МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное автономное образовательное учреждение высшего образования «Пермский государственный национальный исследовательский университет»

Механико-математический факультет

Кафедра информационных технологий

Отчет о практике  
ГРУППОВАЯ ПРОЕКТНАЯ РАБОТА

по направлению

02.03.02 Фундаментальная информатика и информационные технологии

Работу выполнили студенты   
3 курса группы ММ/О ФИТ-2020 НБ

|  |  |  |
| --- | --- | --- |
| Коноплева Д.Д. |  | |
| Колпащиков М.Н | |  |
| Коваль А.Е. |  | |
| Лязгин Д.С. |  | |
| Япаров А.Э. |  | |

Консультант:

Генеральный директор ООО «Нейрософт Глобал»

|  |  |
| --- | --- |
| Степанов В.А. |  |

Руководитель практики:

доцент кафедры информационных технологий ПГНИУ

|  |  |
| --- | --- |
| Остапенко Е.Н. |  |

Пермь 2023

**Оглавление**

[Раздел 1. Командная работа 4](#_Toc138886000)

[Раздел 2. Проектная работа 6](#_Toc138886001)

[2.1. Выбор темы проекта и ее актуальность 6](#_Toc138886002)

[2.1.2 Уникальность проекта 8](#_Toc138886003)

[2.2. Требования к программе 8](#_Toc138886004)

[2.2.1. Требования к составу выполняемых функций 8](#_Toc138886005)

[2.2.2. Требования к организации входных данных 8](#_Toc138886006)

[2.2.3. Требования к организации выходных данных 9](#_Toc138886007)

[2.2.4. Требования к организации хранения данных 9](#_Toc138886008)

[2.2.5. Требования к графическому интерфейсу 9](#_Toc138886009)

[2.2.6. Требования к программным средствам, используемым программой 9](#_Toc138886010)

[2.3. ИТ-технологии для реализации проекта 9](#_Toc138886011)

[2.4. Выбор методологии разработки проекта 9](#_Toc138886012)

[Раздел 3. Выполненная работа по проекту 10](#_Toc138886013)

[3.1 Спринт 1 (02.06-08.06) 10](#_Toc138886014)

[3.1.1 Цель 10](#_Toc138886015)

[3.1.2 Планирование задач 10](#_Toc138886016)

[3.1.3 Выполненные задачи 10](#_Toc138886017)

[3.1.4 Ретроспектива 15](#_Toc138886018)

[3.2 Спринт 2 (09.06-15.06) 15](#_Toc138886019)

[3.2.1 Цель 15](#_Toc138886020)

[3.2.2 Планирование задач 15](#_Toc138886021)

[3.2.3 Выполненные задачи 16](#_Toc138886022)

[3.2.4 Ретроспектива 18](#_Toc138886023)

[3.3 Спринт 3 (16.06-22.06) 18](#_Toc138886024)

[3.3.1 Цель 18](#_Toc138886025)

[3.3.2 Планирование задач 19](#_Toc138886026)

[3.3.3 Выполненные задачи 19](#_Toc138886027)

[3.3.4 Ретроспектива 21](#_Toc138886028)

[3.4 Спринт 4 (23.06-29.06) 21](#_Toc138886029)

[3.4.1 Цель 21](#_Toc138886030)

[3.4.2 Планирование задач 22](#_Toc138886031)

[3.4.3 Выполненные задачи 22](#_Toc138886032)

[3.4.4 Ретроспектива 25](#_Toc138886033)

[3.5 Выполненный функционал системы 25](#_Toc138886034)

[Библиографический список 26](#_Toc138886035)

[Приложение 1. Личный отчет по проекту Коноплевой Д.Д 27](#_Toc138886036)

[Приложение 2. Личный отчет по проекту Колпащикова М.Н 28](#_Toc138886037)

[Приложение 3. Личный отчет по проекту Коваль А.Е. 29](#_Toc138886038)

[Приложение 4. Личный отчет по проекту Лязгина Д.С. 30](#_Toc138886039)

[Приложение 5. Личный отчет по проекту Япарова А.Э. 31](#_Toc138886040)

Раздел 1. Командная работа

Командная работа - это совместные усилия группы для достижения общей цели или выполнения задачи наиболее эффективным способом [1].

Команда - это группа лиц, объединённая общими мотивами, интересами, идеалами, действующая сообща. Участники команды объединены поддержкой друг друга и несут коллективную ответственность за результат деятельности всей команды.

В настоящее время существует достаточно много различных методологий разработки программного обеспечения.

Методология разработки ПО - это система принципов, а также совокупность идей, понятий, методов, способов и средств, определяющих стиль разработки программного обеспечения [2].

В последнее время в качестве методологии разработки ИТ-проектов выбирают agile (гибкие) методологии. Такие методологии стали популярны из-за их способности быстро адаптироваться к изменениям в проекте и требованиям заказчика. Они основаны на принципах коллаборации, коммуникации и итеративной разработки, что позволяет достигать более высокой эффективности работы команды.

Наиболее популярные виды agile методологий:

1. Scrum - это agile методология, основанная на итеративном и инкрементальном подходе к разработке. Команда работает в коротких спринтах (1-2 недели).

Спринт – это временной цикл разработки, в течение которого команда должна выполнить запланированные задачи.

Каждый спринт начинается с планирования бэклога спринта. Вся команда изучает бэклог продукта. На его основе составляются задачи, которые необходимо выполнить в пределах одного цикла. Заканчивается спринт обсуждением полученных результатов, демонстрацией готового продукта – ревью и анализом причин успехов и неудач спринта – ретроспективой.

Бэклог спринта - это список задач, которые команда планирует выполнить/протестировать за время спринта.

Бэклог продукта - это приоритетный список функциональности, который должен содержать продукт [4].

В классическом Scrum существует 3 базовых роли:

* Product owner (Владелец продукта) - является связующим звеном между командой разработки и заказчиком. Его задача - составление бэклога продукта. максимальное увеличение ценности разрабатываемого продукта и работы команды.
* Scrum master сопровождает команду от спринта к спринту, следит за соблюдением скрам-процессов, проводит ежедневные встречи и помогает добиваться поставленных целей, коммуницирует с членами команды, учит их взаимодействовать и решать задачи, общаясь и находя точки соприкосновения [5].
* Development team (команда разработки) - то люди, работающие над элементами бэклога спринта. Они имеют все необходимые компетенции, чтобы каждый спринт создавать работающий инкремент продукта.

2. Kanban - это методология, основанная на визуальном управлении проектом. Команда использует доску задач, на которой отображаются все задачи проекта и их статусы. Это помогает команде лучше организовывать свою работу и быстрее реагировать на изменения.

Основное Scrum и Kanban — это спринты. В Scrum предусмотрены четко организованные периоды работы с конкретными задачами на период, а в Kanban участники команды могут получать новые задачи хоть каждый день. Scrum-команды выполняют работу на время, в Kanban задачи поступают в непрерывном режиме.

3. Microsoft Solutions Framework (MSF) - методология разработки программного обеспечения, предложенная корпорацией Microsoft. MSF опирается на практический опыт Microsoft и описывает управление людьми и рабочими процессами в процессе разработки решения.

Методология MSF считает, что успешная работа команды над проектом существенным образом зависит от ее структуры и распределения зон ответственности ролевых групп внутри команды.

Главная особенность модели команды в MSF является то, что она не имеет официального лидера, все отвечают за проект в равной степени.

Определение ролей в команде:

* Управление программой (program manager) – отвечает за управление проектом, за то, что ожидания заинтересованных сторон будут верно поняты и проведены через проект.
* Архитектура продукта – отвечает за систему в целом, вырабатывает архитектуру решения, включая сервисы, технологии и стандарты, которые будут использованы в ходе работы над решением.
* Разработка – отвечает за проектирование и осуществление реализации.
* Тестирование – отвечает за качество решения с точки зрения заказчика и будущих пользователей.
* Управление выпуском (release manager) – отвечает за гладкое внедрение решения в инфраструктуру заказчика.
* Удовлетворение потребителя (user experience) – отвечает за понимание потребностей пользователей и их надлежащую реализацию в решении.
* Управление продуктом (product manager) – отвечает за понимание того как, и успешное получение бизнес-отдачи от внедрения разрабатываемого решения, которое в результате сможет получить заказчик.

Раздел 2. Проектная работа

**2.1. Выбор темы проекта и ее актуальность**

Тема проекта: проектирование и создание системы автоматизации документооборота.

Довольно часто возникает необходимость создания официальных документов. Видов документов существует очень много, например, заявления, договоры, акты, отчёты и прочее. Большинство этих документов имеют чётко определённую, или даже стандартизованную, структуру, поэтому зачастую нет нужды формировать их с нуля. Достаточно один раз создать шаблон документа, а затем лишь наполнять его необходимыми данными.

Информационная система «ОкиДоки» – система автоматизации документооборота, позволит пользователям по выбранному шаблону получать готовые документы, заполняя при этом только содержательные данные.

Программа предназначена для пользователей, работающих с документами в различных сферах деятельности, например, сфера услуг и образования, различные сферы деятельности небольших компании или ИП.

**2.1.1. Анализ рынка и потребностей пользователя**

Наиболее удачными решениями для индивидуальных пользователей являются текстовые процессоры: Microsoft Word и LibreOffice Writer, а также сервисы, предлагающие встроенную коллекцию шаблонов R6R [7] и Eforms [8]. Достоинствами данных решений является простота использования и умеренная гибкость настройки. Среди главных недостатков следует выделить отсутствие поддержки пользовательских шаблонов у систем уровня R6R и Eforms. Интерфейс систем R6R представлен на рисунке 1.

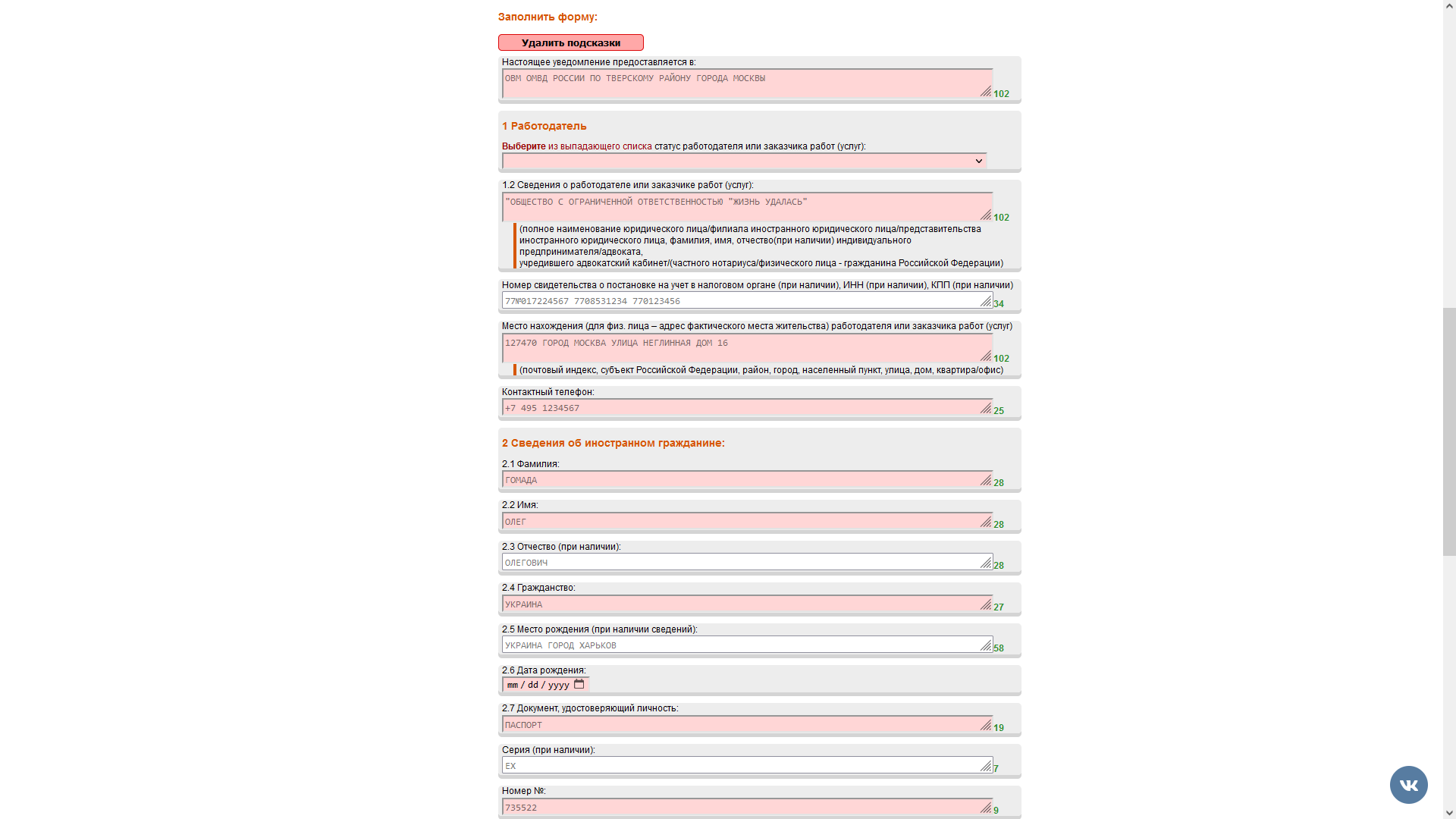


Рисунок 1 – Интерфейс системы R6R

Системы DocWebService и 1С представляют собой решения корпоративного уровня. Среди их достоинств высокий уровень гибкости, поддержка пакетного режима работы, широкие возможности по автоматизации и администрированию. Недостатком этих решений можно считать необходимость дополнительных трудозатрат на подготовку шаблона, поскольку готовых решений не предоставляется. Интерфейс системы представлен на рисунке 2.

