Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.03 Программирование в компьютерных системах

**ОТЧЕТ**

**ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**(преддипломной)**

Выполнил:

обучающийся учебной группы № 4ПКС-120

М.А. Иванов

*(И.О. Фамилия)*

Проверил:

руководитель практики от колледжа:

И.В. Сибирев

*(И.О. Фамилия)*

**Москва**

**20­­­­24**

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc166664592)

[Глава 1 ПРЕДПРОЕКТНОЕ ИССЛЕДОВАНИЕ 5](#_Toc166664593)

[1.1 Описание предметной области 5](#_Toc166664594)

[1.2 Сравнительный анализ программ-аналогов 6](#_Toc166664595)

[1.3 Постановка задачи 7](#_Toc166664596)

[1.4 Характеристика инструментальных средств разработки 9](#_Toc166664597)

[Глава 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММЫ 10](#_Toc166664598)

[2.1 Анализ требований и определение спецификаций программного обеспечения 10](#_Toc166664599)

[2.2 Проектирование программного обеспечения 12](#_Toc166664600)

[2.3 Разработка программного обеспечения 15](#_Toc166664601)

[2.4 Тестирование программы 16](#_Toc166664602)

[ЗАКЛЮЧЕНИЕ 17](#_Toc166664603)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ИНТЕРНЕТ-РЕСУРСОВ 19](#_Toc166664604)

# **ВВЕДЕНИЕ**

В ходе выполнения практики разрабатывался индивидуальный проект по разработке информационной системы «Планировщик задач».

Современные возможности информационных технологий предоставляют уникальные перспективы для разработки высокоэффективных информационных систем. В рамках дипломного проекта предлагается создание информационной системы «Планировщик задач» с использованием фреймворка Django и языка программирования Python.

Цель проекта: повышение уровней организации рабочего процесса, эффективного управления временными ресурсами и повышении личной продуктивности.

Для выполнения целей проекта нужно выполнить следующие задачи:

- Разработать привлекательный и интуитивно понятный пользовательский интерфейс системы с применением технологии Django.

- Написать программный код, обеспечивающий эффективное управление логикой работы информационной системы, включая функции планирования и отслеживания задач.

- Создать механизм сохранения и загрузки данных о состоянии системы, включая сохранение планов, задач и другой важной информации для возможности последующего восстановления.

- Разработать механизм отображения результатов задач в отдельном окне для удобства пользователей.

Объектом исследования является веб-приложение.

Предметом исследования выступают технологии разработки информационной системы «Планировщик задач».

Основные методы исследования включают анализ литературных источников по теме проекта, изучение образовательных методик, а также сравнительный анализ существующих аналогичных информационных систем.

Актуальность проекта обосновывается не только растущей потребностью в эффективных инструментах управления задачами, но и важностью создания средств, способствующих повышению продуктивности и оптимизации рабочего процесса.

Основное функциональное предназначение разрабатываемого программного средства в рамках дипломного проекта заключается в предоставлении пользователям возможности эффективного планирования и управления задачами через инновационную информационную систему.

Для реализации данного проекта будут использованы инструментальные средства и языки программирования, такие как Python, Django, система управления базами данных (СУБД), diagrams.net. Дополнительно, для установки и визуализации будет задействован Actual Installer, а также инструменты Microsoft Word и Microsoft PowerPoint.

Таким образом, дипломный проект направлен на создание информационной системы «Планировщик задач», разработанной с использованием Django и Python. Цель проекта - предоставить пользователям увлекательный инструмент для эффективного управления своими задачами, способствуя осознанному развитию стремления к знаниям и достижению высоких результатов в учебе.

# **Глава 1 ПРЕДПРОЕКТНОЕ ИССЛЕДОВАНИЕ**

## **1.1 Описание предметной области**

Важной составной частью в современной информационной технологии и сфере управления временем является разработка информационной системы «Планировщик задач». Эта система создается с целью обеспечения эффективного управления задачами, предоставляя пользователям инструмент для оптимизации и структурирования их рабочих процессов.

В современном образовании и бизнесе становится все важнее использование технологий, которые не только упрощают задачи, но и обеспечивают более эффективное использование времени. Информационная система, разрабатываемая в рамках данного дипломного проекта на базе Django и Python, объединяет в себе функциональность управления задачами и образовательные элементы, предоставляя пользователю возможность не только эффективно организовывать свои дела, но и улучшать навыки планирования.

