**СОДЕРЖАНИЕ**

[**Раздел 1. Проектирование и разработка веб-приложений 3**](#__RefHeading___Toc128827928)

[**1.1. Общие сведения о предприятии** 3](#__RefHeading___Toc128827929)

[**1.2. Анализ предметной области** 3](#__RefHeading___Toc128827930)

[**1.3. Требования заказчика** 3](#__RefHeading___Toc128827931)

[**1.4. Анализ аналогов** 3](#__RefHeading___Toc128827932)

[**1.5. Техническое задание** 3](#__RefHeading___Toc128827933)

[**1.6. Программное обеспечение веб-сервера** 3](#__RefHeading___Toc128827934)

[**1.7. Разработка интерфейса веб-приложения** 4](#__RefHeading___Toc128827935)

[**1.8 Разработка функционала веб-приложения** 4](#__RefHeading___Toc128827936)

[**1.9. Резервное копирование веб-приложения** 4](#__RefHeading___Toc128827937)

[**1.10. Сравнительный анализ услуг хостинга** 4](#__RefHeading___Toc128827938)

[**Раздел 2. Оптимизация веб-приложений** 5](#__RefHeading___Toc128827939)

[**2.1. Контекстное наполнение приложения** 5](#__RefHeading___Toc128827940)

[**2.2. Показатели работы веб-приложения и их анализ** 5](#__RefHeading___Toc128827941)

[**2.3. Мониторинг работы веб-приложения** 5](#__RefHeading___Toc128827942)

[**2.4. Мероприятия по оптимизации.** 5](#__RefHeading___Toc128827943)

[**Раздел 3. Обеспечение безопасности веб-приложений** 6](#__RefHeading___Toc128827944)

[**3.1. Источники угроз безопасности** 6](#__RefHeading___Toc128827945)

[**3.2. Методы защиты доступа к данным и защиты кода** 6](#__RefHeading___Toc128827946)

[**3.3. Диагностика наличия вредоносных программ** 6](#__RefHeading___Toc128827947)

[**3.4. Мероприятия по защите приложения** 6](#__RefHeading___Toc128827948)

[**3.5. Выводы о безопасности.** 6](#__RefHeading___Toc128827949)

[**ВЫВОДЫ** 7](#__RefHeading___Toc128827950)

[**Список литературы** 9](#__RefHeading___Toc128827951)

[Приложение 10](#__RefHeading___Toc128827952)

# Раздел 1. Проектирование и разработка веб-приложений

## 1.1. Общие сведения о предприятии

Структура колледжа представляет собой систему, состоящую из нескольких взаимосвязанных между собой подразделений. Каждое подразделение выполняет свою уникальную функцию в рамках работы колледжа.

Основной целью колледжа является обеспечение качественного образования студентов и подготовка их к будущей профессиональной деятельности. В рамках этой цели колледж ставит перед собой несколько задач:

1. Разработка и реализация учебных программ, соответствующих современным требованиям рынка труда.
2. Создание условий для качественного преподавания студентов, в том числе подготовка и повышение квалификации преподавательского состава.
3. Обеспечение финансовой устойчивости колледжа через планирование и эффективное использование бюджетных средств.
4. Организация дополнительных образовательных мероприятий и программ, включая стажировки и практику для студентов.

Структура колледжа включает несколько подразделений, взаимодействующих между собой. Главное подразделение координирует работу всего колледжа и связывает основные и вспомогательные подразделения. Академическое подразделение занимается организацией преподавания и разработкой учебных программ. Финансовое подразделение занимается финансовым планированием и распределением бюджетных средств. Вспомогательное подразделение обеспечивает поддержку основных подразделений и при необходимости выполняет дополнительные функции, такие как техническая поддержка и снабжение.

Оргструктура колледжа является ключевым фактором успеха для образовательных учреждений. Успешные колледжи выстраивают эффективную организационную структуру, оптимизируют бюджетные процессы и уделяют большое внимание качеству преподавания и обслуживания студентов.