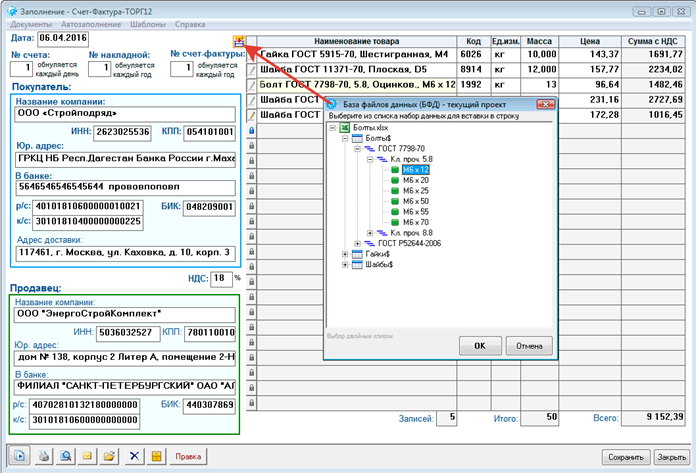


Рисунок 2 – Интерфейс системы DocWebService

Анализ потребностей пользователя представлен в виде value proportions на рисунке 3.

Value proportions (ценностное предложение) — это сообщение, где кратко изложены причины, по которым клиенты выбирают ваши товары или услуги. В нём отражают наиболее очевидные выгоды, которые получит клиент, если купит ваш продукт.

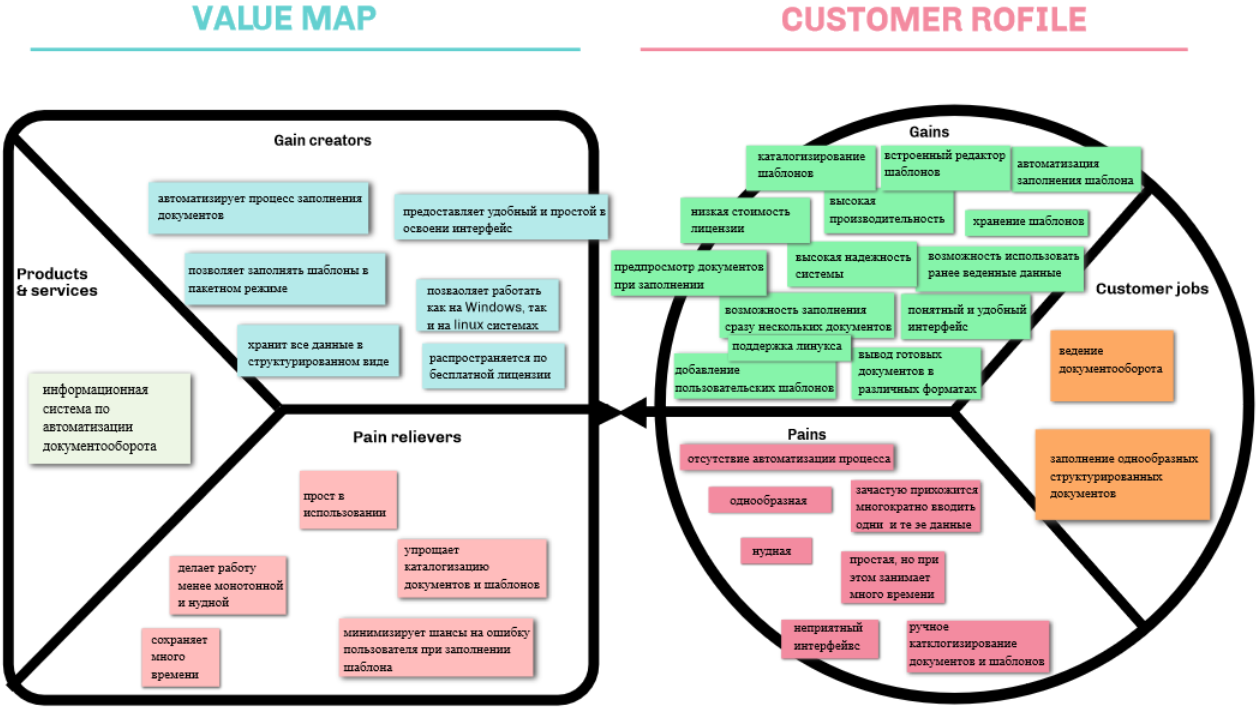


Рисунок 3 – Инфографика value proportions

**2.1.2 Уникальность проекта**

Исходя из представленного выше анализа была выявлена уникальность нашего проекта. Данная система отличается от аналогичных систем следующими особенностями:

* Встроенная коллекция шаблонов, освобождающая от необходимости подготавливать их самостоятельно
* Поддержка большого количества форматов входных и выходных данных
* Встроенный учёт шаблонов
* Удобный интерфейс
* Кроссплатформенность
* Поддержка полного цикла с создания шаблона до готового документа
* Создание, хранение и редактирование шаблонов
* Бесплатная лицензия

**2.2. Требования к программе**

### **2.2.1. Требования к составу выполняемых функций**

1. Создание документа по шаблону.
2. Хранение шаблонов в базе данных
3. Хранение готовых документов.
4. Вывод заполненных шаблонов в различных форматах
5. Загрузка пользовательских шаблонов в базу данных
6. Выбор шаблона по категориям
7. Пакетный режим создания документов по шаблону
8. Предпросмотр документа при заполнении
9. Преобразование формата
10. Добавление/Удаление пользовательских категорий
11. Удаление шаблона
12. Извлечение полей из загружаемого в систему файла для формирования шаблона
13. Поддержка графического пользовательского интерфейса
14. Хранение ранее введенных данных для дальнейшего использования в базе данных
15. Редактирование шаблонов

**2.2.2. Требования к организации входных данных**

Входными данными в системе могут быть файлы для шаблонов документов и данные необходимые для заполнения шаблона. Требования к данным, вводимым при создании документа, определяются спецификой шаблона.

Файлы для шаблонов принимаются системой в следующих форматах: doc, docx, rtf, html, txt.

В загружаемом файле поля для ввода данных должны быть обозначены угловыми скобками.

### **2.2.3. Требования к организации выходных данных**

Поддерживаемые форматы выходных данных: doc, docx, odt, ooxml, rtf, pdf, html, txt, md. Готовые документы хранятся на устройстве пользователя в выбранной им директории. Директория выбирается пользователем в системе.

### **2.2.4. Требования к организации хранения данных**

В системе могут храниться шаблоны документов. Шаблоны хранятся в базе данных в формате XML.

### **2.2.5. Требования к графическому интерфейсу**

Программа должна обеспечивать взаимодействие с пользователем посредством графического пользовательского интерфейса (GUI).

* Отображение базы данных шаблонов в виде древовидной структуры
* Предпросмотр документа при заполнении
* Использование исключительно панелей инструментов для навигации
* Интерфейс заполнения шаблона в виде экранной формы
* В пакетном режиме предусмотрена возможность перехода от одного экземпляра к другому
  + 1. **Требования к программным средствам, используемым программой**

Системные программные средства, используемые ИС, должны быть представлены локализованной версией операционной системы Windows и Linux.

**2.3. ИТ-технологии для реализации проекта**

Для управления проектом были выбраны следующие инструменты:

* + YouTrack – онлайн-доска, инструмент для планирования задач, отслеживания процесса разработки проекта, оценки и учета времени [9].
  + GitHub – сервис для разработчиков проекта, позволяющий удобно вести коллективную разработку IT-проектов. Здесь можно публиковать и редактировать свой код, комментировать чужие наработки [10].
  + Discord – инструмент для коммуникации команды, проведения ежедневных совещаний и встреч.

Язык программирования: С#.

Используемые библиотеки и Фреймворки:

* + Avalonia UI
  + Aspose Words for .NET
  + LiteDB

## **2.4. Выбор методологии разработки проекта**

Для разработки проекта была выбрана Scrum-методология.

Разработку проекта было решено разделить на четыре спринта длительностью в 1 неделю.

Раздел 3. Выполненная работа по проекту

3.1 Спринт 1 (02.06-08.06)

3.1.1 Цель

Цель спринта:

* + - исследовать предметную область документооборота,
    - проанализировать существующие аналоги и потребности пользователей,
    - выявить актуальность проекта,
    - подготовить ТЗ,
    - оценить риски проекта,
    - исходя из требований проекта исследовать возможные ИТ-решения,
    - разработать архитектуру системы,
    - подготовить прототип интерфейса,
    - подготовить шаблоны документов.

3.1.2 Планирование задач

Бэклог спринта:

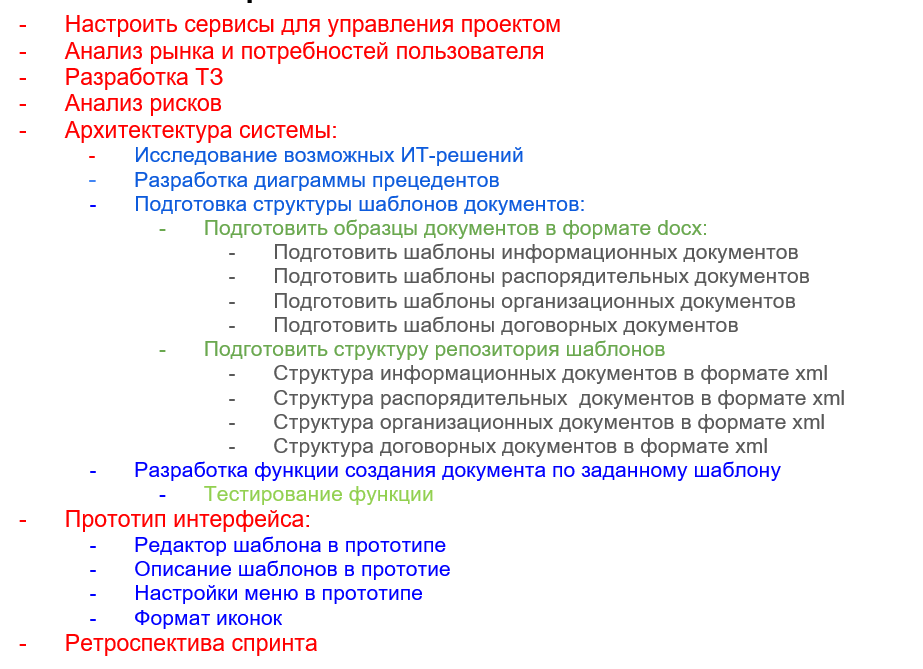


Рисунок 4 – Бэклог спринта 1

3.1.3 Выполненные задачи

* + - Проведен анализ предметной области
    - Изучена библиотек работы с документами
    - Изучен формат Open Office XML
    - Выбран библиотеки и Фреймворки для разработки проекта
    - Построена диаграммы прецедентов. Представлена на рисунке 5.
    - Проведен анализ рынка и потребностей пользователя. Представлен в пункте 2.1.1.
    - Разработана первая версия прототипа интерфейса системы. Представлена на рисунках 6, 7 и 8.
    - Разработана функции для формирования полей для заполнения шаблона
    - Настроена инфраструктуру: организация системы управления проектом и процессов в YouTrack, организация каналов связи для встреч и коллективной разработки в GitHub
    - Подготовлено ТЗ
    - Проведен анализ рисков. Представлен на рисунке 9.
    - Подготовлены системные шаблоны. Пример представлен на рисунке 10.

