ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

"ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ"

Факультет ИСП

Кафедра ПИ им Л.П. Фельдмана

ОТЧЕТ

по преддипломной практике

по теме: «Разработка мессенджера»

Руководители: Выполнил:

Студент гр. ПИ-20б Щедрин С.В. Безуглый В.В.

Донецк 2024

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc7551)

[1.1 Общая характеристика задачи 5](#_Toc27163)

[1.2 Проблемы автоматизации 8](#_Toc2767)

[1.3 Анализ существующих систем 8](#_Toc14281)

[1.4 Анализ существующих программных средств реализации 15](#_Toc23436)

[1.5 Постановка задача 18](#_Toc10336)

[2 АНАЛИЗ ТРЕБОВАНИЙ 19](#_Toc12536)

[3 АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 21](#_Toc17319)

[ЗАКЛЮЧЕНИЕ 27](#_Toc22819)

[СПИСОК ЛИТЕРАТУРЫ 28](#_Toc30536)

# ВВЕДЕНИЕ

Ни одна деятельность человека или компании не обходится без общения.

Каждой организации необходима платформа для безопасного общения между сотрудниками. Технологии позволяют оптимизировать, упростить и улучшить этот сложный процесс. Внедрение собственной платформы позволяет свести к минимуму возможность перехвата важных данных.

Преимущества уникального чат-мессенджера: экономия времени, более эффективное использование ресурсов, повышение прозрачности общения в компании, сохранение истории каждого взаимодействия, гибкость.

Концепция чат-мессенджера подразумевает особый способ общения между людьми по всему миру. Данная тема очень актуальна, так как чат решает многие проблемы любого учреждения.

При выполнении данной работы необходимо не только изучить принципы работы существующих систем, но и реализовать "каркас" чата и модули для работы с базой данных. При этом необходимо учитывать универсальность системы, так как данный продукт может использоваться как в некоммерческих целях, так и на предприятиях.1 СИСТЕМНЫЙ АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## Общая характеристика задачи

Чат-мессенджер – это многопользовательская система для формирования, передачи и хранения сообщений[1].

Задачи чат-мессенджера:

1. **Функциональность чата. О**сновная цель сайта - обеспечить пользователей средствами обмена сообщениями в режиме реального времени. Это включает в себя возможность отправки текстовых сообщений, использование смайликов, возможность отправки файлов (изображений, аудио, видео и документов), а также функции, такие как групповые чаты и приватные сообщения.
2. **Регистрация и аутентификация.** Пользователи должны иметь возможность создания учетной записи и входа в систему. Для безопасности и конфиденциальности информации важно реализовать механизм аутентификации с использованием паролей или других методов (например, двухфакторной аутентификации).
3. **Безопасность.** С учетом важности конфиденциальности переписки, необходимо обеспечить безопасность передаваемых данных, в том числе с использованием шифрования сообщений.
4. **Интерфейс пользователя.** Веб-интерфейс должен быть интуитивно понятным и удобным для использования. Это включает в себя разработку удобного дизайна, легкость навигации и поддержку мобильных устройств.
5. **Хранение сообщений.** Важно иметь эффективную систему хранения сообщений для обеспечения доступности переписки пользователей и возможности просмотра истории сообщений.
6. **Масштабируемость.** Сайт должен быть способен обрабатывать большое количество пользователей и сообщений, поэтому важно учитывать масштабируемость при разработке архитектуры системы.
7. **Мобильная совместимость.** Учитывая популярность мобильных устройств, важно обеспечить полноценную работу чат-мессенджера на мобильных платформах, возможно, с помощью мобильных приложений или адаптивного веб-дизайна.

Основная цель коммуникации – облегченение общения пользователей в сети.

Основные этапы работы с сообщениями:

1) обработка сообщения;

2) сохранение сообщения в БД;

3) отправка сообщения получателю.

Рассмотрим на примере диаграммы состояния жизненный цикл сообщения в системе (см. рис. 1).



Рисунок 1.1 – Состояния сообщения на различных этапах работы коммуникации

В рамках дипломного проекта предусмотрена разработка модуля коммуникации. Принцип работы следующий:

Для отправки сообщения требуется:

1. аккаунт отправителя;
2. отправитель(по умолчанию);
3. получатель(можно выбрать из графического интерфейса);
4. текст сообщения;
5. вложение (опционально).

На данном этапе проводится последовательное сохранение сообщения. Получатель может авторизироваться на сайте и получить новое сообщение.

## Проблемы автоматизации

Мессенджер решает множество проблем учреждения, таких как экономии времени на коммуникацию, сохранность данных и т.д.. Достигается всё за счёт автоматизации многих действий пользователя.

Системы коммуникации помогают выстроить простую и безопасную цепочку действий пользователя, приводящую к ожидаемому результату. Автоматизация различных процессов избавляет от недочетов, слабых мест в чате.

Мессенджер повышает производительность работы, так как позволяет:

1. коммуницировать с любого места;
2. надёжно хранить файлы на сервере;
3. фиксировать историю историю диалогов;
4. снизить число ошибок из-за несогласованности действий.

## Анализ существующих систем

1. WhatsApp (2009) на данный момент подразделение Facebook:

"WhatsApp" — это легендарный мессенджер, разработанный Ян Кумом и выпущенный в 2009 году. Эта самый обычный мессенджер в базовом его понимании. [2].

Мессенджер представляет собой мобильное, десктопное приложение для общения между пользователями в записной книжке смартфона. Дизайн приложения создан настолько простым, что с ним разберётся даже не самый продвинутый пользователь интернета.