Основной целью является предоставление инструмента для эффективного планирования и управления задачами. Система предлагает пользователю функции создания задач, установки приоритетов, определения сроков и отслеживания выполнения. Важным элементом является также интеграция образовательных компонентов, предоставляющих пользователю советы по эффективному управлению временем и задачами.

Дизайн и визуальное оформление информационной системы играют ключевую роль в создании удобной и привлекательной среды для пользователя. Интерфейс должен быть интуитивно понятным и легким в использовании.

Центральным элементом разработки системы является база данных, которая должна быть тщательно структурирована и постоянно обновляться, чтобы обеспечить актуальность и разнообразие функционала системы. Разработка логики системы включает в себя определение правил

планирования, механизм сохранения данных о задачах и определение уровней сложности для повышения эффективности использования системы.

## **1.2 Сравнительный анализ программ-аналогов**

Прежде чем приступать к реализации, проведем анализ и сравним существующие приложения «Планировщик задач». Анализ существующих информационных систем для планирования задач в таблице 1:

Таблица 1. Существующие планировщики задач

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Todoist** | **Trello** | **Microsoft To-Do** | **Any.do** |
| Функционал |  |  |  |  |
| Отправка задач | + | + | + | + |
| Прикрепление файлов | + | + | - | + |
| Установка сроков | + | + | + | + |
| Приоритет задач | + | + | + | + |
| Группировка задач | + | + | + | + |
| Интерфейс |  |  |  |  |
| Наличие тем | + | + | + | + |
| Смена языка | + | + | + | + |
| Стоимость |  |  |  |  |
| Бесплатная установка | + | + | + | + |
| Платная подписка | + | + | + | + |
| Кроссплатформенность |  |  |  |  |
| Мобильное приложение | + | + | + | + |
| Десктоп приложение | + | + | + | + |
| Веб-версия приложения | + | + | + | + |
| Хранилище задач | Сервер | Облако | Облако | Сервер |
| Синхронизация между устройствами | + | + | + | + |
| Распределение задач по проектам | + | + | + | + |
| Оповещения и напоминания | + | + | + | + |

При разработке информационной системы "Планировщик задач" будут учтены лучшие практики и функциональности из рассматриваемых веб-приложений для эффективного планирования и управления задачами.

## **1.3 Постановка задачи**

Определение постановки задачи для разработки информационной системы «Планировщик задач» состоит в определении входной и выходной информации и других аспектов приложения.

Входной информацией для системы является: логин пользователя,

тематика задачи, детали задачи (название, описание, срок выполнения); уровень приоритета задачи.

Выходной информацией для пользователя является: количество выполненных задач, статистика по выполненным задачам (количество выполненных, количество невыполненных), время и дата выполнения каждой задачи.

Логика работы системы:

Цель системы - обеспечить эффективное планирование и управление задачами пользователя. Пользователь создает, отслеживает и завершает задачи в соответствии с их приоритетом и сроками выполнения.

Правила взаимодействия:

Пользователь создает задачу, указывая ее детали и приоритет.

Система предоставляет возможность отслеживания статуса выполнения задачи.

Пользователь может изменять статус задачи (выполнена, отложена, отменена).

Система предоставляет статистику выполненных и невыполненных задач.

Функциональные требования:

Обеспечить возможность создания и управления задачами.

Реализовать систему приоритетов для задач.

Иметь механизм отслеживания сроков выполнения задач.

Предоставить возможность просмотра и редактирования созданных задач.

Описание задач и их обработка:

Создание задачи: Пользователь должен иметь возможность создать новую задачу, указав ее детали и уровень приоритета.

Редактирование задачи: Пользователь может вносить изменения в уже созданные задачи (название, описание, сроки).

Отслеживание выполнения: Система должна отображать статус выполнения каждой задачи.

Уведомления: Система должна предоставлять пользователю уведомления о приближающихся сроках выполнения задач.

Оценка и статистика:

Разработать механизм оценки эффективности выполнения задач.

Позволить пользователю просматривать статистику по выполненным и невыполненным задачам.

Актуализация контента:

Регулярно обновлять систему новыми функциональности и улучшения.

Предоставить пользователям актуальные средства планирования.