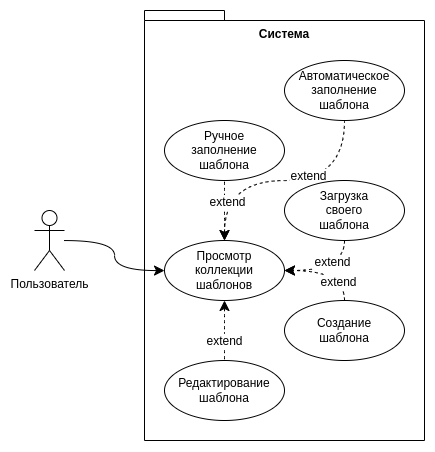


Рисунок 5 – Диаграмма прецедентов

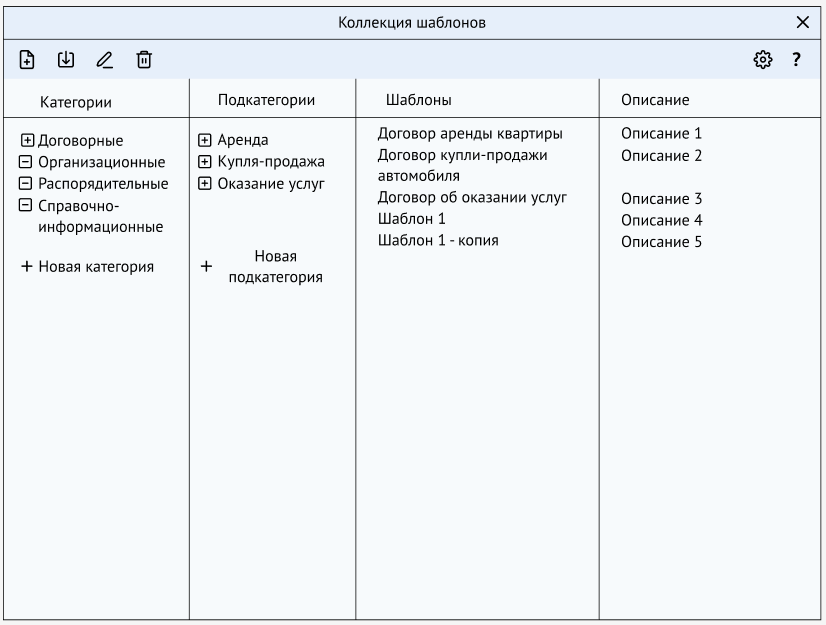


Рисунок 6 – Меню в прототипе интерфейса.

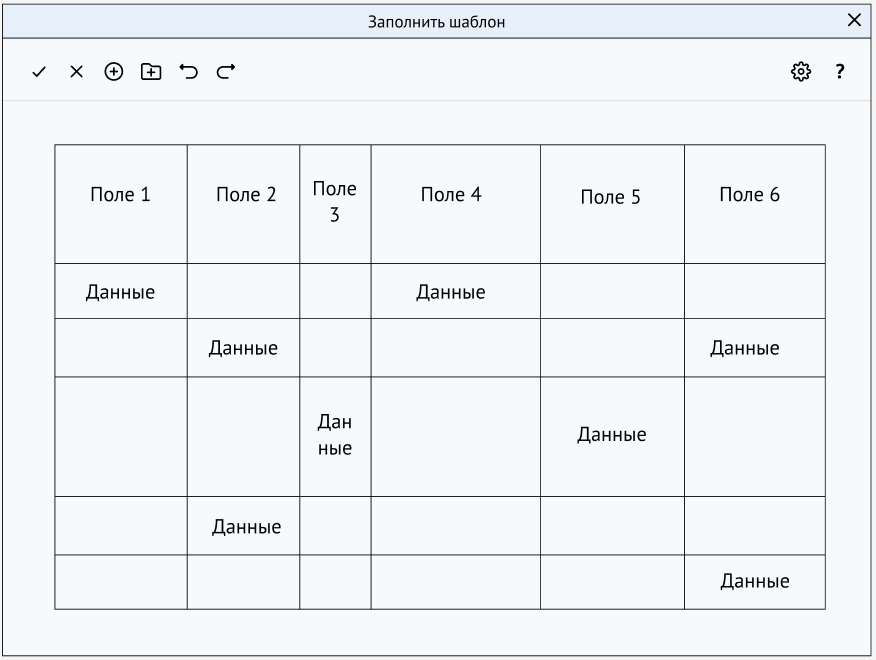


Рисунок 7 – Заполнение шаблона в прототипе интерфейса.

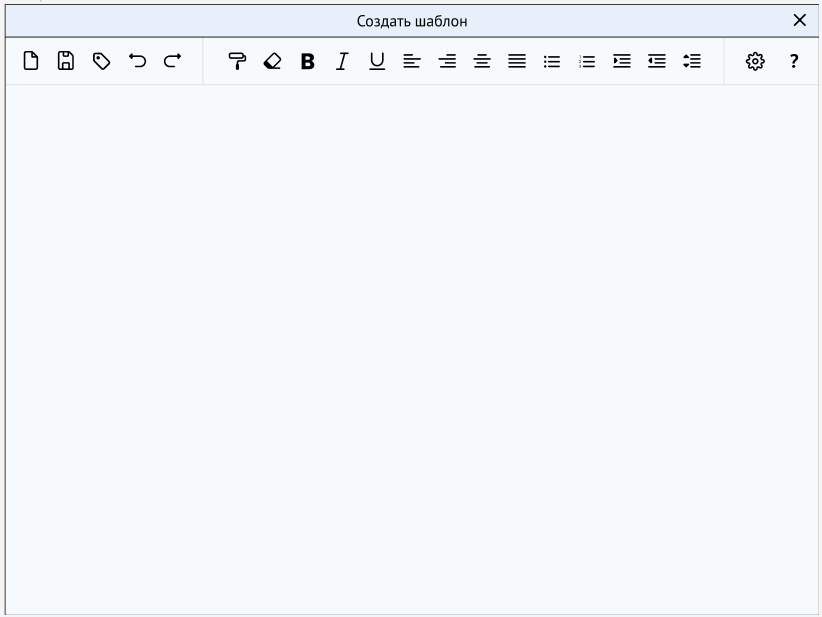
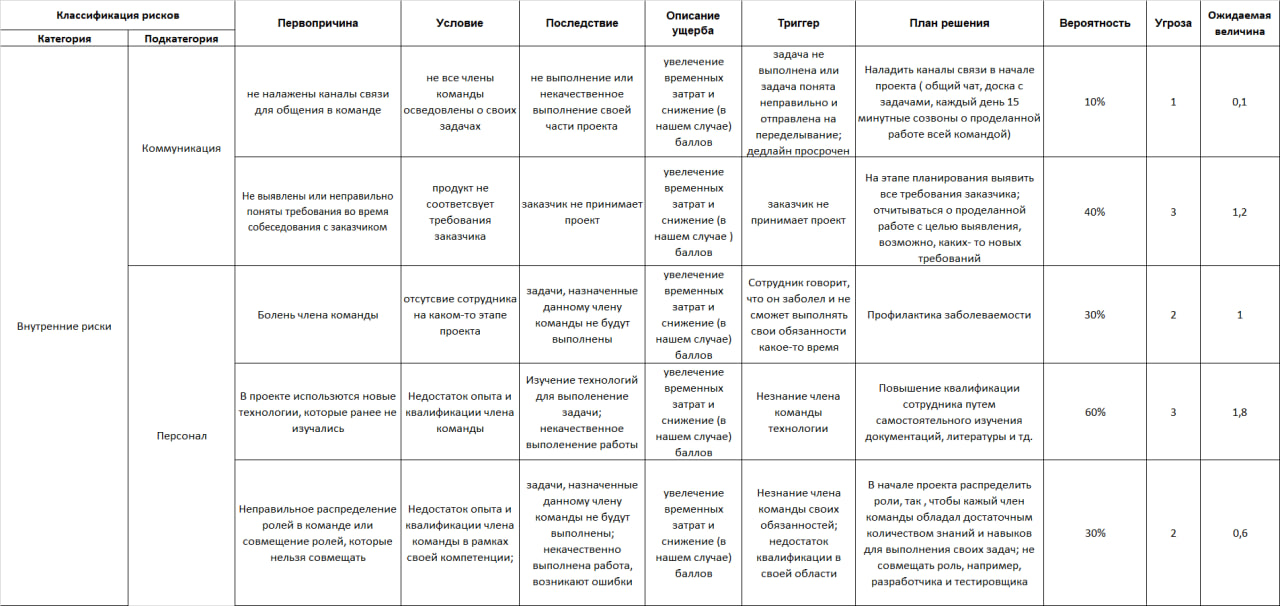
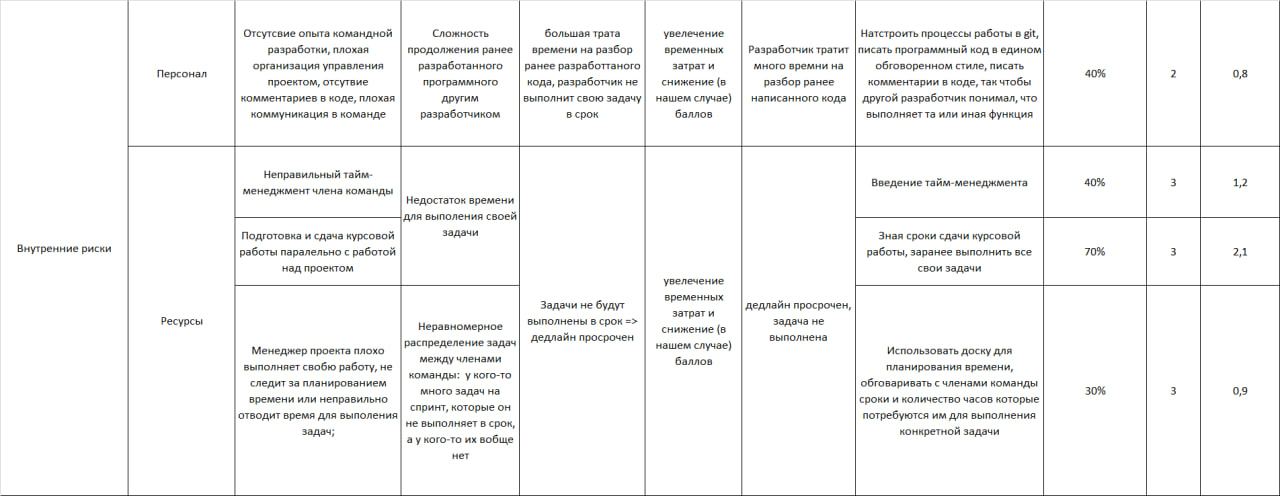