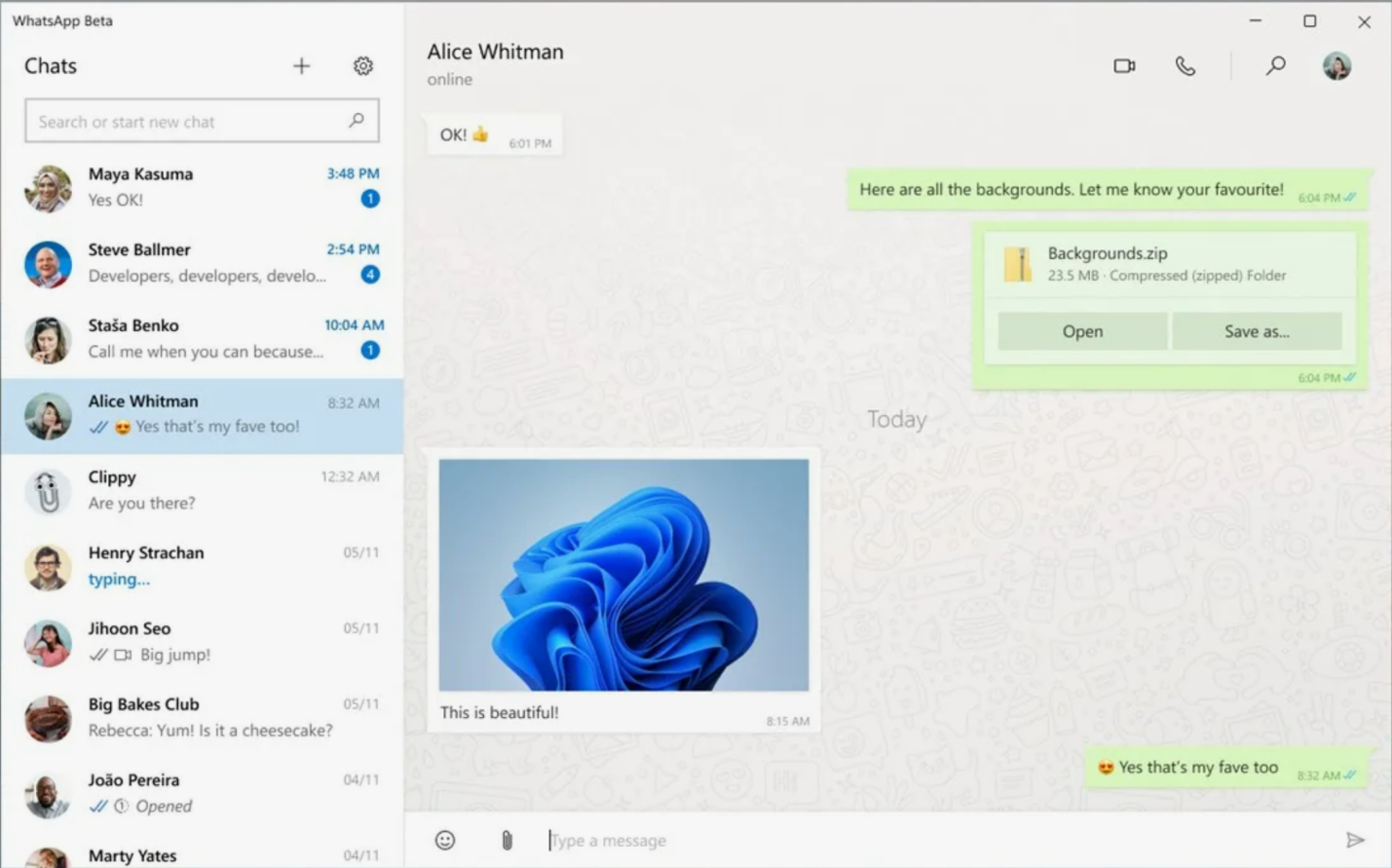


Рисунок 1.1 – Интерфейс WhatsApp

Чат предлагает относительно скудный набор инструментов. Разработчики не торопятся добавлять современные функции для своих пользователей. Так ,например, свет не увидели функции самоудаляющихся сообщений.

В свое время "WhatsApp" впечатляла тем, что был похож на приложение «Сообщения» на Android и IOS и можно было без оплаты оператору сотовой связи связываться с родственниками и друзьями в понятном интерфейсе.

Основатели компании внедрили очень простую и удобную систему регистрации в приложении, только по номеру телефона, без сбора конфиденциальных данных пользователей. Переписка является приватной, никто не сможет ее расшифровать.

Кум и Эктон говорили, что не планировали создавать компанию, а хотели лишь создать продукт, которым будет удобно пользоваться абсолютно каждому человеку. Важно отметить, что основатели WhatsApp были противниками рекламы, их девиз: «Никакой рекламы! Никаких игр! Никаких уловок!». Они считали, что маркетинг пускает пыль в глаза, не давая возможности сфокусироваться на самом продукте. Они не тратили средства на привлечение пользователей, полагаясь на органический рост.

2. WeChat(2010) работа исследовательского центра Tencent:

За каждым великим делом стоит великая личность. В случае с WeChat это программист Аллен Чжан , чье имя на родине при жизни обросло легендами. Чжан родился в 1969 году, в крестьянской семье. Его с детства тянуло к науке и он, помимо занятий в школе, постоянно учился сам, поглощая книгу за книгой. Это позволило мальчику из глухой деревни поступить в Хуачжунский университет науки и технологий (Ухань) и получить степень магистра телекоммуникационной инженерии в 1994-м году.

Изначально над проектом работала команда из семи человек во главе с Чжаном. К ноябрю 2010-го у них был готов прототип, носивший китайское название Weixin. В основном, приложение копировало Kik, мало чем отличалось от конкурентов и было запущено в январе 2011-го без особого шума и рекламы.

В начале 2012-го в мессенджере появилась одна из важнейших составляющих будущей экосистемы – QR-код. Каждый пользователь получал уникальный код и мог, просканировав код на смартфоне другого человека, добавить его в друзья, посмотреть каталог товаров, меню, расписание. Через три месяца (14 после первоначального запуска) количество подписчиков выросло до 100 млн.