Эксплуатационные требования:

Интерфейс пользователя: Разработать интуитивно понятный интерфейс для комфортного взаимодействия с системой.

Поддержка платформ: Обеспечить работу системы на операционной системе Windows 10.

Устойчивость и безопасность: Гарантировать стабильную работу и защиту от сбоев и ошибок.

Подключение к сети интернет: Обеспечить возможность подключения к сети для синхронизации данных.

Поддержка СУБД: Интегрировать систему управления базами данных (СУБД) для эффективного хранения и доступа к данным.

## **1.4 Характеристика инструментальных средств разработки**

Идет активное использование современных инструментальных средств разработки, обеспечивающих эффективное создание и функционирование информационных систем планировщиков задач.

Язык программирования Python: В разработке информационной системы используется язык программирования Python, выбранный в качестве основного для реализации бизнес-логики, взаимодействия с базой данных и обработки данных.

Фреймворк Django: используется фреймворк Django для построения веб-приложения. Django обеспечивает удобную архитектуру, встроенную административную панель, а также инструменты для разработки безопасных и масштабируемых веб-приложений.

Интегрированная среда разработки: для разработки системы применяется интегрированная среда разработки, такая как PyCharm или Visual Studio Code, предоставляющая удобные средства для написания, отладки и тестирования кода на языке Python.

Интерфейс пользователя: для построения пользовательского интерфейса системы используется HTML, CSS.

Система управления базами данных: Применяется СУБД, совместимая с Python и Django, для эффективного хранения, управления и извлечения данных. Например, PostgreSQL или MySQL могут быть использованы в качестве надежных и масштабируемых решений.

Инструменты для построения диаграмм и схем: для визуализации архитектуры системы, построения диаграмм баз данных и диаграммы взаимодействия используются инструменты вроде draw.io или аналогичные.

Программные продукты: В процессе разработки могут использоваться инструменты для создания инсталлятора (например, Inno Setup), для документации и презентаций (например, Microsoft Word, PowerPoint), а также для обработки графических изображений, если это необходимо.

# **Глава 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММЫ**

## **2.1 Анализ требований и определение спецификаций программного обеспечения**

На рисунке 1 представлена диаграмма вариантов использования «Планировщика задач»



Рисунок 1. Диаграмма вариантов использования «Планировщика задач»

Код:

|  |
| --- |
| @startuml  left to right direction  skinparam packageStyle rectangle  actor Пользователь as user  actor Администратор as admin  rectangle "'Планировщик задач'" {  user -- (Регистрация/Вход)  user -- (Создание задачи)  user -- (Просмотр задач)  user -- (Редактирование задачи)  user -- (Удаление задачи)  user -- (Отметить задачу как выполненную)  user -- (Добавление комментариев к задаче)    admin -- (Управление пользователями)  admin -- (Настройки системы)  }  @enduml |

На рисунке 2 представлена функциональная диаграмма верхнего уровня

«Планировщика задач»

