

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Кафедра «Прикладная математика и информатика»
(наименование)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Разработка программного обеспечения

(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему «Разработка веб-приложения для оптимизации процессов электронного документооборота»

Обучающийся

Е.С. Иевлева

(И.О. Фамилия)

(личная подпись)

Руководитель

к. э. н., доцент, Т.А. Раченко

(ученая степень, звание, И.О. Фамилия)

Консультант

к. ф. н., доцент, М.В. Дайнеко

(ученая степень, звание, И.О. Фамилия)

Тольятти 2025

Аннотация

Тема выпускной квалификационной работы: «Разработка веб-приложения для оптимизации процессов электронного документооборота».

Работа была выполнена студентом Тольяттинского Государственного Университета, кафедры прикладной математики и информатики, группы ПИБ-2106а, Иевлевой Екатериной Сергеевной.

Объект исследования – система управления документами предприятия ООО «Ворлдинтертех Рус». Предмет исследования – автоматизация работы с документами на предприятии ООО «Ворлдинтертех Рус».

Целью исследования является создание эффективного программного обеспечения, способствующего улучшению процессов управления документами в организации. Основные задачи включают в себя:

1. Определение функциональных требований к ПО, анализ бизнес-процессов ООО «Ворлдинтертех Рус».
2. Проектирование базы данных, выбор технологий и платформ для реализации.
3. Создание базы данных и привязка к системе, создание прототипа приложения.
4. Реализация функций авторизации и аутентификации.
5. Внедрение, проверка эффективности и удобства пользования.

Актуальность исследования обусловлена растущей необходимостью повышения эффективности управления информацией на предприятии.

Результатом работы является приложение «ClipFolio» для работы с файлами.

Бакалаврская работа содержит пояснительную записку объемом 52 страницы, включая 21 иллюстраций, 5 таблиц, список литературы из 26 наименований, включая 7 зарубежных.

Abstract

The title of the graduation work is *Developing a web application for optimizing electronic document workflow processes*.

The graduation work consists of an introduction, 3 parts, 21 figures, 5 tables, a conclusion, and a list of 26 references including 7 foreign sources.

The aim of this graduation work is to create effective software contributing to improving the document workflow processes in an organization.

The object of the graduation work is the document management system of OOO “Worldintertech Rus” (LLC under the laws of the Russian Federation).

The subject of the graduation work is the automation of working with the documents at the company under research.

The graduation work may be divided into several logically related parts which are: studying the subject area, designing and developing the software, as well as testing and evaluating its effectiveness.

The first part analyzes the software analogues on the market and defines the functional and non-functional requirements.

In the second part, the software and database structure is described, the choice of the development tools is explained, and the specified functions are implemented.

The third part deals with testing and debugging the code, describing the results obtained after implementing the application at the enterprise. In this part, the application effectiveness and convenience are evaluated as well.

In conclusion, it should be emphasized that the implementation of the new software can have a positive influence on changing the work culture of the company.

The developed application has great development prospects and can become not just a convenient tool, but also an entire corporate system.

The work is of interest for a wide circle of readers interested in developing tools for optimizing business processes and Java applications.

Оглавление

Введение	5
Глава 1 Постановка задачи на разработку программного обеспечения для информационной системы предприятия	8
1.1 Значение программного обеспечения для информационной системы предприятия	8
1.2 Основные функции программного обеспечения для информационной системы предприятия	13
1.3 Обзор существующих решений и их анализ	17
1.4 Требования к функционалу и интерфейсу приложения	21
Глава 2 Процесс разработки программного обеспечения	25
2.1 Проектирование программного обеспечения	25
2.2 Выбор технологий и инструментов для разработки	29
2.3 Реализация функциональных требований	32
Глава 3 Оценка эффективности разработанного программного обеспечения	36
3.1 Тестирование и отладка приложения	36
3.2 Оценка удобства и функциональности приложения	39
3.3 Анализ результатов использования программного обеспечения	42
Заключение	46
Список используемой литературы	49

Введение

В современном мире информационные технологии проникают во все сферы деятельности человека, из-за чего оптимизация бизнес-процессов становится одной из ключевых задач для предприятий различного масштаба. Одной из наиболее значимых областей, где технологии могут существенно улучшить эффективность работы, является электронный документооборот.

Объектом ВКР является система управления документами предприятия ООО «Ворлдинтертех Рус».

Предметом ВКР является автоматизация работы с документами на предприятии ООО «Ворлдинтертех Рус».

Актуальность исследования обусловлена растущей необходимостью повышения эффективности управления информацией на предприятии. По мере того как компании расширяют своё цифровое присутствие и переходят на безбумажные технологии, растёт спрос на эффективные решения для хранения, поиска и управления документами. Широкое распространение облачных систем управления документами ещё больше ускорило этот переход, предоставляя компаниям масштабируемые, экономичные и доступные решения. Согласно исследованию, проведенному TAdviser, рынок СЭД, стабильно растёт с каждым годом на 10%. В связи с этими фактами, эффективный документооборот становится ключевым фактором успешной деятельности бизнес-организаций. Он позволяет сократить временные затраты на рутинные задачи и минимизировать ошибки, связанные с ручной обработкой документов.

Целью исследования является создание эффективного программного обеспечения, способствующего улучшению процессов управления документами в организации. Основные задачи включают в себя:

- анализ предметной области (рынка ЭДО) и определение функциональных и нефункциональных требований к ПО;

- проектирование структуры ПО и базы данных, выбор технологий и платформ для реализации;
- реализацию заявленных функции (в т.ч. функций авторизации и аутентификации);
- внедрение, проверку эффективности и удобства пользования.

Оглавление работы подчеркивает системный подход к созданию решения, что позволяет не только решить задачи оптимизации, но и обеспечить интеграцию приложения в существующую систему предприятия. Таким образом, данное исследование имеет значительное практическое значение и отвечает на актуальные вызовы, стоящие перед современными предприятиями в сфере управления документами.

В первой главе рассматривается постановка задачи, включая требования пользователей и спецификации системы. Во второй — рассмотрен процесс разработки веб-приложения на базе Java EE. В третьей – проводится оценка эффективности внедренного решения, анализируя его влияние на скорость и качество документооборота в организации. Таким образом, работа направлена на создание эффективного инструмента для автоматизации документооборота, что является актуальной задачей для современных предприятий.

В данной работе будет рассмотрен процесс разработки веб-приложения на базе Java EE, предназначенного для оптимизации процессов электронного документооборота на предприятии. Мы начнем с постановки задачи на разработку программного обеспечения, в которой будут определены основные требования к системе, ее функции и цели. Важно отметить, что успешная реализация проекта требует четкого понимания текущих бизнес-процессов и выявления узких мест, которые необходимо оптимизировать.

Следующим этапом будет процесс разработки программного обеспечения, который включает в себя проектирование архитектуры системы, выбор технологий и инструментов, а также реализацию и тестирование приложения. В данной главе мы подробно рассмотрим этапы разработки, начиная от анализа требований и заканчивая внедрением готового решения.

Особое внимание будет уделено принципам разработки, таким как модульность, повторное использование компонентов и тестирование, что позволит обеспечить высокое качество конечного продукта.

В третьей главе будет проведена оценка качества работы предложенного программного решения и оценка его эффективности. Мы определим необходимые на данном этапе виды тестирования, и проведем их с помощью. Мы рассмотрим различные метрики и показатели, которые помогут определить, насколько успешно приложение справляется с поставленными задачами. Оценка эффективности будет включать в себя анализ времени, затраченного на обработку документов, уровня удовлетворенности пользователей, а также экономических показателей, таких как снижение затрат на бумажный документооборот и повышение производительности сотрудников.

В заключении работы оценивается эффективности проведенной работы и разработанного приложения. Это позволит определить его влияние на качество работы ООО «Ворлдинтертех Рус», а также выявить возможные направления для дальнейшего совершенствования системы.

Глава 1. Постановка задачи на разработку программного обеспечения для информационной системы предприятия

1.1. Значение программного обеспечения для информационной системы предприятия

Система электронного документооборота (СЭД) представляет собой комплекс программных и аппаратных средств, которые обеспечивают создание, хранение, обработку, передачу и архивирование электронных документов [2]. Она позволяет значительно упростить взаимодействие между различными подразделениями компании, а также между компанией и ее клиентами или партнерами.

Информационная система предприятия (ИСП) – это комплекс программных, технических, организационных и методических средств, предназначенных для сбора, хранения, обработки, анализа и передачи информации, необходимой для управления бизнес-процессами компании [4].

Программное обеспечение служит основой для реализации информационных систем [6], которые, в свою очередь, обеспечивают сбор, хранение, обработку и передачу информации в рамках организации. Оно позволяет автоматизировать рутинные операции [5], минимизировать количество ошибок, связанных с человеческим фактором, и значительно ускорить процессы [7]. В современном мире информационные технологии играют ключевую роль в функционировании предприятий, обеспечивая эффективное управление, автоматизацию процессов и оптимизацию ресурсов. Согласно опросу, проведенному ОФД в 2021 году, около 63% опрошенных российских компаний используют ЭДО в работе (рисунок 1).

Проникновение ЭДО в российский бизнес

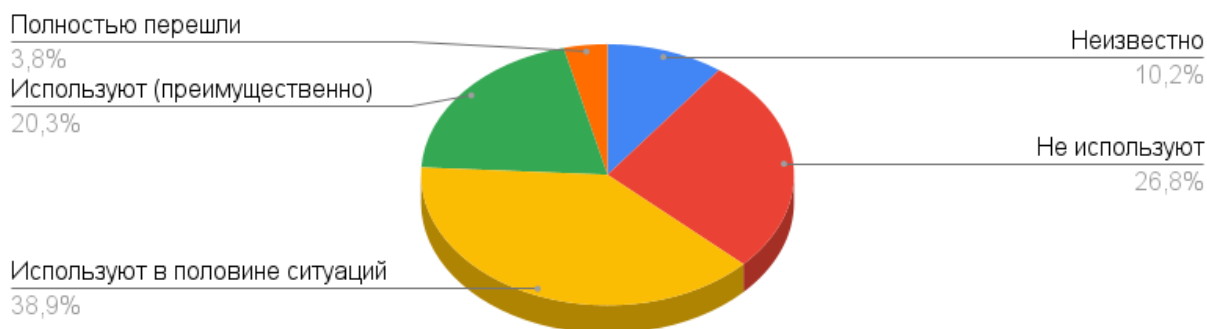


Рисунок 1 – Статистика проникновения ЭДО среди опрошенных компаний, 2021г.

Особенно это актуально для компаний, работающих в сфере высоких технологий, таких как ООО «Ворлдинтертех Рус», где требуется постоянное обновление и адаптация к быстро меняющимся условиям рынка. Разработка программного обеспечения является важной задачей, которая требует комплексного подхода и тщательной проработки всех аспектов, связанных с функционированием бизнеса [8]. В данном контексте особое внимание следует уделить значению программного обеспечения для информационной системы, а также его роли в оптимизации процессов электронного документооборота.

В связи с тем, что компания ведет активную деятельность в различных сферах, требуется оптимизация рутинных действий. Так как на данный момент в компании не внедрена система электронного документооборота, то процесс подписания договоров, счетов, актов и других необходимых документов занимает много времени, а ручная перепись информации из документов может вызывать ошибки.