Рисунок 8 – Создание шаблона в прототипе







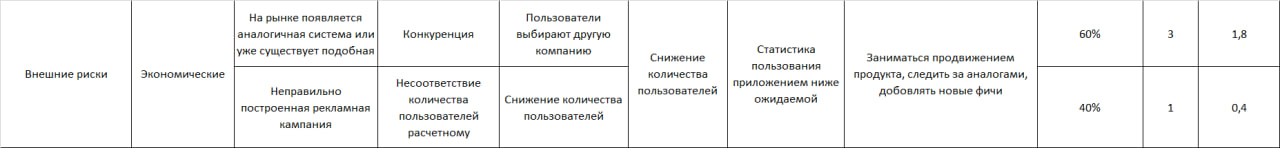


Рисунок 9 – Анализ рисков

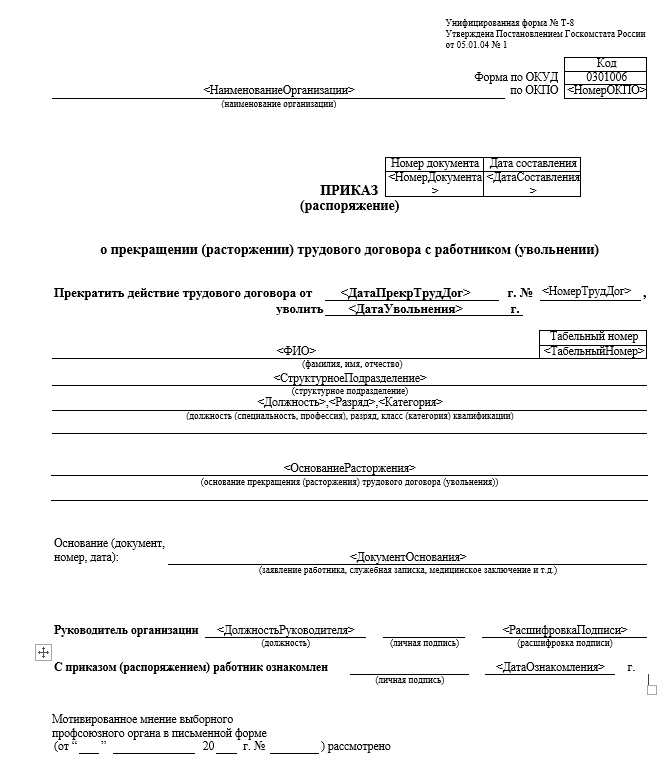


Рисунок 10 – Пример подготовленного шаблона

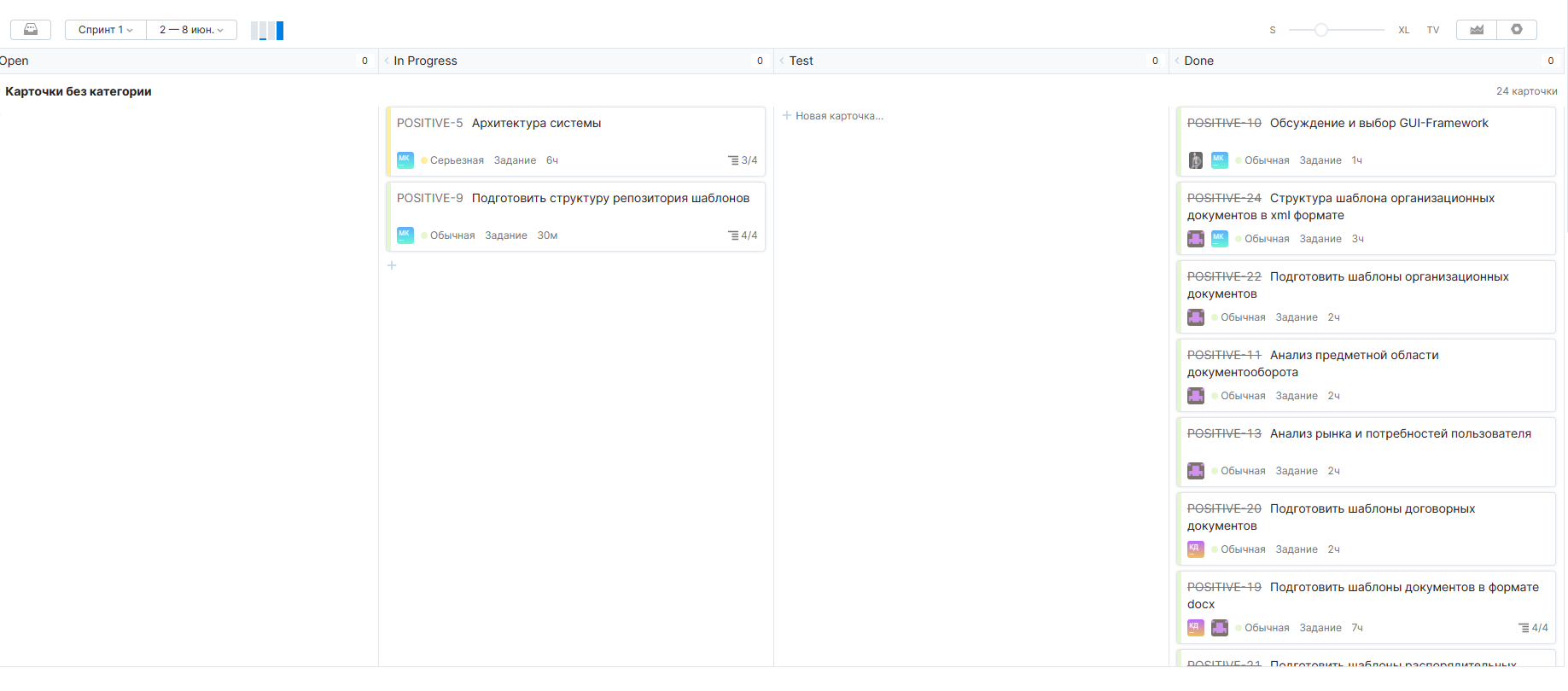


Рисунок 11 – Выполнение задач спринт 1

* + 1. Ретроспектива

Что было положительного?

* Хорошая коммуникация в команде
* Понятные задачи и сроки выполнения
* Дружный коллектив
* Все заняты своими задачами, каждый привносит вклад в разработку проекта

Что было отрицательного?

* Ежедневные 15-минутные совещания затягиваются по времени
* Недостаточное выделение времени проекту
* Разброс идей и предложений, нет целостности понимая, как будет работать
* Изначально не было разбиения задач на более мелкие

Как исправить?

* Проводить отдельно технические совещания и встречи
* Начать рабочий день раньше
* Завести матрицу идей
* Декомпозиция задачи

3.2 Спринт 2 (09.06-15.06)

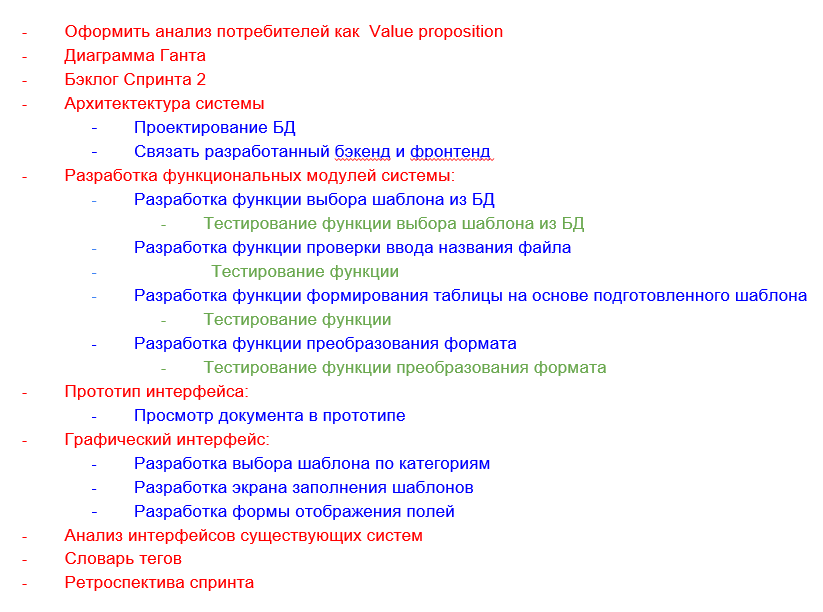
3.2.1 Цель

Цель спринта:

* + - Зафиксировать стандартный набор описаний полей шаблонов в словаре
    - Разработать первоначальный графический интерфейс
    - Связать бэкенд и фронтенд
    - Создание БД для хранения шаблонов и данных пользователей

3.2.2 Планирование задач

Бэклог спринта:



3.2.3 Выполненные задачи

* + - Построена модель БД. Представлена на рисунке 12.
    - Разработан словарь тегов
    - Разработан первоначальный графический интерфейс системы. Представлен на рисунке 13, 14.
    - Проведен анализ по улучшению интерфейса нашей системы