## 1.2. Анализ предметной области

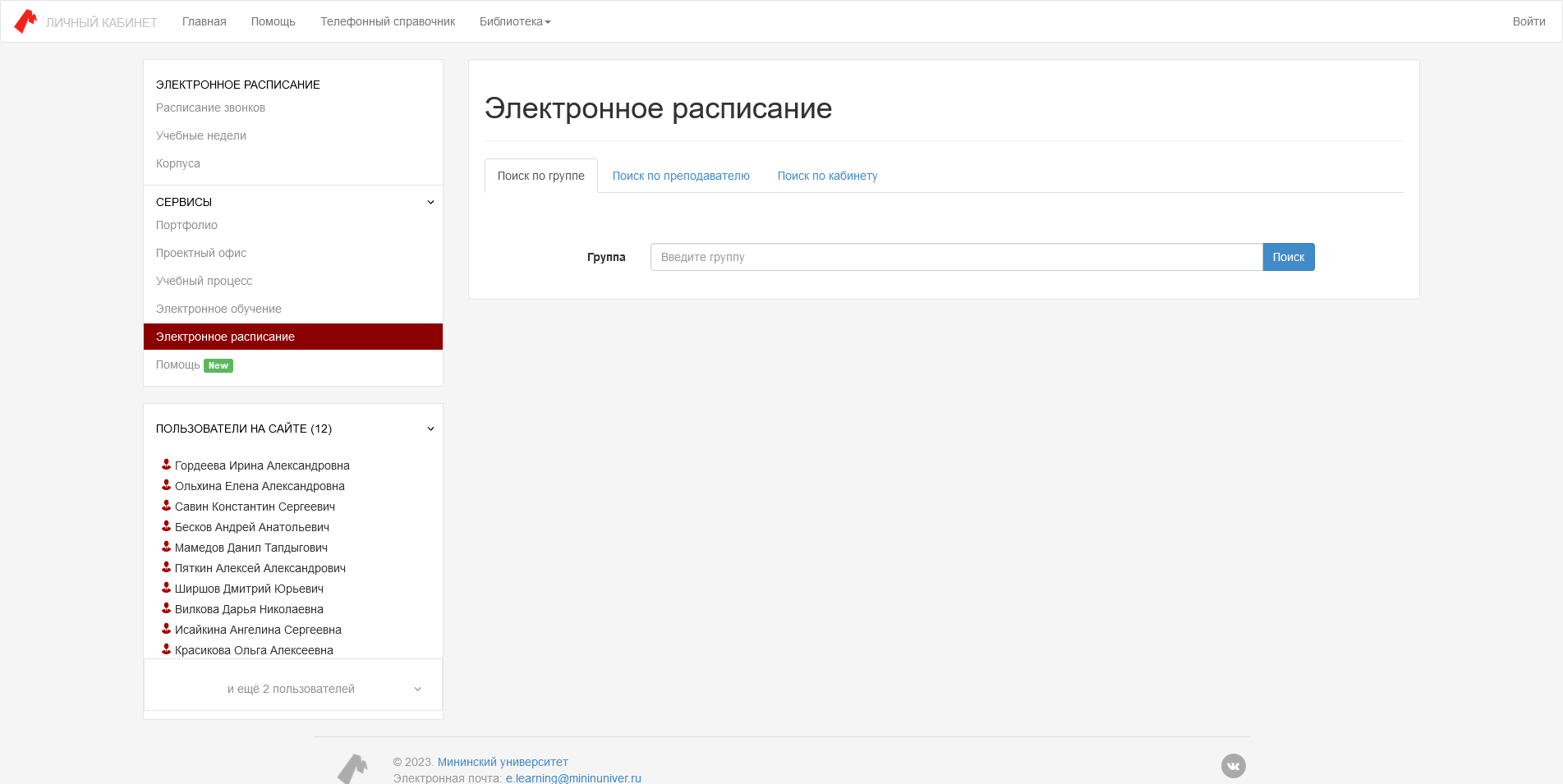
*Концептуальная модель IDEF(1 уровень декомпозиции) и DFD(1 уровень декомпозиции)*