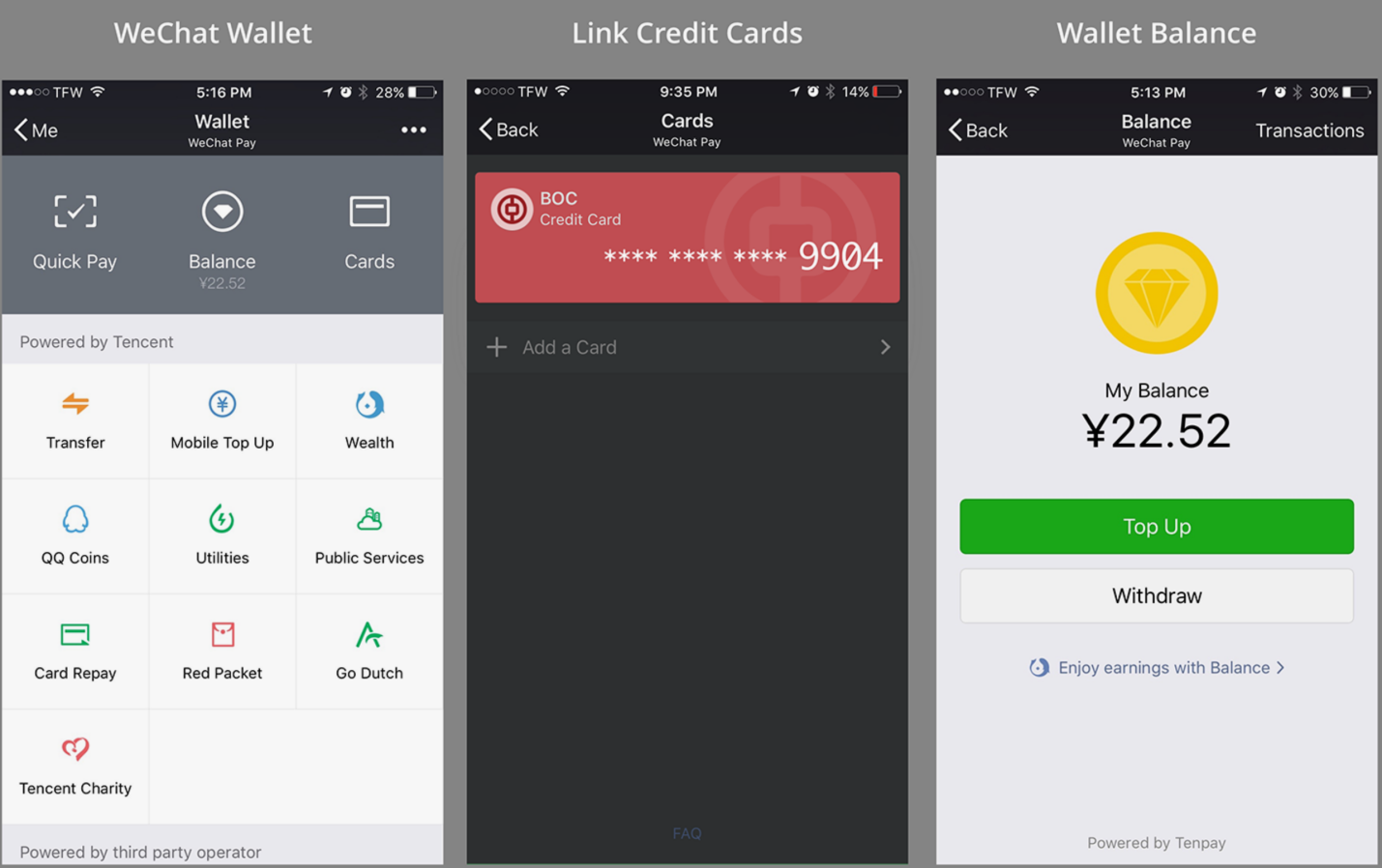


Рисунок 1.2 – Интерфейс WeChat

По данным за 2021-й год WeChat активно пользуются 1,2 миллиарда человек ежемесячно, каждый из которых тратит на него в среднем 77 минут в день, а 17% ежедневно проводит в приложении по 4 часа и больше.

3. "Telegram"(2013) Павел Дуров:

Telegram - это популярный чат-инструмент, доступный на 32 языках. За десять с лишним лет существования Telegram постоянно входит в пятерку самых скачиваемых приложений в мире, а к 2023 году число его пользователей достигнет 800 миллионов в месяц.

Со слов самого Павла Дурова, идея создания мессенджера пришла к нему еще в 2011 году. В тот момент на фоне проблем с правоохранительными органами он осознал, что у него нет безопасного способа коммуникации с близкими. Брат Павла, Николай Дуров, разработал криптографический протокол для шифрования переписки [MTProto](https://ru.wikipedia.org/wiki/MTProto" \t "https://martrending.ru/smm/_blank), который лег в основу Телеграм.

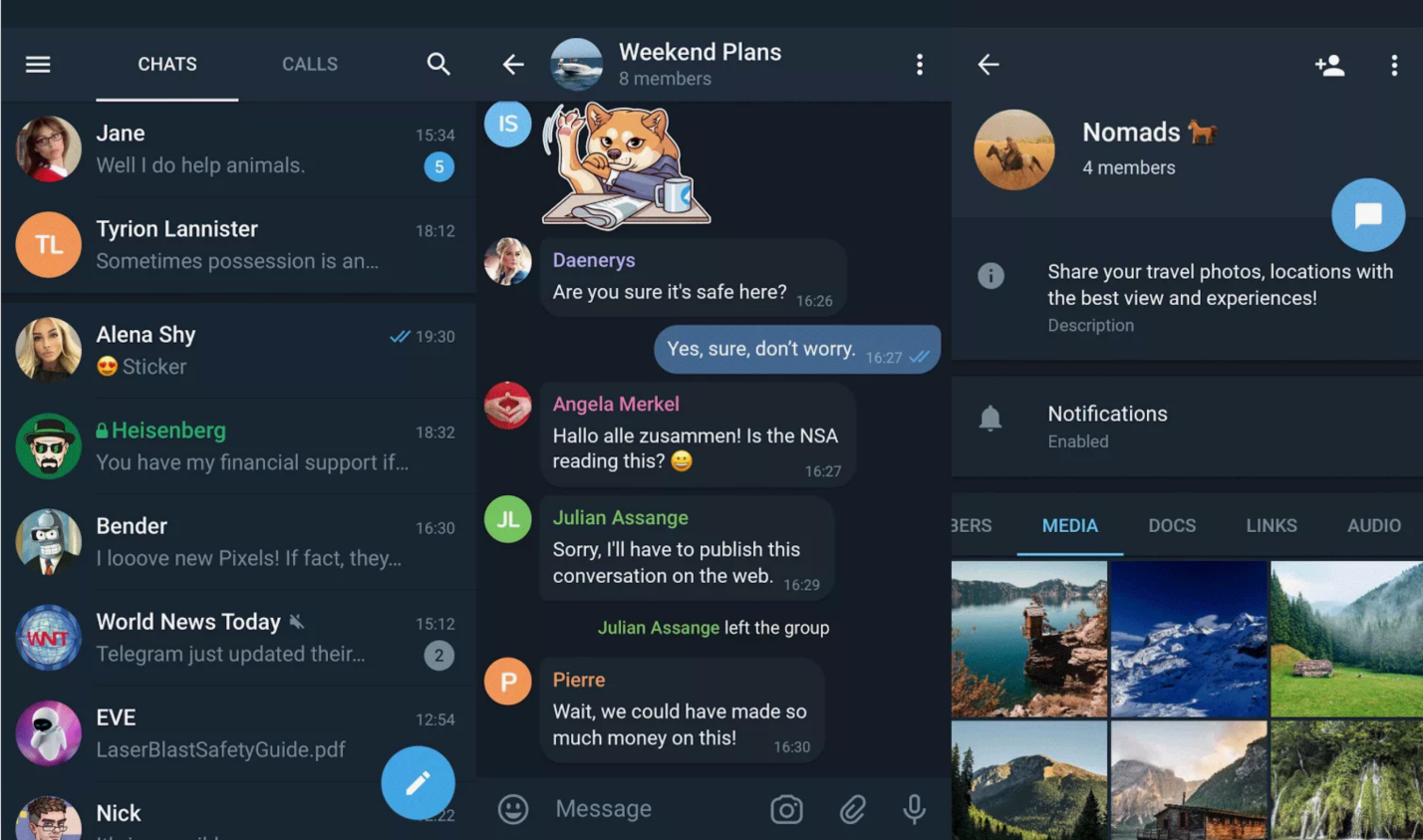


Рисунок 1.3 – Интерфейс Telegram

Одной из ключевых особенностей "Telegram" является защищенность. Разработанный протокол является безопасным способом передавать сообщений. Это и легло в основу незаконной деятельности внутри мессенджера.