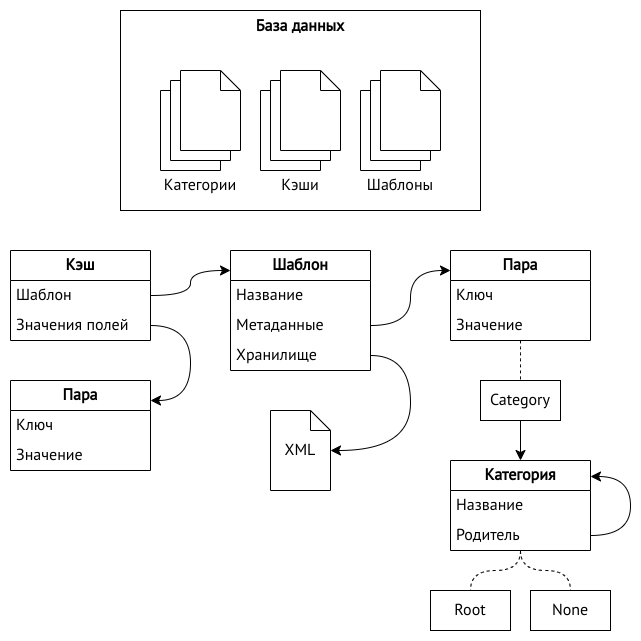


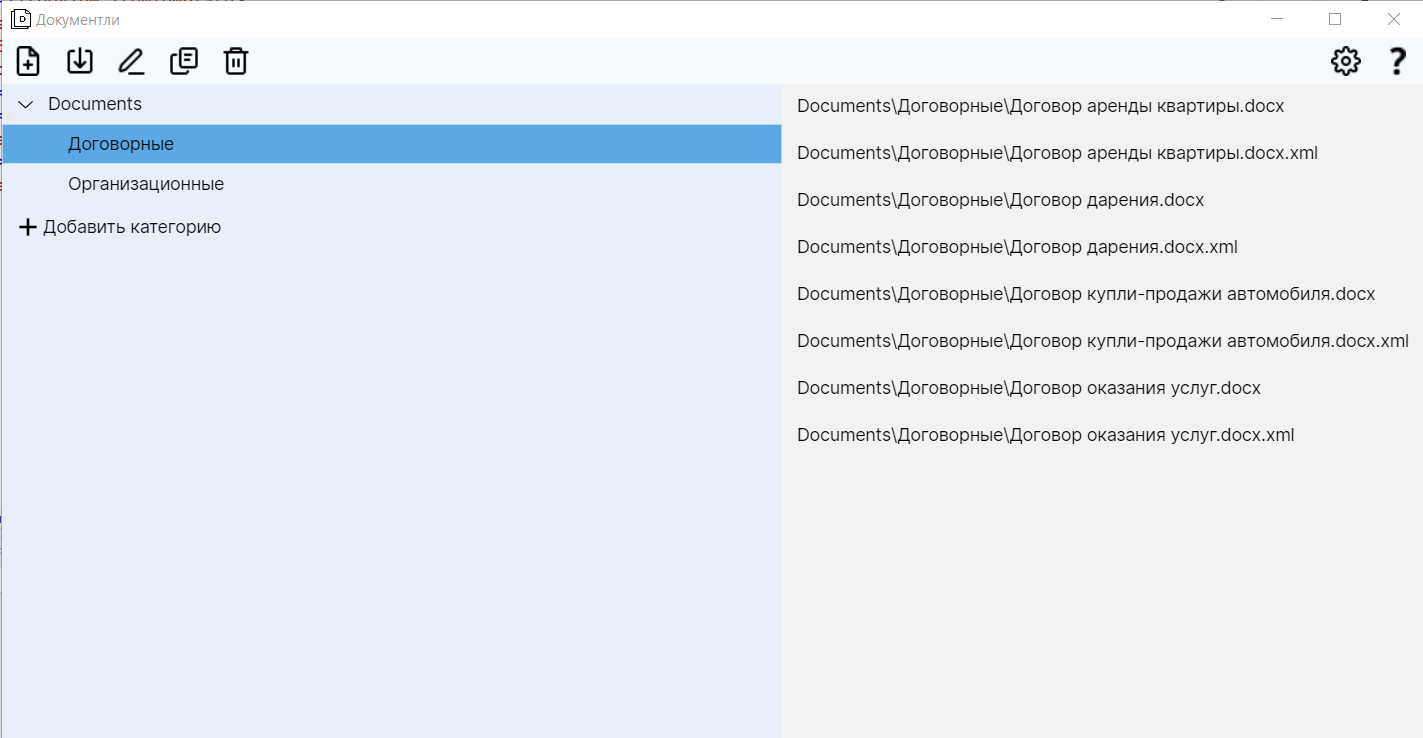
Рисунок 12 – Модель БД

Рисунок 13 – Коллекция шаблонов

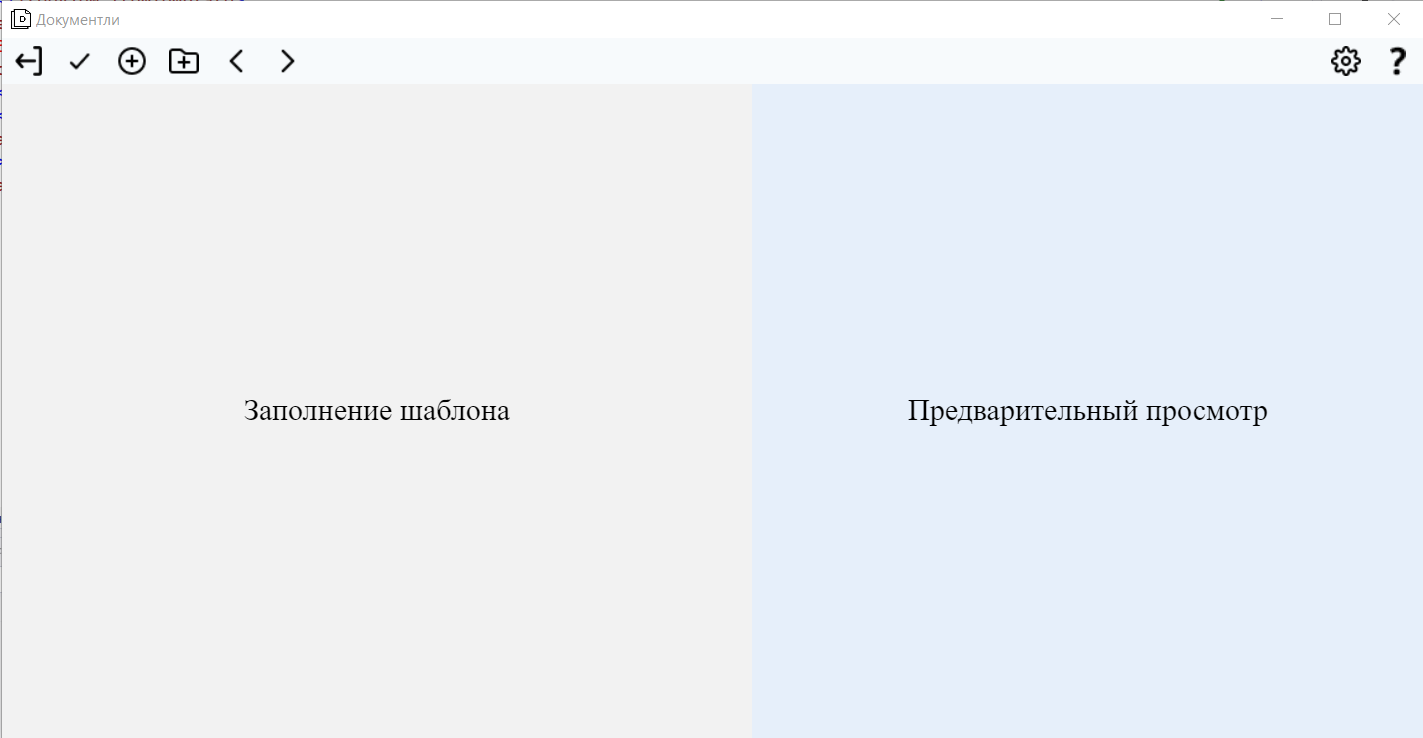


Рисунок 14 –Заполнение шаблона

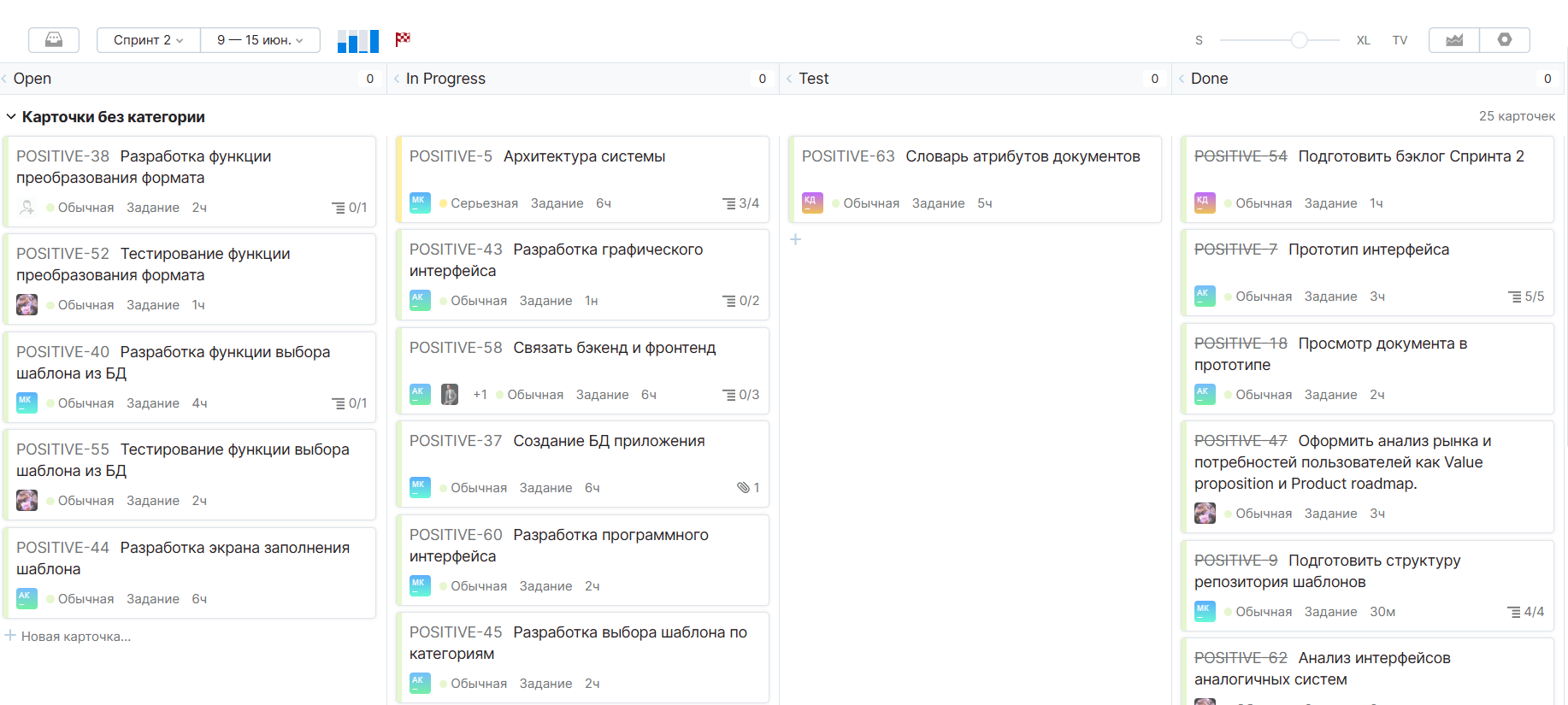


Рисунок 15 – Выполнение задач спринт 2

3.2.4 Ретроспектива

Что было положительного?

* Хорошая коммуникация в команде
* Появилось разделение 15-минуток и встреч
* Декомпозиция задач
* Появилось общее понимание разрабатываемой системы, так как все требования описаны в ТЗ

Что было отрицательного?

* Нехватка времени на выполнение задач из-за высокой загруженности
* Команда не до конца была погружена в разработку проекта
* Не все задачи в спринте были выполнены

Как исправить?

* Удвоить продуктивность работы команды в два раза
* грамотно оценивать и планировать время выполнения задач

3.3 Спринт 3 (16.06-22.06)

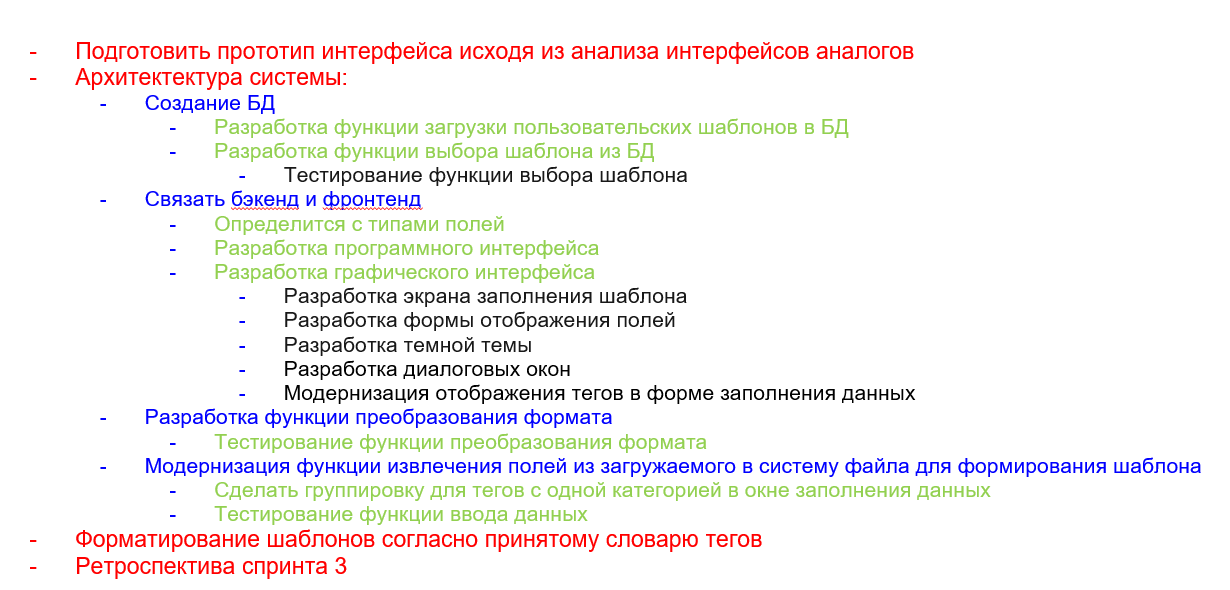
3.3.1 Цель

Цель спринта:

* Создать БД
* Связать БД с бэком и фронтом
* Доработать прототип интерфейса согласно анализу интерфейсов
* Разработка функциональных модулей программы: преобразование формата, выбор шаблона по категориям из БД, формирование полей ввода данных и заполнение данных в шаблон
* Повысить продуктивность спринта

