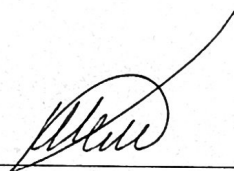


МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ  
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

**КУРСОВОЕ ПРОЕКТИРОВАНИЕ**  
Формальный документооборот в коммерческой  
деятельности компаний использующих  
медицинские информационных систем  
Б-ИН-21-136150. 8847-с. 23. 000. КП

Студент  
гр. БИН-21-1



С.Д. Шемилин

Руководитель  
доцент, к.т.н.



Д.И. Харитонов

Владивосток 2025

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВВГУ»)

Институт информационных технологий и анализа данных  
Кафедра информационных технологий и систем

Индивидуальное задание  
на производственную технологическую (проектно-технологическую) практику

Студенту гр. БИН-21-1 Шемилину Сергей Дмитриевичу

1. Тема работы: Формальный документооборот в коммерческой деятельности компаний использующих медицинские ИС

Утверждена приказом № 8847-с от 03.10.2024 г.

2. Срок сдачи работы: 16.01.2025 г.

3. Техническое задание

3.1 Цель

Спроектировать и разработать прототип модуля автоматического формирования документов.

3.2. Технические требования

Спроектировать прототип модуля автоматического формирования документов с использованием инструментов веб-разработки, разработать программный интерфейс для возможности создания, редактирования и удаления отчетной формы, способу хранения и обработки данных, модуль должен формировать и выводить готовый документа, запрашиваемый пользователем.

4. Курсовой проект в обязательном порядке представляется:

а) пояснительной запиской,

б) графическими материалами:

в) действующий прототип модуля автоматического формирования документов.

5. Содержание пояснительной записки:

- Анализ предметной области формального документооборота в коммерческой деятельности компаний использующих медицинские ИС;
- Составление технического задания на разработку модуля автоматического формирования документов;
- Разработка прототипа модуля автоматического формирования документов;
- Заключение;
- Список использованных источников;
- Приложения.

6. Срок сдачи отчета на кафедру: 16.01.2025

Руководитель,  
Доцент, кандидат технических наук



Харитонов Д.И.

Задание получил:

Шемилину С.Д.

## Аннотация

Тема курсового проектирования: «Формальный документооборот в коммерческой деятельности компаний использующих медицинские ИС».

Представленная курсовая работа состоит из трех глав, заключения, списка использованных источников.

В первой части курсового проекта рассматривается предметная область медицинских ИС, деятельности компаний использующих МИС и формального документооборота. Дана характеристика и принципы использования МИС, причинах его использования в государственных и частных учреждениях, а также объяснение и различие формального и неформального документооборота.

Во второй части составлено техническое задание будущего программного решения на основе Государственных стандартов серии 34 (ГОСТ 34). На основе Государственного стандарта, была составлена общая информация о системе, характеристика и необходимые требования к разработке конечного программного продукта.

В третьей части описывается проведенный анализ разработки прототипа будущего программного продукта, выбора средств и инструментом разработки. Рассматривается непосредственная реализация самого прототипа, описывающая шаги разработки и проведения тестовых сценариев с целью выявления определения сильных и слабых сторон проектного решения еще на ранних этапах разработки.

В заключительной части приводятся основные выводы и анализ полученный в ходе курсового проектирования.

Общий объем работы 44 страницы.

## Содержание

1 Анализ предметной области формального документооборота в деятельности компаний использующих МИС	5
1.1 Медицинские информационные системы	5
1.2 Деятельность компаний использующих МИС	7
1.3 Формальный документооборот	8
2 Техническое задание на разработку модуля автоматического формирования документооборота	10
2.1 Общие сведения об информационной системе	10
2.2 Назначение и цели создания модуля автоматического формирования документооборота	10
2.3 Характеристика модуля автоматизации формирования документооборота	11
2.4 Требования к модулю автоматического формирования документооборота	11
2.4.1 Требования к модулю в целом	11
2.4.2 Требования к стандартизации и унификации	13
2.4.3 Требования к функциям, выполняемым модулем автоматизированного формирования документооборота	13
2.4.4 Требования к видам обеспечения доступа к модулю автоматического формирования документооборота	16
2.5 Состав и содержание работ по созданию модуля	18
2.6 Порядок контроля и приемки модуля автоматического формирования документооборота	19
2.7 Требования к документированию	19
3 Разработка прототипа модуля автоматизированного формирования документооборота	20
3.1 Анализ алгоритма работы прототипа	20
3.2 Выбор средств разработки и инструментов для реализации прототипа	24
3.3 Разработка пользовательского интерфейса прототипа	26
3.4 Разработка программного кода прототипа	30
3.4.1 Разработка клиентской части прототипа	30
3.4.2 Разработка серверной части прототипа	32
3.5 Тестирование разрабатываемого прототипа	33
Заключение	37
Список использованных источников	38
Приложение А	40
Приложение Б	43

# 1 Анализ предметной области формального документооборота в деятельности компаний использующих МИС

## 1.1 Медицинские информационные системы

Медицинская информационная система (МИС) – это специализированный программно-аппаратный комплекс направленный на эффективную работу сотрудников медицинского персонала, оптимизированный контроль рабочих процессов и автоматизированную обработку большого количества данных[1].

Использование МИС должна соответствовать самой структуре работы медицинского учреждения, в котором оно используется. Каждое отделение, занимающееся определенной спецификой работы имеет требования, необходимые для решения поставленных задач и одним из основных требований является сбор, хранение и формирование документов и отчетных форм.

Сама медицинская информационная система является сложно структурированной системой, разделенной на различные модули, в соответствии организационной структурой самого медицинского учреждения. Из-за большого объема ведения информации, его строгой отчетности и формализации документов в медицинских организациях, необходимо, чтобы модуль МИС, отвечающий за вывод отчетной формы в виде запрашиваемого документа, мог соответствовать следующим требованиям:

- сбор и хранение информации;
- обработка информации;
- возможность доступа к запрашиваемой информации, как из другого отделения, так и из другой медицинской организации;
- формирование и вывод запрашиваемой информации в соответствии с установленными нормативными документами.

Обеспечение данных требований позволяет сотрудникам медицинской организации своевременно получать информацию о пациенте и его тактике лечения[2].

Субъекты, имеющие непосредственное участие в различных процессах и его результатах, являются медицинский персонал, отвечающий за оказание приема пациента и оказание медицинской помощи, гражданский персонал, отвечающий за административную работу медицинской организации, а также сотрудники МИС, отвечающие за внедрение и настройку ведения документооборота в самих медицинских организациях.

Разбирая более детально процесс работы медицинского персонала, он основывается на заполнении медицинской информации о пациенте в ходе приема, а именно:

- основные данные о приеме;
- жалобы пациента;

- анамнез пациента;
- объективный статус пациента;
- информация о диагнозе;
- информация о проведенных исследованиях;
- история болезни пациента;
- направления на консультацию, исследования, другие приемы;
- протокол оказанного приема;
- заключение по итогу оказанного приема.

Несмотря на это, стоит также учитывать и специфичные приемы других специалистов, которые могут иметь отличимые медицинские данные в ходе оказания приема. Например врач-онколог, заполняющий дополнительные сведения о злокачественных новообразованиях, либо врачебная комиссия, в составе врачей, различных специализаций, проводящая медицинский осмотр с заполнением медицинских данных о состоянии пациента и проведении итогового заключения группы здоровья, и т.д. Из этого следует потребность в возможности комбинированного сбора информации, для последующего формирования соответствующего документа.

Область работы административного персонала заключается в сборе большого количества статистических данных для проведение дальнейшего анализа. Основная нагрузка заключается в фильтрации большого количества данных и формировании. При этом область входных параметров может быть разная, в зависимости от требуемой статистической отчетной формы, для дальнейшей формы[3].

Для формирования отчетной формы документа необходима предварительная настройка, в результате которой будет зависеть ее поведение. Во время настройки отчетных форм, следует учитывать специфику отчетной формы, а именно:

- для каких целей настраивается;
- что должно входить в конструктор отчетных форм;
- на основании какого приказа должен соответствовать итоговый документ;
- какие входные параметры должны быть для формирования документа;
- какие есть возможности будущей настройки отчетных форм.

Данная специфика тесно связана с деятельностью медицинской организации в МИС, так как в ходе поддержки и сопровождения, стандарты ведения документации с течением времени меняется и для этого важна своевременная поддержка и сопровождение имеющихся документов в требуемом виде.

## 1.2 Деятельность компаний использующих МИС

Согласно постановлению Правительства РФ от 09.02.2022 № 140 (ред. от 04.03.2024) «О единой государственной информационной системе в сфере здравоохранения» (вместе с «Положением о единой государственной информационной системе в сфере здравоохранения»)[4] указаны положения об использовании «Единой государственной информационной системы в сфере здравоохранения» (ЕГИСЗ) для обмена данными в электронном виде. Данное положение и использование ЕГИСЗ позволяет компаниям, использующие МИС обмениваться между собой данными, необходимыми для решения рабочих процессов, независимо от того, какую информационную систему использует каждое из направлений.

Использование МИС государственными и частными компаниями включает в себя несколько направлений[5]:

- медицинские учреждения;
- страховые компании;
- государственные органы;
- разработчики программного обеспечения.

В каждом из описанных направлений, у каждой компании имеются свои поставленные цели и задачи по использованию и автоматизированной работы с данными. Несмотря на использование различных компаний, в частности медицинских организаций различных программных решений МИС, требования и стандарты использования являются едиными.

Деятельность медицинских учреждений подразумевает работу с медицинскими данными, необходимыми для оказания медицинской помощи пациентам. Процессы работы медицинских учреждений состоят из:

- ведения электронных медицинских карт;
- лабораторных исследований;
- диагностических услуг;
- бухгалтерского учета.

Работа страховых компаний, в рамках МИС необходима для автоматизации процессов страхования и регулирования страховых случаев. Возможность взаимодействия таких компаний с медицинскими организациями позволяет организовать такие процессы, как:

- обработки страховых полисов;
- оценка результатов исследований пациента.

Государственные органы, такие как министерство здравоохранения используют МИС для анализа и управлению здоровьем населения, а также регулированием медицинских

учреждений. Процессы государственных органов состоят из:

- сбор статистики и анализ состояния здоровья населения;
- планирование ресурсов;
- регулирование медицинских организаций;
- управление процессами вакцинации.

Основной деятельностью компаний-разработчиков является разработка и поддержка МИС и его модулей, в частности модуля ведения электронного документооборота. Процессы работы разработчиков программного обеспечения состоит из:

- разработки и внедрения МИС;
- техническая поддержка.

Приведенные выше процессы данных направлений являются взаимосвязанными и требуют соблюдения требований формирования электронных документов. Следует учитывать специфичность поддерживаемых форматов использования и сбора отчетных форм в электронном виде для дальнейшей работы с ними.

### 1.3 Формальный документооборот

Документооборот является одним из основных способов взаимодействия, как между медицинскими организациями, так и между его отделениями. Организация документооборота позволяет выполнять[6]:

- организованную передачу информации;
- высокое качество работы медицинского персонала;
- высокое качество предоставления медицинских услуг;
- оперативное и правильное решение в тактике лечения пациента;
- возможность дополнения новой информации к существующей.

