

Воронежский Государственный Университет  
наименование организации – разработчика ТЗ на АС

УТВЕРЖДАЮ

*Клейменов Илья Валерьевич (Ассистент,  
Воронежский Государственный  
Университет)*

Руководитель (должность, наименование  
предприятия – заказчика АС)

Личная                      Расшифровка  
подпись                      подписи

Печать

Дата 24.03.2023

УТВЕРЖДАЮ

*Иванников Сергей Петрович  
(Студент, Воронежский Государственный  
Университет)*

Руководитель (должность, наименование  
предприятия – разработчика АС)

Личная                      Расшифровка  
подпись                      подписи

Печать

Дата 24.03.2023

Веб-приложение для публикации статей  
наименование вида АС

Свободная интернет-энциклопедия с большой базой знаний  
наименование объекта автоматизации

WikiWorld  
сокращенное наименование АС

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На 2-29 листах

Действует с 24.03.2023

СОГЛАСОВАНО

*Тарасов Вячеслав Сергеевич (Старший  
преподаватель, Воронежский Государственный  
Университет)*

Руководитель (должность, наименование  
предприятия – заказчика АС)

Личная                      Расшифровка  
подпись                      подписи

Печать

Дата 24.03.2023

## Содержание

1 Общие сведения.....	4
1.1 Понятия и термины .....	4
1.2 Наименование системы .....	5
1.2.1 Полное наименование системы и название приложения .....	5
1.2.2 Краткое наименование системы.....	5
1.3 Основания для проведения работ .....	6
1.4 Наименование исполнителя и заказчика приложения .....	6
1.4.1 Наименование заказчика .....	6
1.4.2 Наименование исполнителя.....	6
1.5 Плановые сроки начала и окончания работ .....	6
1.6 Сведения об источниках и порядке финансирования работ .....	6
1.7 Порядок оформления и предъявления заказчику результатов работ по созданию приложения .....	6
2 Назначения приложения.....	8
3 Требования к приложению.....	9
3.1 Общие требования .....	9
3.1.1 Требования к архитектуре.....	9
3.1.2 Требование к эргономике и технической эстетике .....	9
3.1.3 Требования к защите информации.....	9
3.1.4 Требования по патентной части .....	9
3.1.5 Требования к используемым технологиям.....	9
3.2 Требования к функциям приложения .....	11
3.2.1 Регистрация и авторизация .....	11

3.2.2 Просмотр статьи.....	12
3.2.3 Просмотр тематик.....	13
3.2.4 Возможности прослушивания текстов статей .....	14
3.2.5 Скачивание контента .....	14
3.2.6 Поиск статей.....	14
3.2.7 Личный кабинет пользователя .....	14
3.2.8 Возможность выхода из аккаунта .....	15
3.2.9 Добавление статьи .....	15
3.2.10 Редактирование статьи .....	16
3.2.11 Просмотр своих статей в личном кабинете.....	16
3.2.12 Администрирование .....	16
3.3 Дополнительные требования .....	19
3.3.1 Требование к масштабируемости и открытости.....	19
3.3.2 Требование к поддерживаемым языкам .....	20
4 Группы пользователей .....	21
5 Перспективы развития, модернизации приложения .....	22
6 Состав и содержание работ по созданию приложения .....	23
7 Порядок контроля и приемки приложения .....	25
8 Требования к составу содержания работ по вводу системы в эксплуатацию .....	26
9 Требования к документированию .....	27
10 Источники разработки .....	28

## 1 Общие сведения

### 1.1 Понятия и термины

- *Клиент* — это аппаратный или программный компонент вычислительной системы, посылающий запросы серверу
- *Сервер* — выделенный или специализированный компьютер для выполнения сервисного программного обеспечения
- *База данных* — это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных (СУБД)
- *HTTP* — это протокол, позволяющий получать различные ресурсы, например, HTML-документы. Протокол HTTP лежит в основе обмена данными в Интернете
- *SQL-запросы* — это наборы команд для работы с реляционными базами данных
- *Дизайн-макет* — это схематичное изображение финальной идеи с указанием всех деталей. В нем указываются концепция, шрифты, тексты, изображения, расположение всех элементов и общая картина продукта
- *Аутентификация* — процедура проверки подлинности, например, проверка подлинности пользователя путем сравнения введенного им пароля с паролем, сохраненным в базе данных
- *Авторизация* — предоставление определенному лицу или группе лиц прав на выполнение определенных действий
- *Фреймворк* — программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта

- *SQL-инъекция* — внедрении в запрос произвольного SQL-кода, который может повредить данные, хранящиеся в БД или предоставить доступ к ним
- *HTTPS* — расширение протокола HTTP для поддержки шифрования в целях повышения безопасности
- *Пользователь* — человек, который использует приложение
- *Аккаунт или учетная запись* — это персональная страница пользователя или личный кабинет, который создается после регистрации на сайте
- *Frontend* — клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса
- *Backend* — программно-аппаратная часть сервиса, отвечающая за функционирование его внутренней части
- *REST* — архитектурный стиль взаимодействия компонентов распределённого приложения в сети
- *API* — описание взаимодействия одной компьютерной программы с другой
- *Статья* — это публикация, которая содержит информацию об определенной теме
- *Тематика статьи* — это конкретная тема или предмет, о котором написана статья. Она определяет, какую информацию следует включить в статью и как организовать ее содержание

## **1.2 Наименование системы**

### **1.2.1 Полное наименование системы и название приложения**

Полное наименование: свободная интернет-энциклопедия, которая содержит множество статей на различные темы.

Название приложения: WikiWorld.

### **1.2.2 Краткое наименование системы**

Краткое наименование приложения: WikiWorld.

### **1.3 Основания для проведения работ**

На основании следующих документов создается данное приложение:

- Техническое задание
- Дизайн-макет приложения созданный в Miro

### **1.4 Наименование исполнителя и заказчика приложения**

#### **1.4.1 Наименование заказчика**

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Заказчик: Ассистент Клейменов Илья Валерьевич. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

#### **1.4.2 Наименование исполнителя**

Исполнитель: студент Иванников Сергей Петрович. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Сильнов Сергей Денисович. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Саратов Иван Русланович. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

### **1.5 Плановые сроки начала и окончания работ**

Плановый срок начала работ – март 2023 г.

Плановый срок окончания работ – июнь 2023 г.

### **1.6 Сведения об источниках и порядке финансирования работ**

Финансирование работ осуществляется из личных средств Исполнителя.