16 апреля 2018 года на территории России начался процесс ограничения доступа к мессенджеру Telegram.

Предпосылкой блокировки мессенджера стали поправки в законе , имеющие антитеррористическую направленность.

Это не отменяет факта популярности мессенджера на территории СНГ. Telegram является тем, что представляет собой безопасный мессенджер.

Сравнение основных характеристик аналогов систем по основным функциональным характеристикам представлено в таблице 1.1.

Таблица 1.1 ⸺ Анализ функциональных характеристик программных систем

|  |  |  |  |
| --- | --- | --- | --- |
| Характеристики / Игры | WhatsApp | WeChat | Telegram |
| Год выпуска | 2009 | 2010 | 2013 |
| Отправка сообщений | Есть | Есть | Есть |
| Отправка голосовых сообщений | Есть | Есть | Есть |
| Поддержка платформ | Windows, MacOS, Android, iOS, S40, KaiOS, WEB | Windows, MacOS, Android, iOS, S40/J2ME, Windows Phone, Symbian, BlackBerry | Windows, MacOS, Android, iOS, WEB |
| Функция самоуничтожения сообщений | Доступна для отдельных диалогов | Нет | Есть |
| Цифровые звонки и видеозвонки | Есть | Есть | Есть |

Продолжение таблицы 1.1 ⸺ Анализ функциональных характеристик программных систем

|  |  |  |  |
| --- | --- | --- | --- |
| Защита сообщений | Присутствует. Сообщения хранятся у пользователей. | Присутствует | Присутствует |
| Платежная система | Нет | Есть. Самая популярная в Китае платежная система. | Есть. Криптовалюты GRAM,TON. |

На основе проведённого анализа мессенджеров следует отметить, что ни один из них не является идеальным, но у каждого есть свои преимущества, которые выделяют его на фоне других.

Например, WhatsApp, хоть и не хранит письма на сервере и в случае утери доступа к устройству теряются и переписки но является безопасным от взлома серверов WhatsApp. WeChat несмотря на лучшую платёжную систему, не поддерживает базовую на текущее время функцию самоудаляющихся сообщений. Telegram, хотя и имеет криптовалютные наработки но не раз попадались на обмане пользователей и привлекались к ответственности.

## Анализ существующих программных средств реализации

За время существования чатов было создано множество проектов. Как следствие, существует множество вариантов решений, связанных с используемыми инструментами.

В большинстве случаев используется клиент-серверная система - основные модули управления сообщениями и данными располагаются на специальном сервере, а клиентская часть представляет собой интерфейс для взаимодействия пользователя с системой;

Клиент-серверный подход выбран для реализации чатов. Все данные хранятся и обрабатываются на сервере, а пользователь манипулирует ими с помощью клиентского приложения. Однако для создания клиентского приложения мы используем веб-технологии.

Преимущества такого подхода - быстрая обработка данных и надежность системы. Самое главное преимущество веб-технологий - их кроссплатформенность. Доступ к сайтам можно получить практически с любого устройства.

В качестве СУБД для создания серверной части рабочего процесса была выбрана MySQL[3].

MySQL - это объектно-реляционная система управления базами данных. Ключевой особенностью объектно-реляционной базы данных является поддержка определяемых пользователем объектов и их поведения, таких как типы данных, функции, операции, домены и индексы. Это делает выбранную СУБД очень гибкой и надежной.

Клиентская часть реализована с помощью веб-технологий, таких как HTML, CSS и JavaScript. Рассмотрим, почему были выбраны именно эти технологии.

HTML - это язык гипертекстовой разметки, который используется для отображения веб-страниц. Все содержимое хранится в статических файлах.

Преимуществами HTML-приложений являются:

1. легкий вес;
2. экономичное потребление ресурсов;
3. высокая стабильность (ошибка на одной странице не влияет на весь сайт);
4. поддерживается широким спектром браузеров [4].

CSS - это язык для обозначения стиля сайта, предназначенный для улучшения процесса взаимодействия с веб-страницами. Каскадные таблицы стилей задают внешний вид страниц.

CSS также обладает рядом преимуществ:

1. простота использования;
2. экономия времени (один и тот же код можно применять к разным страницам);
3. позволяет адаптировать страницы под разные устройства;
4. глобальные веб-стандарты;
5. приложения могут хранить CSS локально в автономном кэше;
6. CSS не зависит от платформы клиента [5].

JavaScript присутствует на странице для обработки действий пользователя, и его положительными качествами являются:

1. он быстр для конечного пользователя (действия выполняются локально, а не на сервере);
2. он использует модель DOM и предоставляет множество функций для различных объектов на странице;
3. универсальность.

Предполагается, что сервер будет написан на языке программирования Golang.

Golang обладает рядом неоспоримых преимуществ:

1. высокая скорость и общая производительность;
2. простой синтаксис (сходства C++ и Python);
3. многозадачность и широкие возможности разработки;
4. язык разработан как серверный.

Поэтому выбор технологии рационален и дает возможность писать качественные, гибкие системы.

## 1.5 Постановка задача

В результате проведенного анализа существующих мессенджеров, выявленных недостатков и учтённых требований пользователей к подобным системам, принято решение о создании чата-мессенджера для демонстрации основных возможностей при разработке веб сайтов на Golang, с использованием базы данных MySQL.

Цель работы – разработка веб приложения, представляющего собой мессенджер, демонстрирующую основные механики при создании веб сайта, такие как: дизайн вёрстки, вёрстка веб-сайта, разработка БД, интеграция бэкэнд-составляющей.

Задачи работы:

1. разработка архитектуры системы (определение структуры приложения);
2. создание интерфейса пользователя (проектирование простого и интуитивно понятного пользовательского интерфейса, включая страницы входа и страницы чатов);
3. разработка программных модулей бэкэнд (реализация механики обмена сообщениями, хранения сообщений в базе данных);
4. тестирование разработанной системы (проведение тестов, включающих проверку чата на работоспособность, корректность работы интерфейса и обработку сценариев использования);
5. формулировка выводов и областей дальнейшего совершенствования системы.