В медицинских организациях обработка огромного объема информации, строгая отчетность, формализация документов и высокий уровень требований оказания услуг, занимающая время и требование к минимизации ошибок на этапе заполнения документов оказывает большое влияние сотрудников медицинского персонала. Исходя из общей характеристики, возникает потребность автоматизации на разных этапах процесса медицинских организаций[7].

Высокая степень формализации и большое количество потока документооборота затрагивает различные аспекты бизнес-процессов. На регламентирующих норм, постановлений, распоряжений, законов и приказов позволяет стандартизировать данный



аспект. В рамках медицинских учреждений имеется большое количество регламентирующей документации:

– ФЗ от 21 ноября 2011 г. № 323 «Об основах охраны здоровья граждан в Российской Федерации»:[8]

– ФЗ от 29 ноября 2010 г. № 326 «Об обязательном медицинском страховании в Российской Федерации»:[9]

– ФЗ от 6 апреля 2011 г. № 63 «Об электронной подписи» (ред. от 01.07.2011г.);[10]

– приказ Минздрава РФ от 5 августа 2022 г. № 530н «Об утверждении унифицированных форм медицинской документации, используемых в медицинских организациях, оказывающих медицинскую помощь в стационарных условиях, в условиях дневного стационара и порядков их ведения»:[11]

– приказ Минздрава РФ от 7 сентября 2020 г. № 947н «Об утверждении Порядка организации системы документооборота в сфере охраны здоровья в части ведения медицинской документации в форме электронных документов»:[12]

– приказ Минздрава России от 15.12.2014 № 834н «Об утверждении унифицированных форм медицинской документации, используемых в медицинских организациях, оказывающих медицинскую помощь в амбулаторных условиях, и порядков по их заполнению» (ред. от 18.04.2024);[13]

– постановление Правительства РФ от 1 июня 2021 г. № 852 «О лицензировании медицинской деятельности (за исключением указанной деятельности, осуществляемой медицинскими организациями и другими организациями, входящими в частную систему здравоохранения, на территории инновационного центра «Сколково») и признании утратившими силу некоторых актов Правительства Российской Федерации».[14]

Разбирая дальше документооборот, он может быть также и неформальным. Неформальные документы являются документами, составленными в произвольной или свободной форме. Такие документы довольно сложно как-то систематизировать и по большей части составляются самими организациями[15].

Несмотря на это, неформализованный вид документооборота используется также активно, как формализованный. Такие документы чаще всего в рамках:

- заявления;
- служебных записок;
- писем;
- неформализованных соглашений;
- спецификаций;
- отчетов.

Несмотря на это данный вид документов так или иначе стараются также стандартизировать. Однако установленный образец документа отличается от установленного образца документа в такой же смежной организации.

Анализируя, указанные выше причины, потребность в использовании информационных систем только растет. Основным направлением информационных систем, является автоматизация документооборота как в государственных, так и в частных организациях.

Стратегия информатизации и перехода на электронный документооборот, позволяет снизить нагрузку на сотрудников организации, повышая эффективность работы и самих бизнес-процессов.

## **2 Техническое задание на разработку модуля автоматического формирования документооборота**

### **2.1 Общие сведения об информационной системе**

Полное наименование: Модуль «Отчеты» медицинской информационной системы.

Краткое наименование системы: модуль «Отчеты» МИС.

Заказчик – АО «БАРС Груп».

Разработчик – студент группы БИН-21-1 Шемилин Сергей Дмитриевич.

Плановые сроки начала и окончания работ с 03.02.2025 г. По 26.04.2025 г.

Порядок оформления и предъявления результатов работ предъявляется поэтапно в соответствии с ТЗ[16].

### **2.2 Назначение и цели создания модуля автоматического формирования документооборота**

Модуль МИС «Отчеты» предназначена для автоматического формирования формальных документов, используемых в текущей информационной системе, его предварительной настройки и последующего использования.

Целью обновления модуля «Отчеты» является:

– улучшение и модернизация текущего модуля «Отчеты» МИС, предназначенного для автоматизированного сбора данных и предоставлении в виде отчета из модели баз данных.

В рамках текущей цели, необходимо решить следующие задачи:

- анализ текущего модуля информационной системы;
- разработка модели базы данных и организация хранения;
- разработка API для взаимодействия пользовательской части с базой данных;

- создание алгоритма передачи данных между шаблоном и переменными;
- разработка системы синхронизированного изменения текстового редактора и графического редактора, при работе в одном из них;
- разработка более удобной системы по созданию шаблонов отчета со стороны верстки на языках HTML, CSS и JS;
- разработка более удобного графического интерфейса системы для создания шаблонов в виде документа, как в MS Office Word.

Поставленные задачи являются необходимой частью реализации поставленной цели для создания автоматического формирования документооборота

## 2.3 Характеристика модуля автоматизации формирования документооборота

Объектом автоматизации является модуль «Отчеты» МИС компании АО «БАРС Групп». Основная деятельность которой является разработка, внедрение, сопровождение и техническая поддержка программного обеспечения в государственных и частных организациях[17].

Условиями объекта автоматизации является настройка отчетных форм и использование автоматического формирования формальных документов в модуле «Отчеты» МИС используется пользователями МИС.

## 2.4 Требования к модулю автоматического формирования документооборота

### 2.4.1 Требования к модулю в целом

Структура модуля «Отчеты» МИС должна представлять собой систему, включающую в себя следующие подсистемы:

- подсистема сбора, обработки, хранения и загрузки данных;
- подсистема настройки отчетных форм;
- подсистема формирования и визуализации отчетных форм в виде собранного документа.

Подсистема сбора, обработки, хранения и загрузки данных должна:

- предоставлять источник данных для записи, чтения, изменения и удаления связанных в модулем «Отчеты» МИС;

Подсистема настройки отчетных форм должна:

- иметь возможность создания, изменения и удаления отчетной формы;

- возможность настройки входных параметров для дальнейшего формирования документа.

Подсистема формирования и визуализации отчетных форм в виде собранного документа должна:

- иметь на основе настроенной, вызываемой отчетной формы, корректно собирать документ;
- визуализировать сформированный документ в соответствии настроенными входными параметрами.

Пользователи, работающие в МИС, должны обладать базовыми навыками работы с компьютером. Численность и квалификация персонала к модулю «Отчеты» МИС определяется с учетом следующих требований:

- модуль «Отчеты» МИС предоставляет возможность выполнения настроек отчетных форм как одному, так и нескольким сотрудникам МИС в ходе производственной необходимости;
- пользователи МИС использующие данный модуль для автоматического формирования документов, имеющие навыки выполнения рабочих процессов с использованием компьютера и обладающие знанием программных средств в объеме пользовательской документации.

Требуемый режим работы персонала:

- в соответствии производственным календарем
- полный рабочий день с 9:00 до 18:00;
- основной перерыв на обеденное время должен составлять 1 час.

Надежность модуля «Отчеты» МИС должен обеспечиваться бесперебойной работой и предоставлением функциональных возможностей по настройке и формированию сбора запрашиваемого документа.

Диагностика и профилактика проводится два раза в месяц, в соответствии с плановыми техническими работами МИС. Проверка целостности данных и нарушений проводится по мере необходимости.

Модернизация модуля «Отчеты» МИС может происходить путем модернизации самой МИС, в рамках изменения или дополнения функциональности, при обновлении МИС на актуальной продуктивной версии.

Обеспечение информационной безопасности модуля «Отчеты» МИС должно удовлетворять требованиям:

- защита модуля системы должна обеспечиваться комплексом программно-технических средств и поддерживающих их организационных мер;

- защита модуля системы должна обеспечиваться на всех этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ;

- программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики модуля системы (надежность, быстродействие, возможность изменения настроек).

Средства антивирусной защиты должны быть установлены на всех рабочих местах пользователей и администраторов системы. Средства антивирусной защиты рабочих мест пользователей и администраторов должны обеспечивать:

- централизованное управление сканированием, удалением вирусов и протоколированием вирусной активности на рабочих местах пользователей;

- централизованную автоматическую установку клиентского ПО на рабочих местах пользователей и администраторов;

- централизованное автоматическое обновление вирусных сигнатур на рабочих местах пользователей и администраторов;

- администрирование антивирусных продуктов.

Указанные выше требования и их декомпозиция позволяет более углубленно уточнить необходимый перечень содержания и способов работы с модулем «Отчеты» МИС.

#### 2.4.2 Требования к стандартизации и унификации

В процессе работы модуля «Отчеты» МИС должны использоваться программные средства с учетом удобства применения в рамках основной МИС.

Данные о настройках отчетных форм и автоматизированного формирования документа должны храниться в облачной базе данных, соединенной с основной МИС и все изменения настроек также должны сохраняться.

Интерфейс должен соответствовать общему стилю интерфейса основной МИС.

Выполнение требований по обеспечению лицензионной чистоты в ходе проведения работ по данному техническому заданию должно обеспечиваться Исполнителем.

Исполнитель должен гарантировать не нарушение авторских, исключительных или иных прав на объекты интеллектуальной собственности.

#### 2.4.3 Требования к функциям, выполняемым модулем автоматизированного формирования документооборота

Выполняемые функции модуля «Отчеты» МИС разделяется на три основные части. Каждая из функциональных возможностей данного модуля имеет перечень задач.

В таблице 1 приведен список требуемых функций и задач функции, подлежащих автоматизации.

Таблица 1 – Требуемые функции и задачи функций

Функция	Задача
Подсистема сбора, обработки, хранения и загрузки данных.	Создание, изменение и удаление данных.
	Поиск и загрузка требуемых данных.
Подсистема настройки отчетных форм.	Создание, изменение и удаление данных.
	Изменение и сохранение данных.
	Добавление входных параметров для чтения данных.
Подсистема формирования и визуализации отчетных форм в виде собранного документа.	Чтение и обработка запрашиваемых данных.
	Сборка сформированных данных.
	Визуализация сформированных данных на основе настроенной отчетной формы.

Временной регламент реализации каждой функции задачи приведен в таблице 2.

Таблица 2 – Временной регламент задач

Задача	Требование к временному регламенту
Создание, изменение и удаление данных.	В течении всего периода работы системы. В случае необходимости изменения процессов создания, изменения и удаления данных.
Поиск и загрузка требуемых данных.	В течении всего периода работы системы. В случае необходимости изменения процесса поиска и загрузки данных.
Изменение и сохранение данных.	В течении всего периода работы системы. В случае необходимости изменения процесса изменения и сохранения данных.
Добавление входных параметров для чтения данных.	В течении всего периода работы системы. В случае необходимости изменения процесса входных параметров для чтения данных.
Чтение и обработка запрашиваемых данных.	В течении всего периода работы системы. В случае необходимости изменения процесса чтения и обработки запрашиваемых данных.

## Продолжение таблицы 2

Задача	Требование к временному регламенту
Сборка сформированных данных.	В течении всего периода работы системы. В случае необходимости изменения процесса сборки сформированных данных.
Визуализация сформированных данных на основе настроенной отчетной формы.	В течении всего периода работы системы. В случае необходимости изменения процесса визуализации сформированных данных на основе настроенной отчетной формы.

