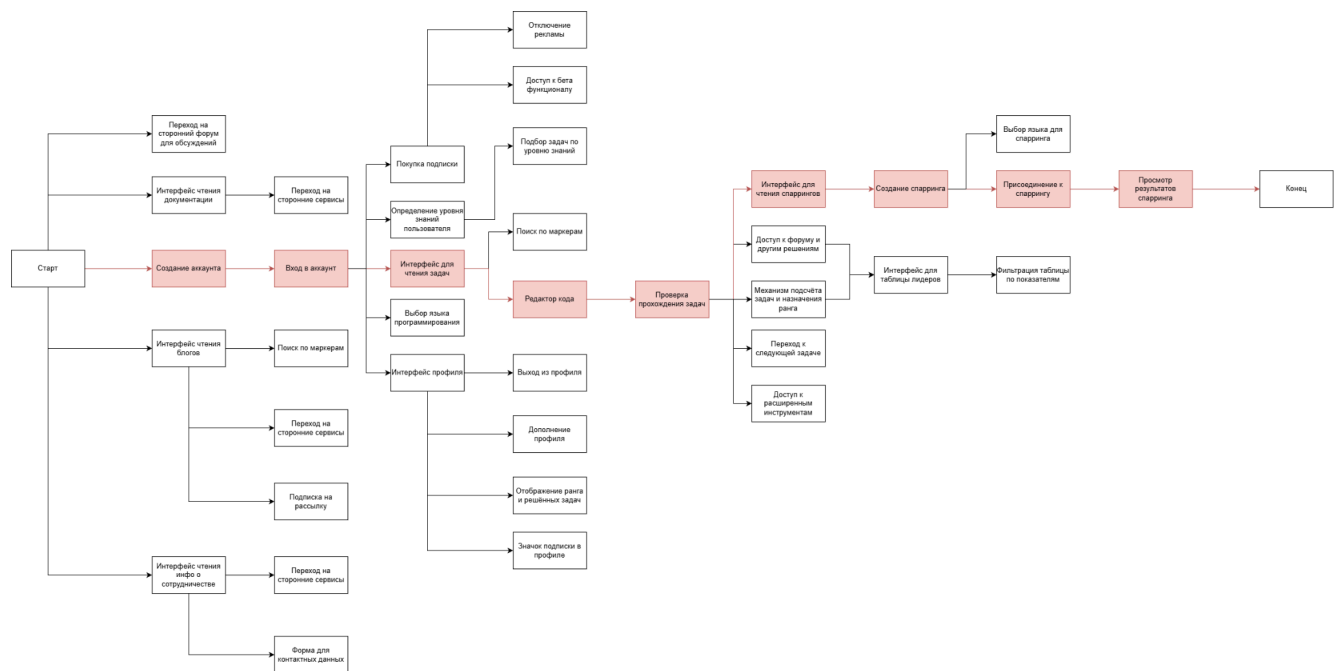
| | | | | | | | |
|---|----------------------|---|----|-----|-----|-----|---------------------|
| 6 | Документация | | 30 | 50 | 40 | 40 | 3,333 3333 33 |
| | | Интерфейс для чтения документации | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Переход на сторонний сервис для корректировок | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| 7 | Ранг пользователя | | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Механизм подсчета задач и назначения ранга | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| 8 | Профиль пользователя | | 60 | 100 | 80 | 80 | 6,666 6666 67 |
| | | Интерфейс профиля | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Выход из профиля | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Дополнение профиля | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Отображение ранга и решенных задач | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| 9 | Свободный спарринг | | 90 | 150 | 120 | 120 | 10 |
| | | Интерфейс для чтения спаррингов | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Создание спарринга | 20 | 40 | 30 | 30 | 3,333 3333 33 |
| | | Присоединение к спаррингу | 20 | 30 | 25 | 25 | 1,666 6666 67 |

| | | | | | | | |
|----|-----------------|-----------------------------------|----|-----|-------|-------|----------------------|
| | | Выбор языка для спарринга | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Просмотр результатов | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| 10 | Обсуждения | | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Переход на сторонний форум | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| 11 | Таблица лидеров | | 40 | 60 | 50 | 50 | 3,333 3333 33 |
| | | Интерфейс для таблицы лидеров | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Фильтрация по показателям | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| 12 | Подписка "Red" | | 90 | 155 | 122,5 | 122,5 | 10,83 3333 33 |
| | | Покупка подписки и выбор срока | 30 | 50 | 40 | 40 | 3,333 3333 33 |
| | | Доступ к расширенным инструментам | 20 | 40 | 30 | 30 | 3,333 3333 33 |
| | | Отключение рекламы | 10 | 20 | 15 | 15 | 1,666 6666 67 |
| | | Доступ к бета-функционалу | 20 | 30 | 25 | 25 | 1,666 6666 67 |
| | | Значок подписки в профиле | 10 | 15 | 12,5 | 12,5 | 0,833 3333 333 |