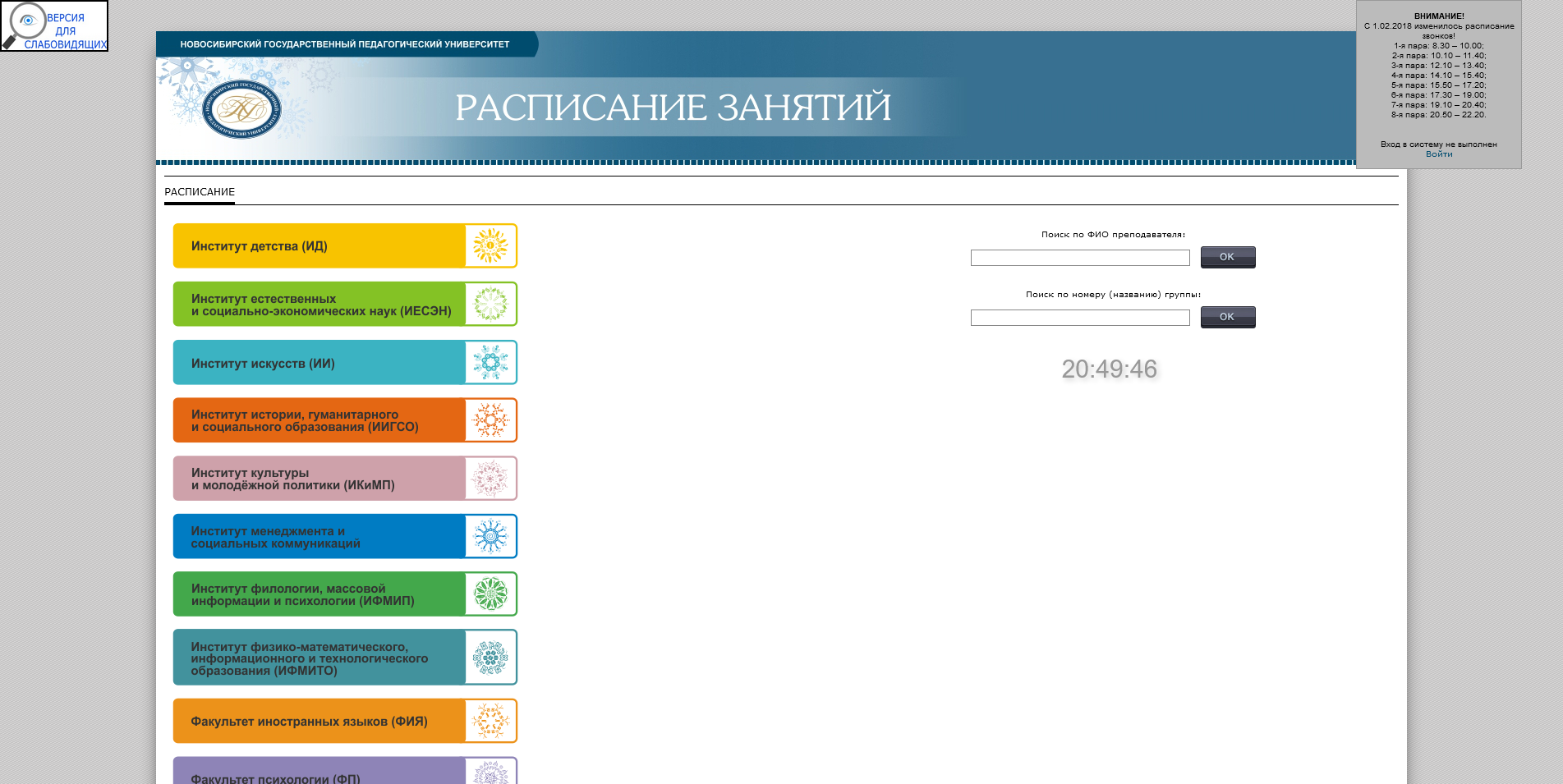
## 1.3. Требования заказчика

*В ходе проведение интервью с заказчиком были выявлены следующие требования:*

* Регистрация
* Авторизация
* Админ панель
* Современный дизайн

## 1.4. Анализ аналогов

Рисунок 1: Электронное расписание mininuniver.ru

Рисунок 2: Электронное расписание nspu.ru

## 1.5. Техническое задание

*При разработке ТЗ придерживаться ГОСТ 19.201-78. Вынести в приложение к ПЗ*

## 1. Наименование продукта

1. "Электронное расписание"

## 2. Назначение

## Программный продукт предназначен для автоматизации составления расписания в учебных заведениях.

## 3. Функциональное назначение

1. Программа должна обеспечивать следующие функции:
2. Администратор, гость и пользователь
3. Составление расписания занятий;
4. Отслеживание изменений в расписании занятий;
5. Просмотр расписания и изменений;
6. Отображение расписания на сайте учебного заведения.

## 4. Технические требования

1. Программа должна работать на операционных системах:
2. Windows 7;
3. Windows 8;
4. Windows 10.
5. Программа должна быть написана на языке программирования python с использованием фреймворка Django drf.
6. Программа должна содержать удобный пользовательский интерфейс.
7. Данные о занятиях и пользователях должны храниться в базе данных SQLite.

## 5. Требования к документации

1. Техническая документация на программный продукт должна соответствовать требованиям ГОСТ 19.201-78 и содержать следующие разделы:
2. Техническое описание программы;
3. Руководство пользователя;
4. Руководство администратора;
5. Программа-инструкция на CD-ROM;
6. Набор тестовых заданий для проверки работоспособности программы.

## 6. Требования к срокам и стоимости

1. Срок выполнения работ - 2 месяцев.
2. Стоимость разработки программного продукта - 0 рублей.

## 7. Требования к порядку приемки и сдачи работ

1. Поставщик должен предоставить в тестовую эксплуатацию электронное расписание сроком на один месяц.
2. После успешной проверки заказчик должен уведомить поставщика о готовности принять работу в эксплуатацию.
3. Весь текст должен быть оформлен согласно ГОСТ 19.201-78.

## 1.6. Программное обеспечение веб-сервера

*Описание названий и версий ПО для работы веб-сервера, описание алгоритма настройки ПО для веб-сервера*