Рутинные действия вынуждают сотрудников отвлекаться от более приоритетных задач. В связи с этим компания теряет часть человеческого

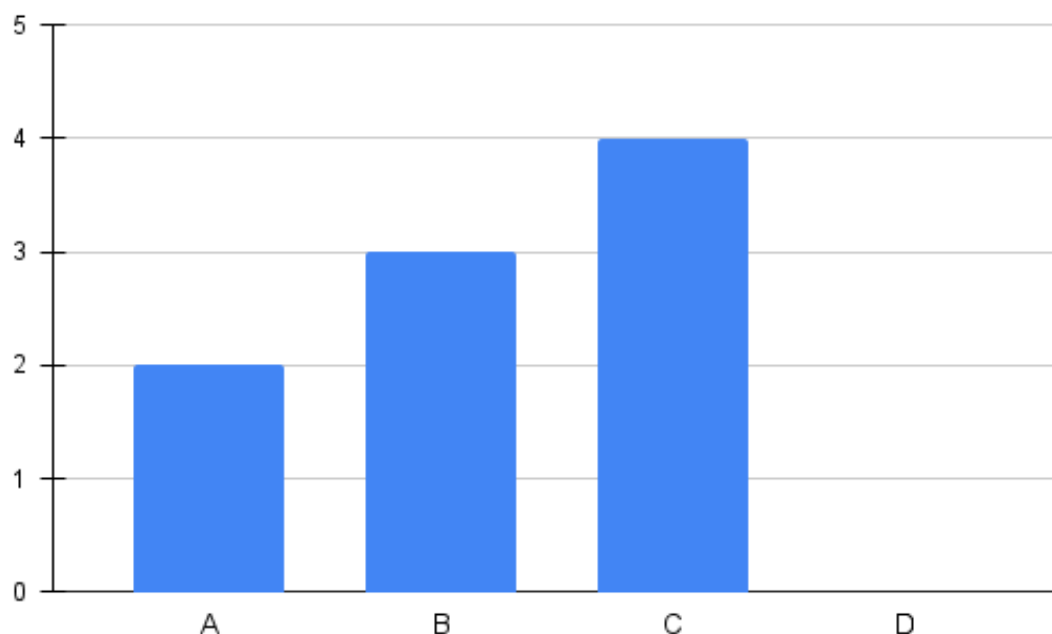
потенциала своих сотрудников. Ниже представлена диаграмма «Причина-следствие» (рисунок 2).



Рисунок 2 – диаграмма «Причина-следствие»

Для того, чтобы оценить, как программное обеспечение может изменить ситуацию и решить выявленные проблемы, было проведено исследование. Мы организовали имитацию рабочего дня нескольких сотрудников, а после этого, опросили их. На диаграмме ниже представлены результаты опроса офисных работников ДО внедрения СЭД в рабочий процесс (рисунок 3).

Сколько времени вы тратили на работу с документами в архивах?

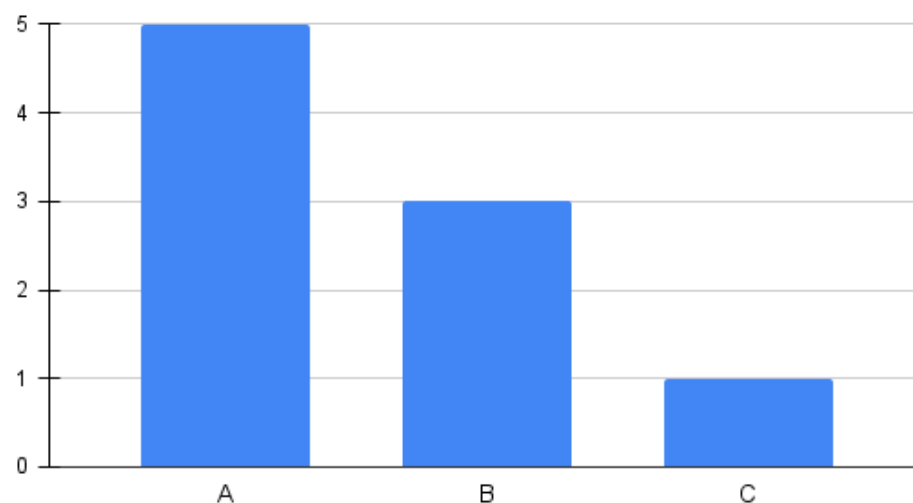


Шкалы: А – более трех часов в день, В – более одного часа в день, С – более двух часов в день, D – менее часа в день.

Рисунок 3 – Статистика распределения времени офисных сотрудников до внедрения СЭД

Следующим действием мы внедрили СЭД в рабочий процесс респондентов и полученные результаты опроса составили в диаграмму (рисунок 4).

Сколько примерно времени в день уходило на работу с документами ПОСЛЕ внедрения приложения?



Шкалы: А – менее одного часа в день, В – более одного часа в день, С – более двух часов в день.

Рисунок 4 – Статистика распределения времени офисных сотрудников после внедрения СЭД

Можно сделать вывод, что внедрение СЭД поможет избежать ошибок, связанных с человеческим фактором и ускорить передачу документов, так как не требуется пересылки оригинальных бумаг. А также помимо вышеперечисленного, СЭД обеспечит удобное и упорядоченное хранение всех документов в одном месте.

В процессе разработки программного обеспечения для информационной системы предприятия необходимо учитывать не только технические, но и организационные аспекты. Важно обеспечить взаимодействие между различными подразделениями компании, которые будут использовать систему. Это может потребовать проведения дополнительных встреч и консультаций с представителями различных отделов, чтобы понять их потребности и ожидания от новой системы. Кроме того, необходимо предусмотреть обучение сотрудников, что позволит им быстро освоить новую систему и начать эффективно ее использовать.

Значение программного обеспечения для ИСП ООО «Ворлдинтертех Рус» невозможно переоценить, так как оно является основой для автоматизации процессов, повышения эффективности работы и обеспечения безопасности данных. Внедрение современного решения для оптимизации документооборота позволит компании не только улучшить текущие бизнес-процессы, но и создать архитектуру, способную адаптироваться к новым вызовам и требованиям рынка.

1.2 Основные функции программного обеспечения для информационной системы предприятия

В контексте задачи, поставленной перед ООО «Ворлдинтертех Рус», необходимо определить основные функции программного обеспечения, которое будет служить основой для создания web-приложения. Важно учитывать, что информационная система должна быть не просто набором инструментов, а интегрированным решением, способным обеспечить взаимодействие всех подразделений компании, а также упростить и ускорить процессы, связанные с обработкой документов.

Одной из ключевых функций программного обеспечения является автоматизация документооборота. Это включает в себя создание, обработку, хранение и передачу электронных документов, что позволяет существенно сократить время, затрачиваемое на выполнение рутинных задач. Кроме того, система должна поддерживать возможность работы с различными форматами файлов, такими как PDF, DOCX, PNG и другими, что обеспечит большую гибкость в работе с документами. Ниже представлена диаграмма, демонстрирующая, как пользователь контактирует с ПО, включая интерфейсы, входные данные и выходные данные (рисунок 5).

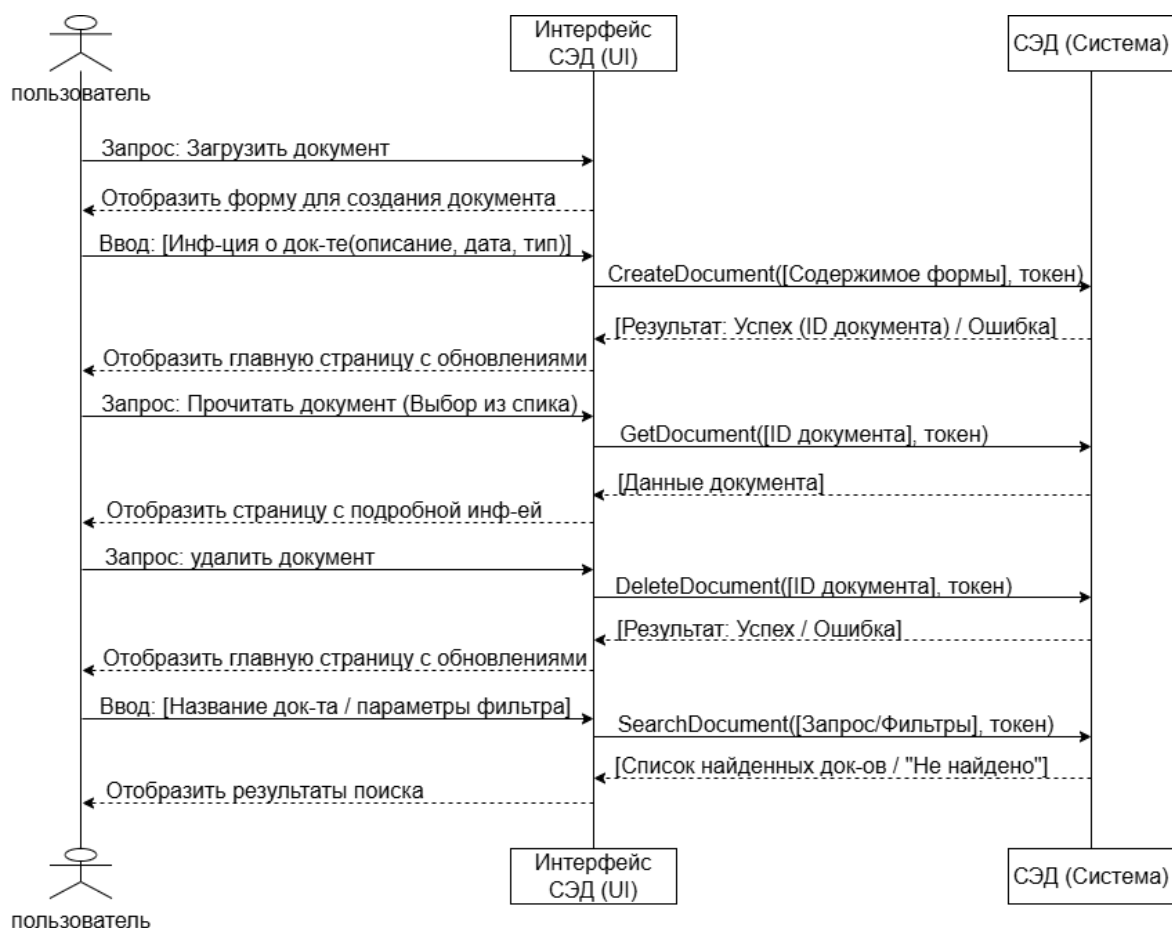


Рисунок 5 – Диаграмма взаимодействия пользователя с системой.
Работа с документами

Система должна обеспечивать высокий уровень безопасности данных [9]. В условиях современного бизнеса защита информации становится все более актуальной. Для этого необходимо реализовать систему аутентификации и авторизации пользователей [19]. В разработанном приложении использовалась архитектура Spring Security [16]. Она строится на фильтрах Servlet (FilterChain), которые обрабатывают каждый HTTP-запрос до попадания в контроллер (рисунок 6).