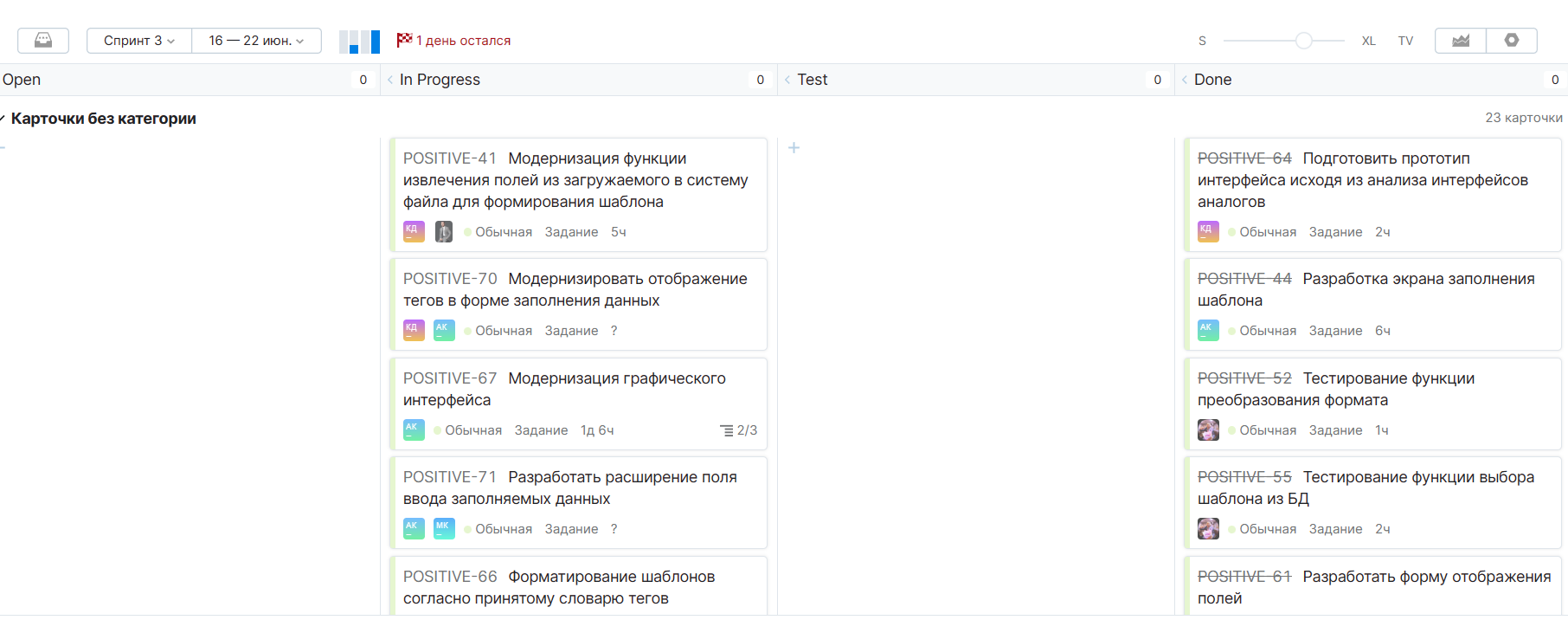
3.3.2 Планирование задач

Бэклог спринта:



3.3.3 Выполненные задачи

* Создана база данных для хранения шаблонов
* Разработана функция загрузки пользовательских шаблонов в БД
* Разработана функция выбора шаблона из БД
* Связан бэкенд и фронтенд
* Разработан экрана заполнения шаблона и формы отображения полей. Показана на рисунке 15 и 16.
* Разработана функция преобразования формата
* Разработана функция удаления/добавления пользовательских категорий в системе
* Разработана функция удаления шаблона из БД
* Проведено тестирование функций выбора шаблона, загрузки шаблона, заполнения данных, преобразования формата и вывода готового документа. План тестирования и чек-лист тестирования представлен на рисунке 17 и 18.



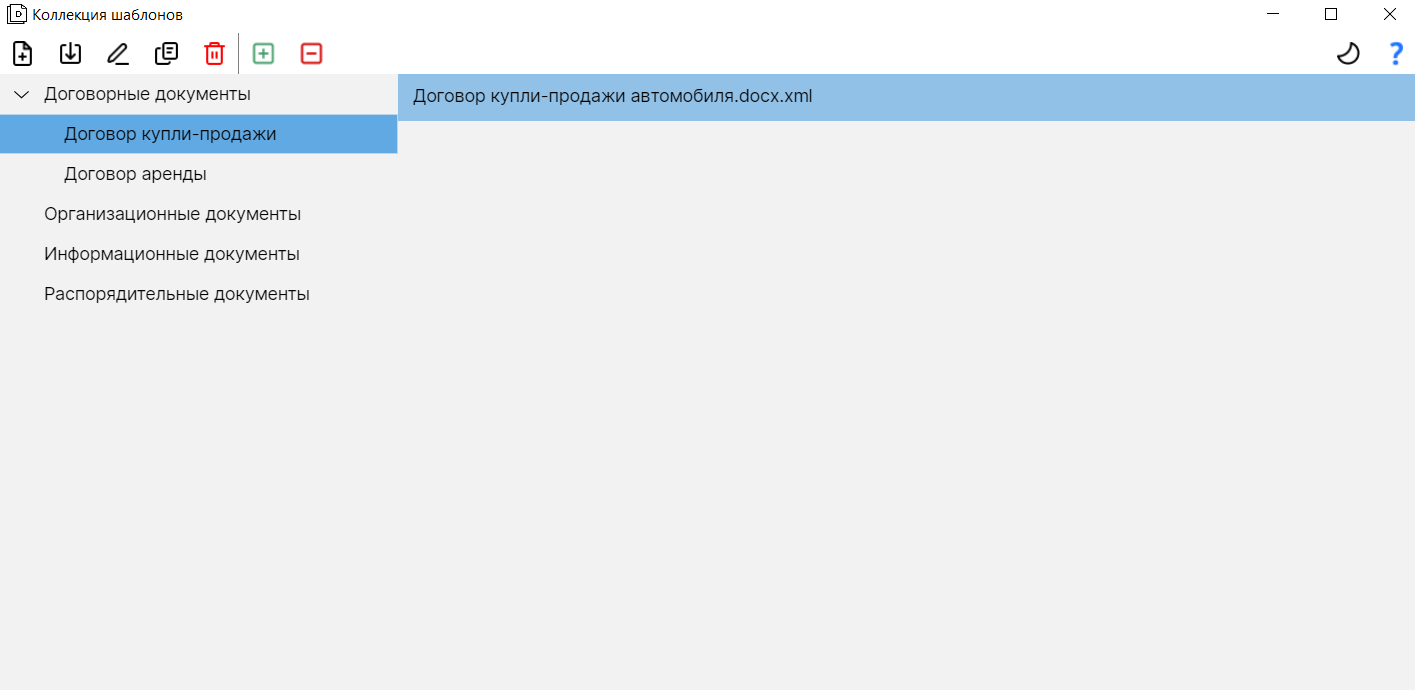


Рисунок 16 – Выбор шаблона в системе

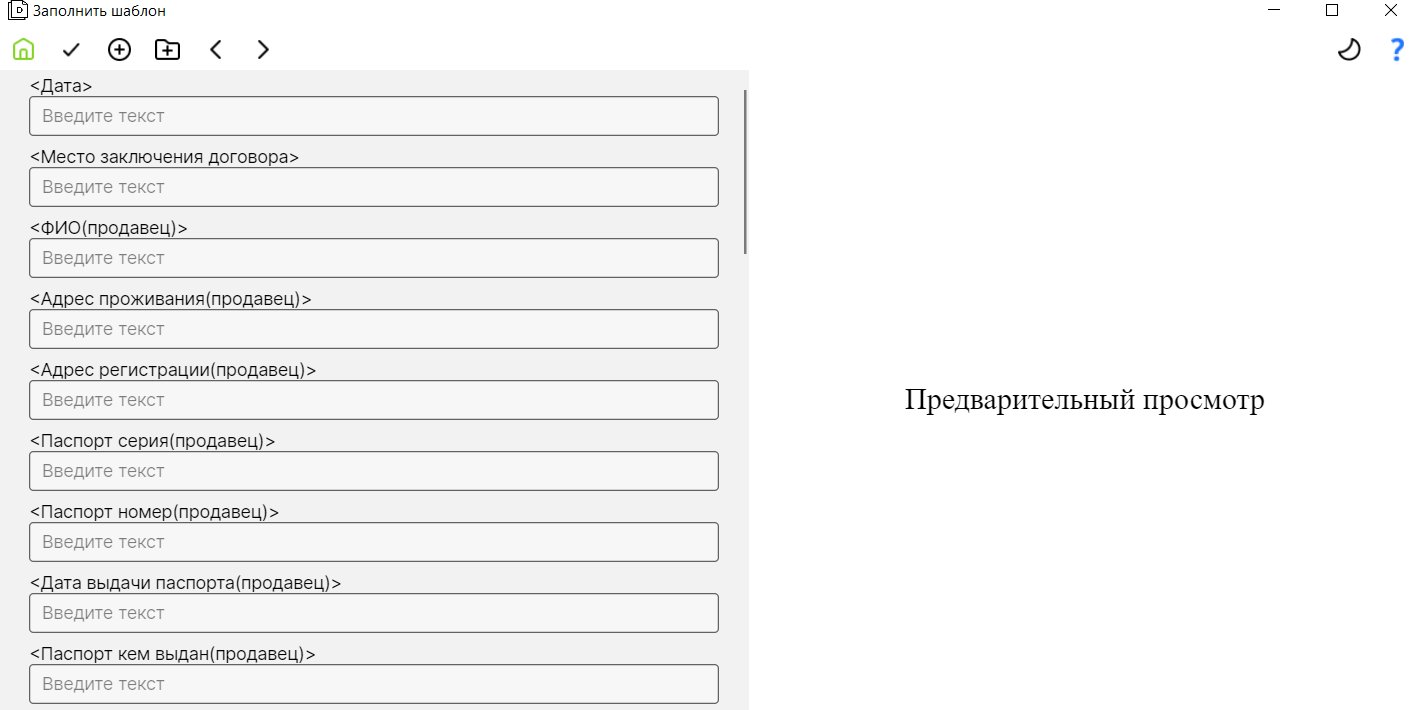


Рисунок 17 – Заполнение шаблона в системе

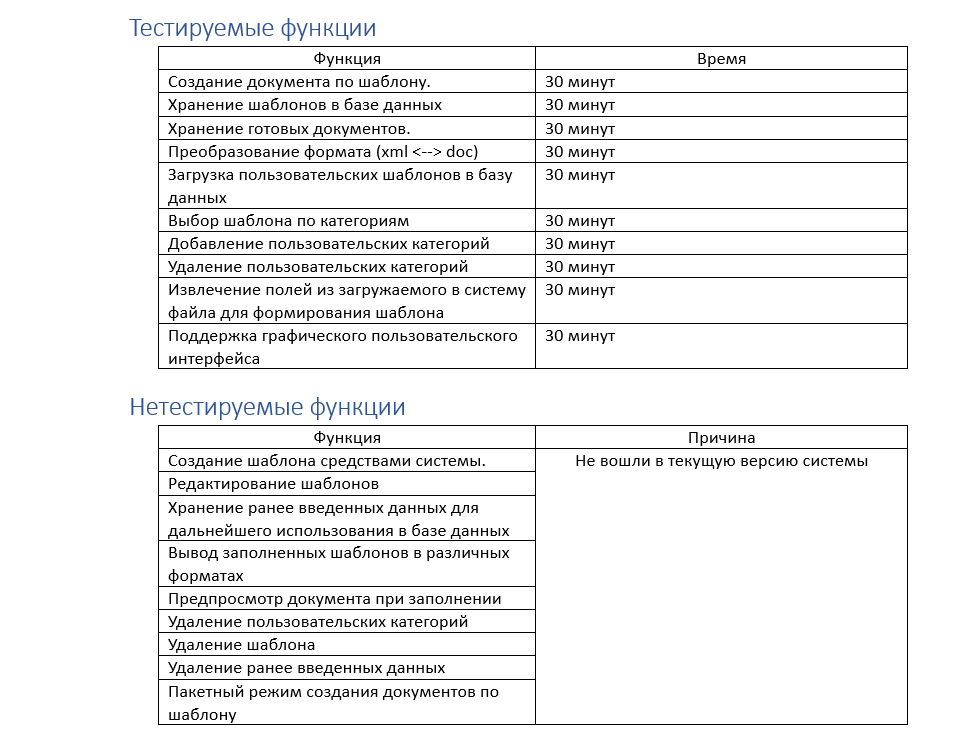


Рисунок 18 – Тестирование функций

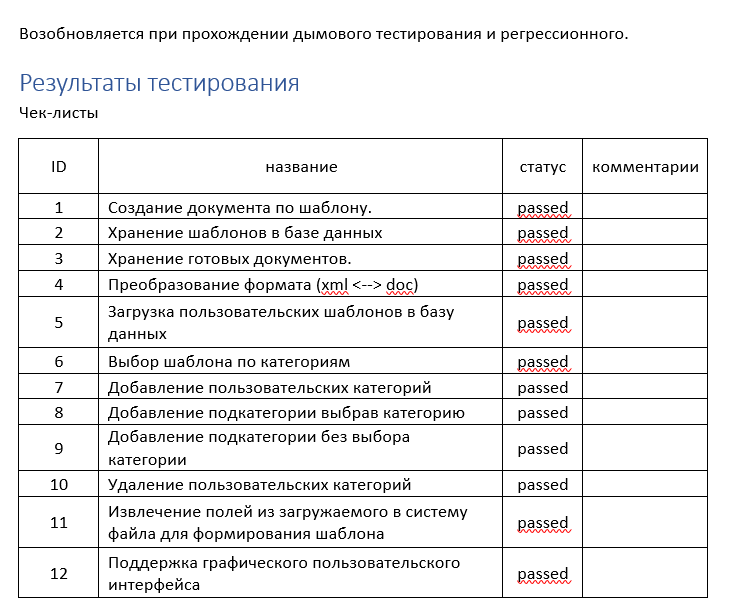


Рисунок 19 – Чек-лист тестирования

3.3.4 Ретроспектива

Что было положительного?