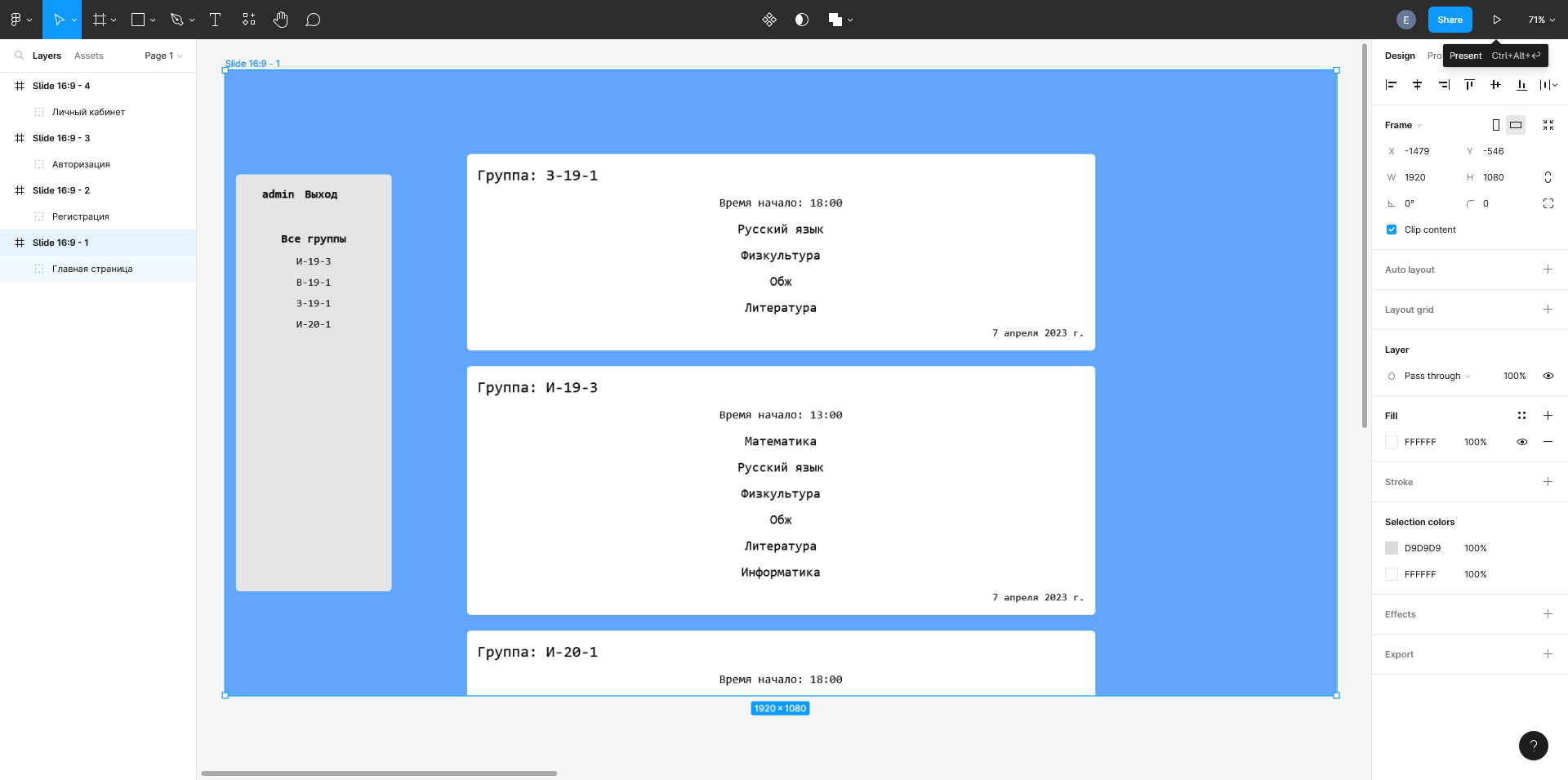
Для разработки Django DRF проекта вам потребуется установить следующее программное обеспечение:

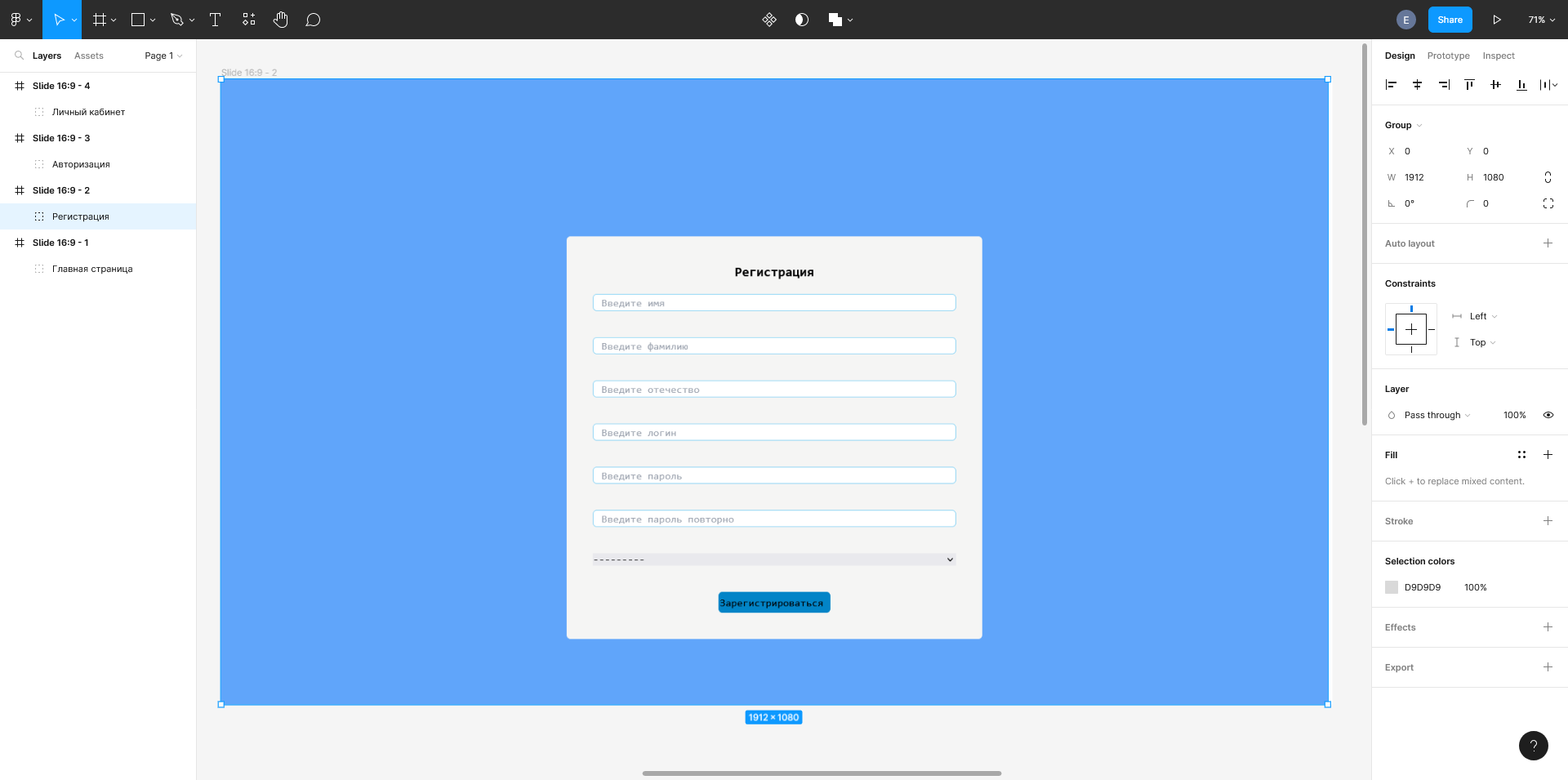
* Python версии 3.6 или выше
* Django версии 3.0 или выше
* Django REST framework версии 3.11 или выше
* Для установки этих зависимостей рекомендуется использовать менеджер пакетов Python - pip.
* Кроме того, для разработки проекта может потребоваться следующее ПО:
* База данных, например, PostgreSQL версии 11 или выше
* Утилита наподобии git для контроля версий

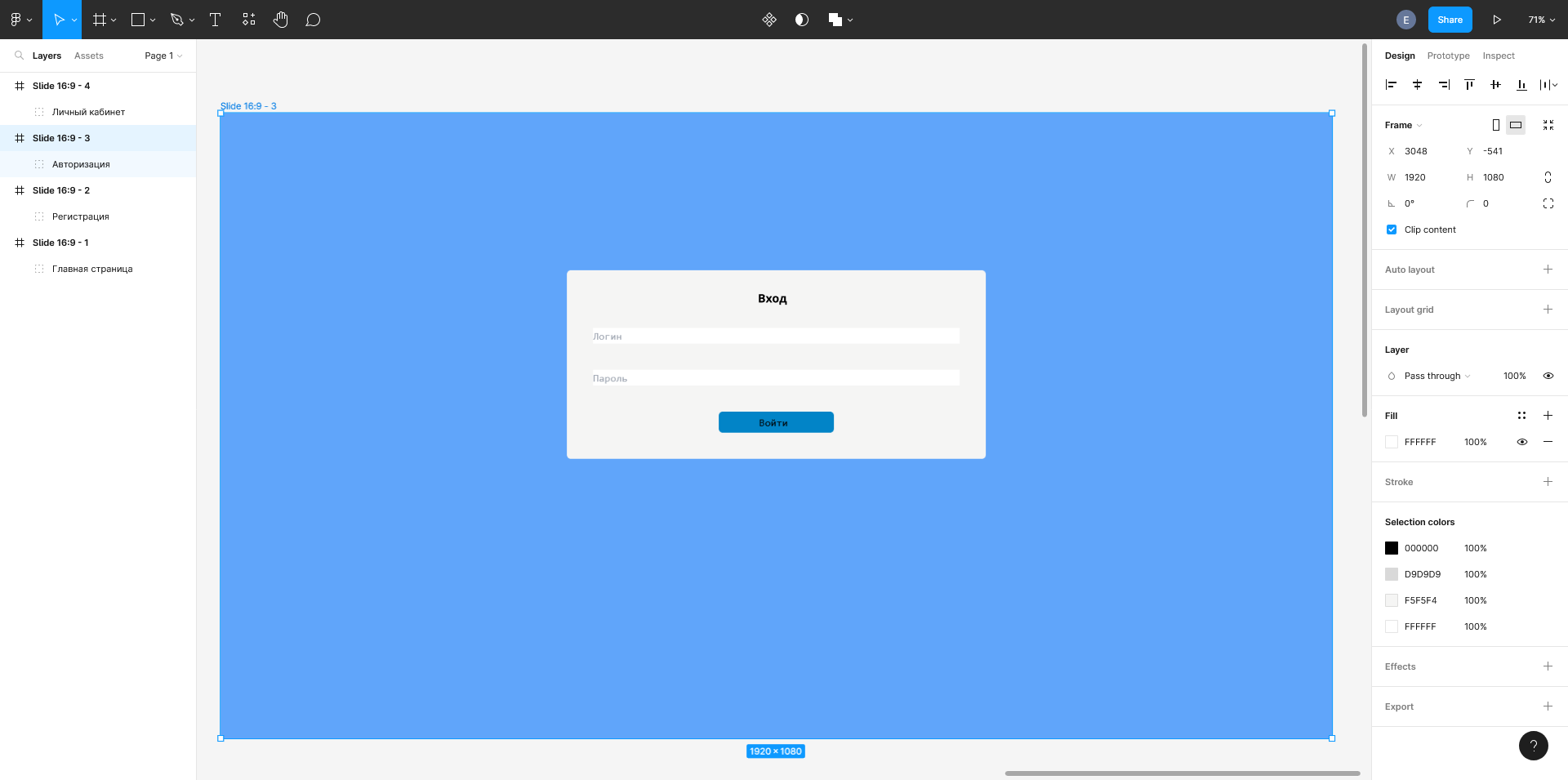
Для настройки ПО для веб-сервера Django DRF на хостинге, вам следует следующим образом:

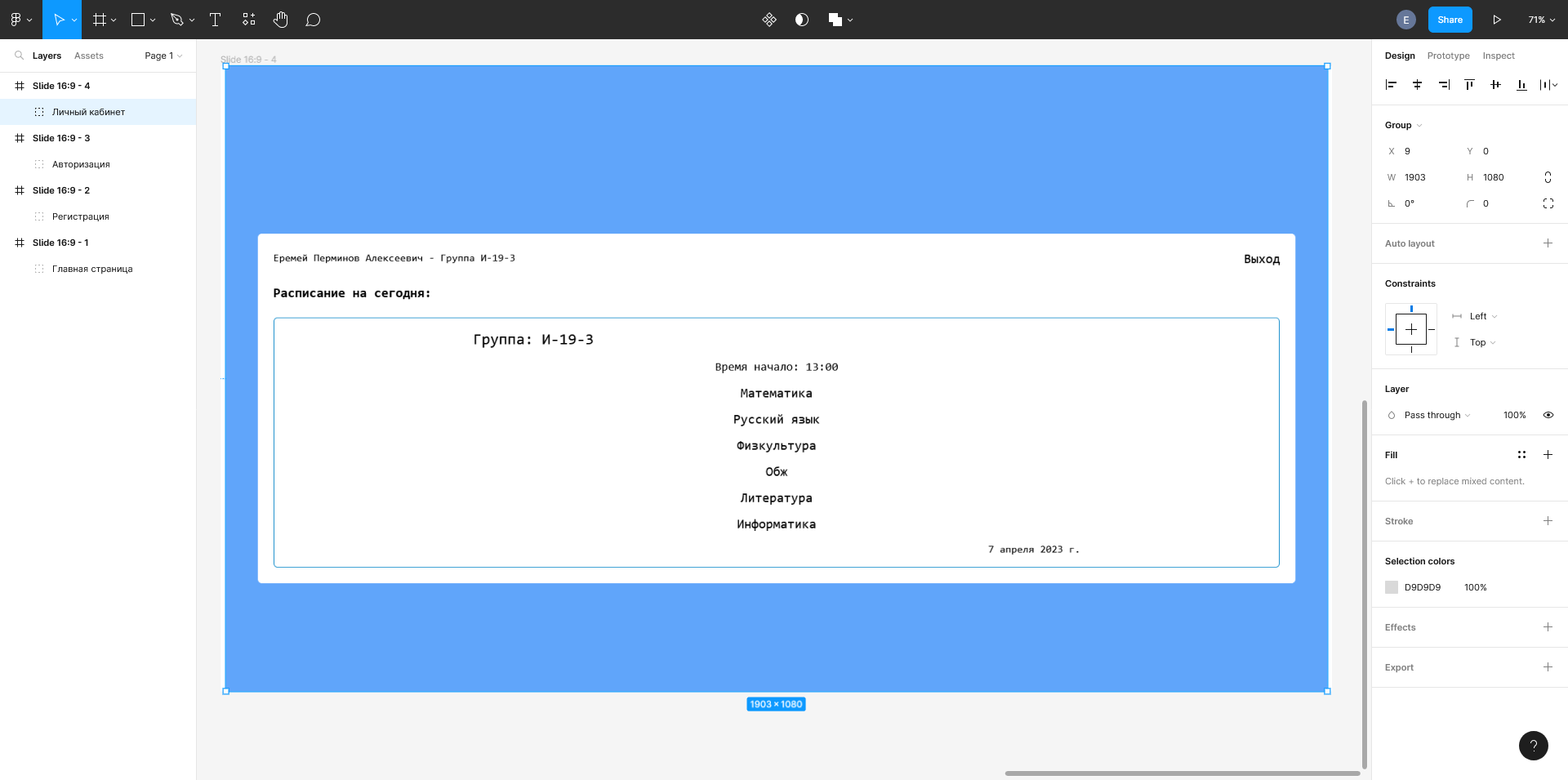
* Приобретите хостинг-аккаунт у провайдера услуг хостинга, который поддерживает Django DRF и имеет поддержку веб-сервера (например, Apache или Nginx) и базы данных (например, MySQL или PostgreSQL).
* Зайдите в панель управления вашего хостинг-аккаунта и найдите опцию для установки веб-сервера и базы данных. В большинстве случаев, провайдер услуг хостинга предоставляет возможность установки необходимых программных компонентов в несколько кликов с помощью панели управления.
* Настройте файл конфигурации веб-сервера, указав каталог, где расположено приложение Django. Для Apache файл конфигурации находится по пути /etc/apache2/apache2.conf. В файле конфигурации укажите каталог, где расположены файлы проекта Django: <Directory /path/to/your/project> Require all granted
* Создайте и активируйте виртуальное окружение с помощью инструмента управления виртуальными окружениями, который, как правило, предоставляет провайдер услуг хостинга.
* Установите Django и необходимые библиотеки с помощью инструмента управления пакетами, которые также, как правило, предоставляет провайдер услуг хостинга.
* Настройте файрвол для веб-сервера, чтобы обеспечить безопасность сетевых подключений.
* Настройте DNS-сервер, если требуется.