$$E = \sum E_i = 2015$$

$$CKO = \sqrt{\sum CKO_i^2} = 31,81980515 \approx 32$$

$$E_{95\%} = E + 2 * CKO = 2078,63961 \approx 2079$$



Критический путь: 310 часов

Долгий путь: 450 часов

Скорости в 310 часов можно достигнуть при идеальной работе и, в таком случае, одна команда из 1-го фронтендера и 1-го бэкендера должна следовать исключительно по критическому пути, а другие 2 команды из 1-го фронтендера и бэкендера должны делать все остальные задачи, от которых критический путь не зависит.

Положим, что рабочий день, в среднем, составляет 6 часов. Тогда, мы получаем 52 рабочих дня, за которые мы сможем завершить этот проект, за вычетом этапа тестирования. Добавив в нашу команду 2-х тестировщиков, мы увеличим минимальное время завершения проекта на ~ 20 часов и получим 55 рабочих дней. Если же мы пойдем по самому негативному сценарию, то наша разработка затянется на 75 рабочих дней без тестировщиков и на 79 дней с тестировщиками.

Метод функциональных точек

Тип оценки

Тип оценки: продукт

Границы продукта

Границами продукта являются реальные функции, реализованные продуктом. В границы системы входят все функции, описанные выше, за исключением тех, которые осуществляют переход на сторонние сервисы (подробнее см. Use Case - диаграмму)

Подсчет функциональных точек, связанных с данными

| № | Название | RET | DET | Сложность | UFP |
|---|-------------------------------------|-----|---|-----------|-----|
| 1 | Регистрация | 2 | Email, имя, пароль, повтор пароля (4) | Low | 7 |
| 2 | Вход | 1 | Email, пароль (2) | Low | 7 |
| 3 | Контактные данные | 2 | Вид деятельности, email, пароль (3) | Low | 7 |
| 4 | Дополнительная информация в профиле | 5 | Настоящее имя, клан, уровень знаний, увлечения, ссылка на соц. сети (5) | Low | 7 |
| 5 | Решение задач | 1 | Код (1) | Low | 7 |
| 6 | Создание спарринга | 2 | Выбор языка программирования, заголовок, текст задания (3) | Low | 7 |
| 7 | Поиск по таблице лидеров | 3 | Звание, выполненные задачи, предложено задач (3) | Low | 7 |

Подсчет функциональных точек, связанных с транзакциями

| № | Название | Тип | FTR | DET | Сложность | UFP |
|----|---|-----|-----|-----|-----------|-----|
| 1 | Регистрация | EI | 1 | 5 | Low | 3 |
| 2 | Вход | EI | 1 | 3 | Low | 3 |
| 3 | Контактные данные | EI | 1 | 4 | Low | 3 |
| 4 | Дополнительная информация в профиле | EI | 2 | 6 | Low | 3 |
| 5 | Просмотр задач | EQ | 1 | 1 | Low | 3 |
| 6 | Решение задач | EI | 1 | 2 | Low | 3 |
| 7 | Создание спарринга | EI | 1 | 4 | Low | 3 |
| 8 | Присоединение к спаррингу | EI | 1 | 1 | Low | 3 |
| 9 | Просмотр таблицы лидеров | EQ | 1 | 1 | Low | 3 |
| 10 | Поиск по таблице лидеров | EI | 1 | 4 | Low | 3 |
| 11 | Просмотр профиля | EI | 1 | 1 | Low | 3 |
| 12 | Переход на сторонний сервис для блогов | EO | 2 | 1 | Low | 4 |
| 13 | Переход на сторонний сервис для общения | EO | 2 | 1 | Low | 4 |

| | | | | | | |
|----|---|----|---|---|-----|---|
| 14 | Переход на сторонний сервис для сотрудничества | EO | 2 | 1 | Low | 4 |
| 15 | Переход на сторонний сервис для редактирования документации | EO | 2 | 1 | Low | 4 |
| 16 | Просмотр блога | EQ | 0 | 1 | Low | 3 |
| 17 | Просмотр документации | EQ | 2 | 2 | Low | 3 |