* Хорошая коммуникация в команде
* Продуктивная работа всей команды
* Большое количество выполненных задач

Что было отрицательного?

Не все задачи в спринте были выполнены, так как добавлялись новые в процессе работы

Как исправить?

Стараться не добавлять новые задачи в текущий спринт или добавлять их, но с учетом того, что они точно будут выполнены

3.4 Спринт 4 (23.06-29.06)

3.4.1 Цель

Цель спринта:

* Разработка функциональных модулей системы: предварительный просмотр документа, пакетный режим создания документов по шаблону, вывод заполненных шаблонов в различных форматах, хранение введённых пользовательских данных
* Тестирование функциональных модулей
* Модернизация графического интерфейса
* Модернизация функции формирования полей шаблона
* Подготовить групповой отчет
* Подготовить руководство пользователя

3.4.2 Планирование задач

Бэклог спринта:



3.4.3 Выполненные задачи

-Разработана функция пакетного режима заполнения шаблона

-Разработана функция вывода готового документа в различных форматах

-Разработана функция предварительного просмотра документа

-Модернизирована функция формирования ролей для заполнения шаблона и отображение названий полей при заполнении.

-Модернизировано отображение полей заполнения шаблона. Заполнение шаблона представлено на рисунке 20.

-Модернизировано отображение названий полей

-Добавлено поле для заполнения даты. Показано на рисунке 21 и 22.

-Добавлены диалоговые окна. Показано на рисунке 23.

-Добавлена статистика количества шаблонов в категориях

-Разработана темная тема, удаление шаблона, переименование категорий

-Связаны преобразование формата и загрузка шаблона в БД

-Разработана справка для пользователей

-Разработан логотип системы. Представлен на рисунке 25.

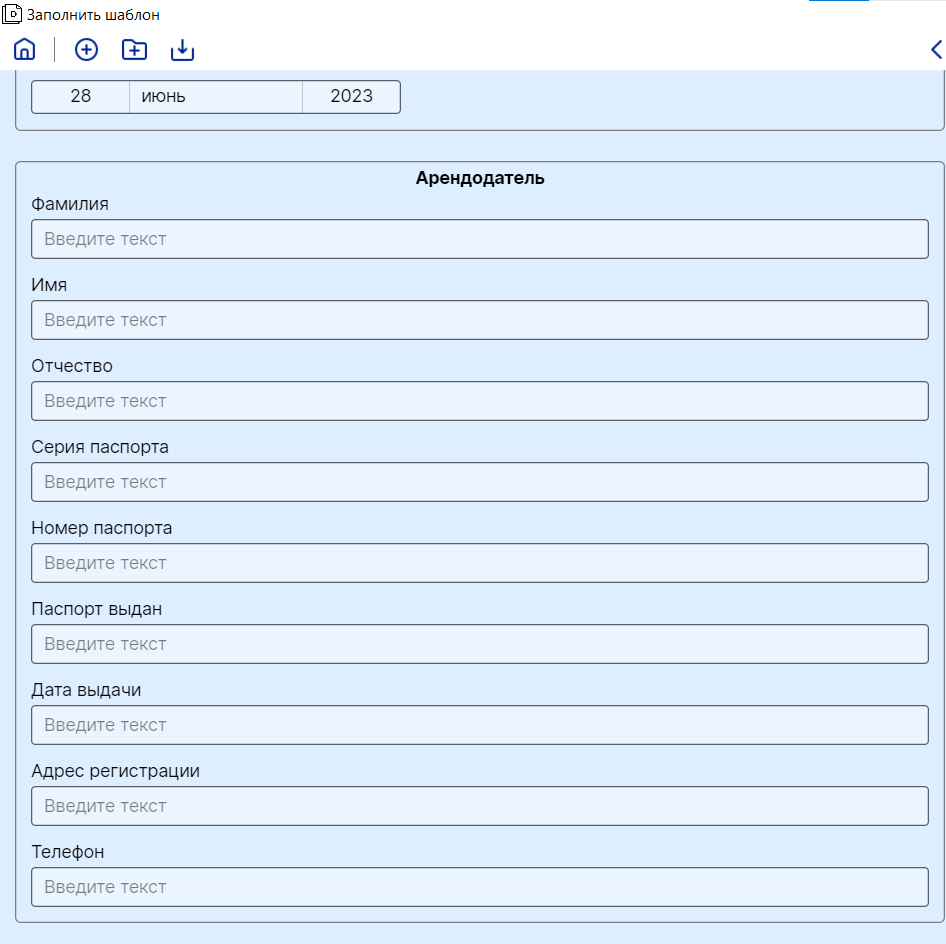


Рисунок 20 – Заполнение шаблона



Рисунок 21 – Отображение полей категории «Дата» в системе

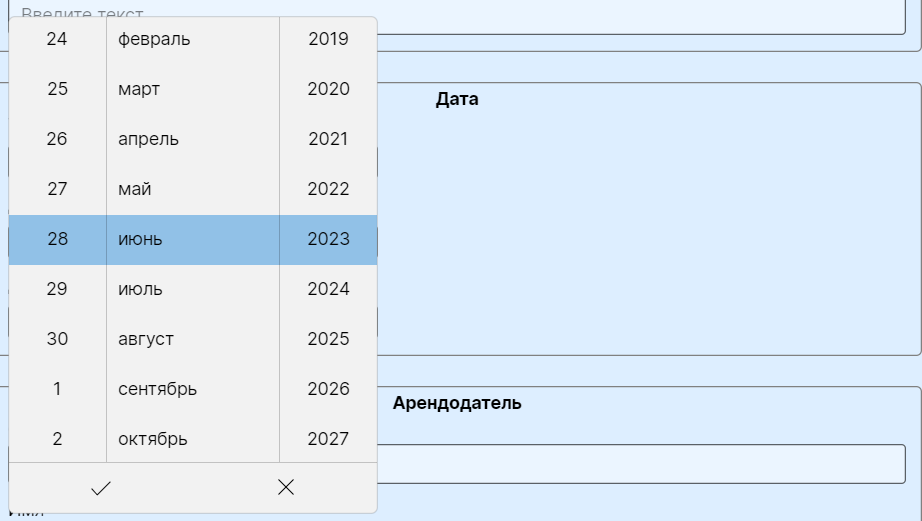


Рисунок 22 – Заполнение полей категории «Дата»