## 1.7. Разработка интерфейса веб-приложения

Рисунок 3: Главная страница

Рисунок 4: Страница регитсрации

Рисунок 5: Страница авторизации

Рисунок 6: Станица профиля

## 1.8 Разработка функционала веб-приложения

Таблица 1: Время

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип поля | Первичный ключ | Внешний ключ |
| id | integer | да |  |
| time | varchar(30) |  |  |

Таблица 2: Предметы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип поля | Первичный ключ | Внешний ключ |
| id | integer | да |  |
| lesson | varchar (70) |  |  |

Таблица 3: Группы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип поля | Первичный ключ | Внешний ключ |
| id | integer | да |  |
| group | varchar (70) |  |  |

Таблица 4: Таблица многи ко многи

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип поля | Первичный ключ | Внешний ключ |
| id | integer | да |  |
| schedule\_id | bigint |  | да |
| lessonschedule\_id | bigint |  | да |

Таблица 5: Расписание

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип поля | Первичный ключ | Внешний ключ |
| id | integer | да |  |
| created | date |  |  |
| group\_id | bigint |  | да |
| time\_id | bigint |  | да |
| lesson | ARRAY |  | да |

## 1.9. Резервное копирование веб-приложения

## *План мероприятий по резервному копированию данных:*

* Определить критически важные данные и ресурсы, которые нуждаются в резервном копировании
* Определить метод резервного копирования: полное, инкрементное или дифференциальное
* Выбрать место для хранения резервных копий, оптимально использовать отдельный сервер или облачный сервис, который обеспечит безопасность данных при возникновении непредвиденной ситуации
* Настроить расписание резервного копирования, чтобы обеспечить ежедневное, недельное или месячное создание резервных копий
* Проверять резервные копии на ежемесячной основе для обеспечения целостности и доступности данных в случае их восстановления.

Описание процесса резервного копирования:

1. Определить данные и ресурсы, которые нуждаются в резервном копировании
2. Выбрать метод резервного копирования: полное, инкрементное или дифференциальное
3. Выбрать место для хранения резервных копий, обеспечить доступность хранилища данных в случае необходимости восстановления
4. Настроить расписание резервного копирования: определить, как часто должны создаваться резервные копии (ежедневно, недельно или месячно)
5. Запустить процесс резервного копирования
6. Проверить созданную резервную копию на наличие ошибок и ее полноту при необходимости восстановления.

Работа с системой контроля версий :

* Установить систему контроля версий Git на локальную машину
* Создать репозиторий в Git и добавить его в проект
* Добавлять изменения в файлы и делать коммиты в Git
* Создавать ветки для новых функций и исправлений ошибок
* Сливать ветки в основную ветку после проверки кода на качество
* *Добавлять удаленные репозитории и загружать изменения в удаленный репозиторий при необходимости.*

## 1.10. Сравнительный анализ услуг хостинга

*Обзор ресурсов с услугами хостинга, критерии отбора, обоснование выбора одного из хостингов, расчет стоимости размещения веб-сайта на хостинге на полгода.*

*На рынке существует множество компаний, которые предоставляют услуги хостинга веб-сайтов. Некоторые из них самые популярные компании в этой области включают в себя:*

* Bluehost
* HostGator
* SiteGround
* Dreamhost
* InMotion Hosting

Критерии отбора:

При выборе хостинга нужно ориентироваться на ряд критериев, включая:

* Надежность и устойчивость системы;
* Скорость работы сайта;
* Возможности масштабирования;
* Уровень безопасности;
* Качество технической поддержки;
* Цена.