### **1.7 Порядок оформления и предъявления заказчику результатов работ по созданию приложения**

Результаты работ предъявляются Заказчику в следующем виде:

- Работающее, согласно настоящему Техническому Заданию, веб-приложение
- Сервер веб-приложения
- Документация к приложению
- Презентация в формате видео с демонстрацией функциональности приложения
- Презентация по Техническому Заданию
- Защита проекта

## **2 Назначения приложения**

Основными назначениями приложения являются:

Предоставление информации:

- Просмотр статей
- Просмотр тематических подборок статей
- Прослушивание статей

Совместное использование знаний:

- Возможность авторизованным пользователям работать над созданием и редактированием статей
- Возможность для администратора редактировать и удалять контент

Экспорт файлов:

- Скачивание аудиофайла со статьей
- Скачивание текстовых материалов статей

Поиск статей и тематик.



### **3 Требования к приложению**

#### **3.1 Общие требования**

##### **3.1.1 Требования к архитектуре**

Приложение должно быть построено на трехуровневой архитектуре: клиент (веб-приложение) – сервер – база данных.

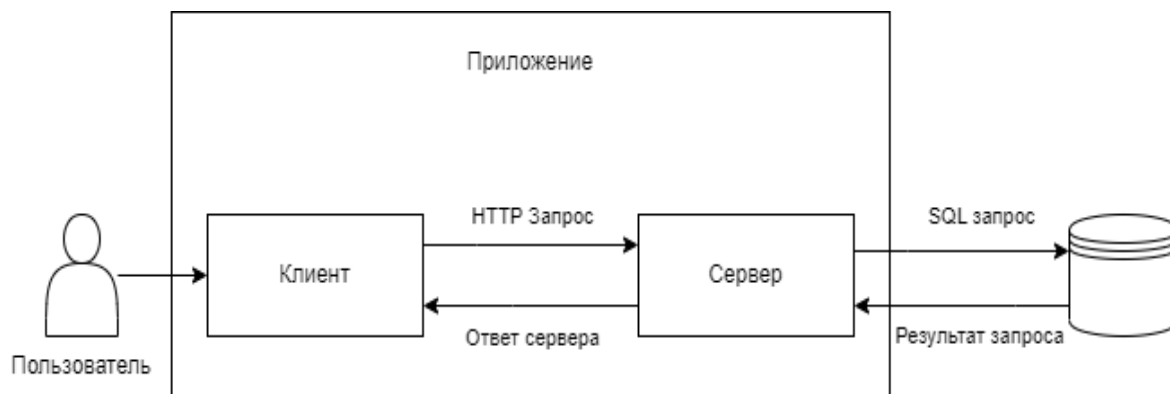


Рисунок 1 - Трех уровневая архитектура приложения

##### **3.1.2 Требование к эргономике и технической эстетике**

Внешний вид приложения должен соответствовать дизайн-макету. Дизайн-макет будет разработан после утверждения настоящего Технического Задания.

##### **3.1.3 Требования к защите информации**

- Обеспечение авторизации и аутентификации пользователей
- Использование механизмов защиты от SQL-инъекций
- Использование протокола передачи данных HTTP

##### **3.1.4 Требования по патентной части**

Приложение должно не нарушать никаких лицензий и патентов. В случае нарушения данного пункта всю ответственность несет сторона Исполнителя.

##### **3.1.5 Требования к используемым технологиям**

Ниже приведен перечень используемых технологий, который в ходе разработки может расширяться.

Backend:

- Python — высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ
- Django REST Framework — это мощный и гибкий инструмент для создания веб-интерфейсов (web APIs)
- PostgreSQL — Данное СУБД было выбрано из-за следующего т.к. является продуктом с открытым исходным кодом, который поддерживается многими серверами. Поддержка различных типов данных. PostgreSQL поддерживает множественные типы данных, такие как числа разной точности, тексты с различными кодировками, изображения, звуки, видео, XML-документы, JSON-объекты и многие другие
- Docker — это программная платформа для быстрой разработки, тестирования и развертывания приложений

#### Frontend:

- HTML — стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере
- CSS — формальный язык описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки
- Bootstrap — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML и CSS шаблоны оформления для типографики, веб форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript расширения

- JavaScript — это высокоуровневый язык программирования, который используется для создания интерактивных веб-страниц и приложений

Инструменты для ведения документации:

- Miro — платформа для совместной работы распределенных команд
- Swagger — это фреймворк для спецификации REST API.
- Draw.io — Бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом. Его интерфейс можно использовать для создания диаграмм, таких как блок-схемы, каркасы, диаграммы UML
- Ramus – графическая среда для проектирования и моделирования сложных систем широкого назначения, который может быть использован для создания диаграмм в формате IDEF0
- Figma – онлайн-сервис для дизайнеров, веб-разработчиков и маркетологов. Он предназначен для создания прототипов сайтов или приложений, иллюстраций и векторной графики

Дополнительный инструментарий:

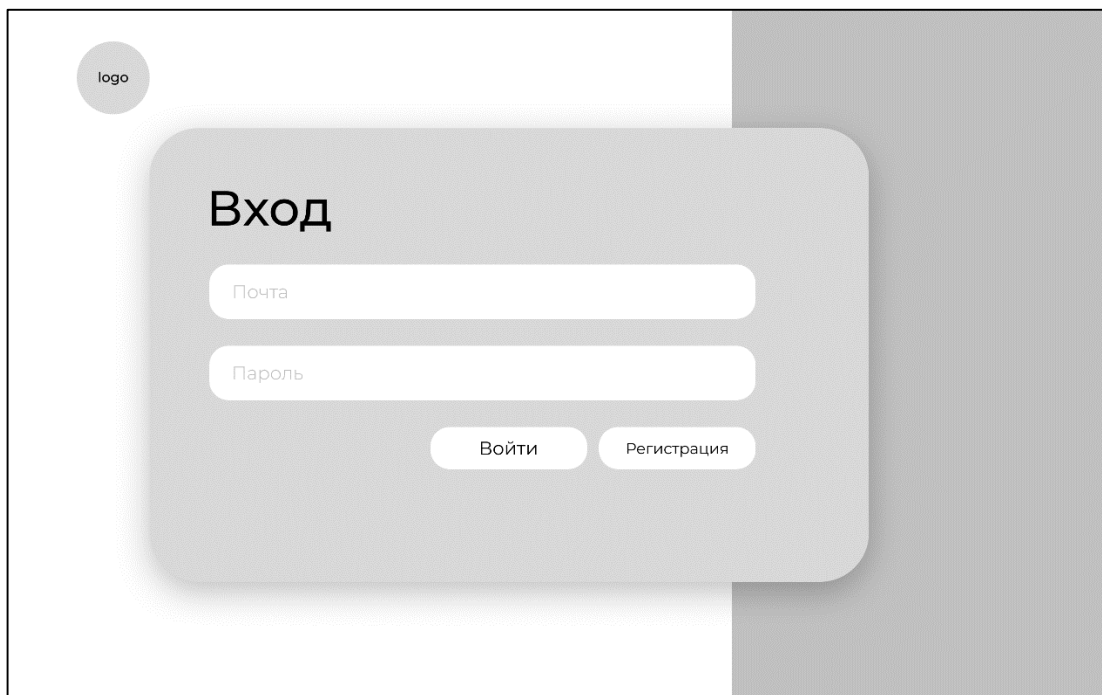
- Git — распределённая система управления версиями.
- GitHub — платформа разработки программного обеспечения с открытым исходным кодом, представляющая систему управления репозиториями кода для Git
- Trello — визуальный инструмент, обеспечивающий эффективность командной работы на любом проекте.