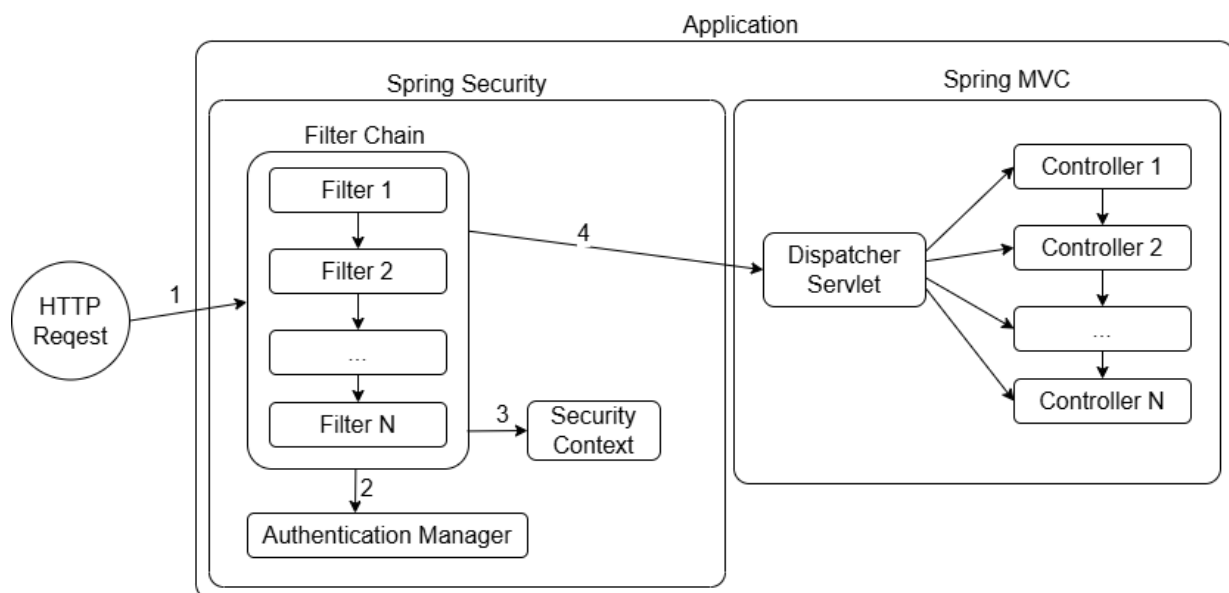


Рисунок 6 – Архитектура Spring Security

Spring Security реализует аутентификацию пользователя по следующему сценарию:

- пользователь вводит логин/пароль на /login;
- Spring Security перехватывает запрос через UsernamePasswordAuthenticationFilter. Далее создает объект Authentication (логин + пароль) и передает его в AuthenticationManager;
- AuthenticationManager использует UserDetailsService для загрузки пользователя из БД (или памяти);
- PasswordEncoder проверяет пароль. Если успешно – создается аутентифицированный объект Authentication и сохраняется в SecurityContextHolder.

Важно подчеркнуть, что во время авторизации пользователя Spring Security действует следующим образом:

- для каждого запроса (/admin, /user) проверяются права через AuthorizationManager;
- если у пользователя нет нужной роли, то возвращается ошибка с кодом 403.

Ниже представлена диаграмма взаимодействия пользователя и ПО (рисунок 7).

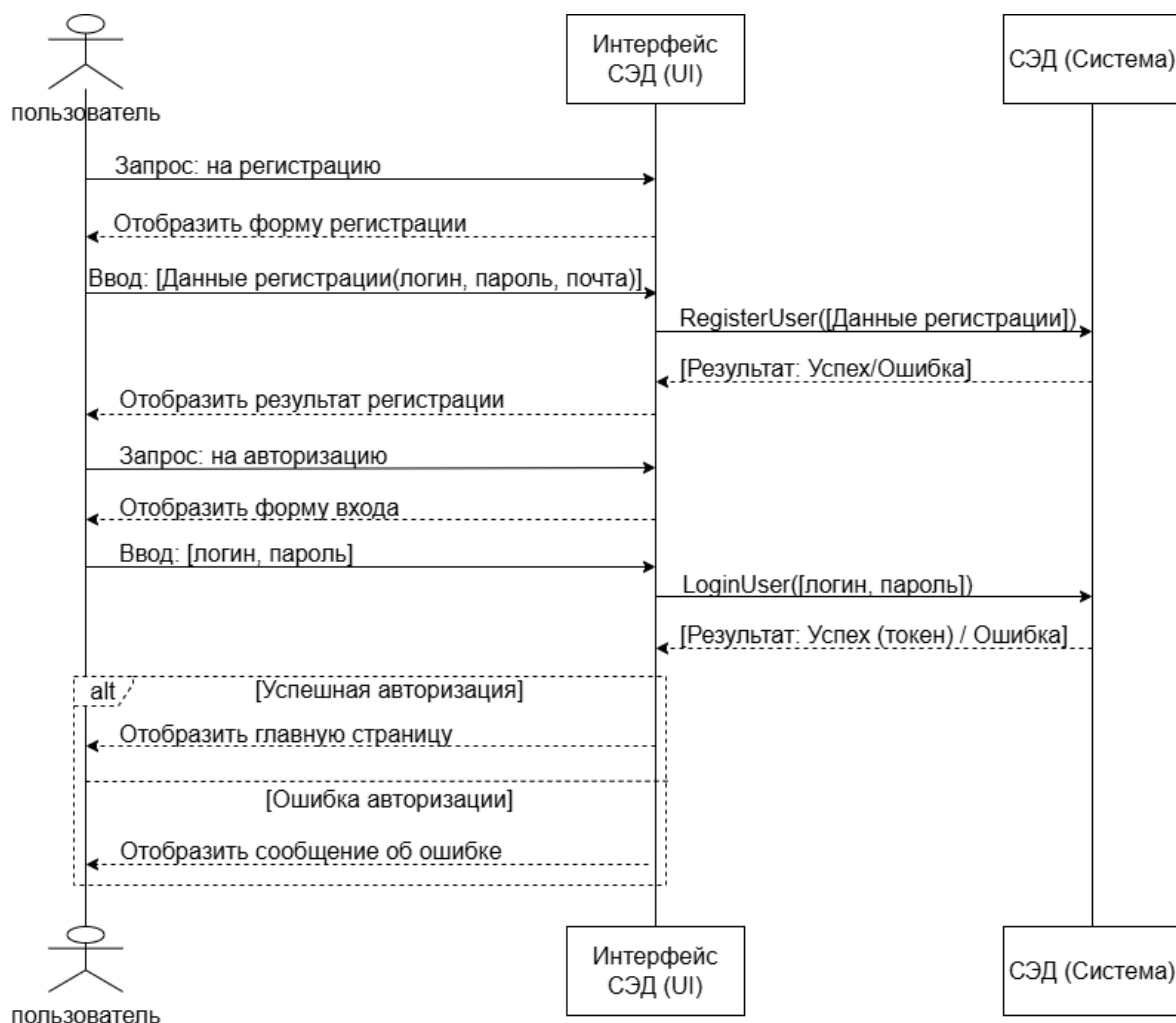


Рисунок 7 – Диаграмма взаимодействия пользователя с системой.
Регистрация и авторизация

В заключение, разработка программного обеспечения для информационной системы предприятия ООО «Ворлдинтертех Рус» должна учитывать все вышеперечисленные функции и аспекты. Это позволит создать мощный инструмент для оптимизации процессов электронного документооборота, который будет способствовать повышению эффективности работы компании в целом. Важно, чтобы система была гибкой, масштабируемой и безопасной, а также соответствовала современным требованиям пользователей и бизнеса.

1.3. Обзор существующих решений и их анализ

В процессе разработки программного обеспечения для ООО «Ворлдинтертех Рус», особенно в контексте оптимизации процессов электронного документооборота, необходимо тщательно проанализировать существующие решения на рынке. Это позволит не только понять текущие тренды и технологии, но и выявить недостатки, которые можно устранить в разрабатываемом приложении. В данном разделе мы рассмотрим основные подходы к автоматизации документооборота, существующие программные решения, их функциональные возможности, а также плюсы и минусы каждого из них.

На сегодняшний день существует множество систем для автоматизации документооборота [1], которые различаются по функционалу, архитектуре, используемым технологиям и целевой аудитории [10]. К числу наиболее популярных решений можно отнести такие системы, как 1С:Документооборот, Microsoft SharePoint, SAP Document Management System [11]. Каждое из этих решений имеет свои особенности, которые могут быть как преимуществами, так и недостатками в зависимости от конкретных потребностей предприятия.

1С:Документооборот является одним из наиболее распространенных решений в России [1]. Оно предлагает широкий спектр функциональных возможностей, включая автоматизацию обработки входящей и исходящей документации, управление задачами и проектами, а также интеграцию с другими продуктами 1С. Система позволяет настроить бизнес-процессы под конкретные нужды компании, что делает её гибкой и адаптируемой. Однако для 1С не подходит предприятию малого бизнеса по следующим причинам:

- Высокая стоимость внедрения и поддержки.
- Необходимость обучения сотрудников.

– Система может быть сложной в настройке и эксплуатации, что требует наличия квалифицированных специалистов.

Microsoft SharePoint, в свою очередь, предлагает мощные инструменты для совместной работы и управления документами. Он позволяет создавать корпоративные порталы, настраивать рабочие процессы и интегрировать с другими продуктами Microsoft. SharePoint хорошо подходит для крупных организаций, где важна возможность масштабируемости и интеграции с существующими системами. Однако для малых и средних предприятий он может оказаться неудобным и убыточным решением по тем же причинам, что и 1С:Документооборот: неудовлетворительная стоимость внедрения и вложения в обучение кадров.

SAP Document Management System (DMS) – это решение, ориентированное на крупные предприятия с высокими требованиями к безопасности и управлению документами [23]. Оно предлагает широкий набор инструментов для хранения, поиска и обработки документов, а также интеграцию с другими модулями SAP. Однако, как и в случае с другими крупными системами, внедрение SAP DMS может потребовать значительных затрат и времени на настройку. Кроме того, система может оказаться избыточной для небольших компаний, которым достаточно более простых решений.

Проведем сравнительный анализ, чтобы помочь пользователям выбрать наиболее подходящее решение для своих нужд (результат представлен в таблице 1). Анализ будет проводиться по следующим критериям: функциональность, стоимость, удобство использования, поддержка и безопасность. Каждый критерий будет оцениваться по шкале от 1 до 5.

Таблица 1 - Сравнительный анализ аналогов

Критерии	1С:Документооборот	SAP Document Management System	Microsoft. SharePoint
----------	--------------------	--------------------------------	-----------------------

Функциональность	5	5	5
Стоимость	2	2	4
Удобство использования	2	2	3
Поддержка	4	4	4
Безопасность	5	4	4

Для наглядности представим результаты сравнительного анализа в виде столбчатой диаграммы (рисунок 8).

Сравнение аналогов ПО

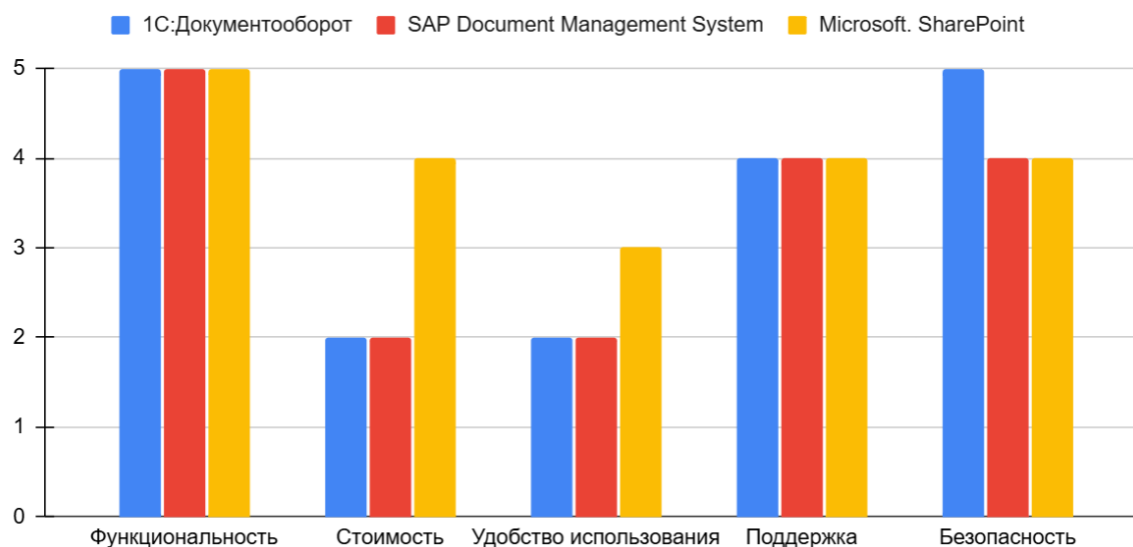


Рисунок 8 - Сравнительный анализ аналогов

На основе проведенного анализа можно сделать выводы о том, какое программное обеспечение подходит для разных категорий пользователей. Представим информацию о целевой аудитории с акцентом на отраслевое распределение в виде столбчатой диаграммы (рисунок 9).