В таблице 3 представлены требования к качеству реализации функциональных задач, в каком виде должны предоставляться, характеристика точности и время выполнения.

Таблица 3 - Требования к качеству реализации функциональных задач

Задача	Форма представления данных	Характеристика точности и времени выполнения
Создание, изменение и удаление данных.	Данные в структуре БД.	Данные должны быть сохранены. Не более 2 минут.
Поиск и загрузка требуемых данных.	Данные в структуре БД.	Данные должны быть найдены и сохранены. Не более 2 минут.
Изменение и сохранение данных.	В стандарте интерфейса МИС. Данные в структуре БД.	Данные должны быть сохранены. Не более 2 минут.
Добавление входных параметров для чтения данных.	В стандарте интерфейса МИС. Данные в структуре БД.	Данные должны должны сохранены. Не более 2 минут.
Чтение и обработка запрашиваемых данных.	В стандарте интерфейса МИС. Данные в структуре БД.	Данные должны быть найдены и обработаны. Не более 2 минут.
Сборка сформированных данных.	В стандарте интерфейса МИС.	Данные должны быть собраны. Не более 2 минут.
Визуализация сформированных данных на основе настроенной отчетной формы.	В стандарте интерфейса МИС.	Данные должны быть собраны. Не более 2 минут.

Таблица 4 – Критерии отказа работы функций

Функция	Критерий отказа	Время восстановления	Коэффициент готовности
Подсистема сбора, обработки, хранения и загрузки данных.	Не выполняется ни одна из задач функции.	3 часа	0.80
Подсистема настройки отчетных форм.	Не выполняется ни одна из задач функции.	5 часа	0.70
Подсистема формирования и визуализации отчетных форм в виде собранного документа.	Не выполняется ни одна из задач функции.	4 часа	0.80

В таблице 4 указан перечень требований и критериев, характеризующий отказ работы функциональности модуля автоматического формирования документооборота.

#### 2.4.4 Требования к видам обеспечения доступа к модулю автоматического формирования документооборота

Требованиями к информационному обеспечению входит база данных, входные данные, выходная документация.

Входной информацией является:

- база данных;
- запрос пользователя – сотрудник МИС.

Выходной информацией является:

- изменение данных в структурных таблицах базы данных;
- данные базы данных;
- настроенная отчетная форма;
- сформированная отчетная форма в виде документа.

Язык системы: Русский.

Допускается использование английского языка, для настройки отчетной формы. Пользовательский интерфейс должен соответствовать общим требованиям МИС, быть



интуитивно понятным и не иметь требования дополнительной специфической подготовки по использованию.

На серверной части модуля «Отчеты» МИС должны быть установлены:

- операционная система: ОС Linux Debian;
- СУБД: PostgreSQL;
- веб-сервер: NGINX.

На рабочем месте пользователя должны быть установлены:

- операционная система: MS Windows 7/8/10 или ОС Linux Debian
- веб-браузер для работы с модулем «Отчеты» МИС: Firefox, Яндекс.Браузер, Google Chrome.

К обеспечению качества программных средств такие требования, как:

- функциональность должна обеспечиваться выполнением подсистемами всех их функций;
- надежность должна обеспечиваться за счет предупреждения ошибок - не допущения ошибок в готовых ПС;
- эффективность должна обеспечиваться за счет принятия подходящих, верных решений на разных этапах разработки ПС и системы в целом;
- сопровождаемость должна обеспечиваться за счет высокого качества документации по сопровождению;
- каждом этапе в разработке ПС должна проводится проверка правильности принятых решений по разработке и применению готовых ПС.

Для минимального функционирования модуля «Отчеты» МИС, составлен список требований к техническому обеспечению:

- процессор: 2 ядра от 1,5 ГГц;
- ОЗУ: от 4 Гб;
- дисковое пространство: от 60 Гб;
- пропускная способность: от 200 Мбит/с;
- отказоустойчивость и стабильность;
- удаленный доступ к файлам и БД;
- возможность одновременного подключения: от 30 соединений.

Основными пользователя модуля «Отчеты» МИС, являются сотрудники МИС.

К организации функционирования данного модуля «Отчеты» МИС и порядку взаимодействия персонала, обеспечивающего эксплуатацию, устанавливаются следующие требования:

- в случае возникновения необходимости внесения функциональных изменений

модуля «Отчеты» МИС, пользователи МИС должны составить заявку с типом «Запрос на изменение» с описанием, согласно установленному регламенту подачи заявок в системе управления проектами;

– отдел, отвечающий за эксплуатацию модуля «Отчеты» МИС, должно не менее чем за 6 часов информировать всех пользователей о переходе на технические работы.

К защите от ошибочных действий пользователя, предъявляются следующие требования:

– для пользователей должна быть запрещена возможность удаления уже настроенных объектов модуля «Отчеты» МИС;

– разработано руководство пользователя для минимизации ошибочных действий.

Необходимо разработать следующие методологические документы:

– руководство пользователя модуля «Отчеты» МИС для сотрудников использующих автоматическое формирование документов;

– руководство администратора модуля «Отчеты» МИС для сотрудников использующих настройку отчетных форм.

Указанные выше требования должны удовлетворять возможностями обеспечения доступа к модулю автоматического формирования документооборота.

## 2.5 Состав и содержание работ по созданию модуля

Перечень документов, предъявляемых по окончанию соответствующих стадий разработки модуля «Отчеты» МИС представлен в таблице 5.

Таблица 5 – Перечень документов стадии разработки модуля «Отчеты» МИС

Наименование стадий и этапов создания системы	Сроки выполнения работ	Результат работ
1. Эскизный проект: 1.1. Разработка проектного решения по модулю и его частям.	03.02.25 - 07.02.25	Описание функций, функций подсистем, алгоритм работы подсистем и их целей.
2. Технический проект: 2.1. Разработка проектного решения модуля и его частям; 2.2. Разработка документации и его частям.	10.02.25 - 04.04.25	Описание ПО, интерфейса, информационной базы
3. Рабочая документация: 3.1. Разработка рабочей документации модуля и его частей.	07.04.25 - 11.04.25	Готовая версия программного решения. Документация программного решения. Руководство пользователя и администратора.

## Продолжение таблицы 5

Наименование стадий и этапов создания системы	Сроки выполнения работ	Результат работ
4. Ввод в действие: 4.1. Проведение предварительных испытаний.	14.04.25 - 26.04.25	Протокол испытаний. Устранение технических недочетов. Внесение изменений в документацию.

Представленная таблица описывает перечень документов, предъявляемых по окончании соответствующих стадий разработки модуля «Отчеты» МИС.

## 2.6 Порядок контроля и приемки модуля автоматического формирования документооборота

Контроль за разработкой автоматизации формального документооборота для модуля «Отчеты» МИС будет производиться с двух сторон. Со стороны руководителя отдела филиала города Владивосток АО «БАРС Групп», а именно руководителя практики предприятия и со стороны руководителя практики высшего учебного заведения «Владивостокский Государственный Университет». Соответственно принимающими сторонами будут вышеупомянутые лица.

Приемка этапа заключается в рассмотрении и оценке проведенного объема работ и предъявленной технической документации в соответствии с требованиями настоящего технического задания.

## 2.7 Требования к документированию

Проектная документация должна включать в себя текстовые и графические материалы (представленные в виде бумажной копии и на цифровом носителе в формате MS Word), не противоречащие с соглашением о неразглашении, но имеющие информацию в рамках будущего проекта автоматизации, а именно:

- описание автоматизируемых функций;
- схема функциональной структуры автоматизации;
- описание технического процесса обработки данных;
- схема логической структуры баз данных;
- схема взаимодействия внутренней логики подсистем модуля «Отчеты» МИС;
- руководство пользователя по использованию автоматизированного сбора документа
- руководство администратора по настройке отчетных форм.

## 3 Разработка прототипа модуля автоматизированного формирования документооборота

### 3.1 Анализ алгоритма работы прототипа

В результате анализа предметной области и составления технического задания к программному решению, следующим этапом проектирования является разработка прототипа для будущего модуля автоматизированного формирования документов. Прототипирование будущего продукта позволит визуализировать и протестировать идеи проекта и выявить еще на ранних этапах разработки потенциальные проблемы и недостатки самих этапов разработки. Однако для его разработки требуется провести анализ алгоритма работы, а так же разработать схему взаимодействия компонентов подсистем между собой[18].

Перед началом разработки прототипа, необходимо составить схему работы и взаимодействия компонентов между собой.

Для этого необходимо разобрать общую логику, в которой необходимо систематизировать последовательность действий работы прототипа. Логика работы прототипа разделяется на две части:

- логика при настройке отчетной формы(рис. 1);
- логика при выводе запрашиваемой отчетной формы(рис. 2).