Суммарное количество невыровненных точек

$$UFP = 49 + 55 = 104$$

Определение значения фактора выравнивания (VAF)

| № | Параметр | DI |
|---|--|----|
| 1 | Обмен данными | 3 |
| 2 | Распределенная обработка данных | 0 |
| 3 | Производительность | 2 |
| 4 | Ограничения по аппаратным ресурсам | 0 |
| 5 | Транзакционная нагрузка | 0 |
| 6 | Интенсивность взаимодействия с пользователем | 2 |
| 7 | Эргономика | 3 |
| 8 | Интенсивность изменения данных | 2 |
| 9 | Сложность обработки | 2 |

| | | |
|----|----------------------------|---|
| 10 | Повторное использование | 5 |
| 11 | Удобство инсталляции | 0 |
| 12 | Удобство администрирования | 1 |
| 13 | Портируемость | 1 |
| 14 | Гибкость | 0 |

$$TDI = \sum DI = 21$$

$$VAF = TDI * 0.01 + 0.65 = 0.86$$

Расчет количества выровненных функциональных точек (AFP)

$$AFP = UFP * VAF = 89.44$$

Метод COSOMO II

Оценка размера продукта в KSLOC

Стек технологий: React, Spring Boot

Разделив функциональность между слоями получим 3/5 Frontend и 2/5 Backend.

$$KSLOC = UFP * SIZE = 104 * 3/5 * 0.047 + 104 * 2/5 * 0.053 = 5.1376$$

| Фактор | Уровень | Значение |
|--------|---------|----------|
| PREC | Normal | 3.72 |
| FLEX | High | 2.03 |
| RESL | Low | 5.65 |
| TEAM | High | 2.19 |
| PMAT | Low | 6.24 |

Оценка уровней множителей трудоемкости

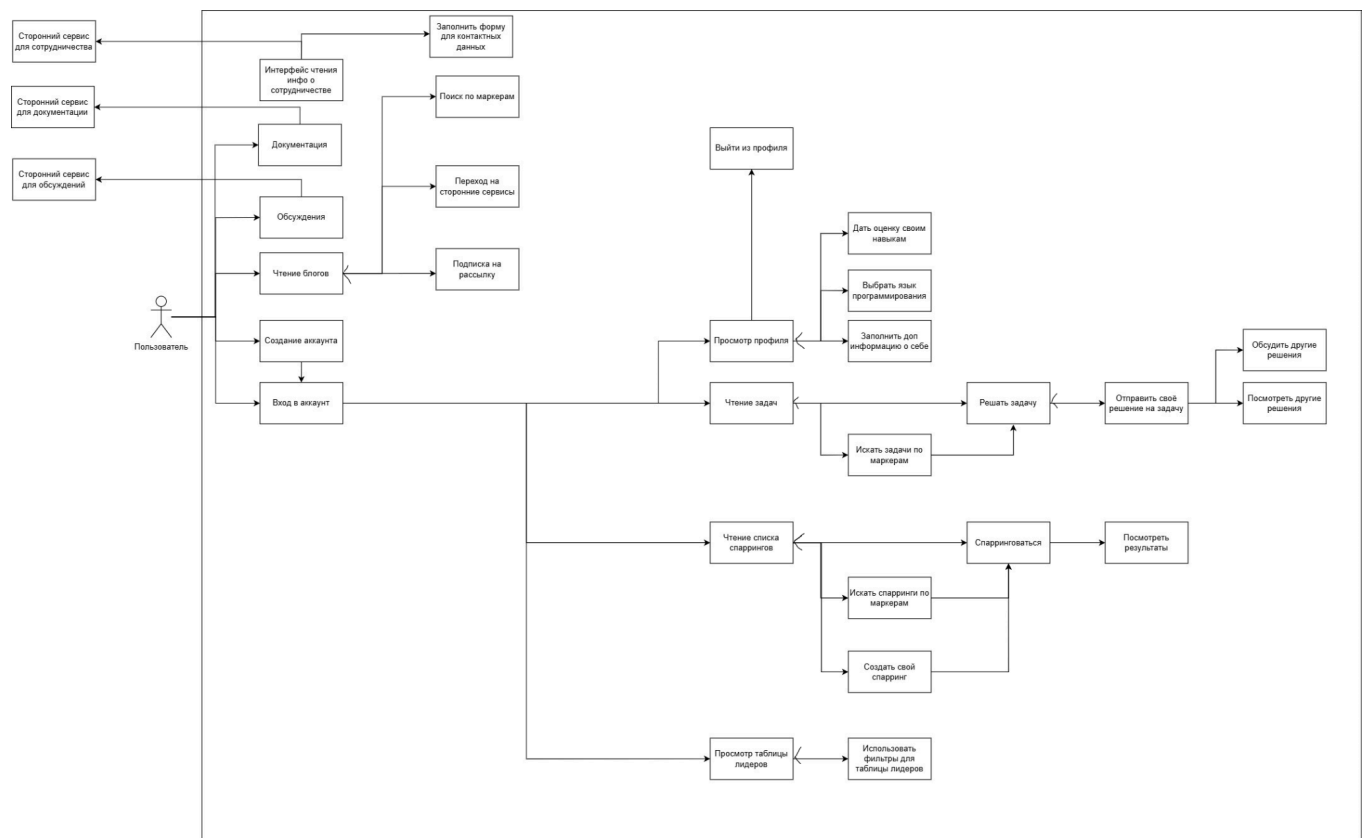
| Фактор | Уровень | Значение |
|--------|---------|----------|
| PERS | Nominal | 1 |
| RCPX | High | 1.33 |
| RUSE | Low | 0.95 |
| PDIF | Nominal | 1 |
| PREX | Nominal | 1 |
| FSIL | Nominal | 1 |
| CSED | Nominal | 1 |