ЦА аналогичных ПО

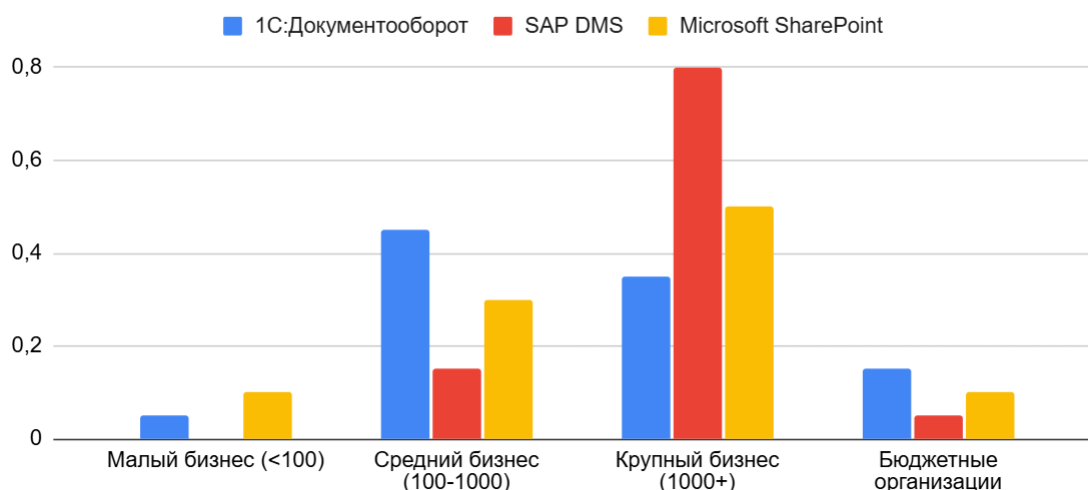


Рисунок 9 - Целевая аудитория аналогов ПО с акцентом на отраслевое распределение

Проанализировав информацию о клиентах представленных аналогов можно сделать следующие выводы:

1С:Документооборот хорошо подойдет для среднего бизнес (45%) и крупных локальных компаний (35%) и бюджетных организаций (школы, больницы) (15%). Малый бизнес в числе его клиентов почти не представлен (5%).

SAP DMS в основном используется крупными корпорациями (промышленность, нефтегаз) (80%) и практически не имеет клиентов малого бизнеса. Слабое проникновение в госсектор (5%) и средний бизнес (15%).

Microsoft SharePoint востребован среди предприятий крупного (50%) и среднего бизнеса (30%). Имеет некоторое количество клиентов в малом бизнесе в основном в сферах стартапов в IT и консалтинга (10%).

Каждое из рассмотренных решений имеет свои сильные и слабые стороны. Выбор подходящего решения зависит от конкретных потребностей и ресурсов пользователя, и на данный момент представляется сложной задачей.

Анализ существующих решений показывает, что на рынке не хватает программного обеспечения, которое бы полностью удовлетворяло

потребности предприятий малого бизнеса, таких как ООО «Ворлдинтертех Рус».

1.4. Требования к функционалу и интерфейсу приложения

В рамках разработки программного обеспечения для ИСП, необходимо детально проработать требования к функционалу и интерфейсу приложения. Важно уточнить, что успех проекта зависит не только от технических решений, но и от тщательной проработки требований, так как они определяют, каким образом пользователи будут взаимодействовать с системой, какие функции она будет выполнять и как будет организован процесс работы с документами. Основываясь на специфике работы ООО «Ворлдинтертех Рус» и потребностей ее сотрудников, выделим ключевые аспекты.

Во-первых, функционал приложения должен обеспечивать полный цикл работы с документами, начиная от их создания и редактирования, заканчивая хранением и архивированием. Во-вторых, важным аспектом является интеграция системы с другими информационными системами, которые используются в организации. В-третьих, приложение должно предоставлять возможность настройки прав доступа для различных категорий пользователей. Четвертым аспектом является необходимость реализовать функции поиска по документам, включая фильтрацию по типу документа. Пятым аспектом является обеспечение удобного и интуитивно понятного интерфейса приложения. Пользователи должны легко ориентироваться в системе, не тратя много времени на обучение. Важно учесть, что приложение должно быть адаптивным и корректно отображаться на различных устройствах.

На основе аспектов сформируем функциональные требования к системе:

- Создание, удаление, чтение документов;
- Разделение ролей и управление пользователями;
- Регистрация и авторизация в системе;
- Поиск и фильтрация документов.

Функциональные требования к системе были формализованы с учетом рекомендаций по проектированию ИС [12], [18]. Представим функциональные требования в виде таблицы, сделав акцент на взаимодействие пользователя с функцией (таблица 2).

Таблица 2 - функциональные требования

Функция	Описание	Пример сценария
Работа с документами	Пользователь может базовый набор операций для работы с документами: - создавать (Create); - просматривать (Read); - удалять (Delete). Пользователь должен иметь возможность внести дополнительную информацию при загрузке документа.	Пользователь заполняет форму и нажимает кнопку «Создать». После чего в базе данных создается новая запись с документом. Перейдя на страницу подробной информации, пользователь может просмотреть документ и дополнительную информацию. Нажав на кнопку «Удалить», пользователь удаляет документ из БД.
Управление пользователями	- Регистрация и аутентификация: Пользователь должен иметь возможность создавать учетные записи и входить в систему. – Управление ролями: Администратор должен иметь возможность удалять простых пользователей и управлять этими аккаунтами.	- Пользователь нажимает кнопку «войти», после чего переходит на страницу входа, вводит данные и попадает на главную страницу. - Пользователь с правами администратора нажимает на кнопку, открывающую панель администратора, напротив ненужного пользователя нажимает кнопку «удалить», после чего этот аккаунт больше недействителен.
Поиск и фильтрация	- Пользователь должен иметь возможность из списка всех документов найти нужный. - Пользователь должен иметь возможность отфильтровать список всех документов для своего удобства.	- На главной странице пользователь вводит название документа в поисковую строку, после чего из списка пропадают все документы, у которых название не совпадает. - На главной странице пользователь нажимает на фильтр и из выпадающего списка выбирает нужный параметр, после чего из списка пропадают все документы, у которых этот параметр не совпадает.

Нефункциональные требования описывают характеристики системы, такие как производительность, безопасность и удобство использования [24]. Определим нефункциональные требования к разрабатываемому приложению:

- приложение должно обрабатывать запросы пользователей в течение 2 секунд;
- приложение должно обеспечивать поддержку одновременной работы не менее 100 пользователей;
- интерфейс приложения должен быть интуитивно понятным и доступным для пользователей без технического образования;
- адаптивный дизайн для работы на различных устройствах (ПК, планшеты, мобильные телефоны).

Определим четкий алгоритм, по которому будут проводиться тесты. Представим его в виде блок-схемы на рисунке 10.

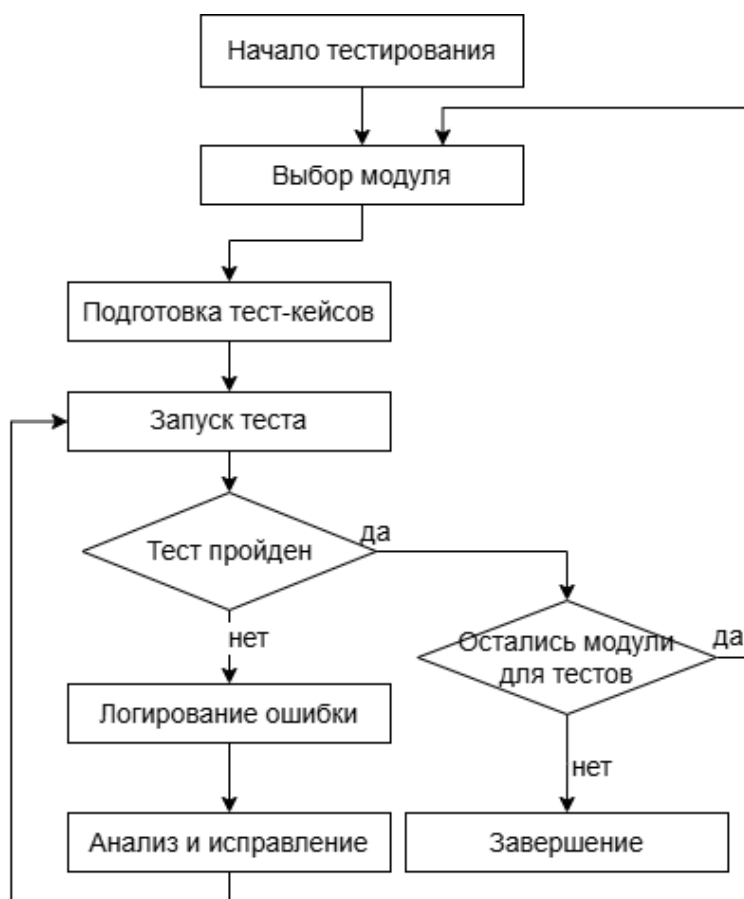


Рисунок 10 – блок-схема тестирования использования ПО

Кроме того, необходимо планировать регулярные обновления системы, которые будут включать в себя исправления ошибок, улучшения функционала

и внедрение новых возможностей, что позволит поддерживать актуальность и конкурентоспособность приложения.

Требования к функционалу и интерфейсу приложения являются ключевыми для его успешной реализации и удовлетворения потребностей пользователей. Визуализация этих требований через таблицы позволило лучше понять и проиллюстрировать основные аспекты, что, в свою очередь, поможет в разработке эффективного и удобного приложения. Для понимания полноты требований дополнительно по FURPS+ представим их в виде таблицы 3:

Таблица 3 - описание требований по FURPS+

Категория	Критерии
Functionality	Хранение, загрузка, скачивание, поиск, интеграция на предприятие
Usability	Интерфейс, простота навигации, обучаемость
Reliability	Стабильность, backup, защита от потери данных
Performance	Скорость загрузки/скачивания, работа с разными типами файлов (PDF, DOCX)
Support	Документация, техподдержка
+Security	RBAC, аудит действий пользователя
+Scalability	Поддержка растущего числа пользователей/файлов

Учитывая все вышеперечисленные аспекты, можно создать эффективное и удобное решение, которое будет соответствовать потребностям пользователей и требованиям современного бизнеса. Важно помнить, что успешная реализация проекта зависит не только от технических решений, но и от тщательной проработки требований, которые должны быть основаны на понимании специфики работы организации и потребностей ее сотрудников.

Глава 2. Процесс разработки программного обеспечения

2.1. Проектирование программного обеспечения

В рамках данной главы особое внимание будет уделено этапу проектирования программного обеспечения. Проектирование программного обеспечения – это процесс, в ходе которого разрабатывается архитектура и структура системы, определяются ее компоненты и их взаимодействия, а также формируются требования к функциональности и производительности. Этот этап является критически важным, так как от качества проектирования зависит успешность реализации всего проекта, его соответствие требованиям пользователей и возможность дальнейшего масштабирования и модификации системы.

Прежде чем перейти к выбору технологий, следует приступить к более детальному рассмотрению компонентов системы. Ниже представлена диаграмма компонентов в нотации UML, которая наглядно демонстрирует структуру Java-приложения, его основные компоненты и их взаимосвязи (рисунок 11) [13]. Цель диаграммы компонентов - показать, как различные части системы взаимодействуют друг с другом, чтобы реализовать заявленные функции: создание, удаление, чтение документов; разделение ролей и управление пользователями; регистрация и авторизация; поиск и фильтрация документов.