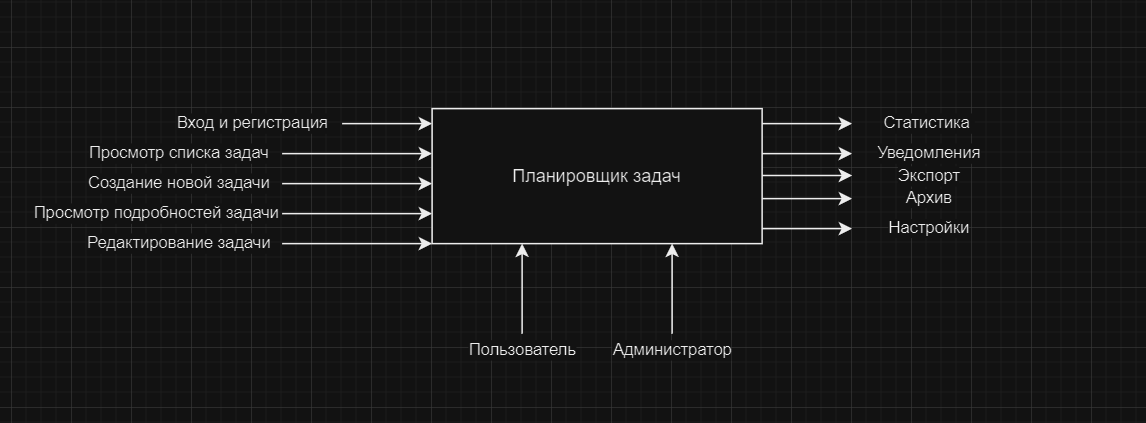


Рисунок 2. Функциональная диаграмма верхнего уровня

На рисунке 3 представлена ER-диаграмма

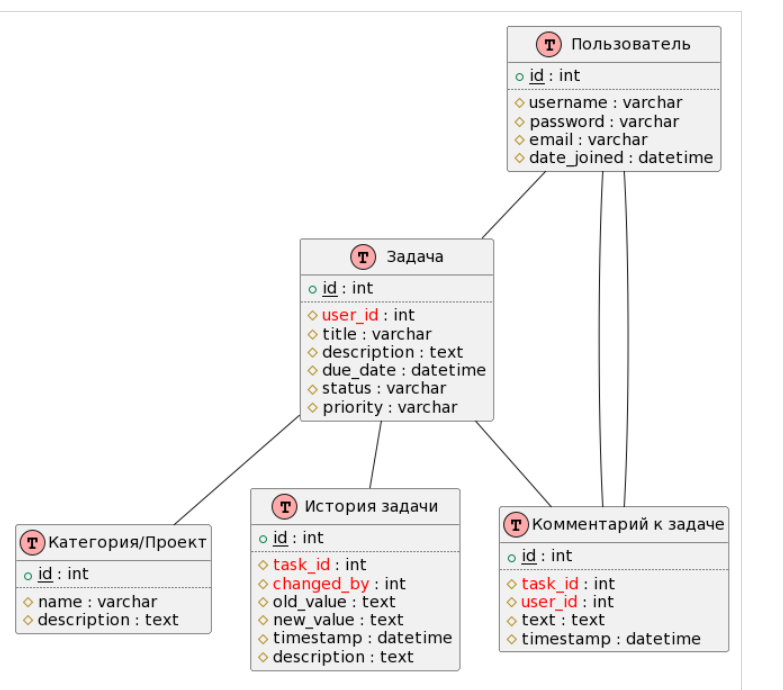


Рисунок 3. ER-диаграмма «Планировщика задач»

## **2.2 Проектирование программного обеспечения**

Для понимания устройства программы, необходимо построить две схемы.

Структурная схема программы применяется для иллюстрации устройства программы и взаимодействия её компонентов. Это средство делает процесс чтения и осмысления кода более простым, демонстрируя как осуществляется последовательность операций и какие связи существуют между различными блоками кода.

Функциональная схема программы служит для наглядного представления взаимодействий между разными функциями и подпрограммами, а также показывает, как происходит передача данных внутри программы.

На рисунке 4 представлена структурная схема «Планировщика задач»

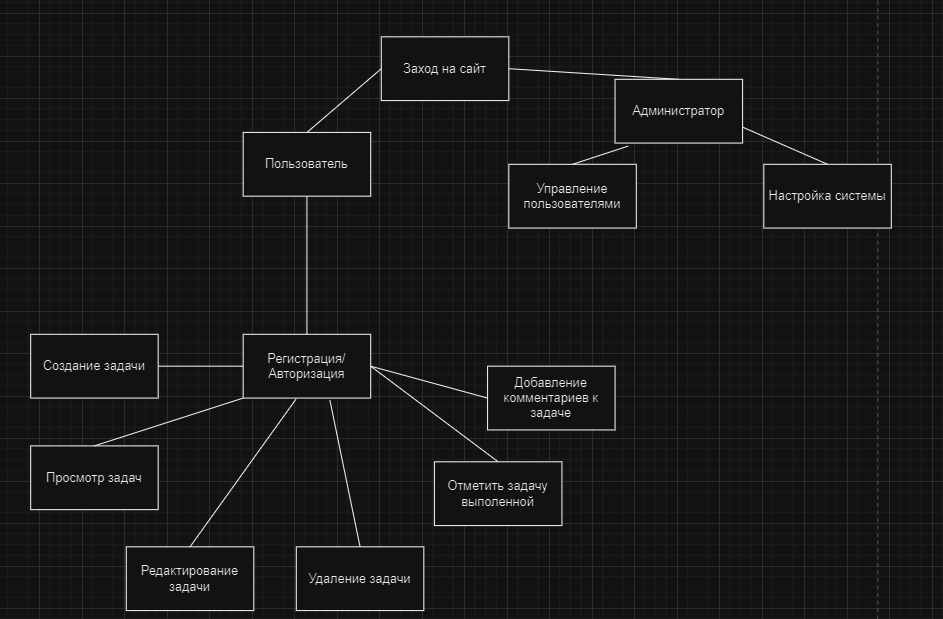


Рисунок 4. Структурная схема «Планировщика задач»