Оценка трудоемкости

$$E = 0.91 + 0,01 \cdot (3.72 + 2.03 + 5.65 + 2.19 + 6.24) = 1.1083$$

$$PM = 2.94 \cdot 5.1376^{(1.1083)} \cdot (1 \cdot 1 \cdot 1 \cdot 1 \cdot 1.33 \cdot 0.95) = 22.785 \text{ ч./мес.} = 3645 \text{ ч./час}$$

Use Case Points



Оценка веса прецедентов

| Сложность | Вес | Количество | Затраты |
|-----------|-----|------------|---------|
| Low | 5 | 21 | 105 |
| Medium | 10 | 5 | 50 |
| High | 15 | 1 | 15 |

UUCW = 170

Оценка веса акторов

| Сложность | Вес | Количество | Затраты |
|-----------|-----|------------|---------|
| Low | 1 | 0 | 0 |
| Medium | 2 | 1 | 2 |
| High | 3 | 1 | 3 |

UAW = 5

Оценка веса технических факторов

| Фактор | Вес | Балл | Затраты |
|---|-----|------|---------|
| Распределенная система | 2 | 1 | 2 |
| Цели времени отклика/производительности | 1 | 3 | 3 |
| Эффективность для конечного пользователя | 1 | 3 | 3 |
| Внутренняя сложность обработки | 1 | 4 | 4 |
| Возможность повторного использования кода | 1 | 2 | 2 |

| | | | |
|--------------------------------------|-----|---|-----|
| Легко установить | 0.5 | 2 | 1 |
| Легко использовать | 0.5 | 3 | 1.5 |
| Переносимость на другие платформы | 2 | 1 | 2 |
| Техническое обслуживание системы | 1 | 1 | 1 |
| Одновременная/параллельная обработка | 1 | 2 | 2 |
| Функции безопасности | 1 | 2 | 2 |
| Доступ для третьих лиц | 1 | 1 | 1 |
| Обучение конечного пользователя | 1 | 2 | 2 |

$$TCF = 0.6 + TF / 100 = 0.865$$

Оценка веса факторов окружения

| Фактор | Вес | Балл | Затраты |
|--|-----|------|---------|
| Знаком с моделью проекта, которая используется | 1.5 | 2 | 3 |
| Опыт применения | 0.5 | 2 | 1 |
| Опыт в веб разработке | 1 | 3 | 3 |
| Возможность ведущего аналитика | 0.5 | 2 | 1 |
| Мотивация | 1 | 3 | 3 |
| Стабильные требования | 2 | 2 | 4 |
| Частичная занятость | -1 | 3 | -3 |

| | | | |
|----------------------------------|----|---|----|
| Сложность языка программирования | -2 | 3 | -6 |
|----------------------------------|----|---|----|

$$ECF = 1,4 + (-0,03 * EF) = 1.22$$

Расчёт UCP

$$UCP = (UUCW + UAW) * TCF * ECF = 184.6775$$

Расчёт PF

За основу берутся все лабораторные по БЛПС.