# 2 АНАЛИЗ ТРЕБОВАНИЙ

Требование к программному обеспечению – это функциональная или нефункциональная потребность, которая должна быть реализована в системе.

Функциональная потребность подразумевает предоставление пользователю определённой услуги. Требование к программному обеспечению также может быть нефункциональным, это может быть требование к производительности и интерфейсу [7].

Рассмотрим бизнес-требования для программного обеспечения для системы обмена сообщениями:

1. система должна обеспечить обмен информацией между пользователями;
2. пользователь должен иметь возможность покинуть свой аккаунт;
3. система должна снизить время на коммуникацию корпоративных пользователей;
4. система должна оптимально использовать серверное время;
5. система должна оптимально;
6. система должна оптимально использовать ресурсы устройств пользователей.

Следующий этап анализа требований – формирование функциональных требований к системе:

1. информация о сообщениях при получении в списке должна содержать информацию: время отправки, вложение (если есть), текст сообщения;
2. должна быть удобная система поиска пользователей;
3. разработать систему отправки файла с базы данных пользователю ( на случай запроса пользователем вложения );
4. должно быть предусмотрено примитивное шифрование всего контента в обязательном автоматическом порядке;
5. в системе должно быть руководство пользователя;
6. необходима возможность создания неограниченного количества сообщений;
7. должна быть возможность регистрации и авторизации уже зарегистрированных пользователей;
8. необходимо обеспечить сохранность данных при потере соединения с сервером;
9. должна быть возможность сохранения активных диалогов на сервере;

Рассмотрим не фукнциональные требования к интерфейсу:

1. система должна иметь простой и понятный интерфейс;
2. форма авторизации должна содержать два текстовых поля для ввода логина и пароля, а также кнопку для входа;
3. сообщение об ошибке выводить в виде отдельной страницы;
4. на странице чата сделать меню в шапке, в котором реализовать выход из аккаунта и переключение цветовой темы.
5. реализовать добавление смайликов к сообщению.

# 3 АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Следующие вопросы оказывают существенное влияние на архитектуру разрабатываемого приложения

Параллелизм, включая вопросы организации процессов, подходы и методы.

Обработка ошибок (особые случаи) и отказоустойчивость. Проблему этой темы можно сформулировать очень просто: как предотвратить сбои или гарантировать дальнейшую функциональность системы в случае сбоя.

Взаимодействие и представление - взаимодействие между пользователем и системой, т.е. представление информации пользователю и реакция системы на действия пользователя [8].

Для создания приложения была выбрана архитектура клиент-сервер, поскольку она имеет множество преимуществ.

При создании приложения очень важно представить разрабатываемую архитектуру графически. Для реализации этого используются язык UML.

Унифицированный язык моделирования (UML) - это графический язык для визуализации, спецификации, построения и документирования систем, в которых основную роль играет программное обеспечение С помощью UML можно создавать детальные планы создаваемой системы, включая системные функции, бизнес-процессы и другие концептуальные элементы. детальные планы создаваемой системы, включая концептуальные элементы, такие как системные функции, бизнес-процессы и т. д. [9].

Для начала построим абстрактную модель разрабатываемого модуля – диаграмму вариантов использования. Основываясь на требованиях к системе, можем создать обзор приложения.

В системе предусмотрено один вида пользователей: собеседник. Собеседник получает весь функционал продукта(кроме конфиденциальной информации других пользователей).

Составим диаграмму вариантов использования чата на примере двух пользователей (см. рис. 3.1).

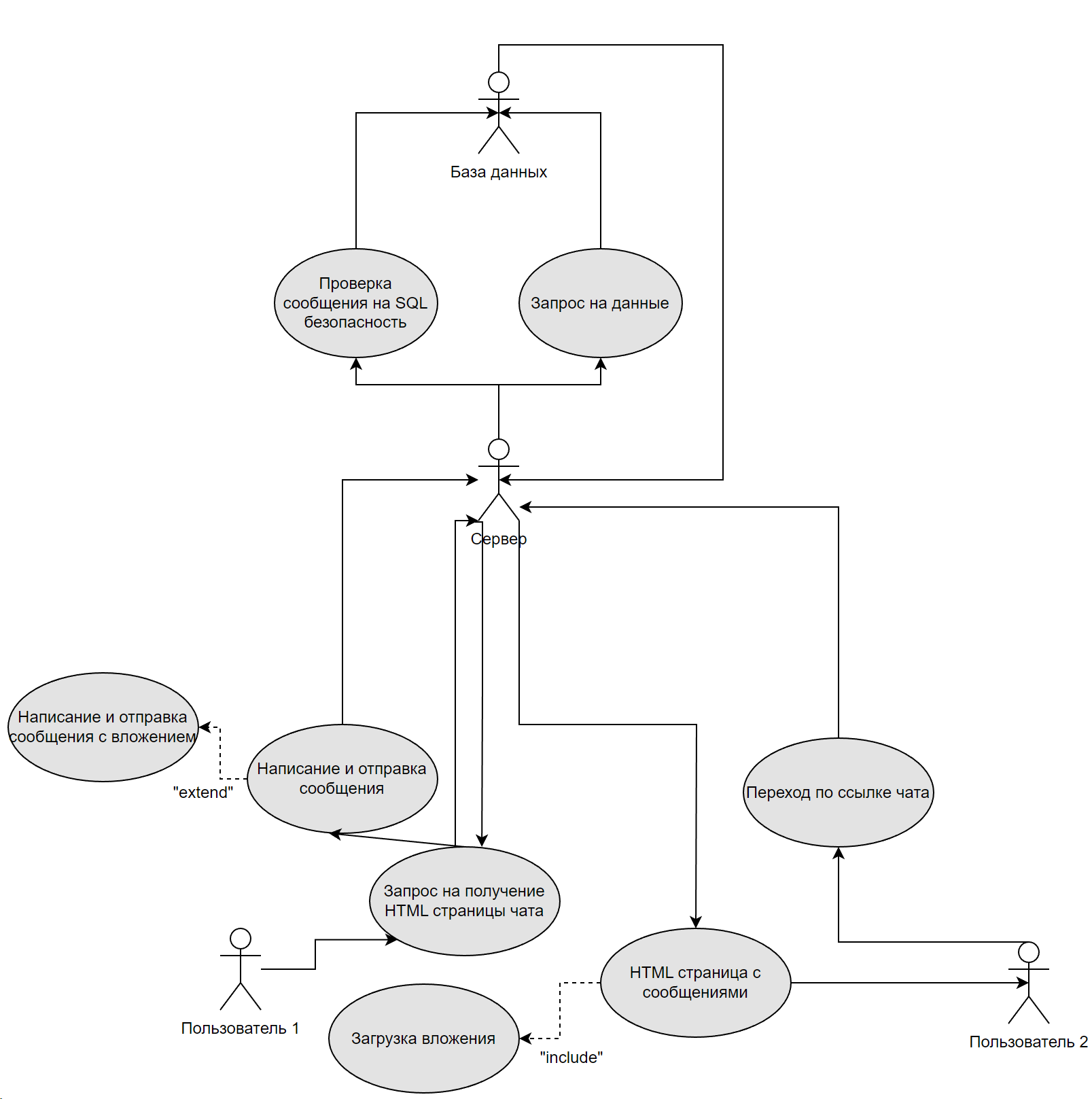


Рисунок 3.1 – Диаграмма прецедентов для пользователей чата

Стоит выделить какие участники предусмотрены и что им позволено делать:

1. «Пользователь 1» – отправитель сообщения.
2. «Пользователь 2» – получатель сообщения.