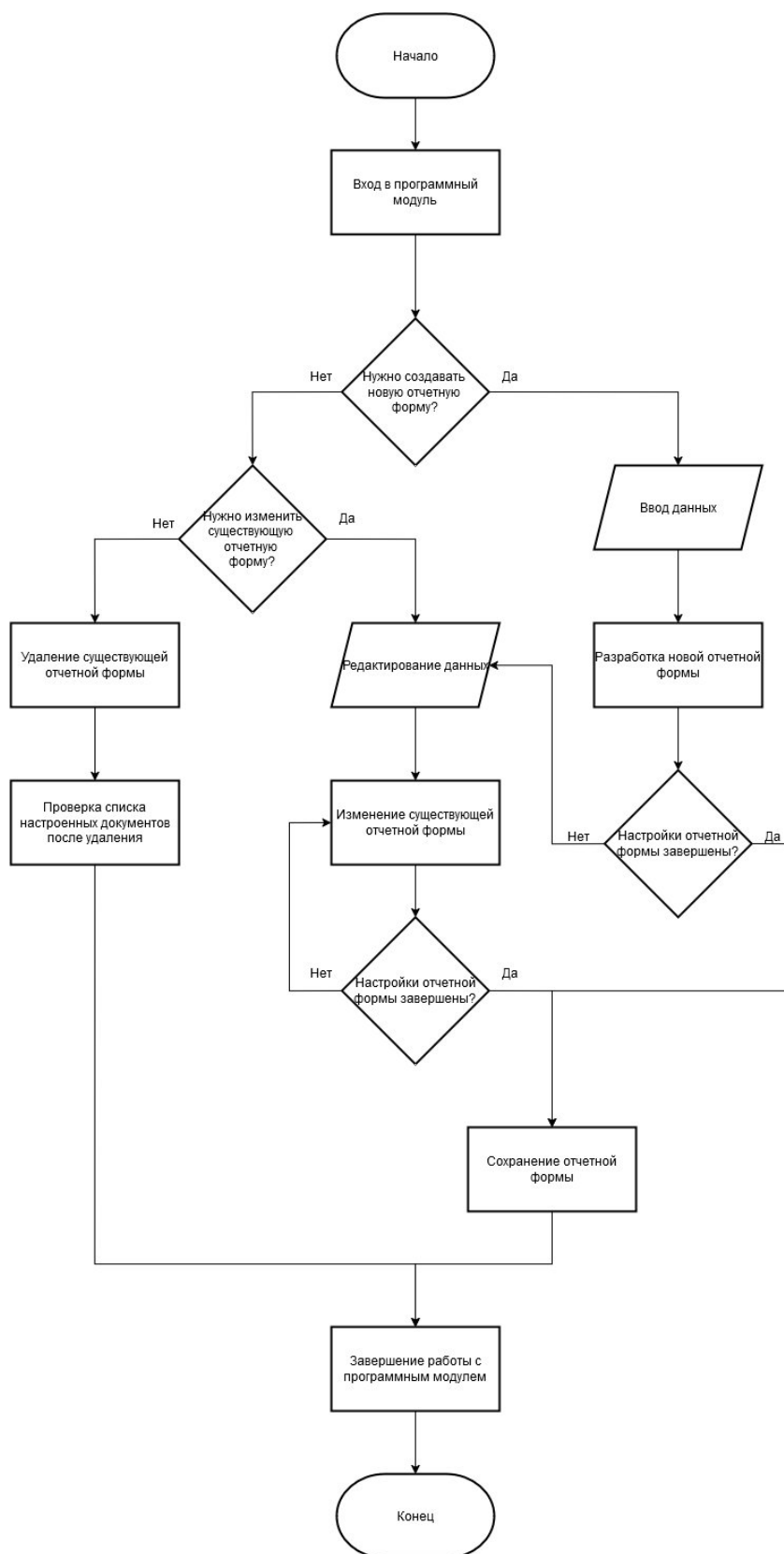


Рисунок 1 – Настройка отчета

Первая блок-схема описывает шаги действия прототипа в ходе проведения и изменения настроек отчетной формы. В ней отражена схема настройки отчетных форм и его действия, в зависимости взаимодействия с пользователем.

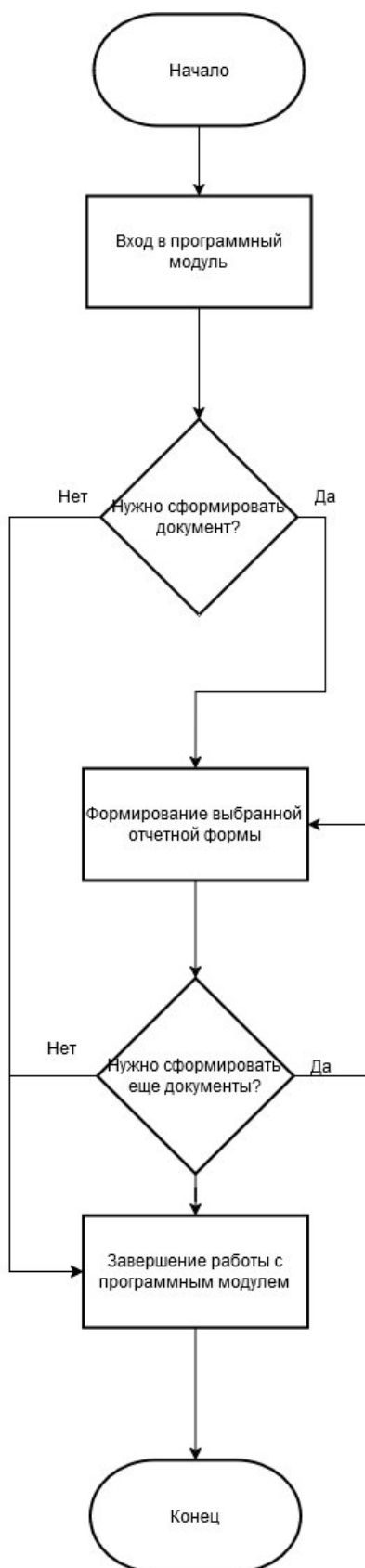


Рисунок 2 – Получения сформированного документа

Вторая блок-схема описывает шаги действия прототипа в ходе формирования запрашиваемого документа и его поведение в случае отсутствия запрашиваемой отчетной формы.

Далее, в ходе разрабатываемого прототипа необходимо понимать, как будут взаимодействовать между собой компоненты системы и какие данные будут передаваться между собой. Для этого необходимо разработать архитектуру верхнего уровня.

Архитектура верхнего уровня позволяет описать компоненты системы, каждая из которых должна отвечать за определенную часть функционирования программного продукта, интерфейсов взаимодействия между компонентами, набором правил, по которым будут работать компоненты системы, а также шаблон проектирования[19].

В ходе анализа были определены следующие компоненты прототипа и шаблон проектирования(рис. 3):

- база данных;
- серверная часть;
- клиентская часть;
- программный интерфейс (API).



Рисунок 3 – Шаблон проектирования MVC

Данная схема описывает взаимодействие пользователя с базой данных, через API, реализующая внутреннюю логику работы с полученными данными, обработкой и вывода запрашиваемой информации.

Одним из главных преимуществ данного подхода в том, что он позволяет независимо изменять каждый компонент, тем самым масштабировать его в будущем, благодаря чему позволяет экономить время и ресурсы разработки.

В результате проведенного анализа была установлена логика работы прототипа с двух сторон, а также разработана архитектура взаимодействия компонентов между собой. Данные

схемы позволяют более ясно определить будущую стратегию и стек разработки будущего прототипа.

### 3.2 Выбор средств разработки и инструментов для реализации прототипа

Поскольку основной задачей является разработка прототипа будущего программного продукта, из этого следует необходимость в проведение анализа инструментов и средств разработки. Выбор инструментов разработки позволит, также определить стек технологий будет выполняться процесс разработки.

Для выбора будущих инструментов разработки, были выделены следующие критерии:

- язык программирования;
- база данных;
- набор необходимых библиотек для ускорения разработки;
- среда разработки;

Поскольку разработка прототипа основывается на использования веб-технологий, то необходимо ее разделить на две части:

- фронтенд;
- бэкенд.

Данное разделение обусловлено разными областями разработки с набором своих библиотек и языков программирования.

Фронтенд является видимой частью программного обеспечения, которая при взаимодействии с пользователем, она реагирует на его действия, которые выводит на экран.

В рамках веб-технологий, у фронтенда имеется уже устоявшийся стек разработки, в который входит:

- HTML;
- CSS;
- JavaScript.

Данный стек является основой формирования веб-страниц браузера. На его основе имеются различные библиотеки, расширяющие возможности разработки. Однако, несмотря на это, для разработки прототипа не требуется использование какой-либо библиотеки, так как настройка и формирование отчетной формы будет происходить на основе базового стека веб-разработки.

Бэкенд является невидимой частью программного обеспечения, в основе которой заложена логика работы самой программы. Бэкенд отвечает за обработку данных и хранение данных, и за функциональность программного обеспечения. У него нет заранее определенного



языка программирования и поэтому необходимо провести анализ его выбора, для последующей разработки.

Перед проведением анализа и определения выбора языка программирования, необходимо выделить следующие критерии:

- динамическая типизация;
- управление памятью;
- производительность;
- поддержка сообществом;
- кроссплатформенность;
- сложность;
- наличие библиотек.

Наиболее популярными языками программирования являются:

- Java;
- Python;
- PHP;
- C#.

В ходе анализа была составлена сравнительная таблица языков программирования и критериев выбора:

Таблица 6 – Таблица сравнения языков программирования и критерии выбора

Критерии	Java	Python	PHP	C#
Динамическая типизация	Нет	Да	Неявная	Нет
Автоматическое управление памятью	Да	Да	Нет	Да
Производительность	Высокая	Средняя	Средняя	Высокая
Поддержка сообществом	Высокая	Высокая	Высокая	Высокая
Кроссплатформенность	Windows, Linux, Mac OS	Windows, Linux, Mac OS	Windows, Linux, Mac OS	Windows. Linux (ограниченно), Mac OS (ограниченно)
Наличие библиотек	Высокая	Высокая	Средняя	Средняя

По результатам анализа было принято решение выбрать язык программирования Python. Основным отличием данного языка является наличие большого количества библиотек и документации, высокая совместимость с другими операционными системами и простота написания программ.

На основе данного языка имеется уже готовая библиотека Django. Данная библиотека имеет уже готовый набор инструментов для быстрого развертывания веб-приложений, разработки новых компонентов за счет шаблонизированной системы загрузки форм, упрощенного способа работы с базой данных, а также возможности структурированной организации внутри веб-приложения, повышая читаемость и понятность, как файлов, так и самого кода.

### 3.3 Разработка пользовательского интерфейса прототипа

Для реализации будущего прототипа автоматизированного сбора отчетных форм, необходимо реализовать программный интерфейс, с которым будет взаимодействовать пользователь. Основные требования к пользовательскому интерфейсу являются:

- легковесность;
- интуитивная понятливость;
- отсутствие отвлекающих и мешающих элементов;
- возможность сразу приступить к работе.
- наличие поддержки русского языка;
- доступность работы на любом размене монитора компьютера.

В ходе проведения анализа, был разработан следующий пользовательский интерфейс представленный на рисунке ниже (рис. 4).

ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить

Рисунок 4 – Пользовательский интерфейс

Прототип пользовательский интерфейс состоит из четырех основных частей:

- панель управления для форматирования отчетных форм;
- окно текстового редактора и нижним полем для названия отчетной формы;
- список сохраненных в базе данных документов, для последующего формирования и использования в документообороте;
- отдельное окно для вывода сформированного документа (рис. 5).

<p><a href="#">Назад</a></p> <p>к приказу Министерства здравоохранения РФ от 15 декабря 2014 г. N 834н</p>				
<p>Наименование медицинской организации _____</p>		<p>Код формы по ОКУД _____</p> <p>Код организации по ОКПО _____</p> <p>Медицинская документация</p> <p>Учетная форма N 025/у</p> <p>Утверждена приказом Минздрава России от 15 декабря 2014 г. N 834н</p>		
<p>Адрес _____</p>				
<p>МЕДИЦИНСКАЯ КАРТА ПАЦИЕНТА, ПОЛУЧАЮЩЕГО МЕДИЦИНСКУЮ ПОМОЩЬ В АМБУЛАТОРНЫХ УСЛОВИЯХ N _____</p>				
<p>1. Дата заполнения медицинской карты: число ____ месяц _____ год ____</p>				
<p>2. Фамилия, имя, отчество _____</p>				
<p>3. Пол: муж. - 1, жен. - 2      4. Дата рождения: число ____ месяц _____ год ____</p>				
<p>5. Место регистрации: субъект Российской Федерации _____</p> <p>район _____ город _____ населенный пункт _____</p> <p>улица _____ дом _____ квартира _____ тел. _____</p>				
<p>6. Местность: городская - 1, сельская - 2</p>				
<p>7. Полис ОМС: серия _____ N _____      8. СНИЛС _____</p>				
<p>9. Наименование страховой медицинской организации _____</p>				
<p>10. Код категории льготы _____      11. Документ _____: серия _____ N _____</p>				
<p>12. Заболевания, по поводу которых осуществляется диспансерное наблюдение:</p>				
<p>Дата начала диспансерного наблюдения</p>	<p>Дата прекращения</p>	<p>Диагноз</p>	<p>Код по МКБ-10</p>	<p>В</p>

Рисунок 5 – Окно вывода сформированного документа

Для начала настройки отчетной формы, пользователю необходимо начать настраивать отчетную форму. Возможностей создания будущего документа у него два:

- разработка документа традиционным способом, похожим как при работе в программе MS Office Word;
- разработка документа с помощью использования языков веб-разметки, очень сходим с веб-программированием (рис. 6).