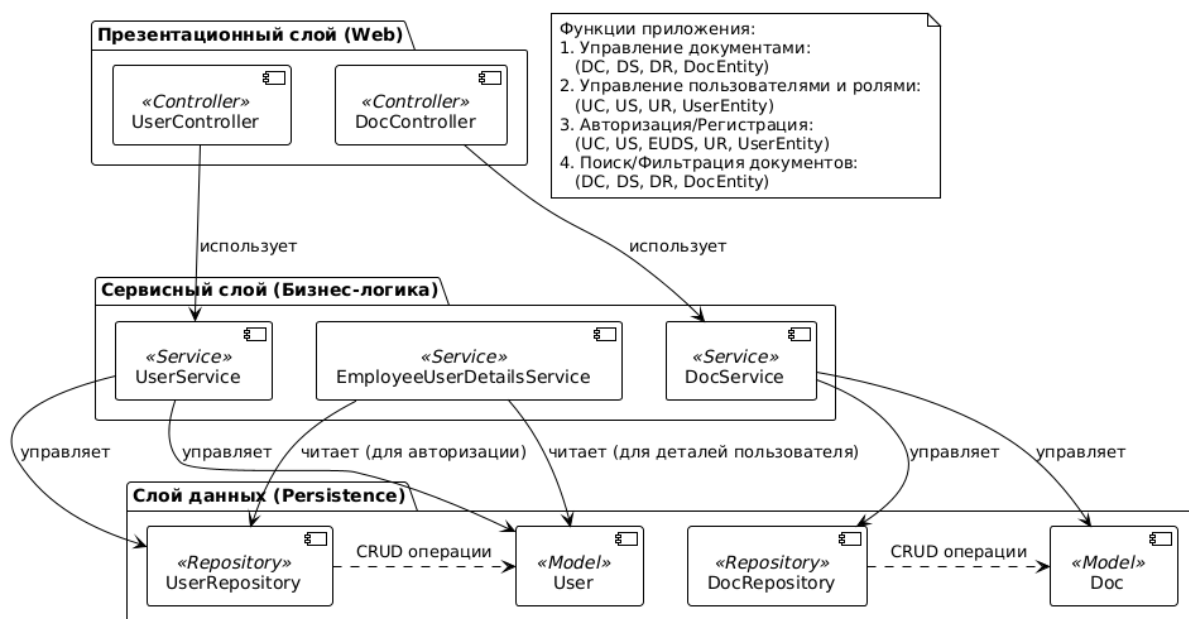


Рисунок 11 – Диаграмма компонентов

После того, как были определены основные компоненты системы, следует рассмотреть проектирование базы данных. Чтобы понять, как организованы данные и как они связаны друг с другом представим диаграмму отношений. Для хранения информации реализуется связь между сущностями User, Role и Doc, как показано на ER-диаграмме (рисунок 12) [17]. Диаграмма сфокусирована на трёх ключевых сущностях: User (пользователь), Role (роль в системе) Doc (документ), их атрибутах и связях между ними. Связь «многие ко многим» (Many-to-Many) между User и Role означает, что один пользователь может иметь несколько ролей, и одна роль может быть назначена нескольким пользователям.

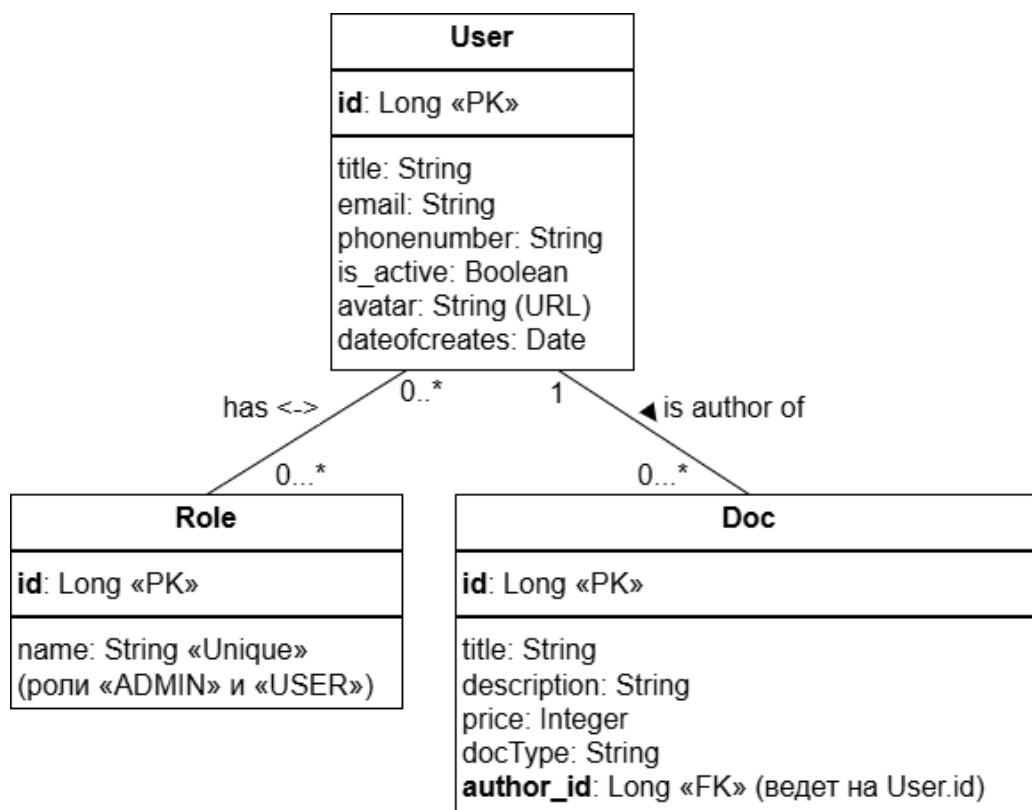


Рисунок 12 - ER-диаграмма

Для повышения удобства использования системы необходимо реализовать интуитивно понятный интерфейс [14], [15]. Пользователи должны иметь возможность легко ориентироваться в приложении, находить необходимые функции и документы. Это можно достичь путем применения современных подходов к дизайну: стиля минимализм, акцент на контент, отсутствие лишних элементов. Макет интерфейса представлен на рисунке 13.

CLIPFOLIO

Поиск документов

Фильтр

Поиск по названию документа

Без фильтра

Найти

3 Общие требования к созданию докумен

- 3.1 Документы могут создаваться на бумажном носителе и в электронной форме с соблюдением установленных правил оформления документов.
- 3.2 При создании документов на двух и более страницах верхняя и последняя страницы пронумерованы.
- 3.3 Требования к созданию документов на расстоянии 10 мм от верхнего края листа.
- 3.4 Требования к созданию документов на лицевой и оборотной стороне листа. При документировании документов лицевой лист должен быть равен.
- 3.5 Предпочтительные гарнитуры и размеры шрифтов для реквизитов документов:
- Times New Roman № 12, 14;
- Arial № 12, 14;
- Courier № 12, 14;
- Сайбл № 14 и приближенные к ним.
- При составлении таблиц документов используются шрифты меньшего размера.

Название: Акт1_000_Лиса

14 990

Тип: Акт об оказанных услугах

3 Общие требования к созданию докумен

- 3.1 Документы могут создаваться на бумажном носителе и в электронной форме с соблюдением установленных правил оформления документов.
- 3.2 При создании документов на двух и более страницах верхняя и последняя страницы пронумерованы.
- 3.3 Требования к созданию документов на расстоянии 10 мм от верхнего края листа.
- 3.4 Требования к созданию документов на лицевой и оборотной стороне листа. При документировании документов лицевой лист должен быть равен.
- 3.5 Предпочтительные гарнитуры и размеры шрифтов для реквизитов документов:
- Times New Roman № 12, 14;
- Arial № 12, 14;
- Courier № 12, 14;
- Сайбл № 14 и приближенные к ним.
- При составлении таблиц документов используются шрифты меньшего размера.

Название: Акт3_000_Лиса

5 670

Тип: Акт об оказанных услугах

3 Общие требования к созданию докумен

- 3.1 Документы могут создаваться на бумажном носителе и в электронной форме с соблюдением установленных правил оформления документов.
- 3.2 При создании документов на двух и более страницах верхняя и последняя страницы пронумерованы.
- 3.3 Требования к созданию документов на расстоянии 10 мм от верхнего края листа.
- 3.4 Требования к созданию документов на лицевой и оборотной стороне листа. При документировании документов лицевой лист должен быть равен.
- 3.5 Предпочтительные гарнитуры и размеры шрифтов для реквизитов документов:
- Times New Roman № 12, 14;
- Arial № 12, 14;
- Courier № 12, 14;
- Сайбл № 14 и приближенные к ним.
- При составлении таблиц документов используются шрифты меньшего размера.

Название: Договор_230899

80 500

Тип: Договор

Рисунок 13 - Дизайн-макет главного экрана приложения

Таким образом, проектирование программного обеспечения – это сложный и многогранный процесс, который требует внимательного подхода и учета множества факторов. На этапе проектирования веб-приложения необходимо учитывать как требования пользователей, так и технические аспекты реализации. Качественное проектирование, основанное на анализе требований, выборе подходящих технологий и инструментов, а также предусмотрении вопросов безопасности и тестирования, является залогом успешной реализации проекта и его соответствия ожиданиям пользователей. Каждый из этапов проектирования требует тщательной проработки и внимания к деталям, что в конечном итоге приведет к созданию надежного и эффективного программного продукта.

2.2. Выбор технологии и инструментов для разработки

Выбор технологий и инструментов для разработки программного обеспечения является одним из ключевых этапов в процессе создания любого веб-приложения, особенно когда речь идет о таких сложных системах, как электронный документооборот.

Java – это объектно-ориентированный язык программирования, который изначально был разработан с акцентом на переносимость и безопасность. Java обладает множеством преимуществ, включая активное сообщество разработчиков, что делает его идеальным выбором для создания масштабируемых и надежных приложений. Данный язык программирования имеет ряд преимуществ:

- стабильность: Java существует десятилетиями и зарекомендовала себя как одна из самых стабильных платформ для корпоративных приложений;
- управление памятью: автоматическая сборка мусора в Java снижает риски утечек памяти, что критично для долго работающих серверных приложений, таких как СЭД;
- многопоточность: встроенные средства для создания многопоточных приложений позволяют эффективно обрабатывать параллельные запросы от множества пользователей, что важно для СЭД;
- обработка исключений: строгая система обработки исключений в Java помогает создавать более отказоустойчивый код.

Несмотря на то, что у Java есть очевидные преимущества, существует множество других платформ, которые тоже можно рассматривать для создания СЭД, удовлетворяющей описанным требованиям. Для того, чтобы окончательно убедиться в том, что Java является лучшим выбором для реализации проекта, проведем анализ доступных средств для разработки и представим в виде таблицы 4. Для проведения сравнительного анализа были

отобраны следующие платформы: Python (Django/ Flask), Node.js (Express/ NestJS), Ruby on Rails, PHP (Laravel/ Symfony), «.NET» (ASP.NET Core).

Таблица 4 - Сравнительный анализ с другими популярными платформами

Платформа	Преимущества	Недостатки (в контексте СЭД с акцентом на Java/Spring)	Почему Java/Spring Boot может быть лучше для СЭД
Python (Django/ Flask)	Быстрая разработка для MVP и небольших проектов. Огромное количество библиотек (в т.ч. для ML/AI, что может быть плюсом в будущем для СЭД). Легкий для изучения.	Производительность Python (из-за GIL) может быть узким местом при очень высоких нагрузках без дополнительных ухищрений (асинхронность, многопроцессорность). Динамическая типизация может привести к ошибкам на этапе выполнения в больших проектах.	Java/Spring изначально спроектированы для высоконагруженных корпоративных систем с упором на многопоточность и надежность. Строгая типизация Java предпочтительнее для критичных систем, где цена ошибки высока. Spring Security часто считается более зрелым для сложных сценариев безопасности.
Node.js (Express/ NestJS)	Высокая производительность для I/O-bound операций благодаря неблокирующему вводу-выводу. JavaScript на фронте и бэкенде (единый язык). Большая npm-экосистема.	Однопоточная природа (требует использования cluster-модуля или PM2 для утилизации всех ядер CPU). «Callback hell» (хотя Promises и async/await это улучшили) или сложность управления асинхронностью в больших проектах. Динамическая типизация (TypeScript решает, но добавляет шаг).	Многопоточная модель Java более привычна для традиционных корпоративных нагрузок и может быть проще в управлении для CPU-bound задач. Экосистема Java и Spring более ориентирована на enterprise-решения с готовыми паттернами для сложных бизнес-процессов и безопасности.
Ruby on Rails	Очень высокая скорость разработки (convention over configuration). Зрелая экосистема, много «гемов».	- Производительность может быть проблемой для очень больших нагрузок. Сообщество и популярность снизились по сравнению с пиком.	Java/Spring обеспечивают лучшую производительность для требовательных СЭД. Больше количество Java-специалистов. Широкая поддержка для долгосрочных проектов.