## **3.2 Требования к функциям приложения**

### **3.2.1 Регистрация и авторизация**

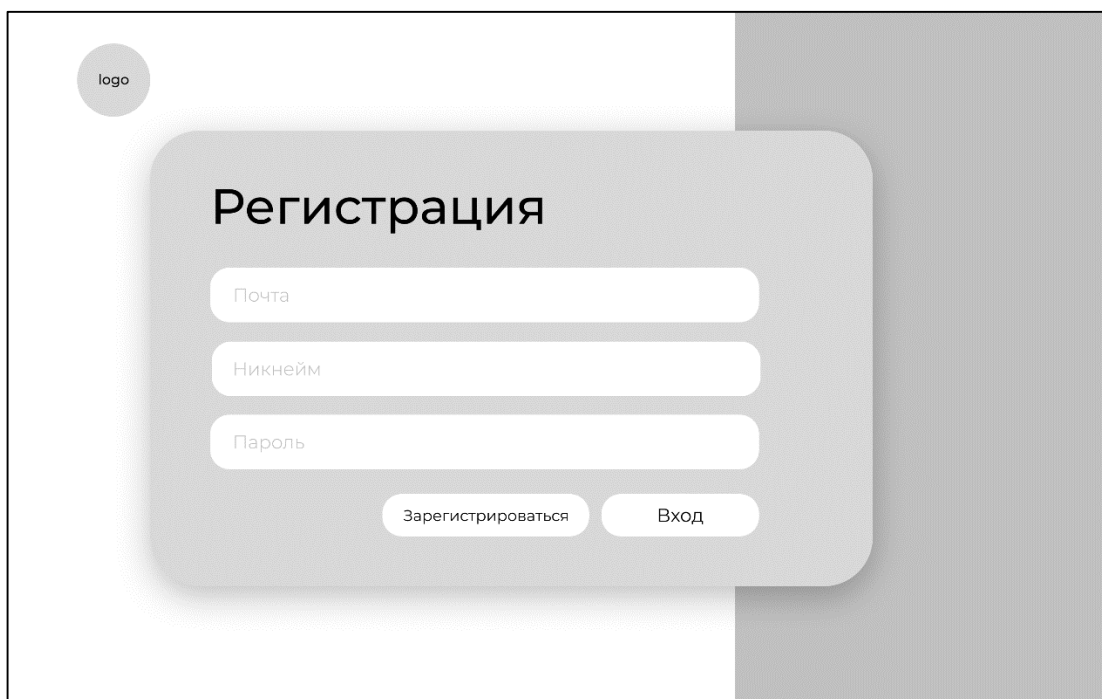
Пользователь при наличии аккаунта может войти систему, введя свои данные: адрес электронной почты и пароль. В ином случае пользователь может зарегистрировать свой аккаунт в системе, введя следующие данные: адрес электронной почты, имя пользователя и пароль. Также при отсутствии

аккаунта пользователь может воспользоваться основной функциональностью приложения. На данном этапе разработки не будут производиться никаких интеграций с почтовыми сервисами и операторами сотовой связи.



The image shows a login form mockup. It features a light gray background with a dark gray vertical bar on the right. In the top left corner, there is a circular logo placeholder labeled "logo". The main form is a rounded rectangle with a light gray background and a subtle drop shadow. It contains the title "Вход" (Login) in a bold, dark font. Below the title are two input fields: "Почта" (Email) and "Пароль" (Password). At the bottom of the form are two buttons: "Войти" (Login) and "Регистрация" (Registration).

Рисунок 2 - Макет страницы со входом



The image shows a registration form mockup. It features a light gray background with a dark gray vertical bar on the right. In the top left corner, there is a circular logo placeholder labeled "logo". The main form is a rounded rectangle with a light gray background and a subtle drop shadow. It contains the title "Регистрация" (Registration) in a bold, dark font. Below the title are three input fields: "Почта" (Email), "Никнейм" (Nickname), and "Пароль" (Password). At the bottom of the form are two buttons: "Зарегистрироваться" (Register) and "Вход" (Login).

Рисунок 3 - Макет страницы регистрации

### 3.2.2 Просмотр статьи

Пользователь (авторизованный и неавторизованный) имеет возможность просматривать выбранную статью.

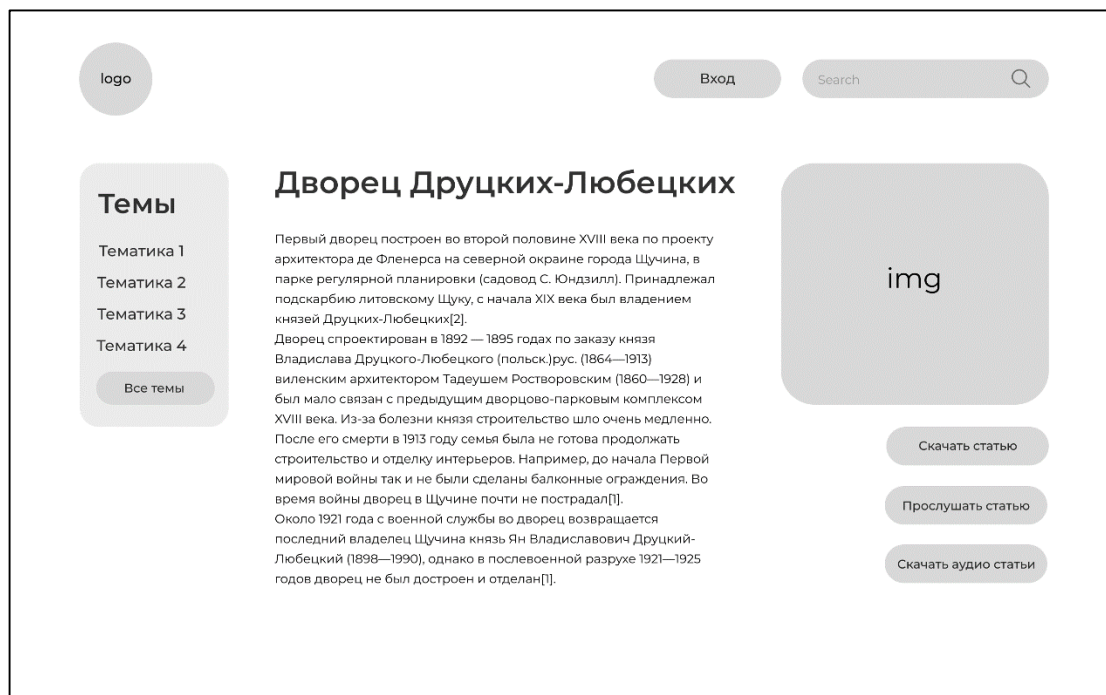


Рисунок 4 - Макет страницы со статьей

### 3.2.3 Просмотр тематик

Пользователь (авторизованный и неавторизованный) имеет возможность просматривать список тематик.