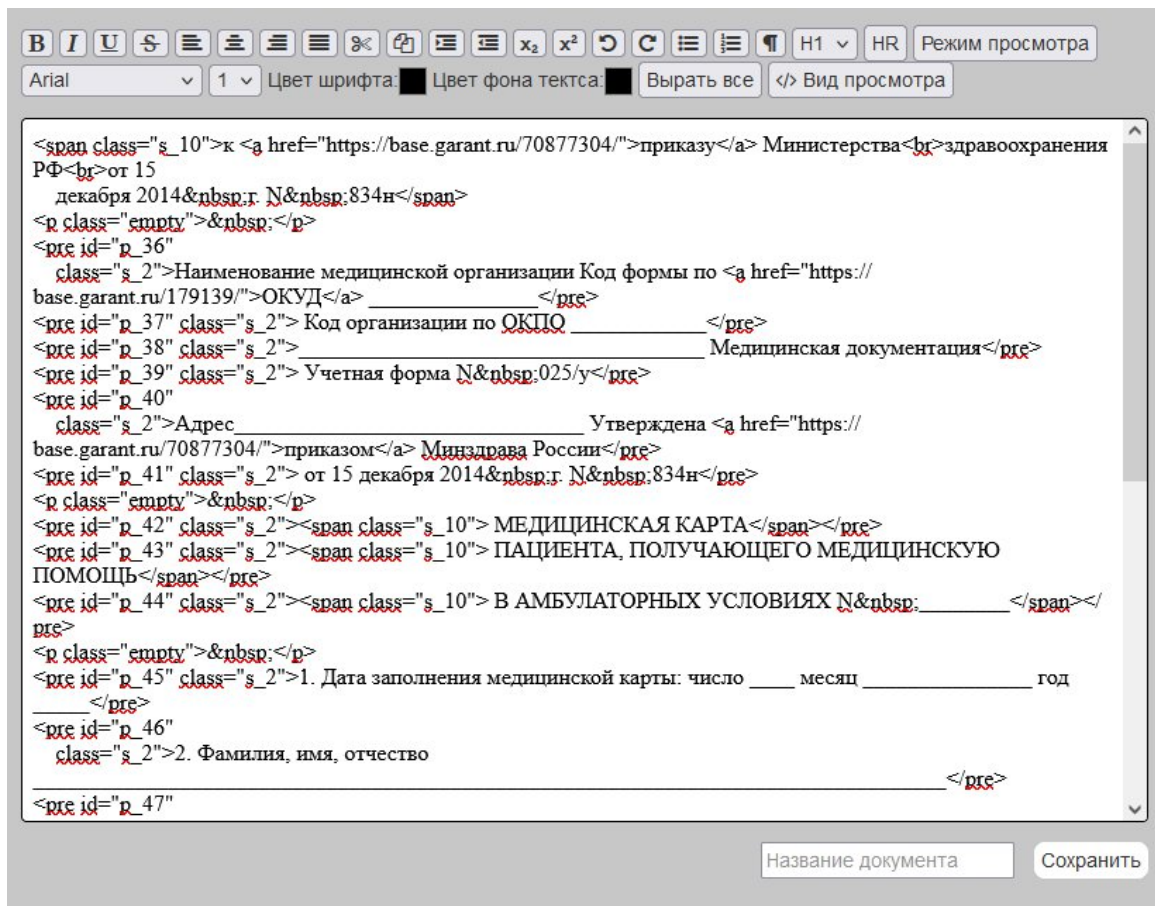


Рисунок 6 – Второй способ использования редактора

Данный способ переключения вида документа находится на панели управления текстом, не требующий дополнительной перезагрузки вкладки браузера.

На панели управления расположены необходимые кнопки для полноценного редактирования и форматирования текста в нужный вид, как показано на рисунке ниже (рис.7)

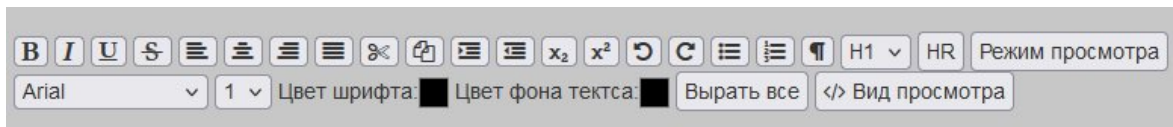


Рисунок 7 – Панель управления

Для исключения потери настроек и изменений документа, разработана функция сохранения отчетных форм в базу данных, а так же вывода списка уже имеющихся в системе, документов (рис. 8).

Сохраненные отчеты		
ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить
23	Образец справки 025/у	Сформировать Изменить Удалить

Рисунок 8 – Список сохраненных документов

Список сохраненных отчетных форм, позволяет оперативно формировать и получать готовые документы для последующей работы с ними. При проведении анализа, были разработаны механизмы сборки, изменения и удаления отчетных форм. Возможность сохранения, изменения и удаления документов позволяет пользователю оперативно выполнять рабочие процессы, без выполнения дополнительных действий.

Интерфейс формирования документа реализован очень. Пользователь, при нажатии на кнопку «Сформировать», будет перемещен на следующую страницу, в которой имеется возможность предпросмотра документа, а также его печати в бумажном виде (рис. 9).

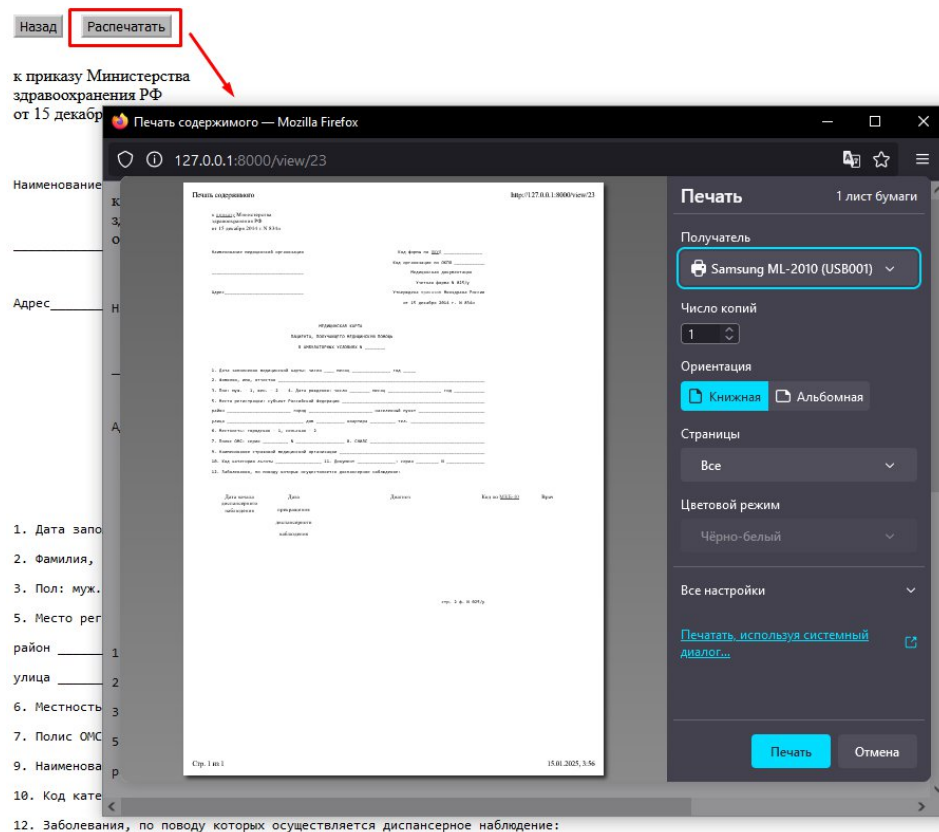


Рисунок 9 – Функционал печати документа

Сам интерфейс печати документа основан на интерфейсе браузера на рабочем месте пользователя.

### 3.4 Разработка программного кода прототипа

Перед непосредственным началом разработки программного кода прототипа, в результате заранее проведенного анализа, процесс разработки был разделен на две части:

- разработка функциональной клиентской части;
- разработка функциональной серверной части.

#### 3.4.1 Разработка клиентской части прототипа

На рисунке ниже (рис. 10) представлена часть программного кода верстки элементов пользовательского интерфейса.

```

3   <html>
14  <body onload="enableEditMode();">
15  <div class="panel">
20      <button onclick="execCmd('justifyLeft');"><i class="fa fa-align-left"></i></button>
21      <button onclick="execCmd('justifyCenter');"><i class="fa fa-align-center"></i></button>
22      <button onclick="execCmd('justifyRight');"><i class="fa fa-align-right"></i></button>
23      <button onclick="execCmd('justifyFull');"><i class="fa fa-align-justify"></i></button>
24      <button onclick="execCmd('cut');"><i class="fa fa-cut"></i></button>
25      <button onclick="execCmd('copy');"><i class="fa fa-copy"></i></button>
26      <button onclick="execCmd('indent');"><i class="fa fa-indent"></i></button>
27      <button onclick="execCmd('outdent');"><i class="fa fa-dedent"></i></button>
28      <button onclick="execCmd('subscript');"><i class="fa fa-subscript"></i></button>
29      <button onclick="execCmd('superscript');"><i class="fa fa-superscript"></i></button>
30      <button onclick="execCmd('undo');"><i class="fa fa-undo"></i></button>
31      <button onclick="execCmd('redo');"><i class="fa fa-repeat"></i></button>
32      <button onclick="execCmd('insertUnorderedList');"><i class="fa fa-list-ul"></i></button>
33      <button onclick="execCmd('insertOrderedList');"><i class="fa fa-list-ol"></i></button>
34      <button onclick="execCmd('insertParagraph');"><i class="fa fa-paragraph"></i></button>
35      <select onchange="execCommandWithArg('formatBlock',this.value);">
36          <option value="H1">H1</option>
37          <option value="H2">H2</option>
38          <option value="H3">H3</option>
39          <option value="H4">H4</option>
40          <option value="H5">H5</option>
41          <option value="H6">H6</option>
42      </select>
43      <button onclick="execCmd('insertHorizontalRule');">HR</button>
44      <button onclick="toggleEdit();">Режим просмотра</button>
45      <br>
46
47      <select onchange="execCommandWithArg('fontName',this.value);">
48          <option value="Arial">Arial</option>
49          <option value="Comic Sans MS">Comic Sans MS</option>
50          <option value="Cursive">Cursive</option>
51          <option value="Tahoma">Tahoma</option>
52          <option value="Avenir">Avenir</option>
53          <option value="Verdana">Verdana</option>
54      </select>
55
56      <select onchange="execCommandWithArg('fontSize',this.value);">
57          <option value="1">1</option>
58          <option value="2">2</option>
59          <option value="3">3</option>
60          <option value="4">4</option>
61          <option value="5">5</option>
62          <option value="6">6</option>
63          <option value="7">7</option>
64      </select>

```

Рисунок 10 – Код разработки панели управления

Помимо разрабатываемого кода построения программного интерфейса, также разработан код, повышающий функциональность программного интерфейса на языке программирования JavaScript.