Рассмотрим диаграмму вариантов использования для пользователей (см. рис. 3.2).

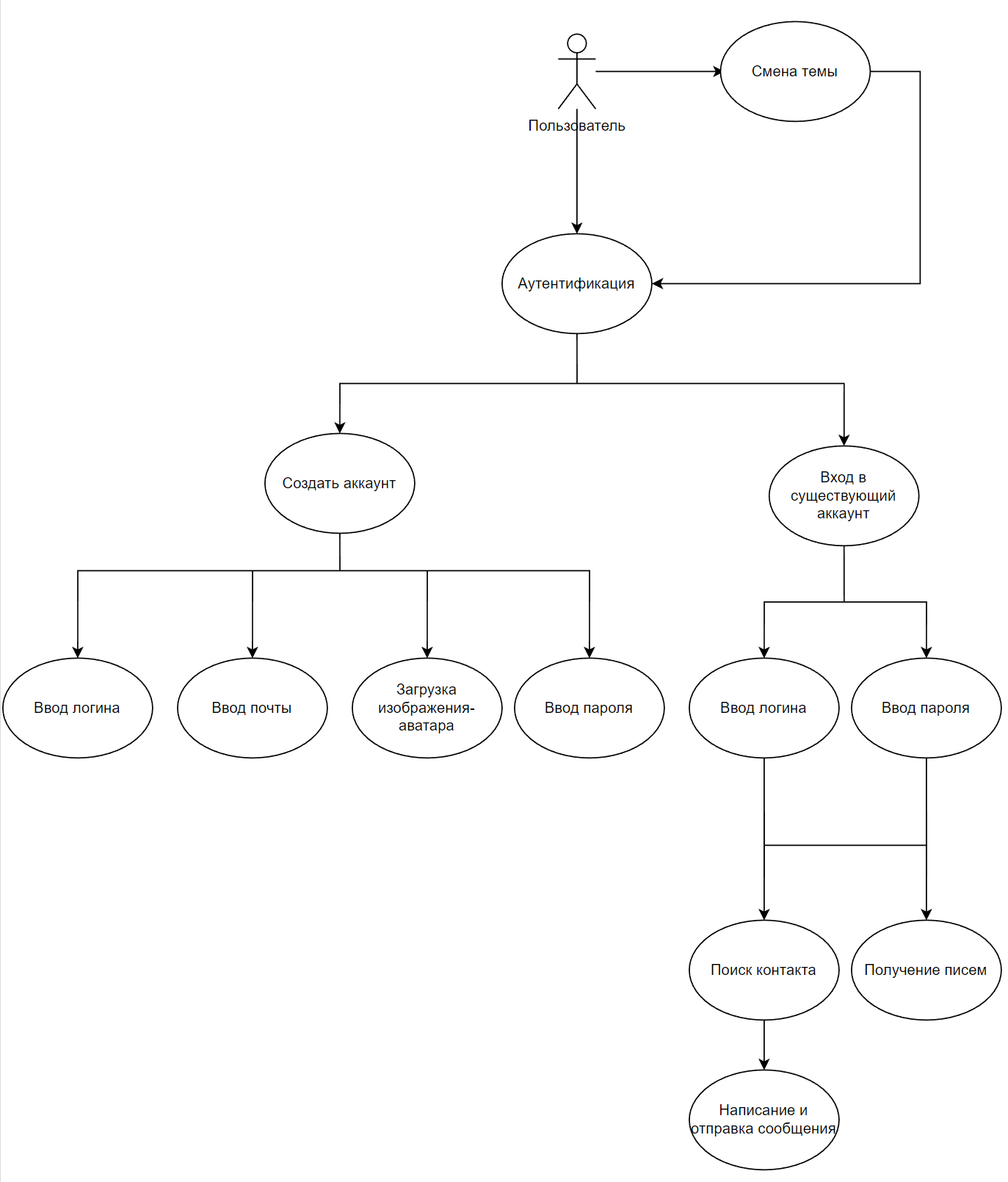


Рисунок 3.2 – Диаграмма вариантов использования для администратора коммуникации

Как видно на диаграмме, пользователь может участвовать как в отправке так и в получении сообщения (по сценарию рисунка 3.1).

Ниже представлена диаграмма состояний. На ней отображается жизненный цикл объекта, начиная с момента его создания и заканчивая разрушением. Таким объектом будет выступать сообщение (см. рис. 3.3).