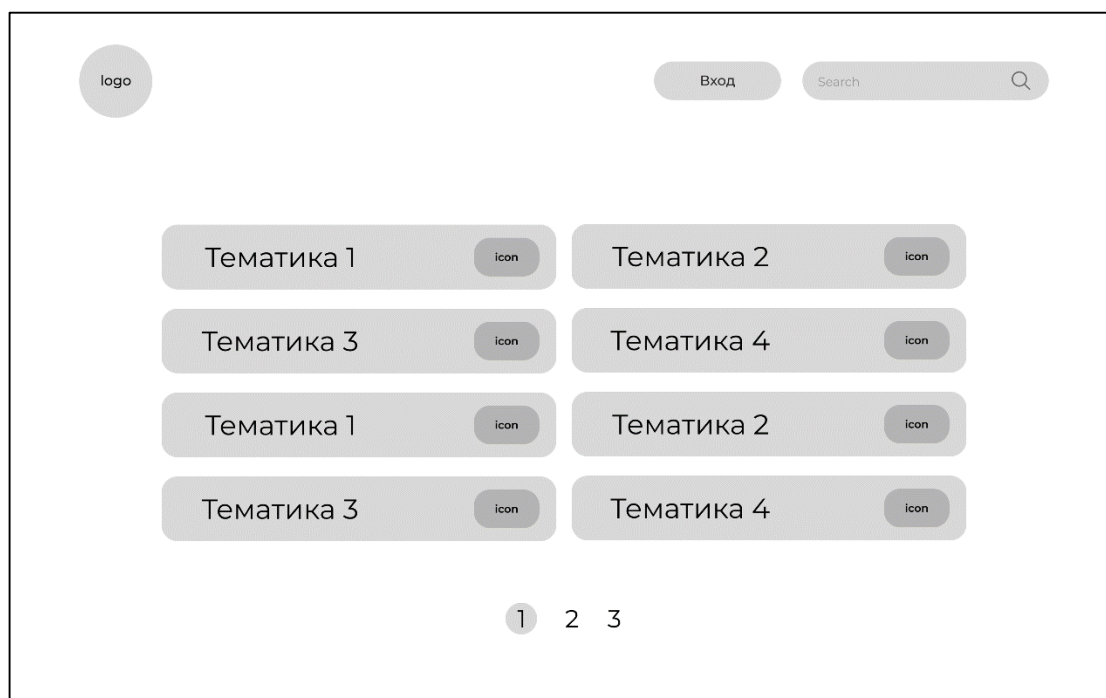


Рисунок 5 - Макет страницы со всеми тематиками

### 3.2.4 Возможности прослушивания текстов статей

Пользователь (авторизованный и неавторизованный) имеет возможность прослушивать статьи в аудиоформате (Рисунок 4).

### 3.2.5 Скачивание контента

Пользователь (авторизованный и неавторизованный) имеет возможность скачивать контент со страницы статьи (Рисунок 4) в текстовом и аудио форматах.

### 3.2.6 Поиск статей

Пользователь (авторизованный и неавторизованный) будет иметь возможность искать статьи, нажав на соответствующее поле ввода и введя ключевые слова, практически на любой странице приложения. После чего откроется выпадающий список с заголовками статей.

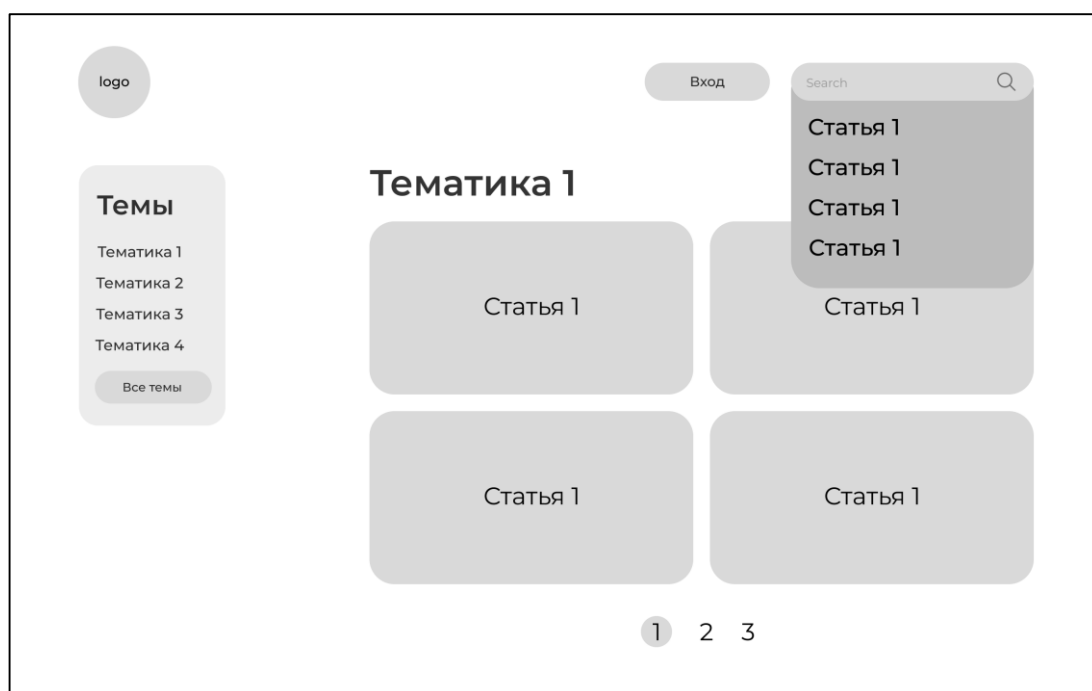


Рисунок 6 - Макет страницы с выпадающим списком поиска

### 3.2.7 Личный кабинет пользователя

Авторизованный пользователь, используя личный кабинет, будет иметь возможность просматривать и редактировать информацию о себе (имя пользователя и смена пароля).