Далее будут представлены рисунок (рис. 11) описывающий функциональную часть программного интерфейса.

```

39
40 $(document).ready(function () {
41
42
43     function method_ajax(form, methodAjax, path) {
44
45         let json = {};
46
47         if ('edit/' === path) {
48
49             var trId = $(form).parent().parent().parent().find('td:nth-child(1)').text();
50             json['id'] = trId;
51             var pathName = path+trId;
52
53         } else if ('del/' === path) {
54
55             var trId = $(form).parent().parent().parent().find('td:nth-child(1)').text();
56             json['id'] = trId;
57             var pathName = path;
58
59         } else if ('save/' === path || "PUT" === methodAjax) {
60
61             var text = TextField.document.getElementsByTagName('body')[0].innerHTML;
62             var name = $(form).parent().find('input')[0].value;
63             json['name'] = name;
64             json['text'] = text;
65             var pathName = path;
66             if ("PUT" === methodAjax) {
67                 json[id] = $('#save_edit')[0].attributes[4].nodeValue;
68             }
69
70         } else if ('view/' === path) {
71
72             var trId = $(form).parent().parent().parent().find('td:nth-child(1)').text();
73             json['id'] = trId;
74             var pathName = path+trId;
75
76         }
77     }

```

Рисунок 11 – Код форматирования и подготовки данных для последующей обработки  
Для более подробного ознакомления с кодовой структурой разрабатываемого программного интерфейса можно ознакомиться в «Приложение А» курсового проекта.

Показанная выше, программная реализация прототипа со стороны пользователя представляет возможность более детально разобраться в актуальности запланированных идей, способов функциональной реализации взаимодействия и отзывчивости программного интерфейса на действия пользователя.

Стоит отметить, что реализация формирования отчетной формы и последующего вывода на экран пользователя, разделен. Соответственно разрабатываемая часть кода была также разделена, для разграничения и организации кодовой и файловой системы.

### 3.4.2 Разработка серверной части прототипа

Реализация серверной части прототипа, ее логики работы и способа взаимодействия с клиентской частью программного решения позволила ответить на вопросы по способу обмена информацией между различными компонентами и способах подхода написания методов взаимодействия и работы с данными.

На представленном рисунке (рис. 12) описана часть программного кода взаимодействия прототипа в целом и его механизмов сохранения, обработки, изменения, удаления и хранения данных. Для более подробного ознакомления с кодовой структурой серверной части разрабатываемого прототипа можно ознакомиться в «Приложение Б» курсового проекта.

```

1  from django.http import JsonResponse, HttpResponseRedirect
2  from django.shortcuts import render
3  from django.views.decorators.csrf import csrf_exempt
4  from .models import *
5  from django.forms.models import model_to_dict
6  import json
7
8  @csrf_exempt
9  def index(request):
10     lst = Reports.objects.all()
11     return render(request, 'index.html', {'table': lst})
12
13  @csrf_exempt
14  def save(request):
15     if request.method == "POST":
16         print(request.body)
17         data = json.loads(request.body.decode('utf-8'))
18         report = Reports()
19         report.name = data['name']
20         report.report = data['text']
21         report.save()
22         report = model_to_dict(report)
23         return JsonResponse({'report':report}, status=200, content_type="application/json")
24     else:
25         return HttpResponseRedirect()
26
27  @csrf_exempt
28  def delete(request):
29     if request.method == "DELETE":
30         idd = json.loads(request.body.decode('utf-8'))
31         print('id', int(idd['id']))
32         data = Reports.objects.filter(id=int(idd['id'])).delete()
33         print('data', data)
34         return JsonResponse({'data':data}, status=200, content_type="application/json")
35     else:
36         return HttpResponseRedirect()

```

Рисунок 12 – Код обработки и вывода данных, методы взаимодействия с базой данных

В ходе разработки серверной части прототипа, были выявлены наиболее слабые места логики работы функциональной части прототипа, связанных с редактированием и представлением данных. Схожесть методов работы сохранения и изменения данных требует дополнительного внимания и дальнейшей оптимизации работы. Логика редактирования и



представления документов, несмотря на схожесть, требуют дополнительного внимания и оптимизации.

### 3.5 Тестирование разрабатываемого прототипа

Тестирование прототипа будущего программного продукта является одной из составных частей разработки и отладки будущего решения. Проведение тестирования прототипа будущей программы позволяет проверить базовую часть функционала с требованиями в заданной области работы.

В рамках тестирования разработанного прототипа, необходимо оценить актуальность реализации, целесообразность дальнейшего развития, функциональные преимущества и недостатки.

Сценарием использования данного программного решения будет являться:

- разработка справки о состоянии пациента на учете в наркологическом диспансере.

Шаги для создания справки (рис. 13-16):

- зайти в программный модуль автоматизированного сбора документов;
- использовать комбинированное создание отчета, как в тестовом формате, так и формате разработки код;
- использовать применяемые стили, доступные на панели инструментов;
- по окончании разработки составить название документа и сохранить.