Продолжение таблицы 4

PHP (Laravel/ Symfony)	Огромное сообщество, низкий порог входа. Множество хостинг-провайдеров. Современные фреймворки (Laravel, Symfony) значительно улучшили качество PHP-разработки.	Исторически сложившаяся репутация проблем с качеством кода (хотя современные фреймворки это исправляют). Производительность может уступать Java/Node.js в некоторых сценариях. Менее сильная типизация по сравнению с Java.	Java/Spring предлагают более строгую архитектуру и инструменты для построения сложных, надежных и безопасных систем, что критично для СЭД. Управление зависимостями и сборка в Java (Maven/Gradle) часто считаются более мощными для больших проектов.
.NET (ASP.NET Core)	Высокая производительность. Мощная платформа от Microsoft, хорошая интеграция с продуктами Microsoft. C# – сильный, типизированный язык. Кроссплатформенность.	Экосистема теснее связана с Microsoft (хотя и стала более открытой). Может быть дороже в плане лицензий (хотя .NET Core бесплатен).	Java/Spring – это полностью открытое решение с огромным выбором библиотек и инструментов от различных вендоров. Если нет сильной привязки к экосистеме Microsoft, Java/Spring предлагает большую гибкость и часто более широкое сообщество вне Windows-мира. .NET и Java здесь очень сильные конкуренты.

В дополнение к языку программирования Java, был выбран Spring Boot. Spring Boot - это расширение фреймворка Spring, которое упрощает процесс разработки и настройки приложений на базе Spring. Он предоставляет разработчикам возможность создавать автономные, производительные приложения с минимальными усилиями, благодаря чему существенно сокращается время, необходимое для разработки и развертывания. Одной из основных особенностей Spring Boot является его способность автоматически настраивать приложения, основываясь на конфигурации и зависимостях, что позволяет разработчикам сосредоточиться на бизнес-логике, а не на рутинных задачах настройки.

В качестве базы данных для разрабатываемого приложения была выбрана MySQL, так как работа именно с этой СУБД будет наиболее удобной, учитывая выбор языка программирования и фреймворка. MySQL будет хорошим выбором, потому как имеет следующие характеристики:

- нативная совместимость с экосистемой Spring, так как Spring Data JPA и Hibernate имеют встроенную поддержку MySQL;
- совместимость с Spring Batch для обработки больших данных;
- интеграция с Spring Security упростит разработку функции аутентификации;

MySQL в совокупности со Spring Boot являются хорошим балансом скорости, надежности и простоты интеграции. Такое сочетание наиболее предпочтительно для разработки СЭД, где надежность, безопасность, масштабируемость и долгосрочная поддержка являются ключевыми факторами. А Java в сочетании со Spring Boot представляет собой исключительно сбалансированное и мощное решение.

2.3. Реализация функциональных требований

Функциональные требования представляют собой описание того, что именно должно делать приложение, какие функции оно должно выполнять и как оно должно взаимодействовать с пользователем и другими системами. В случае с электронным документооборотом, функциональные требования могут включать в себя такие аспекты, как создание, редактирование, хранение, поиск и обмен документами, а также управление доступом к ним. Каждое из этих требований должно быть четко сформулировано и задокументировано, чтобы в дальнейшем можно было перейти к их реализации.

Функциональные требования к системе: Создание, удаление, чтение документов. Данная функция реализована в классе Doc.java (рисунок 14).


```

import java.io.IOException;

@Controller
@RequiredArgsConstructor
public class DocsController {
    private final DocService docService;

    @GetMapping("/")
    public String docs(@RequestParam(name = "title", required = false) String title, Model model) {
        model.addAttribute("docs", docService.listDocs(title));
        return "docs";
    }

    @GetMapping("/{doc/{id}}")
    public String docInfo(@PathVariable Long id, Model model) {
        Doc doc = docService.getDocById(id);
        model.addAttribute("doc", doc);
        model.addAttribute("images", doc.getImages());
        return "doc-info";
    }

    @PostMapping("/doc/create")
    public String createDoc(@RequestParam("file1") MultipartFile file1, Doc doc) throws IOException {
        docService.saveDocs(doc, file1);
        return "redirect:/";
    }

    @PostMapping("/doc/delete/{id}")
    public String deleteDoc(@PathVariable Long id) {
        docService.deleteDoc(id);
        return "redirect:/";
    }
}

```

Рисунок 14 - присвоение классу Дос функций для работы с ними

Разделение ролей и управление пользователями. Принцип разделения ролей реализован с помощью создания нового класса Roles, в котором был описан код, указывающий на возможности конкретных ролей.

Функция регистрации и авторизации в системе реализована при помощи следующего кода в классе SecurityConfig.java при помощи расширения WebSecurityConfigurerAdapter (рисунок 15). И вспомогательного кода в классе User.java (рисунок 16).

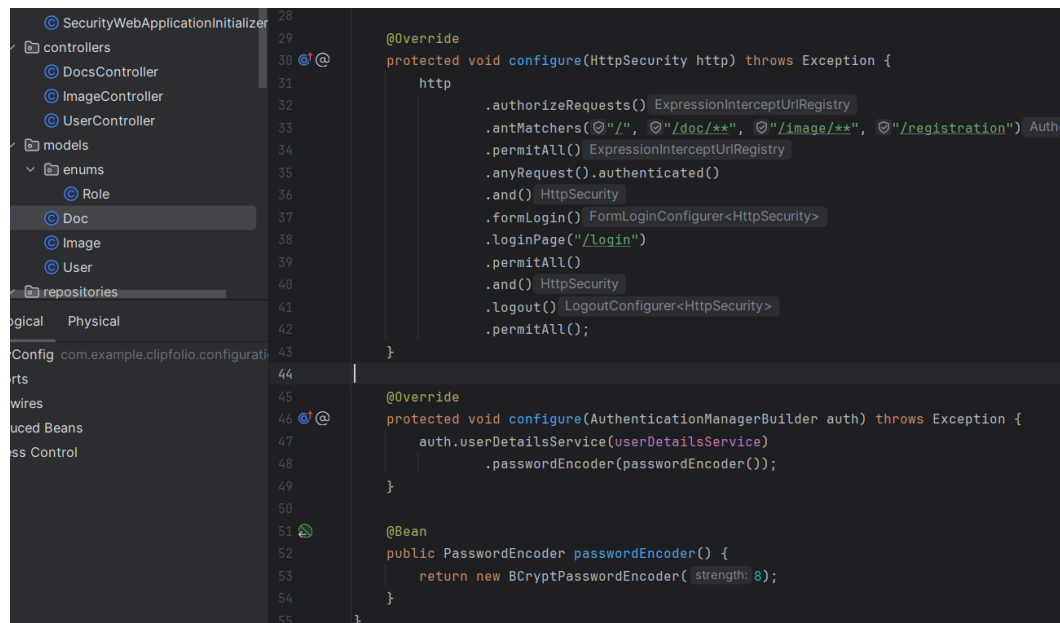


Рисунок 15 – Код для регистрации и авторизация в системе



Рисунок 16 - вспомогательные функции для обеспечения безопасности

Поиск и фильтрация документов была реализована следующим образом при помощи метода Get (рисунок 17).

```
<h4>Посмотри документы здесь</h4>
<form action="/" method="get">
    Поиск по названию: <input type="text" name="title"><br>
    <input type="submit" value="Поиск"/>
</form>
```

Рисунок 17 - добавление функции поиска на главную страницу

При разработке программного кода приложения были реализованы все функциональные требования, указанные в пункте 1.4 «Требования к функционалу и интерфейсу приложения».

В заключение, реализация функциональных требований является одним из самых ответственных и важных этапов разработки web-приложения на базе Java EE для оптимизации процессов электронного документооборота. Этот процесс требует тщательной проработки всех аспектов, начиная от анализа требований и проектирования архитектуры и заканчивая тестированием и документированием. Только при условии качественной реализации функциональных требований можно создать эффективное, безопасное и удобное в использовании приложение, которое будет успешно решать задачи электронного документооборота и удовлетворять потребности пользователей.

Глава 3. Оценка эффективности разработанного программного обеспечения

3.1. Тестирование и отладка приложения

Оценка эффективности разработанного программного обеспечения является важным этапом в процессе его создания и внедрения. В частности, для web-приложения на базе Java EE, предназначенного для оптимизации процессов электронного документооборота, тестирование и отладка приложения занимают центральное место. Эти процессы позволяют выявить и устранить ошибки, улучшить функциональность и повысить общую производительность системы. Тестирование не только обеспечивает соответствие приложения заданным требованиям, но и помогает определить его устойчивость, безопасность и удобство в использовании. Только после того, как все тесты пройдены, приложение можно считать готовым к внедрению.

Тестирование приложения можно разделить на несколько этапов, каждый из которых имеет свои цели и методы. Первоначально, перед началом тестирования, важно определить тестовые сценарии, которые будут охватывать все ключевые функции приложения.

На этапе функционального тестирования (Functional Testing) проверяются все основные функции приложения. Был разработан следующий сценарий тестирования на функциональность:

- а) Работа с документами:
 - 1) Загрузка документа (PDF, DOCX).
 - 2) Поиск по названию/дате/типу.
 - 3) Открытие детальной страницы документа.
- б) Пользовательские операции:
 - 1) Регистрация, авторизация, смена пароля.
 - 2) Редактирование профиля (аватар, контакты).

Инструмент: Selenium (для UI-тестов).

Критерии успеха: 100% покрытие основных сценариев и отсутствие критических багов (блокирующих работу).

Кроме функционального тестирования, важно провести нагрузочное тестирование, чтобы определить, как приложение справляется с высоким объемом запросов. Это особенно актуально для web-приложений, которые могут испытывать пики нагрузки.

Безопасность приложения также является критически важным аспектом, который не следует упускать из виду. Тестирование безопасности (Security Testing) включает в себя проверку на уязвимости, такие как SQL-инъекции, межсайтовые скрипты (XSS) и другие распространенные угрозы. Для этого могут быть использованы специальные инструменты, которые помогают выявить потенциальные уязвимости и предложить пути их устранения [20]. Был разработан следующий сценарий тестирования на безопасность:

- а) Аутентификация:
 - 1) Попытка входа с неверными данными.
 - 2) Bruteforce-атака (защита от перебора паролей).
- б) Данные:
 - 1) SQL-инъекции в поиске.
 - 2) XSS в полях ввода (например, описание документа).
- в) Доступ:
 - 1) Проверка прав доступа (может ли простой User удалить другого пользователя?).

Инструмент: OWASP ZAP.

Критерии успеха: Отсутствие уязвимостей уровня High/Critical (по OWASP Top 10).

Результат тестирования является положительным, так как при последнем анализе не было оповещений об ошибках, что представлено ниже на рисунке 18.