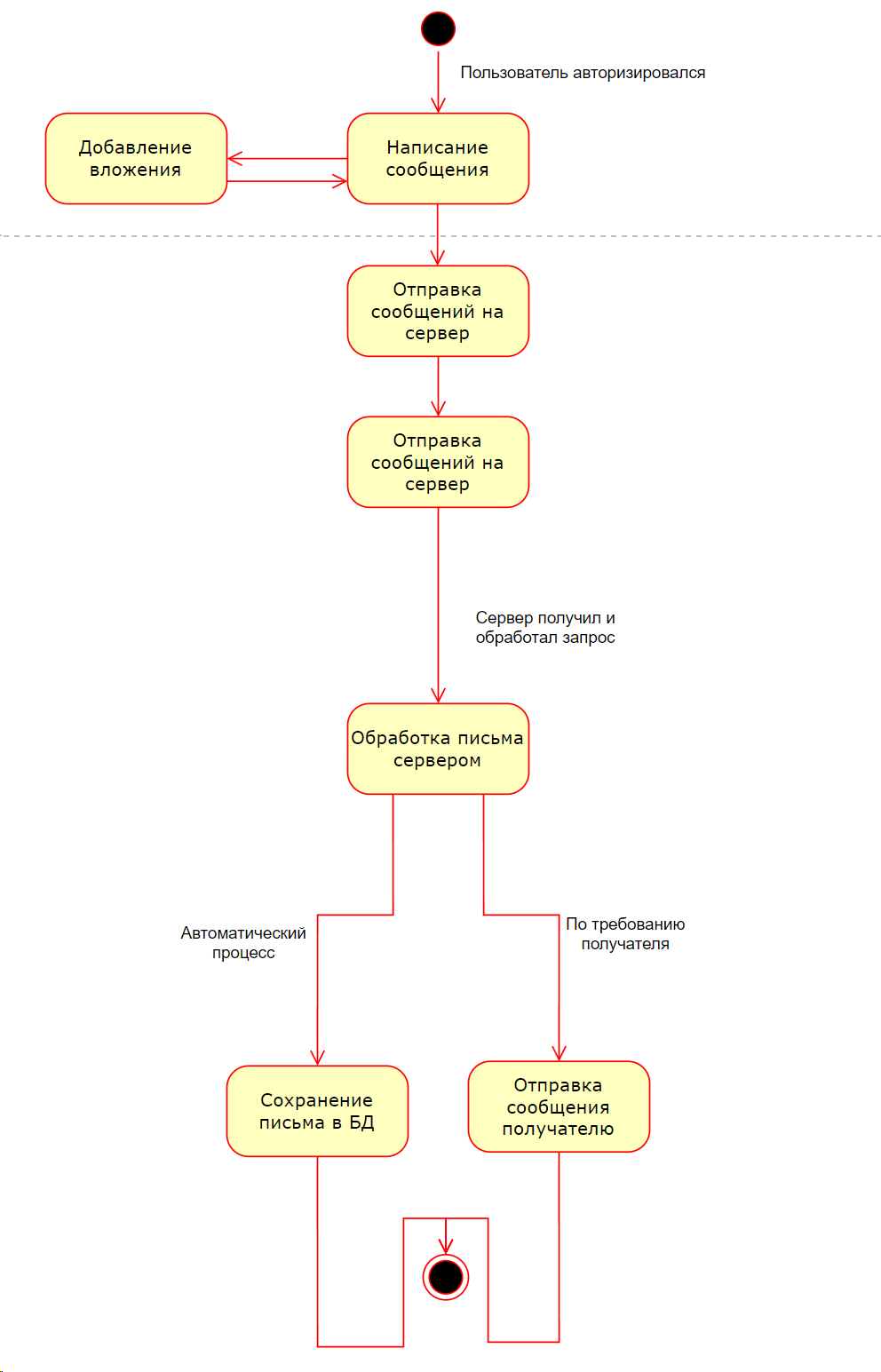


Рисунок 3.3 – Диаграмма состояний

Как и описывалось раннее, автор создаёт сообщение и отправляет сообщение серверу. Сервер отображает сообещние получателю на его странице чата.

Рассмотрим диаграмму последовательности (см. рис. 3.4), которая описывает взаимодействие объектов посредством приёма и передачи сообщений в определённой последовательности во времени.

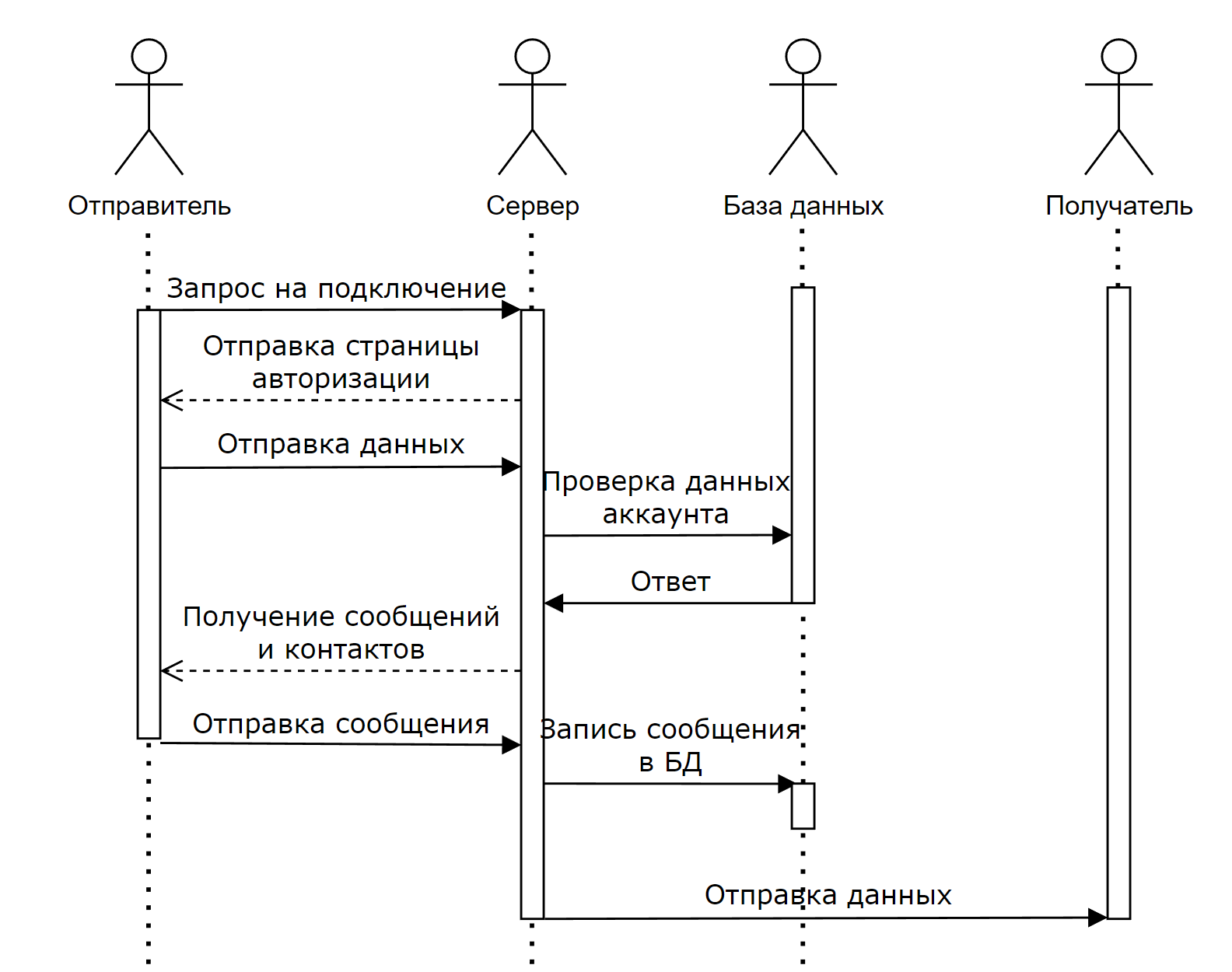


Рисунок 3.4 – Диаграмма последовательности

На данной диаграмме наглядно представлено, как разные участники ПО взаимодействуют друг с другом и чатом. Также показано, когда производится работа с базой данных.