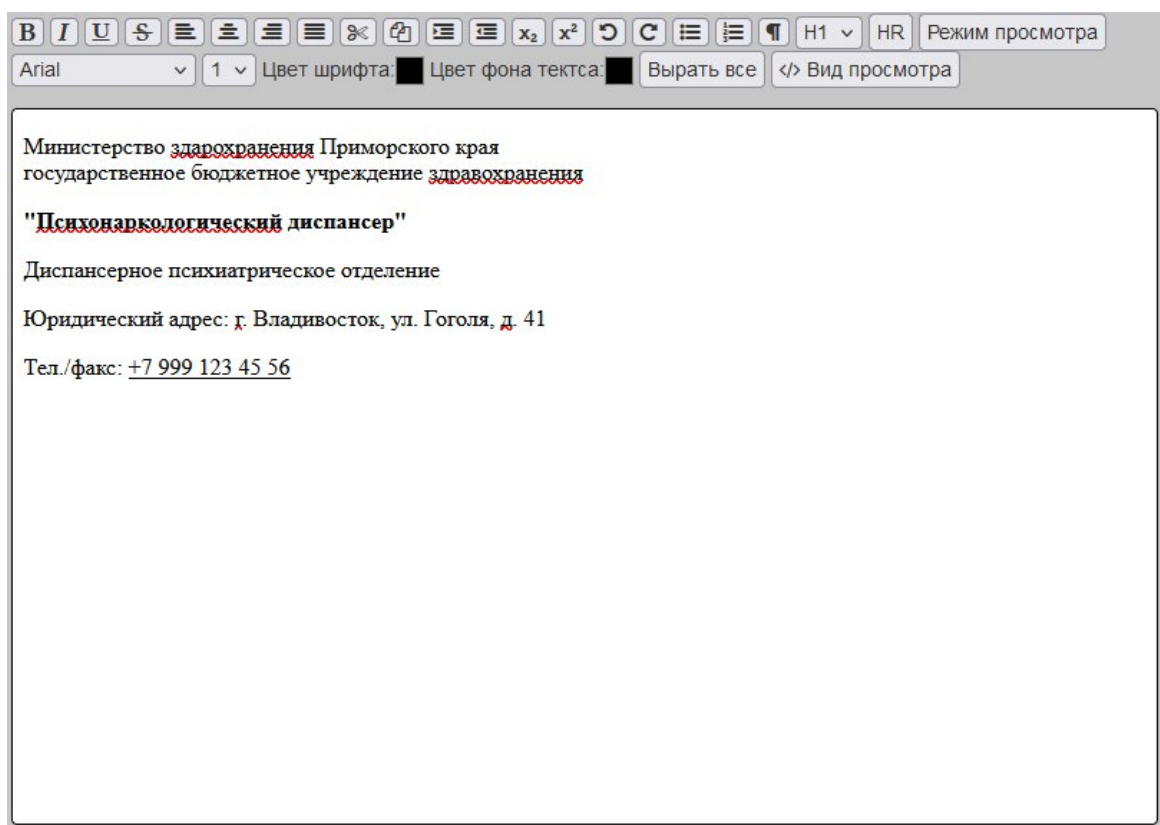


Рисунок 13 – Разработка отчета с текстовым редакторе

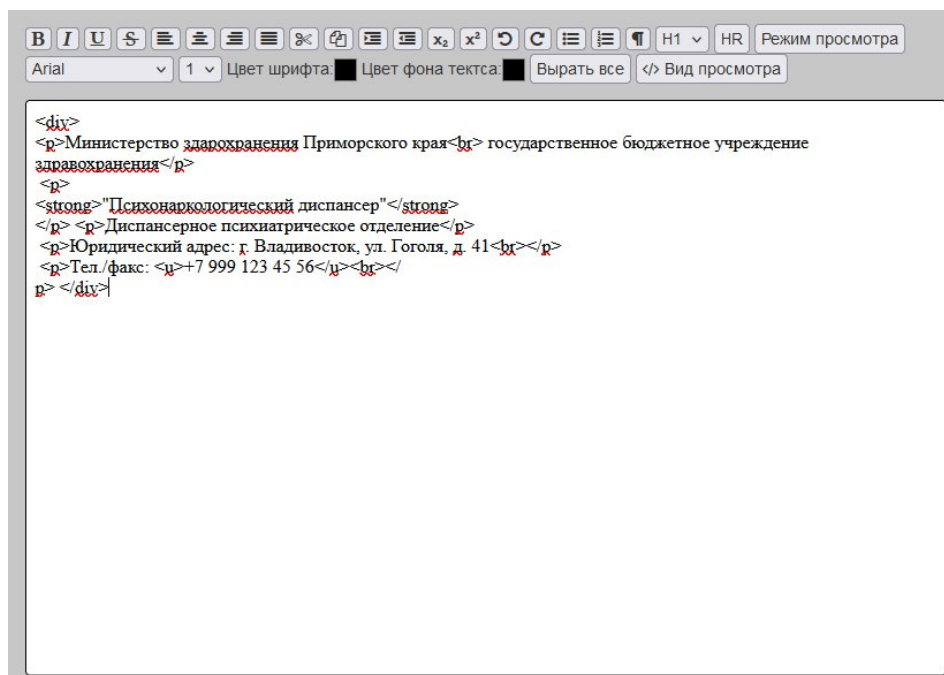


Рисунок 14 – Разработка отчетной формы в формате разработки кода

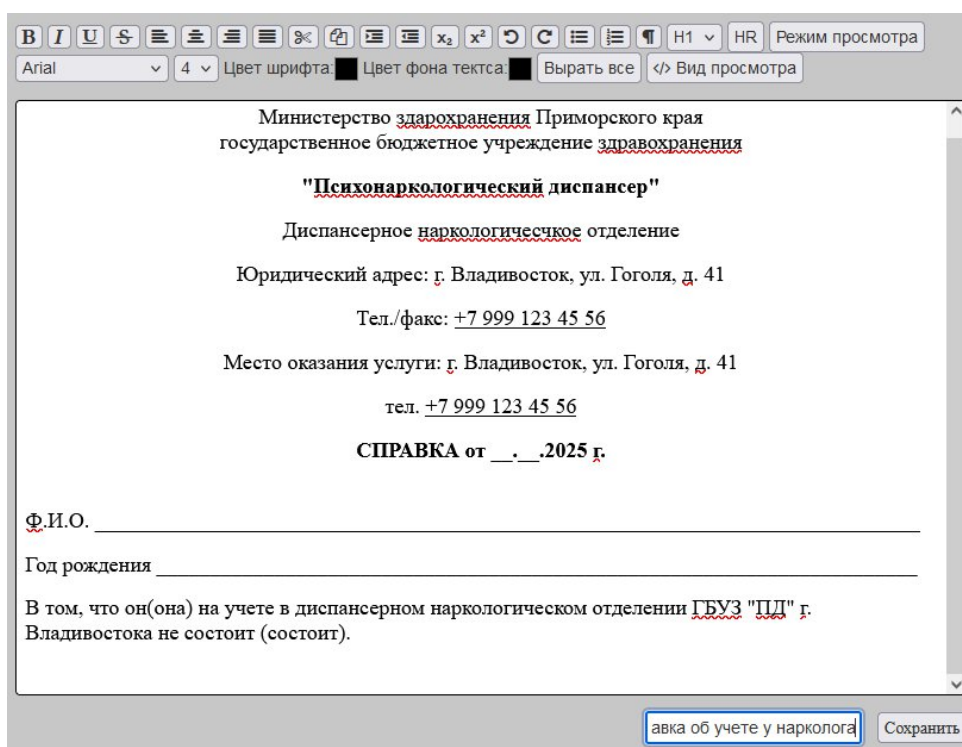


Рисунок 15 – Формирование названия отчета и сохранение

Сохраненные отчеты		
ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить
23	Образец справки 025/у	Сформировать Изменить Удалить
24	<u>Справка об учете у нарколога</u>	Сформировать Изменить Удалить

Рисунок 16 – Проверка успешного сохранения

Далее, необходимо сформировать документа и подготовить его на печать. Необходимые шаги для выполнения (рис. 26-28):

- найти созданную справку;
- нажать на кнопку для формирования документа;
- после ожидания загрузки, нажать на кнопку для печати.

Сохраненные отчеты		
ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить
23	Образец справки 025/у	Сформировать Изменить Удалить
24	<u>Справка об учете у нарколога</u>	Сформировать Изменить Удалить

Рисунок 17 – Нахождение нужного документа и переход к формированию документа

Назад
Распечатать

```

<div><p align="center"><font size="4">Министерство здравоохранения Приморского края<br> государственное бюджетное учреждение
здравоохранения</font></p><p align="center"><font size="4"><strong>Психонаркологический диспансер</strong></font></p>
<p align="center"><font size="4">Диспансерное наркологическое отделение</font></p><p align="center"><font size="4">Юридический
адрес: г. Владивосток, ул. Гоголя, д. 41<br></font></p><p align="center"><font size="4">Тел./факс: <u>+7 999 123 45 56</u></font></
p><div class="rep_title rep_subtitle rep_underline" align="center">
<p><font size="4">Место оказания услуги: <font size="4">г.
Владивосток, ул. Гоголя, д. 41</font></p>
<p><font size="4">тел. <font size="4"><u>+7 999 123 45 56</u></font></p>
</div>
<div class="rep_title" style="margin-bottom: 38px; margin-top: 10px" align="center"><font size="4"><strong>СПРАВКА от __. __. 2025 г.<br></
strong></font></div>
<div class="rep_subtitle rep_subtitle_2">
<p><font size="4">Ф.И.О.
<br></font></p></div>
<div
class="rep_underline" style="width: 100%; height: 20px;"><font size="4">Год рождения
</font></div>
<div class="rep_subtitle rep_subtitle_2"
style="margin-top: 8px; margin-bottom: 16px;">
<p><font size="4">В том, что он(она) на учете в диспансерном наркологическом
отделении ГБУЗ "ПД" г. Владивостока не состоит (состоит).</font></p></div>

```

Рисунок 18 – Автоматическое формирование документа

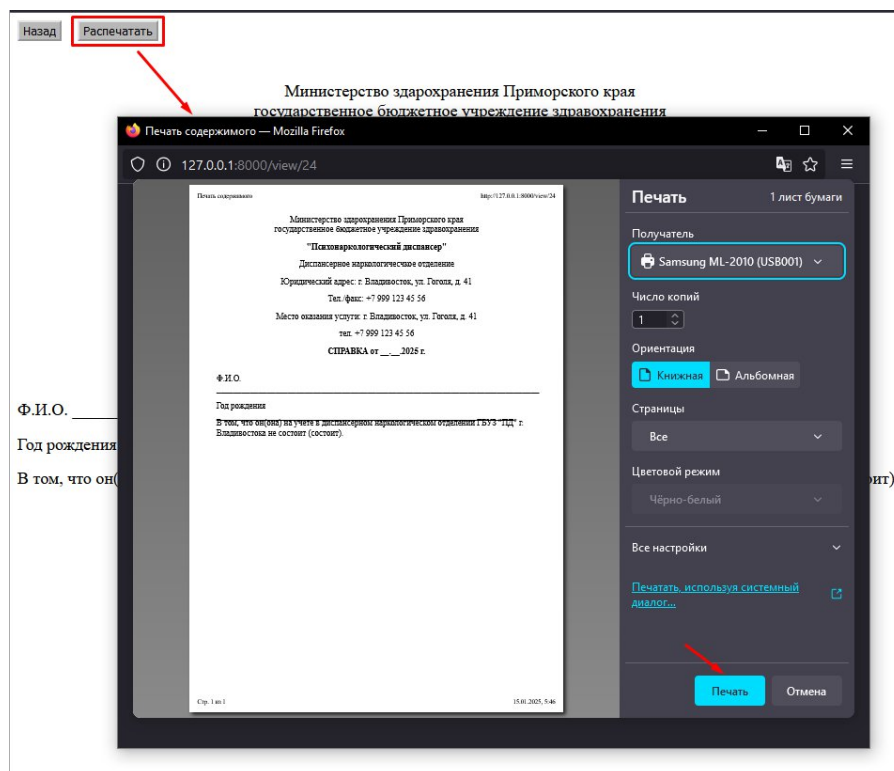


Рисунок 19 – Проверка корректности собранного документа и печать

В рамках тестирования также необходимо проверить корректное удаление отчетной формы. Шаги выполнения по удалению отчетной формы документа (рис. 29-30):

- поиск нужного документа в списке сохраненных;
- нажать на кнопку «Удалить»;
- после удаления убедиться, что документа больше нет в списке.

Сохраненные отчеты		
ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить
23	Образец справки 025/y	Сформировать Изменить Удалить
24	<u>Справка об учете у нарколога</u>	Сформировать Изменить Удалить

Рисунок 20 - Нахождение нужного документа и нажатие на кнопку удаления

Сохраненные отчеты		
ID	Название отчета	Действия
22	Пример документа	Сформировать Изменить Удалить
23	Образец справки 025/y	Сформировать Изменить Удалить

Рисунок 21 - Проверка отсутствия документа в списке сохраненных документов

## Заключение

В ходе курсового проектирования, был проведен анализ предметной области, составлено техническое задание к модулю, а также разработан прототип модуля автоматического формирования документооборота.

Во время изучения предметной области, была рассмотрена актуальность автоматизации формирования документов. Основываясь на общей стратегии развития цифровизации, различных отраслей экономики и тенденции перехода на электронный документооборот, выявлена проблема способов ведения электронной документации и его формирования. Анализ предметной области показал высокий процент потребностей и требований по ведению документооборота.

Перед разработкой прототипа автоматического формирования документооборота было составлено техническое задание. Данное техническое задание позволило упорядочить перечень требований, необходимый к разрабатываемому модулю. Также была определена архитектурная схема, которая проверена в ходе разработки прототипа и оценена с хорошей стороны в рамках организации логики работы компонентов прототипа. В ходе разработки прототипа, были выдвинуты идеи и гипотезы по возможной реализации решения функционирования. Декомпозированное описание кодовой части прототипа позволила выявить некоторые участки методов обработки данных, подлежащие будущей оптимизации и унификации работы.

Первоначальные идеи разработки базового функционала по текстовому редактированию с возможностью сохранения для будущего формирования, как итогового документа, в рамках сценарного тестирования, показала себя с хорошей стороны. Это дает потенциальную возможность дальнейшего развития из прототипа в полноценный программный продукт.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Что такое медицинская информационная система (МИС) и ее помощь в работе клиники [Электронный ресурс] // Первый Бит - Режим доступа: <https://spb.bit-medic.ru/articles/chto-takoe-mis/>.

2 Что такое Медицинская Информационная Система (МИС) [Электронный ресурс] // SyCrel – Режим доступа: <https://sycret.ru/blog/medicinskaya-informacionnaya-sistema>.

3 Структура медицинской организации: принципы эффективности [Электронный ресурс] // Директор клиники – Режим доступа: <https://www.dirklinik.ru/article/470-struktura-meditsinskoj-organizatsii-printsipy-effektivnosti>.

4 Постановление Правительства РФ от 9 февраля 2022 г. № 140 "О единой государственной информационной системе в сфере здравоохранения" [Электронный ресурс] // Гарант – Режим доступа: <https://base.garant.ru/403517946/>.

5 Медицинская информационная система [Электронный ресурс] // Парус – Режим доступа: <https://parus.com/catalog/resheniya-dlya-gosupravleniya/meditsinskaya-informatsionnaya-sistema/>.

6 Документооборот [Электронный ресурс] // Питер Софт – Режим доступа: <https://piter-soft.ru/knowledge/glossary/edo/dokumentooborot.html>.

7 Электронный документооборот в медицинской сфере [Электронный ресурс] // Астрал – Режим доступа: <https://astral.ru/info/elektronnyy-dokumentooborot/elektronnyy-dokumentooborot-v-meditsinskoj-sfere/>.

8 ФЗ от 21 ноября 2011 г. № 323 «Об основах охраны здоровья граждан в Российской Федерации» [Электронный ресурс] // Министерство Здравоохранения РФ – Режим доступа: <https://minzdrav.gov.ru/documents/7025-federalnyy-zakon-323-fz-ot-21-noyabrya-2011-g>.

9 ФЗ от 29 ноября 2010 г. № 326 «Об обязательном медицинском страховании в Российской Федерации» [Электронный ресурс] // Консультант Плюс – Режим доступа: [https://www.consultant.ru/document/cons\\_doc\\_LAW\\_107289/](https://www.consultant.ru/document/cons_doc_LAW_107289/).

10 ФЗ от 6 апреля 2011 г. № 63 «Об электронной подписи» (ред. от 01.07.2011г.) [Электронный ресурс] // Министерство цифрового развития, связи и массовых коммуникаций РФ – Режим доступа: <https://digital.gov.ru/ru/documents/3446/>.