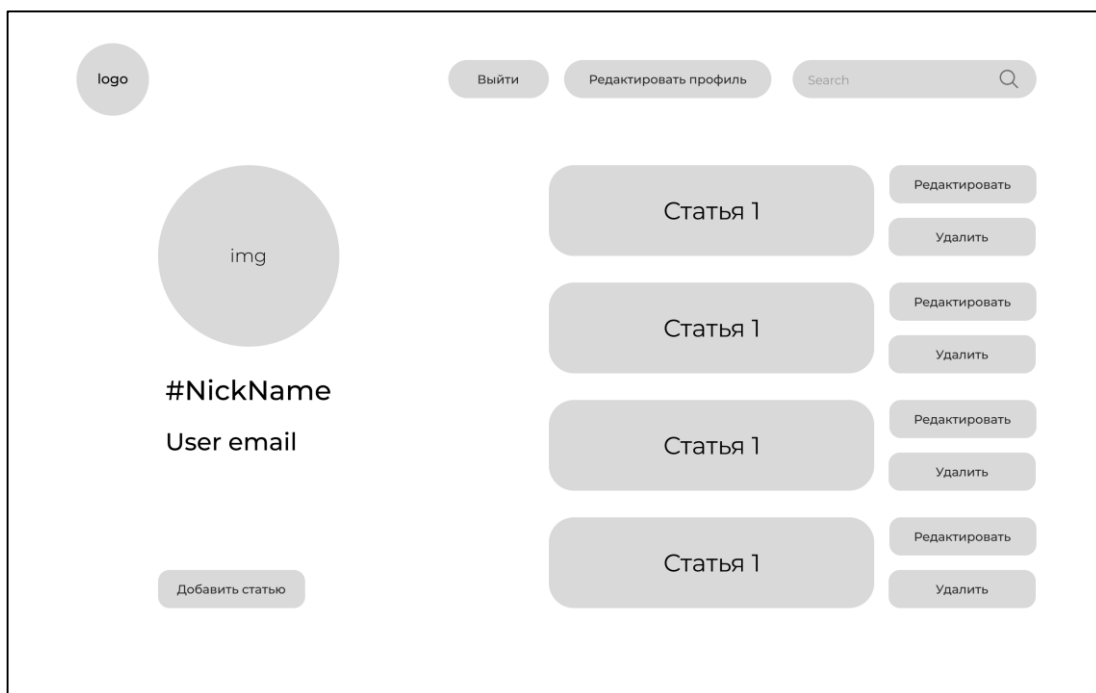


Рисунок 7 - Макет страницы с личным кабинетом

### 3.2.8 Возможность выхода из аккаунта

Авторизованный пользователь имеет возможность выйти из своей учётной записи (Рисунок 7).

### 3.2.9 Добавление статьи

Авторизованный пользователь будет иметь возможность добавить новую статью, нажав на соответствующую кнопку в личном кабинете. После чего откроется страница с формой для заполнения.

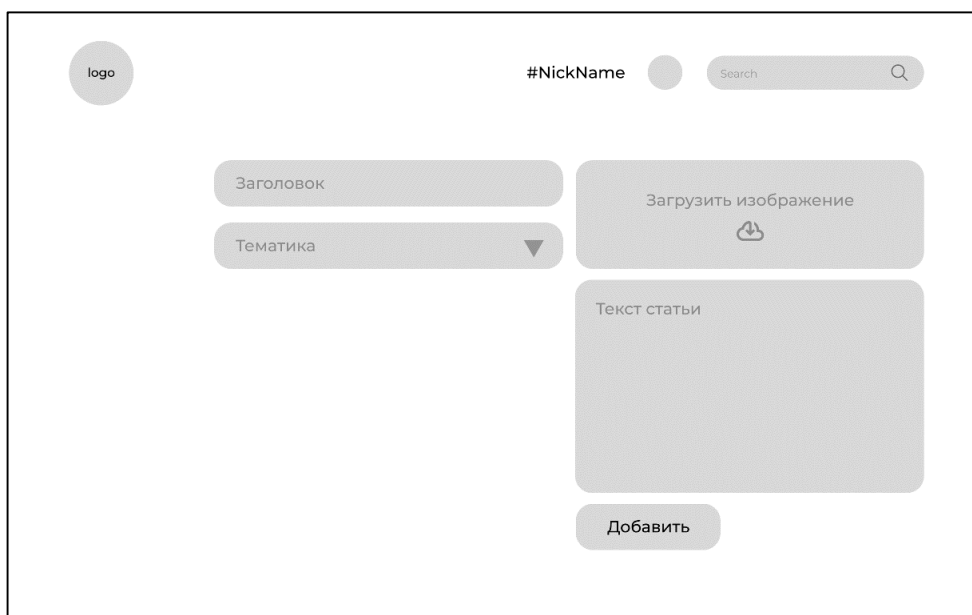


Рисунок 8 - Макет страницы с добавлением статьи

### 3.2.10 Редактирование статьи

Авторизованный пользователь будет иметь возможность редактировать свою статью, нажав на кнопку в личном кабинете. После чего откроется страница с редактированием, где пользователь сможет изменить заголовок, текст статьи, выбрать другую тематику, загрузить другое изображение.