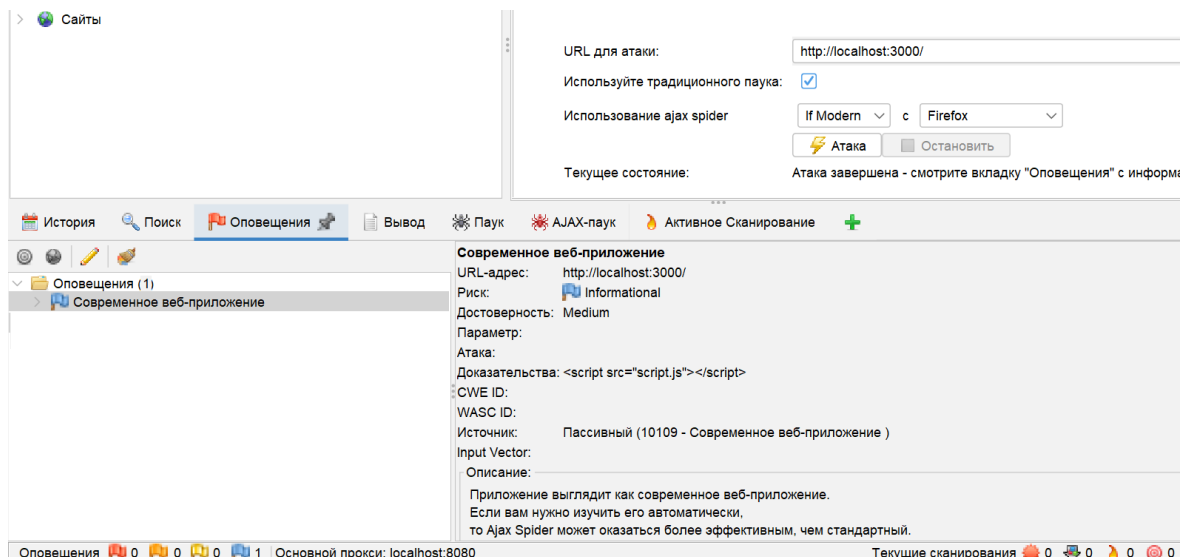


Рисунок 18 - тестирование OWASP ZAP

Важно отметить, что тестирование и отладка — это не одноразовые процессы. Они должны быть частью жизненного цикла разработки программного обеспечения. После развертывания приложения необходимо продолжать мониторинг его работы, чтобы выявлять и устранять новые проблемы по мере их появления.

3.2. Оценка удобства и функциональности приложения

Оценка удобства и функциональности приложения является важным аспектом в процессе разработки программного обеспечения, особенно когда речь идет о web-приложениях, предназначенных для оптимизации процессов электронного документооборота. Удобство и функциональность приложения напрямую влияют на его восприятие конечными пользователями, что, в свою очередь, определяет успех всего проекта. В рамках данной работы мы проведем детальный анализ удобства и функциональности разработанного web-приложения на базе Java EE, акцентируя внимание на различных аспектах, таких как пользовательский интерфейс, взаимодействие с пользователем, производительность, а также на уровне удовлетворенности пользователей. Алгоритм тестирования представлен на рисунке 10. Важно отметить, что при тестировании учитывались не только функциональные, но и нефункциональные требования, такие как надежность, масштабируемость и защищенность системы [25].

Оценка будет проводится по следующим критериям от 1 до 5. Результат представлен в таблице 5:

Таблица 5 – Оценка удобства и функциональности приложения

Критерий	Описание	Оценка
Удобство использования (Usability)	Насколько легко пользователям взаимодействовать с приложением.	5
Функциональность (Functionality)	Соответствие приложения заявленным требованиям и функциональным возможностям	4
Эстетика интерфейса	Визуальная привлекательность и общий дизайн приложения	4
Производительность	Скорость работы приложения и его реакция на действия пользователя	5
Доступность	Уровень доступности приложения для различных групп пользователей, включая людей с ограниченными возможностями	3

Общая удовлетворенность работы приложения составляет 84%.

Не менее важным аспектом является уровень удовлетворенности пользователей. Для этого предварительно разработаем сценарии, которые пользователи должны выполнить в приложении, и после дать обратную связь.

Сценарий 1: «Регистрация нового пользователя и первый вход»

Цель: Проверить возможность нового пользователя зарегистрироваться в системе, подтвердить (если требуется) регистрацию и успешно войти в систему с созданными учетными данными.

Основные шаги (User Flow):

- Перейти на страницу регистрации.
- Заполнить все обязательные поля формы регистрации корректными данными (ФИО, email, логин, пароль, подтверждение пароля).
- Принять условия использования (если есть).
- Нажать кнопку «Зарегистрироваться».
- (Если требуется) Открыть почтовый ящик, найти письмо с подтверждением, перейти по ссылке для подтверждения.
- Перейти на страницу авторизации.
- Ввести логин и пароль, указанные при регистрации.
- Нажать кнопку «Войти».

Ожидаемый результат:

- После шага 4 «Нажать кнопку «Зарегистрироваться»: Система сообщает об успешной отправке данных/необходимости подтверждения.
- После шага 5 «Перейти по ссылке для подтверждения»: Система сообщает об успешном подтверждении учетной записи.
- После шага 8 «Нажать кнопку «Войти»: Пользователь успешно авторизован и перенаправлен на главную страницу/личный кабинет. В системе создана новая учетная запись.

Сценарий 2: «Поиск и фильтрация информации»

Цель: Проверить корректность работы функций поиска и применения фильтров для нахождения релевантной информации.

Основные шаги (User Flow):

- Перейти в раздел, где доступен поиск/фильтрация (например, «Список документов»).

- Ввести поисковый запрос (ключевое слово, существующее в данных) в поле поиска.

- Нажать «Найти» или дождаться автоматического обновления результатов.

- Применить один или несколько фильтров (например, по дате, по типу, по автору).

- Сбросить фильтры/поисковый запрос.

Ожидаемый результат/Критерии успеха:

- После шага 3 «Нажать «Найти»»: Отображаются только те элементы, которые соответствуют поисковому запросу.

- После шага 4 «Применить один или несколько фильтров»: Отображаются только те элементы, которые соответствуют и поисковому запросу (если он был), и примененным фильтрам.

- Если по запросу/фильтрам ничего не найдено, отображается соответствующее сообщение.

- После шага 5 «Сбросить фильтры/запрос»: Отображается полный список элементов (или согласно настройкам по умолчанию).

Инструмент: Google Forms для сбора фидбека.

Критерии успеха: 80% пользователей выполняют задачи без помощи, и средняя оценка удобства $\geq 7/10$.

Результаты опроса показали, что большинство пользователей оценили удобство интерфейса на 100% (рисунок 19). Пользователи отметили, что интерфейс интуитивно понятен, а все необходимые функции доступны в несколько кликов.

Удобно ли расположены документы на Главной странице?	Вся ли необходимая информация отображается на странице документа (дата, тип, стоимость, пользователь, описание)?	Как вы оцениваете переход к профилю пользователя через клик на его имя? Не было ли случайных нажатий?	Насколько понятна форма регистрации в системе (от 1 до 5)?
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5
да, информация быстро считывается	отображается вся информация	удобный переход	5

Рисунок 19 –результаты опроса

В заключение, оценка удобства и функциональности разработанного web-приложения на базе Java EE показала, что мы достигли значительных успехов в создании системы, которая отвечает потребностям пользователей и оптимизирует процессы электронного документооборота. Удобный интерфейс, широкий функционал и высокая производительность делают наше приложение эффективным инструментом для работы с документами.

3.3. Анализ результатов использования программного обеспечения

Оценка эффективности разработанного программного обеспечения является важным этапом в процессе его внедрения и эксплуатации, поскольку она позволяет определить, насколько успешно приложение решает поставленные задачи и удовлетворяет потребности пользователей. В данном разделе будет проведен анализ результатов использования web-приложения,

разработанного на базе Java EE для оптимизации процессов электронного документооборота. Основное внимание будет уделено ключевым аспектам, таким как производительность приложения, его влияние на бизнес-процессы, степень удовлетворенности пользователей, а также выявление возможных проблем и путей их решения.

Для этого были проведены опросы пользователей, которые активно взаимодействовали с приложением в течение рабочего дня. Результаты опросов показали, что внедренное приложение успешно справляется со своими функциями. Они отметили, что система интуитивно понятна и не требует длительного обучения, что особенно важно для сотрудников, не обладающих техническими знаниями. Результаты представлены на рисунке 20.



Рисунок 20 – Положительные результаты опрошенных сотрудников

Тем не менее, несмотря на положительные отзывы, были выявлены и некоторые проблемы, которые требуют внимания. Одной из них является необходимость в дополнительной настройке под конкретные бизнес-процессы организации. Хотя приложение было разработано с учетом общих требований к электронному документообороту, некоторые пользователи выразили желание иметь возможность настраивать определенные функции и параметры под свои нужды. Это может быть связано с особенностями работы конкретных

подразделений, которые могут отличаться от общепринятых стандартов. В связи с этим, разработчикам следует рассмотреть возможность внедрения более гибкой системы настройки, которая позволит пользователям адаптировать приложение под свои специфические требования.

Вторая проблема, которую удалось выявить на данном этапе разработке приложения, это необходимость предварительного просмотра загруженных документов, без необходимости скачивания. По статистике, представленной на рисунке 21, можно сделать вывод, что большая часть пользователей выразила желание пользоваться такой функцией. Поэтому на следующих этапах разработки мы займемся ее реализацией.

Хотели бы вы видеть предпросмотр файла без скачивания?

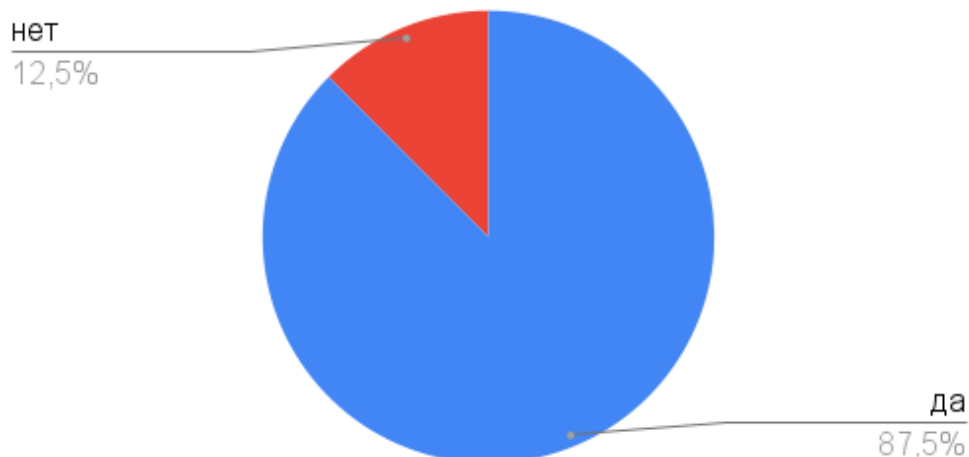


Рисунок 21 – Статистика по результатам опрос на необходимость добавления функции предварительного просмотра документа

Высокая производительность, положительное влияние на бизнес-процессы и высокая степень удовлетворенности пользователей подтверждают успешность реализации проекта. Тем не менее, как и любое программное обеспечение, приложение требует постоянного совершенствования и адаптации к меняющимся условиям и требованиям. Важно продолжать собирать обратную связь от пользователей, проводить регулярные обновления, направленные на повышение надежности, удобства интерфейса и

безопасности системы, чтобы обеспечить максимальную эффективность и удовлетворение потребностей бизнеса в будущем [22], [26]. Таким образом, анализ результатов использования программного обеспечения позволяет не только оценить его текущую эффективность, но и наметить пути для дальнейшего развития и совершенствования, что в конечном итоге приведет к улучшению бизнес-процессов и повышению конкурентоспособности организации. Также стоит учитывать возможные технологические тренды и инновации, которые могут быть интегрированы в приложение для расширения его функциональности. Кроме того, важно развивать систему мониторинга производительности и аналитики, позволяющую выявлять узкие места и оперативно реагировать на возникающие проблемы. Внедрение новых функций должно основываться на приоритетах бизнеса и потребностях конечных пользователей. Таким образом, устойчивое развитие программного продукта требует комплексного и системного подхода, ориентированного на долгосрочную перспективу.