Обоснование выбора одного из хостингов:

После изучения различных предложений на рынке, мы выбрали компанию SiteGround. Она получает высокие оценки за качество своих услуг и обслуживания, а также за свою клиенто ориентированность. Кроме того, SiteGround предоставляет гибкие планы тарифов, которые обеспечивают различные пакеты для разных потребностей и бюджетов.

Расчет стоимости размещения веб-сайта на хостинге на полгода:

Стоимость размещения веб-сайта на хостинге на полгода зависит от выбранного плана тарифов. Например, SiteGround предоставляет три основных плана тарифов: StartUp, GrowBig и GoGeek.

StartUp предлагает базовый пакет функций, включая один сайт и 10 ГБ пространства для хранения. Его стоимость на полгода составляет 4900 руб.

GrowBig предлагает расширенные возможности, включая неограниченное количество сайтов и 20 ГБ пространства для хранения. Его стоимость на полгода составляет 9800 руб..

GoGeek предлагает разнообразные дополнительные функции, включая неограниченное количество сайтов и 40 ГБ пространства для хранения. Его стоимость на полгода составляет 14000 руб..

# Раздел 2. Оптимизация веб-приложений

## 2.1. Контекстное наполнение приложения

***Ниже приведена оптимизация веб-приложений:***

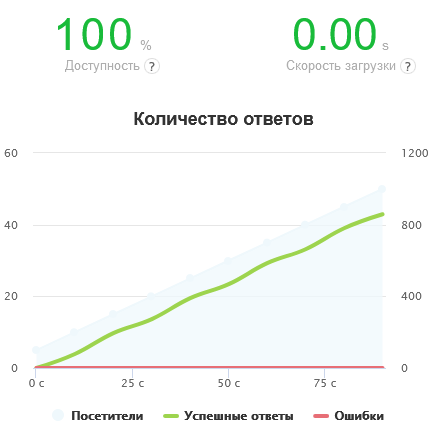
* *Используйте сжатие данных, чтобы уменьшить размер файлов, передаваемых по сети. Например, сжатие кода CSS и JavaScript может улучшить скорость загрузки страницы.*
* *Оптимизируйте изображения, уменьшая их размер и сжимая. Используйте форматы, которые обеспечивают лучшее соотношение качества и размера, например, форматы JPEG или PNG.*
* *Используйте контент-доставщики (Content Delivery Networks - CDN), которые распределяют статический контент на сервера, расположенные ближе к конечным пользователям, улучшая скорость загрузки и снижая задержки.*
* *Оптимизируйте базы данных и запросы к ним, чтобы ускорить доступ к данным и снизить задержки. • Включите кэширование, чтобы ускорить загрузку страниц и экономить пропускную способность сервера, храните часто запрашиваемые ресурсы на стороне пользователя.*
* *Используйте сборщики мусора, которые очищают память, используемую приложением, освобождая ресурсы и уменьшая задержки.*
* *Минимизируйте количество HTTP-запросов, которые отправляются, объединяйте ресурсы, чтобы уменьшить задержки.*
* *Тестируйте производительность приложений, чтобы выявить узкие места и определить, какие улучшения могут быть внесены для улучшения производительности.*

*Эти действия нам помогли оптимизировать сайт и ускорить его.*

## 2.2. Показатели работы веб-приложения и их анализ

*Привести основные показатели работы веб-приложения и описать способы их анализа. Проанализировать любой сайт из рассмотренных выше аналогов, сделать вывод*

Рисунок 7: тестирование на скорость

Рисунок 8: тестрования на запросы

## В ходе тестирования наш сайт оказался устойчивым и эффективным для пользователей. Мы рады, что наша работа приносит пользу и помогает людям. Если у вас есть какие-то вопросы или проблемы, не стесняйтесь обратиться к нам за помощью. Мы всегда готовы помочь вам и обеспечить качественный сервис.

## 2.3. Мониторинг работы веб-приложения

*Показатели веб-приложения показывают, что приложения нуждается в оптимизации, но в то же время поддерживает больше количество специальных возможностей, что способно расширить аудиторию ресурса*

## 2.4. Мероприятия по оптимизации.

*Провести рефакторинг кода веб-приложения. Настроить ключевые слова, файл robots.txt, мета-теги описания, заголовков. Рассчитать стоимость продвижения приложения в сети интернет, указав мероприятия и их стоимость.*

# Раздел 3. Обеспечение безопасности веб-приложений

## 3.1. Источники угроз безопасности

## 3.2. Методы защиты доступа к данным и защиты кода

## 3.3. Диагностика наличия вредоносных программ

## 3.4. Мероприятия по защите приложения

## 3.5. Выводы о безопасности

ВЫВОДЫ

Выводы о достижении целей практики (совершенствование профессиональных навыков и умений, расширение и систематизация знаний на основе изучения деятельности подразделения предприятия, приобретение практического опыта);

## СПИСОК ЛИТЕРАТУРЫ

использованной при составлении отчета (по необходимости).

# Приложение