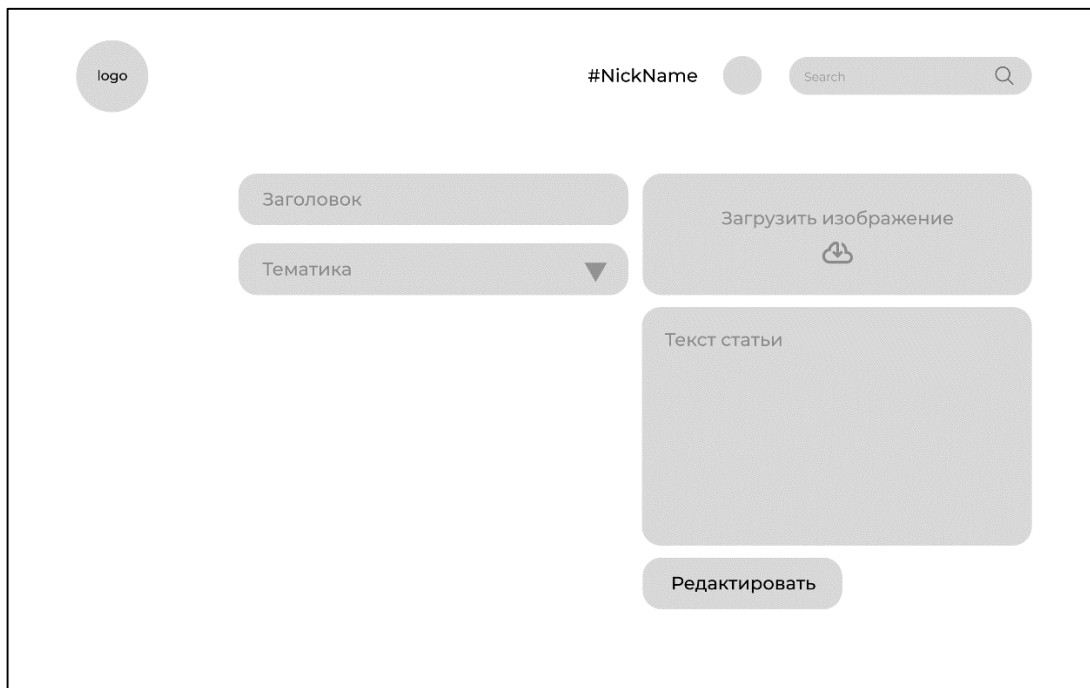


Рисунок 9 - Макет страницы с редактированием статьи

### 3.2.11 Просмотр своих статей в личном кабинете

Авторизованный пользователь будет иметь возможность просматривать список своих статей в личном кабинете (Рисунок 7). Рядом с каждой статьей будут кнопки редактирования и удаления. При нажатии на кнопку удаления, статья будет удалена из списка статей в личном кабинете и базы данных.

### 3.2.12 Администрирование

Администратор системы должен иметь возможность просматривать статьи, иметь возможность их редактировать и удалять, а также добавлять, удалять и редактировать тематики.