Заключение

В процессе работы мы рассмотрели несколько ключевых аспектов, которые играют важную роль в создании эффективного программного обеспечения для информационной системы предприятия. Эти аспекты включают в себя исследование предметной области и бизнес-процессов ООО «Ворлдинтертех Рус», процесс разработки и тестирования программного обеспечения, а также оценку эффективности созданного решения.

По итогам проведенной работы были осуществлены следующие задачи:

- проанализирован существующий документооборот в организации, выявлены его недостатки и определены основные цели, которые должны быть достигнуты с помощью нового программного обеспечения, проведён анализ рынка ЭДО, определены функциональные (регистрация, разделение ролей (RBAC) [21], создание, удаление, чтение документов, поиск и фильтрация) и нефункциональные (безопасность, масштабируемость системы, user-friendly интерфейс, адаптивный под разные устройства) требования к системе;
- в нотации UML разработана архитектура ПО (в виде диаграммы компонентов) и спроектирована структура базы данных (в виде диаграммы Entity-Relationship), подобран стек Java и Spring Boot для backend, Thymeleaf (шаблонизатор Java) для frontend, СУБД MySQL;
- реализованы ключевые функции системы, включая модуль авторизации и аутентификации с соблюдением мер защиты данных;
- система внедрена в рабочую среду, проведено эмпирическое исследование удобства интерфейса и эффективности работы (методом анкетирования сотрудников организации, непосредственно использующих систему в своей профессиональной деятельности), а также проведены тестирование безопасности (с использованием Wireshark) и функциональное тестирование (с использованием Selenium).

Таким образом, реализация СЭД в ООО «Ворлдинтертех Рус» отвечает актуальным требованиям бизнеса, нормативным стандартам [3] и рекомендациям в области цифровизации документооборота [22].

Мы понимаем, что внедрение нового программного обеспечения – это не просто технический процесс, но и изменение культуры работы в организации. У разработанного в ходе работы приложения мы находим следующие пути развития:

- Работа с документами: ООО «Ворлдинтертех Рус» пока молодая компания, которая еще в процессе роста. Поэтому в дальнейшем будет полезно улучшить работу с документами хранением истории изменений каждого отдельного документа. Или реализовать функцию, позволяющую вернуть документ к предыдущей версии.

- Безопасность и контроль: при росте штата сотрудников, компании нужно будет предвидеть различные критические ситуации и нападения извне. Двухфакторная аутентификация нужна для повышения безопасности – она защищает аккаунты даже при утечке пароля. Такая функция будет очень полезна.

Мы уверены, что дальнейшее развитие системы позволит сделать ее еще более эффективной и удобной. Разработанное приложение имеет большие перспективы развития, и может стать не просто удобным инструментом, а целой корпоративной системой.

После завершения разработки мы провели оценку эффективности созданного программного обеспечения. Этот этап является критически важным, так как он позволяет определить, насколько успешно приложение решает поставленные задачи и удовлетворяет потребности пользователей. Мы использовали различные методы оценки, включая анализ производительности системы, пользовательские тестирования и сбор обратной связи от конечных пользователей. Результаты показали, что разработанное приложение значительно ускоряет процессы документооборота, снижает количество ошибок и упрощает взаимодействие между сотрудниками.

В завершение, хочется отметить, что разработка приложения для оптимизации процессов электронного документооборота стала успешным проектом, который не только улучшил внутренние процессы организации, но и повысил уровень удовлетворенности пользователей. Мы надеемся, что полученные результаты будут полезны как для нашей организации, так и для других предприятий, стремящихся оптимизировать свои процессы. документооборота. Разработка и внедрение подобных решений – это важный шаг на пути к цифровой трансформации бизнеса, который позволяет не только повысить эффективность работы, но и создать конкурентные преимущества на рынке. Разработка и внедрение подобных решений – это важный шаг на пути к цифровой трансформации бизнеса, который позволяет не только повысить эффективность работы, но и создать конкурентные преимущества на рынке. Кроме того, успешная реализация проекта служит отличным примером рационального использования современных технологий.

Список используемой литературы

1. Бондарев, А. В. 1С:Документооборот. Версия 3.0. 300 вопросов и ответов : практическое пособие / А. В. Бондарев, Н. Ю. Бурцева. – Москва : 1С-Паблишинг, 2022. – 475 с. – ISBN 978-5-9677-1234-5. – Текст : непосредственный.
2. Бенедикт, Т. Свод знаний по управлению бизнес-процессами. BPMN СВОК 4.0 / Т. Бенедикт, М. Кирхмер, М. Скарсиг ; перевод с английского. – Москва : Альпина Паблишер, 2021. – 504 с. – ISBN 978-5-9614-5678-9. – Текст : непосредственный.
3. Горбачева, Т. Ф. ПМ 05 Проектирование и разработка информационных систем : учебное пособие / Т. Ф. Горбачева. – URL: https://eos.belovokyzgty.ru/pluginfile.php/40958/mod_resource/content/0/37%20ПМ.05%20Проектирование%20и%20разработка%20информационных%20систем.pdf (дата обращения: 22.05.2025). – Текст : электронный.
4. Колисниченко, Д. Н. Программирование для Android / Д. Н. Колисниченко. – Москва : БХВ, 2021. – 288 с. – ISBN 978-5-9775-6632-4. – Текст : непосредственный.
5. Мажуга, П. С. Совершенствование бизнес-процессов мониторинга результативности спортивных организаций : диссертация ... кандидата экономических наук : 08.00.05 / П. С. Мажуга ; Сибирский федеральный университет. – Красноярск, 2018. – 202 с. – URL: https://elib.sfu-kras.ru/bitstream/handle/2311/74254/dissertaciya_mazhuga.pdf?sequence=1 (дата обращения: 22.05.2025). – Текст : электронный.
6. Мартин, Р. Чистый код: создание, анализ и рефакторинг / Р. Мартин. – Москва : Питер, 2022. – 464 с. – ISBN 978-5-4461-2039-5. – Текст : непосредственный.
7. Матвеев, А. А. Проектирование и разработка системы уведомлений студентов о задолженностях : магистерская диссертация / А. А. Матвеев ; Уральский федеральный университет имени первого Президента

России Б. Н. Ельцина. – Екатеринбург, 2024. – 88 с. – URL: <https://elar.urfu.ru/handle/10995/140300> (дата обращения: 22.05.2025). – Текст : электронный.

8. Нечеухин, А. В. Разработка онлайн автоматизированной системы для электронного документооборота : магистерская диссертация / А. В. Нечеухин ; Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. – Екатеринбург, 2014. – 100 с. – URL: <https://elar.urfu.ru/handle/10995/28186> (дата обращения: 22.05.2025). – Текст : электронный.

9. Очиенг, П. Автоматизированная система конфигурирования ПК для компьютерного магазина через web-интерфейс : выпускная квалификационная работа бакалавра / П. Очиенг ; Алтайский государственный университет. – Барнаул, 2017. – 60 с. – URL: http://dspace.bsuedu.ru/bitstream/123456789/32532/1/Ochieng_Avtomatizirovannaya_17.pdf (дата обращения: 22.05.2025). – Текст : электронный.

10. Резник, В. Г. Учебное пособие / В. Г. Резник. – Томск : Томский государственный университет систем управления и радиоэлектроники, 2021. – 120 с. – URL: <https://asu.tusur.ru/learning/090301/d69/090301-d69-lect.pdf> (дата обращения: 22.05.2025). – Текст : электронный.

11. Родионов, М. Г. Рабочая программа дисциплины (модуля) «Средства и системы проектирования ПО» / М. Г. Родионов ; Сибирский институт бизнеса и информационных технологий. – Омск, 2020. – 25 с. – URL: <https://sibit.sano.ru/sveden/files/004388.pdf> (дата обращения: 22.05.2025). – Текст : электронный.

12. Слюта, Н. С. Автоматизация функционирования гостиничного комплекса : выпускная квалификационная работа бакалавра / Н. С. Слюта ; Алтайский государственный университет. – Барнаул, 2018. – 75 с. – URL: http://dspace.bsuedu.ru/bitstream/123456789/37064/1/Saluta_Avtomatizatsiya_18.pdf (дата обращения: 22.05.2025). – Текст : электронный.

13. Филлипс, Б. Android. Программирование для профессионалов / Б. Филлипс. – Москва : Диалектика, 2007. – С. 80–86. – ISBN 978-5-8459-1147-3. – Текст : непосредственный.
14. Фрайман, З. Создание приложений для смартфонов и планшетов под ОС Android. Практический курс / З. Фрайман. – Москва : Едиториал УРСС, 2020. – 504 с. – ISBN 978-5-354-02345-7. – Текст : непосредственный.
15. Фримен, Э. Head First. Паттерны проектирования / Э. Фримен, Э. Робсон. – Москва : Питер, 2021. – 640 с. – ISBN 978-5-4461-1453-0. – Текст : непосредственный.
16. Скин, Д. Kotlin. Программирование для профессионалов / Д. Скин, Д. Гринхол. – Москва : Питер, 2023. – 560 с. – ISBN 978-5-4461-2545-1. – Текст : непосредственный.
17. Сляров, Д. В. Защита персональных данных в интернете: практическое руководство / Д. В. Сляров. – Москва : ДМК Пресс, 2021. – 320 с. – ISBN 978-5-97060-845-2. – Текст : непосредственный.
18. Корнеев, В. П. Кибербезопасность и защита данных: учебник для вузов / В. П. Корнеев, А. А. Михайлов. – Санкт-Петербург : Питер, 2022. – 416 с. – ISBN 978-5-4461-2345-7. – Текст : непосредственный.
19. Stallings, W. Cryptography and Network Security: Principles and Practice / W. Stallings. – 8th ed. – Boston : Pearson, 2023. – 768 p. – ISBN 978-0-13-576-424-2. – Text : direct.
20. European Union Agency for Cybersecurity (ENISA). Guidelines for Securing Personal Data Processing. – 2022. – URL: <https://www.enisa.europa.eu/publications/guidelines-for-securing-personal-data-processing> (дата обращения: 22.05.2025). – Text : electronic.
21. NIST Special Publication 800-63B. Digital Identity Guidelines: Authentication and Lifecycle Management. – 2023. – URL: <https://csrc.nist.gov/pubs/sp/800/63/b/upd3/final> (дата обращения: 22.05.2025). – Text : electronic.

22. Жуков, А. С. Безопасность Java-приложений: от уязвимостей к защите / А. С. Жуков. – Москва : БХВ-Петербург, 2023. – 352 с. – ISBN 978-5-9775-7021-5. – Текст : непосредственный.

23. Haider, A. Opportunities and Challenges in Implementing Electronic Document Management Systems / A. Haider, B. Aryati, B. Mahadi // Asian Journal of Applied Sciences. – 2021. – Vol. 3. – P. 36–39. – URL: <https://www.ajouronline.com/index.php/AJAS/article/view/2239> (дата обращения: 22.05.2025). – Text : electronic.

24. Ziemba, E. Critical Success Factors for Adopting Electronic Document Management Systems in Government Units / E. Ziemba, T. Papaj, D. Descours // Proceedings of the 17th Conference on Computer Science and Intelligence Systems (FedCSIS 2022), 4–7 September 2022, Sofia, Bulgaria. – 2022. – P. 809–813. – Text : electronic.

25. Beheshti, H. M. Selection and critical success factors in successful ERP implementation / H. M. Beheshti, B. K. Blaylock, D. A. Henderson, J. G. Lollar // Competitiveness Review. – 2014. – Vol. 24, № 4. – P. 357–375. – DOI: <https://doi.org/10.1108/CR-10-2013-0082>. – Text : electronic.

26. Dermody, B. Five 2021 Emerging Trends in Records Management and ECM / B. Dermody. – 2021. – URL: <https://www.gflesch.com/blog/document-management-technology> (дата обращения: 22.05.2025). – Text : electronic.