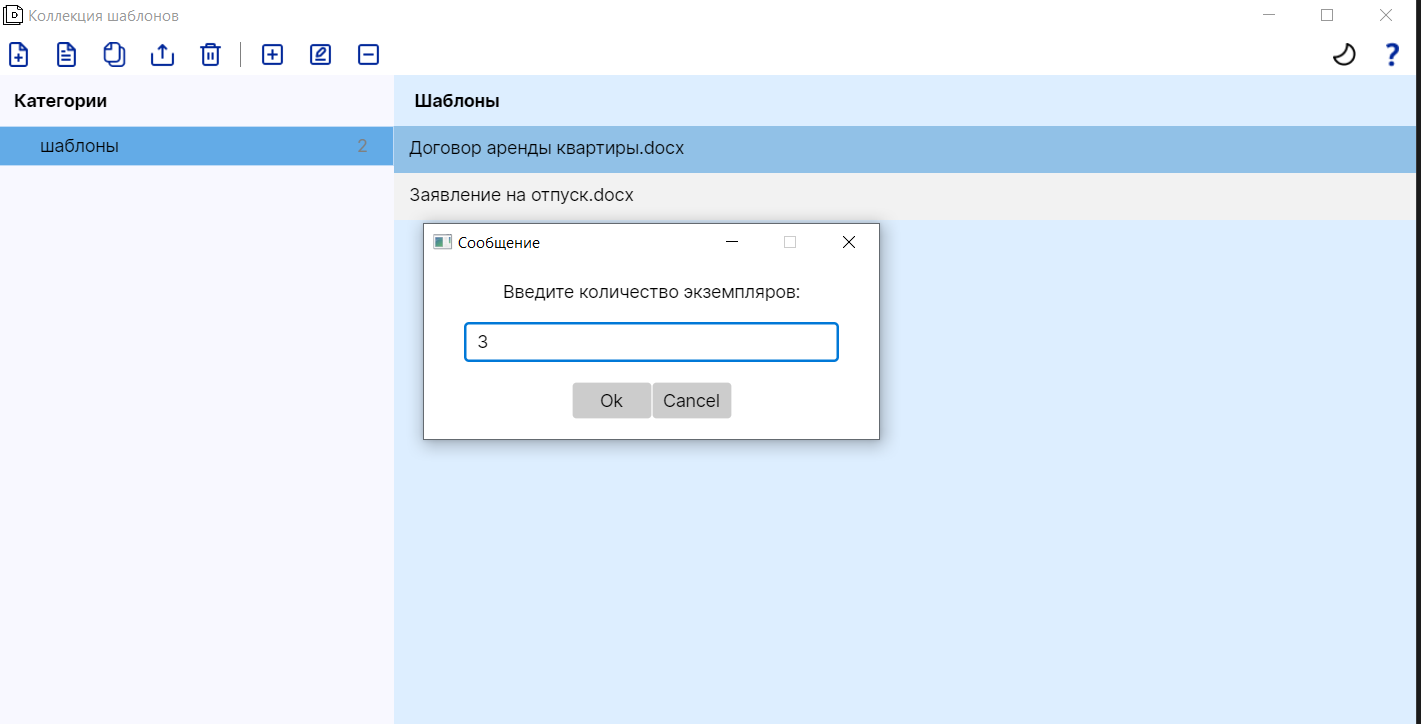


Рисунок 23 – Диалоговое окно при заполнении шаблона

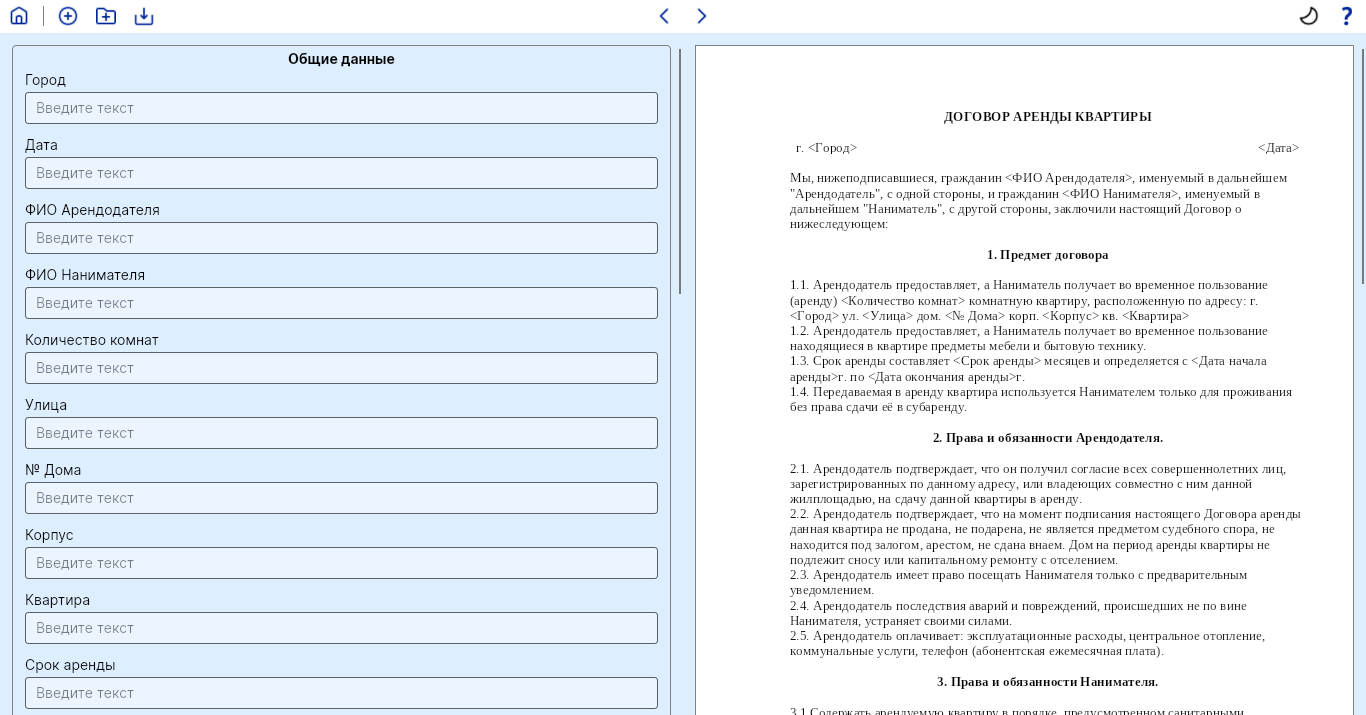


Рисунок 24 – Предпросмотр документа

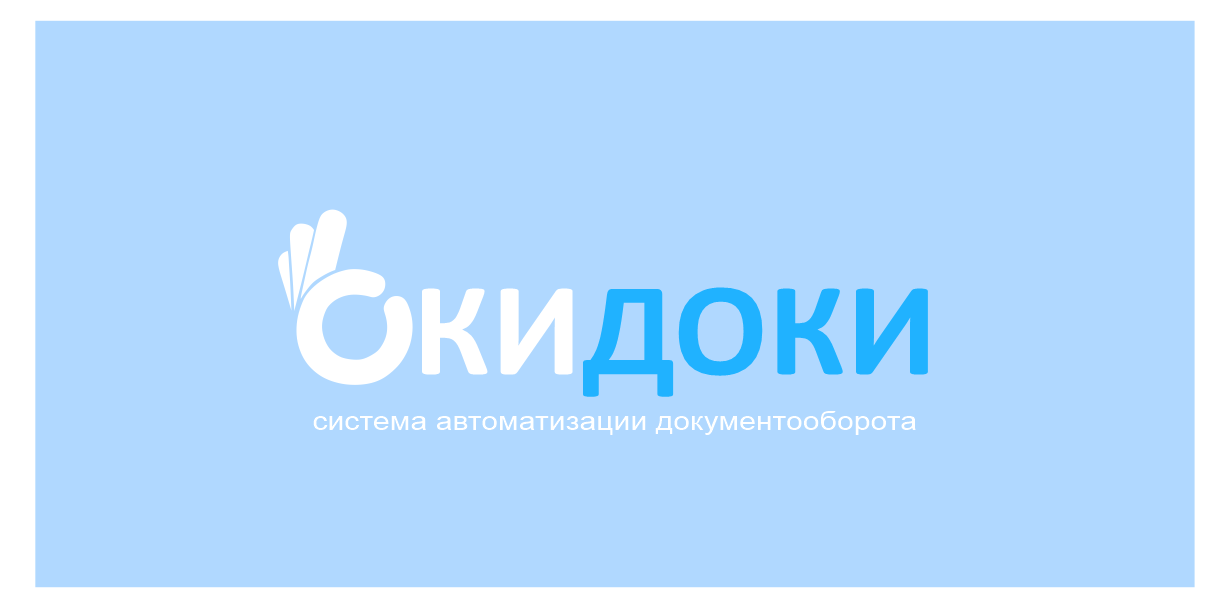


Рисунок 25 – Логотип системы «ОкиДоки»

3.4.4 Ретроспектива

Что было положительного?

* Продуктивная работа всей команды
* Большое количество выполненных задач

Что было отрицательного?

* Много времени затрачено на написание отчета
* Большое количество задач для выполнения, так как это заключительный спринт

## **3.5 Выполненный функционал системы**

Реализованный заявленный функционал системы:

* + 1. Создание документа по шаблону.
    2. Хранение шаблонов в базе данных
    3. Хранение готовых документов.
    4. Вывод заполненных шаблонов в различных форматах
    5. Загрузка пользовательских шаблонов в базу данных
    6. Выбор шаблона по категориям
    7. Пакетный режим создания документов по шаблону
    8. Предпросмотр документа при заполнении
    9. Преобразование формата
    10. Добавление/Удаление пользовательских категорий
    11. Удаление шаблона
    12. Извлечение полей из загружаемого в систему файла для формирования шаблона
    13. Поддержка графического пользовательского интерфейса

Нереализованные функции:

* + 1. Хранение ранее введенных данных для дальнейшего использования в базе данных
    2. Создание шаблона средствами системы
    3. Редактирование шаблонов средствами системы

Библиографический список

1. Командная работа [Электронный ресурс]: Википедия. Свободная энциклопедия. URL: <https://en.m.wikipedia.org/wiki/Teamwork> (дата обращения: 25.06.2023).
2. Методологии разработки ПО [Электронный ресурс]: Хабр. URL: <https://habr.com/ru/sandbox/43802> (дата обращения: 25.06.2023).
3. Управление проектами по Agile методологим [Электронный ресурс]: Хабр. URL: <https://habr.com/ru/companies/otus/articles/710034/> (дата обращения: 25.06.2023).
4. Бэклог продукта [Электронный ресурс]: Википедия. Свободная энциклопедия. URL: [Бэклог продукта — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%91%D1%8D%D0%BA%D0%BB%D0%BE%D0%B3_%D0%BF%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82%D0%B0) (дата обращения: 26.06.2023).
5. Кто такой Scrum Master [Электронный ресурс]: Яндекс Практикум. URL: <https://practicum.yandex.ru/blog/kto-takoy-scrum-master/> (дата обращения: 26.06.2023).
6. Основные принципы MSF [Электронный ресурс]: НОУ ИНТУИТ. <URL:https://intuit.ru/studies/professional_skill_improvements/1419/courses/353/lecture/8416> (дата обращения: 26.06.2023).
7. Автоматическая генерация документов. R6R [Электронный ресурс]. URL: <https://r6r.ru/> (дата обращения: 27.06.2023)
8. Eforms [Электронный ресурс]. URL: <https://eforms.com/> (дата обращения: 27.06.2023)
9. YouTrack [Электронный ресурс]. URL: <https://knplvaa.youtrack.cloud> (дата обращения: 28.06.2023)
10. GitHub [Электронный ресурс]. URL: <https://github.com/teamdevm/teamwork-positive> (дата обращения: 28.06.2023)

Приложение 1. Личный отчет по проекту Коноплевой Д.Д

1. Тема проекта: проектирование и создание системы автоматизации документооборота.

2. Роль(и) в проекте:

* Руководитель проекта
* Менеджер проекта
* Управление продуктом (спринт 4)
* Разработчик (спринт 3)

3. Проделанная работа

* Настроила инфраструктуру: организация системы управления проектом и процессов в YouTrack и GitHub, организация каналов связи для встреч
* Участвовала в планировании и формировании бэклога спринтов
* Распределила задачи спринтов между членами команды
* Занималась планированием времени и контролированием процесса выполнения задач
* Провела анализ рисков
* Подготовила ТЗ
* Провела исследование предметной области документообразования
* Подготовила системные шаблоны документов
* Создавала описание задачи для разработчиков
* Занималась подготовкой документации и отчетов проекта: групповой отчет, презентация, словарь тегов, руководство пользователя
* Модифицировала функцию формирования полей для заполнения

Приложение 2. Личный отчет по проекту Колпащикова М.Н

1. Тема проекта: проектирование и создание системы автоматизации документооборота.

2. Роль(и) в проекте:

* Архитектор системы
* Разработчик

3. Проделанная работа

* сделан анализ инструментальных средств и библиотек,
* разработана диаграмма прецедентов,
* изучен формат Open Office XML,
* создана MVVM модель приложения,
* создана модель базы данных приложения,
* реализован API базы данных,
* реализован интерфейс взаимодействия бэкенда и фронтенда,
* участие в разработке объектной модели полей документа,
* разработан виджет предварительного просмотра документа
* участвовал в планировании и формировании бэклога спринтов

Приложение 3. Личный отчет по проекту Коваль А.Е.

1. Тема проекта: проектирование и создание системы автоматизации документооборота.

2. Роль(и) в проекте

- user expirience (Спринт 1)

- frontend разработчик

3. Проделанная работа

На первом спринте был разработан прототип интерфейса. Результат представлен на рисунках 6, 7 и 8.

На втором спринте началась реализация графического интерфейса на основе данного прототипа. Табличная структура категорий и подкатегорий стала древовидной. Появилась возможность выбирать шаблон по заданной категории. Таблица на форме «Заполнить шаблон». Появилось поле для предпросмотра документа. Интерфейс представлен на рисунках 13 и 14.

На третьем спринте продолжилась реализация графического интерфейса. Была начата разработка тёмной темы. Появились диалоговые окна, которые позволяют пользователю и приложению взаимодействовать друг с другом. Через диалоговые окна были реализованы добавление категорий и подкатегорий, загрузка шаблонов. Стало возможным сохранять документ, созданный на основе шаблона. Интерфейс представлен на рисунках 16 и 17.

На четвертом спринте был реализован графический интерфейс. Также появилась темная тема. Через диалоговые окна были реализованы изменение и удаление категорий, удаление шаблонов. Появился пакетный режим – можно создавать несколько экземпляров на основе одного и того же шаблона. Интерфейс представлен на рисунках 20, 21, 22 и 23.

Приложение 4. Личный отчет по проекту Лязгина Д.С.

1. Тема проекта: проектирование и создание системы автоматизации документооборота.

2. Роль(и) в проекте

* Управление продуктом (Спринт 1,2)
* Тестирование (Спринт 3,4)

3. Проделанная работа

Спринт 1 + 2:

* Провел анализ рынка и потребностей пользователя
* оформил value proposition на основе анализа рынка и потребностей пользователя
* провел анализ предметной области документооборота
* определил с какими шаблонами документов будет работать система
* разрабатывал ТЗ
* подготовил шаблоны документов
* провел анализ интерфейса аналогичных систем
* на основе анализа интерфейсов аналогичных систем выявил ряд предложений по доработкам текущей итерации интерфейса
* участвовал в планировании и формировании бэклога спринтов

Спринт 3 + 4:

* составил тест-план
* подготовил среду для управления тестированием
* провел тестирование функции выбора шаблона из БД
* провел тестирование функции преобразования формата
* провел тестирование ввода данных
* провел тестирование функции проверки ввода названия создаваемого файла
* провел тестирование функции формирования таблицы из подготовленного шаблона
* провел тестирование функции вывода шаблонов в различных форматах
* провел тестирование функции пакетного режима создания документов

Приложение 5. Личный отчет по проекту Япарова А.Э.

1. Тема проекта: проектирование и создание системы автоматизации документооборота.

2. Роль(и) в проекте

* Разработчик

3. Проделанная работа

* функция формирования шапки таблицы по шаблону с
* разделением извлеченных данных на категории;
* функция заполнения документа данными пользователя;
* функция для проверки корректности названия итогового
* документа;
* функция сохранения документа в различных форматах;
* сбор технической информации для модернизации и реализации
* функций;
* разработка логотипа системы.
* участвовал в планировании и формировании бэклога спринтов