Рисунок 10 - Макет главной страницы для администратора

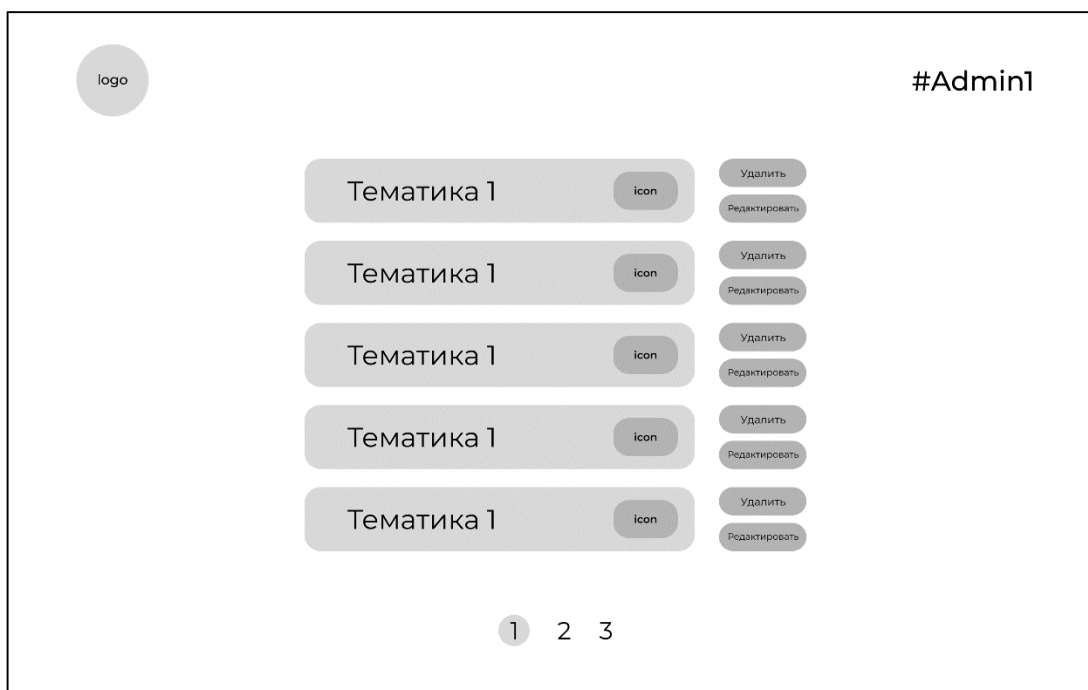


Рисунок 11 - Макет страницы администратора для просмотра тематик

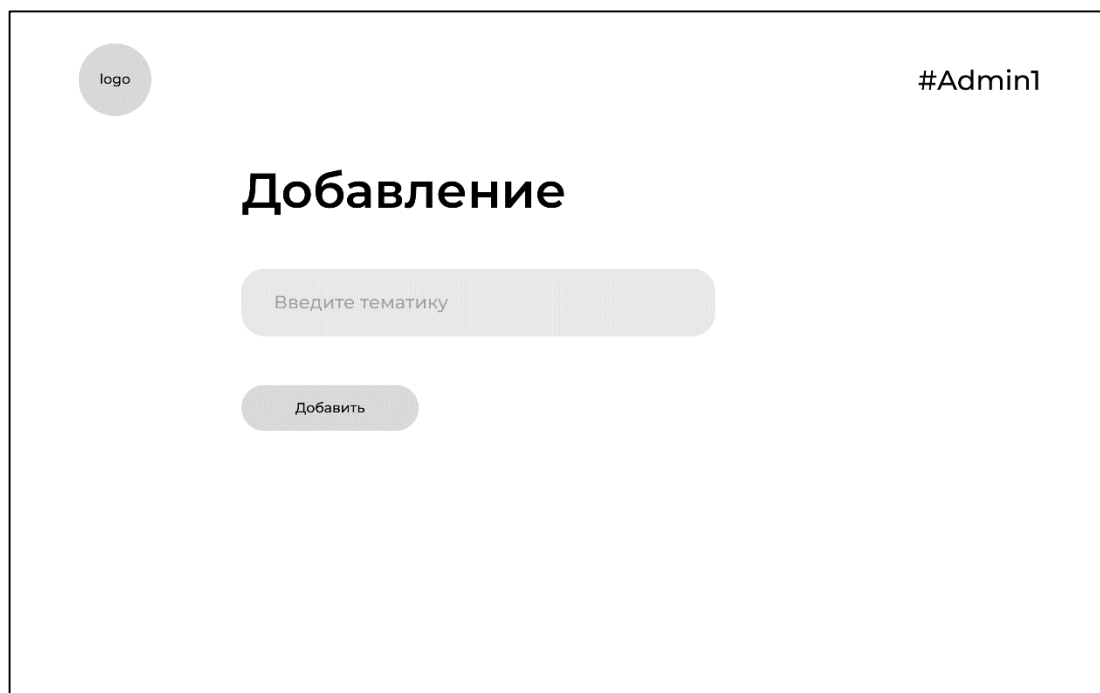


Рисунок 12 - Макет страницы администратора для добавления тематики

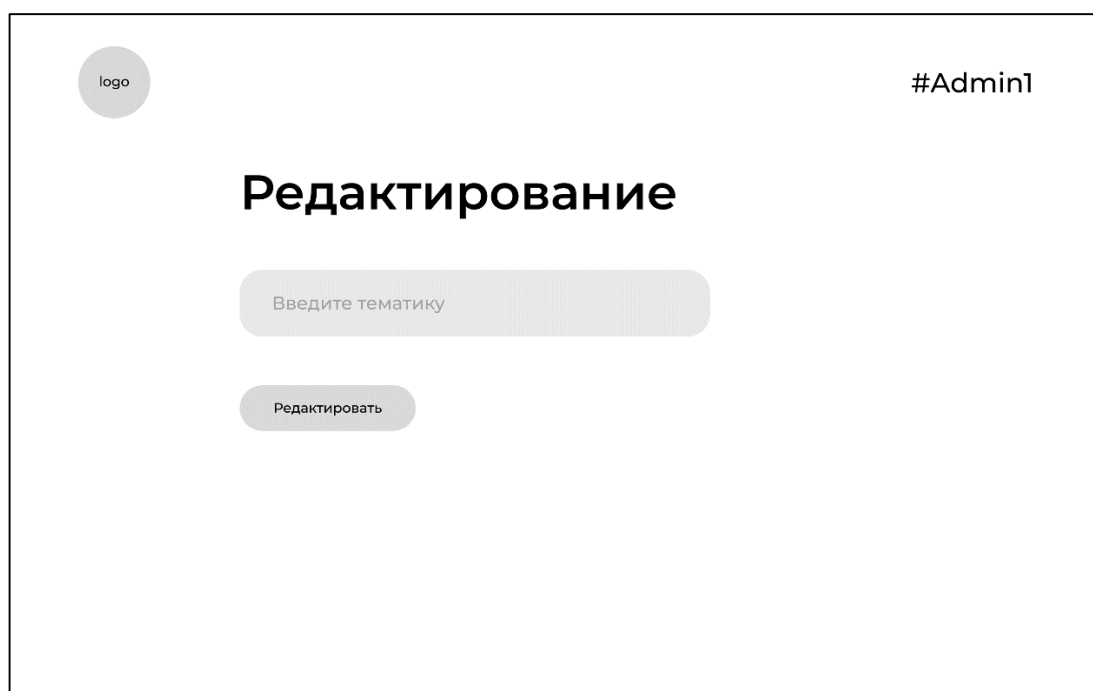


Рисунок 13 - Макет страницы администратора для редактирования  
тематики

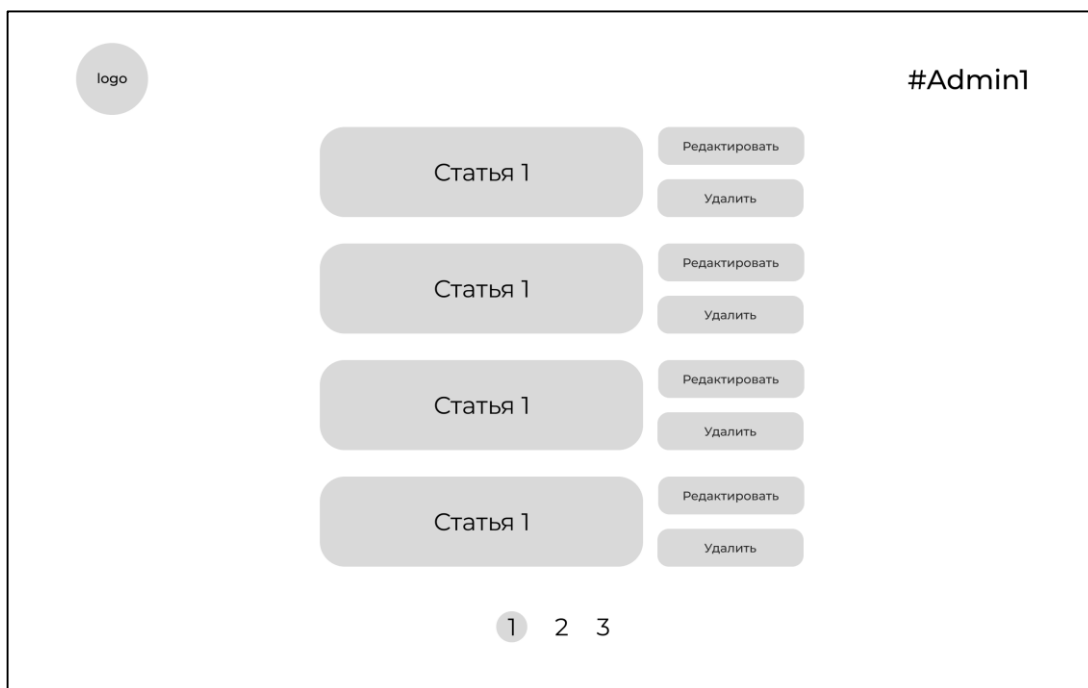


Рисунок 14 - Макет страницы администратора со списком статей

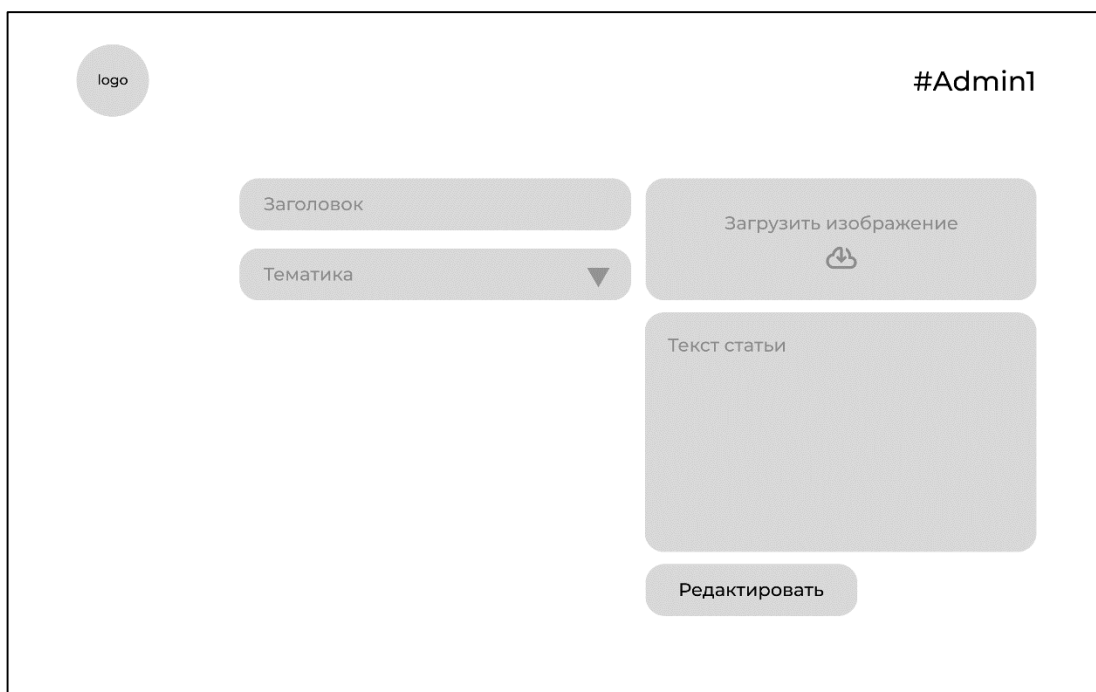


Рисунок 15 - Макет страницы администратора для редактирования статьи

### 3.3 Дополнительные требования

#### 3.3.1 Требование к масштабируемости и открытости

Приложение должно быть легко масштабируемым, т.е. обеспечивать возможность интеграции новой функциональности с минимальным изменением существующей реализации. Исходный код приложения должен находиться в открытом доступе.

### **3.3.2 Требование к поддерживаемым языкам**

Приложение должно поддерживать только русский язык.

#### **4 Группы пользователей**

Приложение должно содержать следующие группы пользователей:

- Неавторизованный пользователь
- Авторизованный пользователь
- Администратор

## **5 Перспективы развития, модернизации приложения**

- Возможность авторизации пользователя, используя сторонние сервисы
- Возможность для пользователей оценивать статьи, написанные другими пользователями
- Возможность загрузки статей из Telegram каналов или сообщества во ВКонтакте
- Подтверждение электронной почты пользователя
- Перенос веб-архитектуры на мобильное устройство

## 6 Состав и содержание работ по созданию приложения

Таблица 1 - Состав и содержание работ по созданию приложения.

Этап	Содержание работ	Порядок приемки и документы	Сроки	Ответственный
Составление технического задания	Разработка функциональных и нефункциональных требований к системе	Утверждение Технического Задания	24.03.2023 г. 21:00 по МСК	Разработка - Исполнитель. Согласование - Заказчик
Техническое проектирование	Разработка дизайн- макета веб-приложения	Описание функциональной схемы в сервисе Miro	24.03.2023 г. 21:00 по МСК	Исполнитель
	Проработка сценариев веб-приложения	Создание UML диаграмм	24.03.2023 г. 21:00 по МСК	Исполнитель
Разработка программной части	Разработка серверного модуля, модуля хранения данных	Приемка осуществляется в процессе испытаний, документация Swagger	Конец мая 2023г.	Исполнитель
	Развертывание серверной части приложения.	Приемка осуществляется в процессе испытаний	Конец мая 2023г.	Исполнитель
	Разработка веб-приложения и интеграция с серверной частью	Приемка осуществляется	Конец мая 2023г.	Исполнитель

		я в процессе испытаний		
Тестирование функциональности приложения	Проверка на соответствие функциональности Техническому Заданию Проверка комплекта документации Доработка и повторные испытания до устранения недостатков	Согласованность с Техническим Заданием Обеспечение тестирования функциональности	Во время и после Разработки	Исполнитель
Эксплуатация	Эксплуатация с привлечением небольшого количества участников (несколько аукционов среди знакомых). Доработки и повторные испытания до устранения выявленных ошибок	Согласованность с Техническим Заданием Сбор отзывов участников эксплуатации	Июнь 2023г.	Исполнитель