Прецеденты

| № | Прецедент |
|----|-----------------------------------|
| 1 | Регистрация |
| 2 | Авторизация |
| 3 | Поиск по категориям |
| 4 | Поиск по производителям |
| 5 | Поиск продукта по названию |
| 6 | Поиск человека по имени |
| 7 | Создание категории |
| 8 | Создание производителя |
| 9 | Создание продукта |
| 10 | Обновление данных о категории |
| 11 | Обновление данных о производителе |
| 12 | Обновление данных о продукте |
| 13 | Удаление данных о категории |
| 14 | Удаление данных о производителе |
| 15 | Удаление данных о продукте |
| 16 | Подписка на уведомления о товаре |

| | |
|----|-----------------------------|
| 17 | Добавление товара в корзину |
| 18 | Удаление товара из корзины |
| 19 | Обновление товара в корзине |
| 20 | Оплата корзины |

Оценка веса прецедентов

| Сложность | Вес | Количество | Затраты |
|-----------|-----|------------|---------|
| Low | 5 | 15 | 75 |
| Medium | 10 | 5 | 50 |
| High | 15 | 0 | 0 |

UUCW = 125

Оценка веса акторов

| Сложность | Вес | Количество | Затраты |
|-----------|-----|------------|---------|
| Low | 1 | 0 | 0 |
| Medium | 2 | 1 | 2 |
| High | 3 | 3 | 9 |

UAW = 11

Оценка веса технических факторов

| Фактор | Вес | Балл | Затраты |
|--|-----|------|---------|
| Распределенная система | 2 | 3 | 6 |
| Цели времени отклика/производительности | 1 | 1 | 1 |
| Эффективность для конечного пользователя | 1 | 3 | 3 |

| | | | |
|---|-----|---|-----|
| Внутренняя сложность обработки | 1 | 3 | 3 |
| Возможность повторного использования кода | 1 | 3 | 3 |
| Легко установить | 0.5 | 2 | 1 |
| Легко использовать | 0.5 | 5 | 2.5 |
| Переносимость на другие платформы | 2 | 2 | 4 |
| Техническое обслуживание системы | 1 | 1 | 1 |
| Одновременная/параллельная обработка | 1 | 3 | 3 |
| Функции безопасности | 1 | 3 | 3 |
| Доступ для третьих лиц | 1 | 1 | 1 |
| Обучение конечного пользователя | 1 | 2 | 2 |

$$TCF = 0.6 + TF / 100 = 0.925$$

Оценка веса факторов окружения

| Фактор | Вес | Балл | Затраты |
|--|-----|------|---------|
| Знаком с моделью проекта, которая используется | 1.5 | 2 | 3 |
| Опыт применения | 0.5 | 1 | 0.5 |
| Опыт в веб разработке | 1 | 2 | 2 |
| Возможность ведущего аналитика | 0.5 | 1 | 0.5 |

| | | | |
|----------------------------------|----|---|----|
| Мотивация | 1 | 2 | 2 |
| Стабильные требования | 2 | 1 | 2 |
| Частичная занятость | -1 | 3 | -3 |
| Сложность языка программирования | -2 | 2 | -4 |

$$ECF = 1,4 + (-0,03 * EF) = 1.31$$

Расчёт UCP

$$UCP = (UUCW + UAW) * TCF * ECF = 164.798$$

Расчёт трудоёмкости

Работа была выполнена ~за 100 часов.

$$PF = E / UCP = 0.607$$

Итог трудоёмкости для проекта

$$E = PF * UCP = 112 \text{ ч./час}$$

Сравнение методов

| Метод | Затраты |
|----------------------|---------|
| Наивный | 2015 |
| PERT | 2079 |
| Функциональных точек | 1521 |
| COCOMO II | 3645 |
| UCP | 112 |

Метод COCOMO II дал самый наивысший результат трудоёмкости, по сравнению с методами PERT и наивным. Стоит подметить, что этот метод единственный среди представленных, который меньше опирается на нашу субъективную оценку и столь

большая разница между этими методами может быть вызвана тем, что мы недооценили некоторые параметры системы или переоценили самих себя. Метод PERT дал результат схожий с наивным, поскольку практически основывается на нашей субъективной оценке. Метод функциональных точек, преимущественно, оценивает трудоёмкость в оценке размеров данных, которые мы получаем от пользователя или передаем пользователю. Метод UCP дал совершенно некорректный результат. Возможно это связано с тем, что мы рассчитываем трудоёмкость не в зависимости от сложности в разработке функционала, а в количестве пользовательских сценариев. Также это может быть связано с неверной оценкой нами выполненных лабораторных работ, поскольку лабораторные работы, всё же, сильно отличаются от реальных проектов и требуют сильно меньше времени на реализацию, чем реальные.

Вывод

В ходе выполнения данной лабораторной работы мы ознакомились с основными методами оценки трудозатрат на разработку. На себе ощутили всю сложность работы продакт-менеджера и сложность оценки трудозатрат, поскольку мы имеем слишком много зависимостей и переменных, от которых наши расчёты могут иметь большую погрешность.