Диаграмма компонентов позволяет создать физическое отражение системы (см. рис. 3.5).

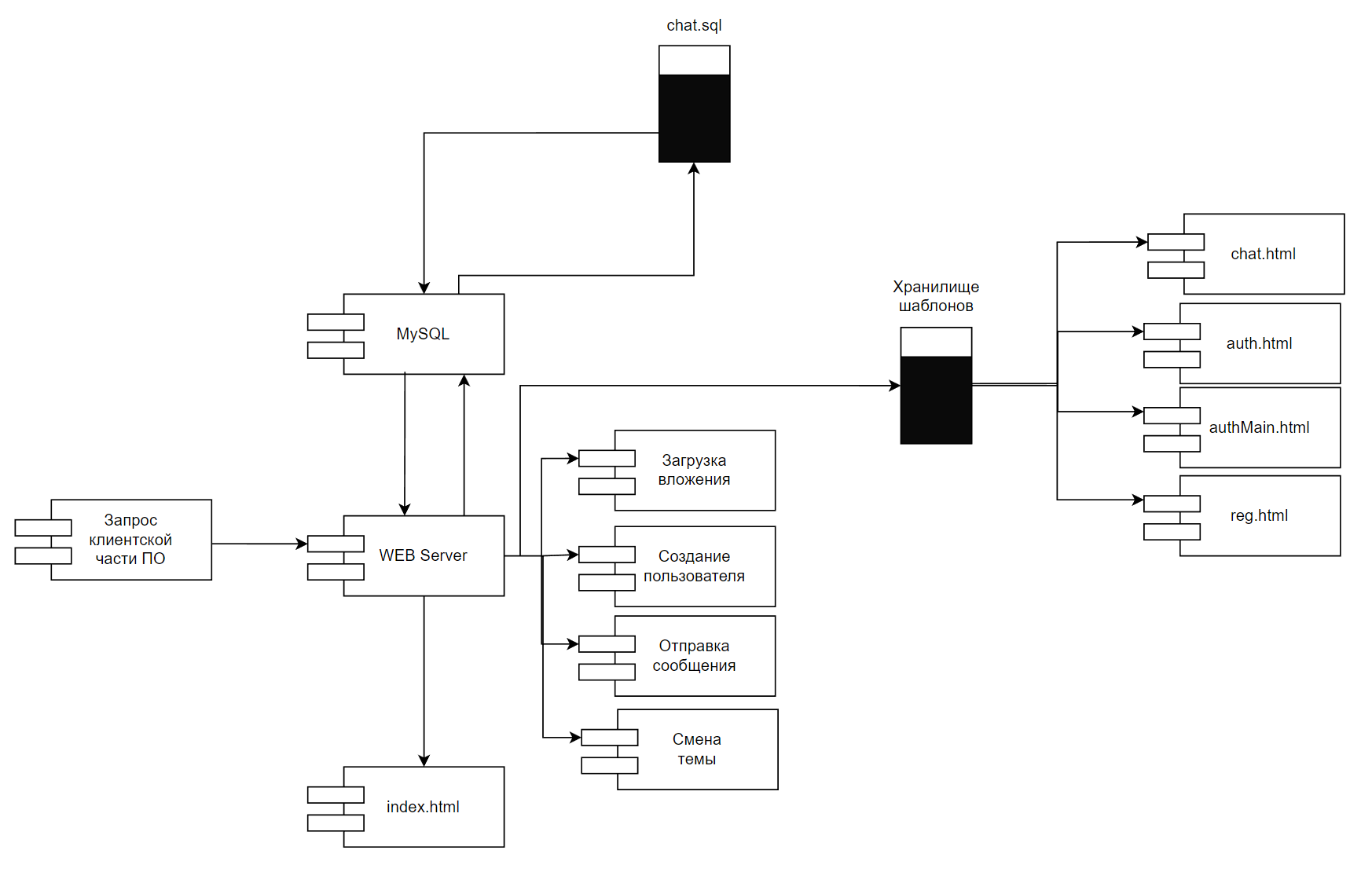


Рисунок 3.5 – Диаграмма компонентов

Продукт состоит из нескольких основных страниц: авторизация, регистрация, главная, чат. К большему числу этих файлов подключены автономные js скрипты для работы локально. Также на схеме видно взаимодействие с базой данных через «MySQL».

На диаграмме не указан css-файл. Он универсален под каждую страницу.

# ЗАКЛЮЧЕНИЕ

В ходе системного анализа предметной области были рассмотрены ключевые аспекты разработки мессенджера. Общая характеристика задачи позволила определить основные цели и задачи работы, направленные на создание функционального и полезного чата.

Анализ аналогичных систем выявил основные особенности существующих чатов, что позволило сделать выводы о преимуществах и недостатках различных подходов к разработке мессенджеров.

Постановка цели и задач работы определила основные направления деятельности, необходимые для успешного завершения проекта. Разработка мероприятий по обеспечению нормальных и безопасных условий труда включала в себя оценку рабочего пространства, выбор и обоснование мероприятий, а также обеспечение пожарной безопасности, вопросы гражданской обороны и безопасности жизнедеятельности человека.

В целом, проведение данного исследования позволило получить ценный опыт и знания в области разработки чатов, а также ознакомиться с вопросами охраны труда, безопасности жизнедеятельности и гражданской обороны в контексте работы над проектами в информационных технологиях.

# СПИСОК ЛИТЕРАТУРЫ

1. [Мессенджер – что это такое, для чего он нужен](https://elama.ru/glossary/messendzher" \o "elama.ru — Мессенджер – что это такое, для чего он нужен, список популярных мессенджеров | Словарь по маркетингу и интернет-рекламе" \t "https://yandex.ru/_blank) – URL: https://elama.ru/glossary/messendzher

2. WhatsApp: история самого популярного мессенджера в мире – URL: https://dzen.ru/a/ZEDARmuMHklH3UfL

3. MySQL: что это, как устроена, чем хороша и как начать с ней работать – URL: https://skillbox.ru/media/code/mysql-chto-eto-kak-ustroena-chem-khorosha-i-kak-nachat-s-ney-rabotat/

4. HTML или CMS – что лучше для коммерческого сайта [Электронный ресурс] – Режим доступа: https://vzh.ru/article/html-ili-cms-chto-luchshe-dlya-kommercheskogo-sajta/ - Загл. с экрана.

5.CSS — Что такое, история создания и преимущества [Электронный ресурс] – Режим доступа: http://proglang.su/css/introduction - Загл. с экрана.