На рисунке 5 представлена функциональная схема «Планировщика задач»

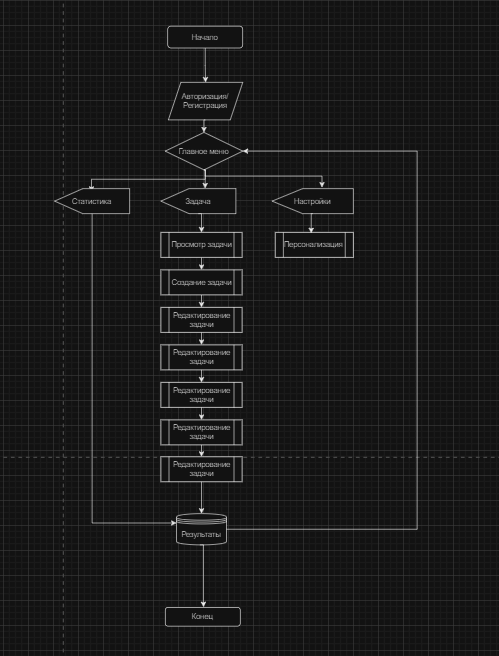
.

Рисунок 5. Функциональная схема «Планировщика задач»

## 2.3 Разработка программного обеспечения

Для создания приложения «Планировщика задач» будет использоваться язык программирования Python и фреймворк Django. Для начала мы запустим сайт и сможем посмотреть две вещи:

- панель управления для администратора.

На рисунке 6 представлена панель управления сайта «Планировщика задач*»*.

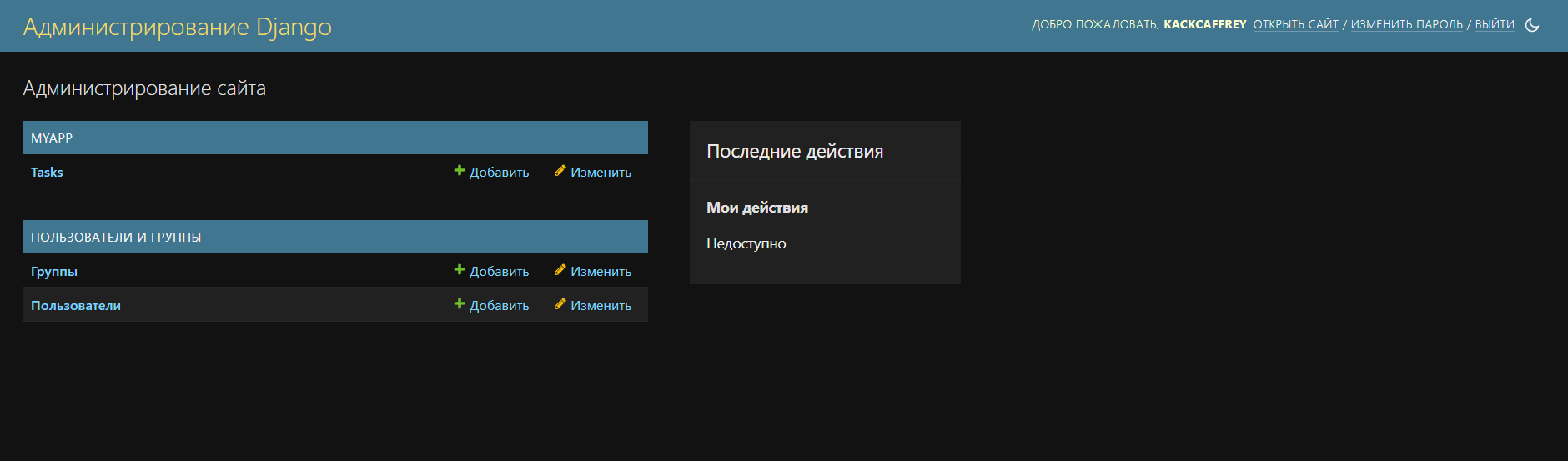


Рисунок 6. Панель управления «Планировщика задач»