Разработка Курсового Проекта	Разработка Курсового Проекта, содержащего информацию о проекте на основе Технического Задания, отчет о проделанной работе	С начала утверждения Технического Задания и до защиты проекта	Июнь 2023г.	Исполнитель



## **7 Порядок контроля и приемки приложения**

Контроль разработки приложения осуществляется путем обсуждения текущего этапа разработки среди исполнителей, а также путем встреч с заказчиком.

Готовое приложение со всей необходимой документацией предоставляется заказчику в обозначенный им срок.

За заказчиком остается осуществление проверки соответствия функциональности приложения согласно данному документу и приемки приложения.

Исполнитель также обязан предоставить заказчику следующий комплект поставки необходимый для защиты проекта:

- Техническое Задание
- Курсовой Проект
- Демонстрационная версия приложения со всеми основными сценариями
- Исходный код приложения
- Аналитику проекта
- Видео-презентацию работы Приложения
- Презентация по Техническому Заданию

## **8 Требования к составу содержания работ по вводу системы в эксплуатацию**

При вводе системы в эксплуатацию необходимо выполнить следующий перечень работ:

Осуществить набор персонала в лице:

- минимум одного главного администратора системы
- тестирование основных сценариев приложения

## **9 Требования к документированию**

Документирование системы ведется в рамках настоящего Технического Задания, составленного в соответствии с ГОСТ 34.602-89.

Документ об отчете проделанной работы над системой будет представлен в рамках Курсового Проекта на основе настоящего Технического Задания.

Документирование основных сценариев работы приложения осуществляется в сервисе Miro.

Документирование серверной части приложения обеспечивается с помощью инструмента Swagger.

## 10 Источники разработки

1. Техническое задание на создание автоматизированной системы [Текст] ГОСТ 34.602-89. – Изд. июнь 2009 г. – Взамен ГОСТ 24.201-85; введ. 24.03.89
2. Документация по использованию PostgreSQL 14[Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/14/index.html> - Заглавие с экрана. – (Дата обращения: 02.03.2023).
3. Документация по использованию Docker [Электронный ресурс]. – Режим доступа: <https://wiki.dieg.info/docker> - Заглавие с экрана. – (Дата обращения: 02.03.2023).
4. Документация по использованию Django Rest Framework [Электронный ресурс]. – Режим доступа: <https://www.django-rest-framework.org/> - Заглавие с экрана. – (Дата обращения: 02.03.2023).

## СОСТАВИЛИ

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Студент	Иванников Сергей Петрович		24.03.2023
Воронежский Государственный Университет	Студент	Сильнов Сергей Денисович		24.03.2023
Воронежский Государственный Университет	Студент	Саратов Иван Русланович		24.03.2023

## СОГЛАСОВАНО

Наименование организации, предприятия	Должность	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Старший преподаватель	Тарасов Вячеслав Сергеевич		24.03.2023
Воронежский Государственный Университет	Ассистент	Клейменов Илья Валерьевич		24.03.2023