11 Приказ Минздрава РФ от 5 августа 2022 г. № 530н «Об утверждении унифицированных форм медицинской документации, используемых в медицинских организациях, оказывающих медицинскую помощь в стационарных условиях, в условиях дневного стационара и порядков их ведения» [Электронный ресурс] // Гарант – Режим доступа: <https://base.garant.ru/405501275/>.

12 Приказ Минздрава РФ от 7 сентября 2020 г. № 947н «Об утверждении Порядка организации системы документооборота в сфере охраны здоровья в части ведения медицинской документации в форме электронных документов» [Электронный ресурс] // Гарант – Режим доступа: <https://www.garant.ru/products/ipo/prime/doc/400083202/>.

13 Приказ Минздрава России от 15.12.2014 № 834н «Об утверждении унифицированных форм медицинской документации, используемых в медицинских организациях, оказывающих медицинскую помощь в амбулаторных условиях, и порядков по их заполнению» (ред. от 18.04.2024) [Электронный ресурс] // Консультант Плюс – Режим доступа: [https://www.consultant.ru/document/cons\\_doc\\_LAW\\_175963/](https://www.consultant.ru/document/cons_doc_LAW_175963/).

14 Постановление Правительства РФ от 1 июня 2021 г. № 852 «О лицензировании медицинской деятельности (за исключением указанной деятельности, осуществляемой медицинскими организациями и другими организациями, входящими в частную систему здравоохранения, на территории инновационного центра «Сколково») и признании утратившими силу некоторых актов Правительства Российской Федерации» [Электронный ресурс] // Консультант Плюс – Режим доступа: [https://www.consultant.ru/document/cons\\_doc\\_LAW\\_385633/](https://www.consultant.ru/document/cons_doc_LAW_385633/).

15 Формализованные и неформализованные документы [Электронный ресурс] // Контур Диадок – Режим доступа: [https://www.diadoc.ru/articles/22582-formalizovannye\\_i\\_neformalizovannye\\_dokumenty](https://www.diadoc.ru/articles/22582-formalizovannye_i_neformalizovannye_dokumenty).

16 ГОСТ 34.602-89 Техническое задание на создание автоматизированной системы [Электронный ресурс] // Project Experience – Режим доступа: [https://www.prj-exp.ru/gost/gost\\_34-602-89.php](https://www.prj-exp.ru/gost/gost_34-602-89.php).

17 АО "БАРС Групп" О компании [Электронный ресурс] // АО "БАРС Групп" – Режим доступа: <https://bars.group/about/>.

18 Прототипирование [Электронный ресурс] // Unisender – Режим доступа: <https://www.unisender.com/ru/glossary/chto-takoe-prototipirovanie-i-zachem-ono-nuzhno>.

19 Верхнеуровневая архитектура [Электронный ресурс] // Хабр – Режим доступа: <https://habr.com/ru/companies/yandex/articles/425611/>.

20 Какой язык программирования мне выбрать? [Электронный ресурс] // Skill Factory – Режим доступа: <https://blog.skillfactory.ru/kakoj-yazyk-programirovaniya-vybrat/>.

21 Django [Электронный ресурс] // Django Project – Режим доступа: <https://www.djangoproject.com/>.

## Приложение А

```

3 <html>
14 <body onload="enableEditMode();"
72 <form id="fr_save">
73 <iframe name="TextField"></iframe>
74 <div class="buttons">
75 <form class="fr_save_edit">
76 <button type="button" id="save_edit" class="fa" style="margin-right: auto; display: none">Сохранить изменения</button>
77 </form>
78 <input type="text" class="name" name="name" style="margin-right: 1em;" placeholder="Название документа">
79 <button type="button" class="fa" id="save">Сохранить</button>
80 </div>
81 </form>
82
83 <div class="history">
84 <p style="text-align: center; margin: 1em auto 2em auto;">Сохраненные отчеты</p>
85 <table>
86 <thead>
87 <tr>
88 <th>ID</th>
89 <th>Название отчета</th>
90 <th>Действия</th>
91 </tr>
92 </thead>
93 <tbody>
94 <{% for row in table %}>
95 <tr name={{ row.id }}>
96 <td>{{ row.id }}</td>
97 <td>{{ row.name }}</td>
98 <td>
99 <form class="fr_view">
100 <button type="button" class="submit fa" class="view">
101 <a href="/view/{{ row.id }}">Сформировать</a></button>
102 </form>
103 <form class="fr_edit">
104 <button type="button" class="submit fa" class="edit">Изменить</button>
105 </form>
106 <form class="fr_del">
107 <button type="button" class="submit fa" class="del">Удалить</button>
108 </form>
109 </td>
110 </tr>
111 <{% endfor %}>
112 </tbody>
113 </table>
114 </div>
115

```

Рисунок А.1 – Код разработки окна текстового редактора и списка сохраненных отчетных форм

```

7 <link rel="stylesheet" href="{% static 'css/editor.css' %}?v={% now '%m/%d/%y %H:%M:%S.%f' %};?}>
8 <title>{{ report.name }}</title>
9 </head>
10 <body style="background-color: #ffff">
11 <button type="button" class="submit" class="view" style="background-color: #b8b8b8; margin-bottom: 2em;">
12 <a href="/">Назад</a>
13 </button>
14 <button class="fa fa-print" style="background-color: #b8b8b8; margin-left: 1em;">
15 <a onClick="javascript:printDiv('frame');" title="Печать">Распечатать</a>
16 </button>
17 <div class="no_vis">{{ report.report }}</div>
18 <div class="frame" id="frame"></div>
19

```

Рисунок А.2 – Код разработки окна сформированной документа и печати



```

1
2 var showingSourceCode = false;
3 var isInEditMode = true;
4
5 function enableEditMode() {
6     TextField.document.designMode = 'On';
7 };
8
9 function execCmd(command) {
10     TextField.document.execCommand(command, false, null);
11 };
12
13 function execCommandWithArg(command, arg) {
14     TextField.document.execCommand(command, false, arg);
15 };
16
17 function toggleSource(){
18     if (showingSourceCode) {
19         TextField.document.getElementsByTagName('body')[0].innerHTML = TextField.document.getElementsByTagName('body')[0].
20             textContent;
21         showingSourceCode = false;
22     }else{
23         TextField.document.getElementsByTagName('body')[0].textContent = TextField.document.getElementsByTagName('body')[0].
24             innerHTML;
25         showingSourceCode = true;
26     };
27 };
28
29 function toggleEdit(){
30     if (isInEditMode) {
31         TextField.document.designMode = 'Off';
32         isInEditMode = false;
33     }else{
34         TextField.document.designMode = 'On';
35         isInEditMode = true;
36     };
37 };
38

```

Рисунок А.3 – Код возможности форматирования создаваемого документа

```

77
78     $.ajax({
79         url: pathName,
80         type: methodAjax,
81         dataType : 'json',
82         data: JSON.stringify(json),
83
84         contentType: 'application/json;charset=UTF-8',
85         success: function (response) {
86             if (methodAjax === "DELETE" || "POST") {
87                 location.reload();
88             } else if (response.report.edit == "edit") {
89                 TextField.document.getElementsByTagName('body')[0].innerHTML = response.report.report;
90                 $('<div name="">.parent().find('input')[0].value = response.report.name;
91                 $('<div name="">#save_edit').css({'display': 'block'}).attr('rep_id', response.report.id);
92             } else if (methodAjax === "PUT") {
93                 location.reload();
94             }
95
96             location.reload();
97             $('<div name="">.name')[0].value = '';
98             //console.log(response)
99         },
100         error: function (response) {
101             console.log("Ошибка добавления", response)
102         }
103     });
104     return false;
105 }
106

```

Рисунок А.4 – Код передачи и приема обработанных данных

```

107
108
109 // событие при нажатии на одну из кнопок
110 $('#save').on('click', function () {
111     method_ajax($(this), "POST", 'save/');
112 });
113
114 $('.fr_save_edit').on('click', 'button', function () {
115     method_ajax($(this), "PUT", 'edit/');
116 });
117
118 $('.fr_view').on('click', 'button', function () {
119     method_ajax($(this), "GET", 'view/');
120 });
121
122 $('.fr_del').on('click', 'button', function () {
123     method_ajax($(this), "DELETE", 'del/');
124 });
125
126 $('.fr_edit').on('click', 'button', function () {
127     method_ajax($(this), "GET", 'edit/');
128 });
129
130 });

```

Рисунок А.5 – Код отслеживания событий, при нажатии на кнопки программного интерфейса

```

1  $('.frame').append($('.no_vis').text());
2  $('.no_vis').css({'display': 'none'});
3
4  function printDiv(divId) {
5      var divContent = document.getElementById(divId).innerHTML;
6      var printWin = window.open('', '_blank', 'width=800,height=600');
7      printWin.document.write('<html><head><title>Печать содержимого</title></head><body>{divContent}</body></html>');
8      printWin.document.close();
9      printWin.addEventListener('load', function() {
10         printWin.print();
11         printWin.close();
12     }, true);
13 }
..

```

Рисунок А.6 – Код формирования и вывода собранного документа

## Приложение Б

```
1 from django.db import models
2
3 class Reports(models.Model):
4     name = models.CharField(max_length=100)
5     report = models.TextField()
6
7
8     def __str__(self):
9         return self.name
10
```

Рисунок Б.1 – Модель базы данных для хранения отчетных форм

```
1 from django.contrib import admin
2 from django.urls import path
3 from report import views as report
4
5 urlpatterns = [
6     path('', report.index),
7     path('save/', report.save),
8     path('del/', report.delete),
9     path('edit/', report.edit),
10    path('edit/<int:rep_id>', report.edit),
11    path('view/', report.view),
12    path('view/<int:rep_id>', report.view),
13    path('admin/', admin.site.urls),
14 ]
```

Рисунок Б.2 – Программный код маршрутизации данных

```

37
38 @csrf_exempt
39 def view(request, rep_id = None):
40     if rep_id != None:
41         report = Reports.objects.get(id=rep_id)
42         report = model_to_dict(report)
43         print('data', report)
44         return render(request, 'view.html', {'report': report})
45
46     elif request.method == "GET":
47         idd = json.loads(request.body.decode('utf-8'))
48         report = Reports.objects.get(id=int(idd['id']))
49         report = model_to_dict(report)
50         print('data', report)
51         return JsonResponse({'id': idd}, status=200, content_type="application/json")
52
53     else:
54         return HttpResponseBadRequest()
55
56 @csrf_exempt
57 def edit(request, rep_id = None):
58     if rep_id != None:
59         data = Reports.objects.get(id=rep_id)
60         data = model_to_dict(data)
61         data['edit'] = 'edit'
62         print('data', data)
63         return JsonResponse({'report': data}, status=200, content_type="application/json")
64
65     elif request.method == "PUT":
66         data = json.loads(request.body.decode('utf-8'))
67         report = Reports.objects.get(id=data['id'])
68         print(report)
69         report.name = data['name']
70         report.report = data['text']
71         report.save()
72         report = model_to_dict(report)
73         return JsonResponse({'report': report}, status=200, content_type="application/json")
74
75     else:
76         return HttpResponseBadRequest()

```

Рисунок Б.3 – Код формирования отчетных форм и логика редактирования документов