- страничка авторизации аккаунта

На рисунке 7 представлена страничка авторизации аккаунта

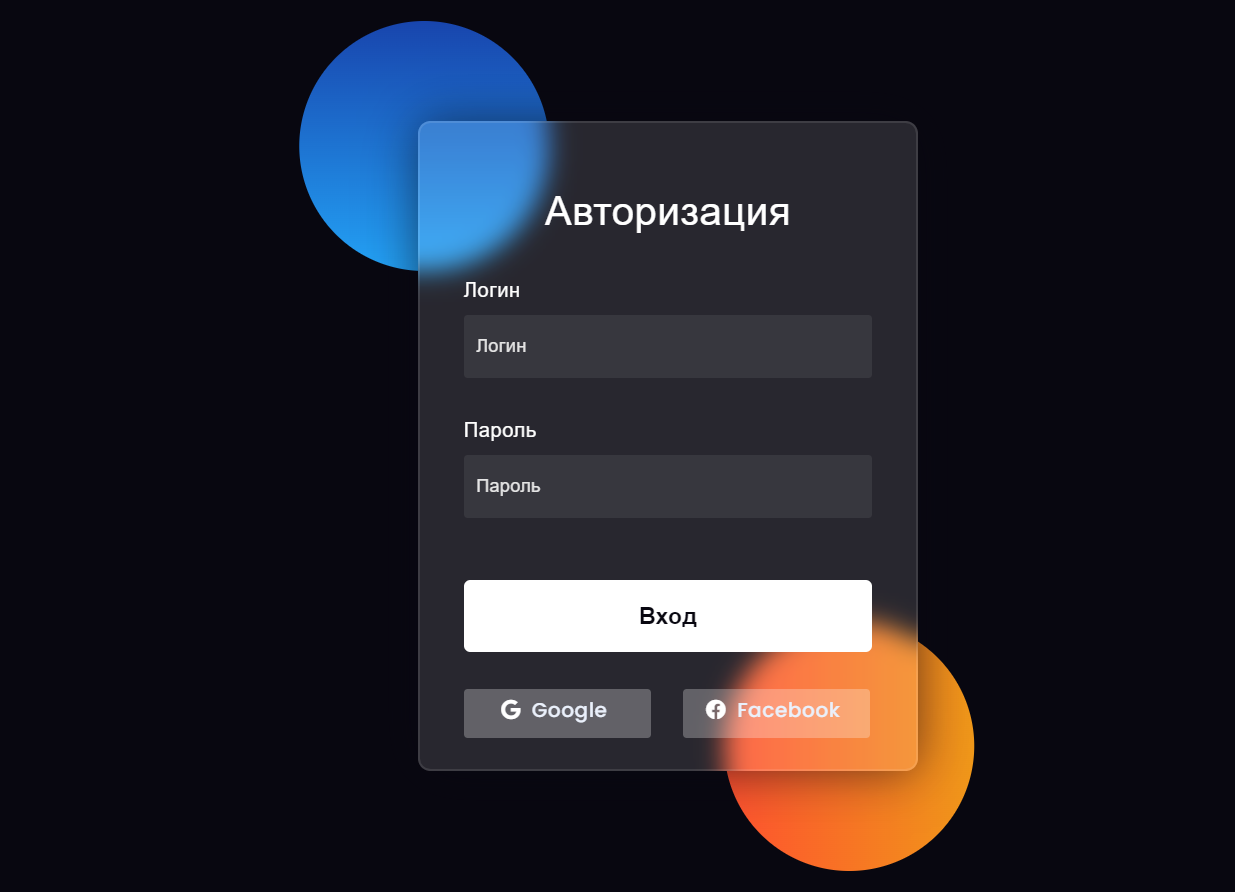


Рисунок 7. Страничка авторизации «Планировщика задач»

## 2.4 Тестирование программы

Для проверки работоспособности сайта, нужно его запустить и перейти в окно авторизации.

На рисунке 8 представлена начальная страничка сайта

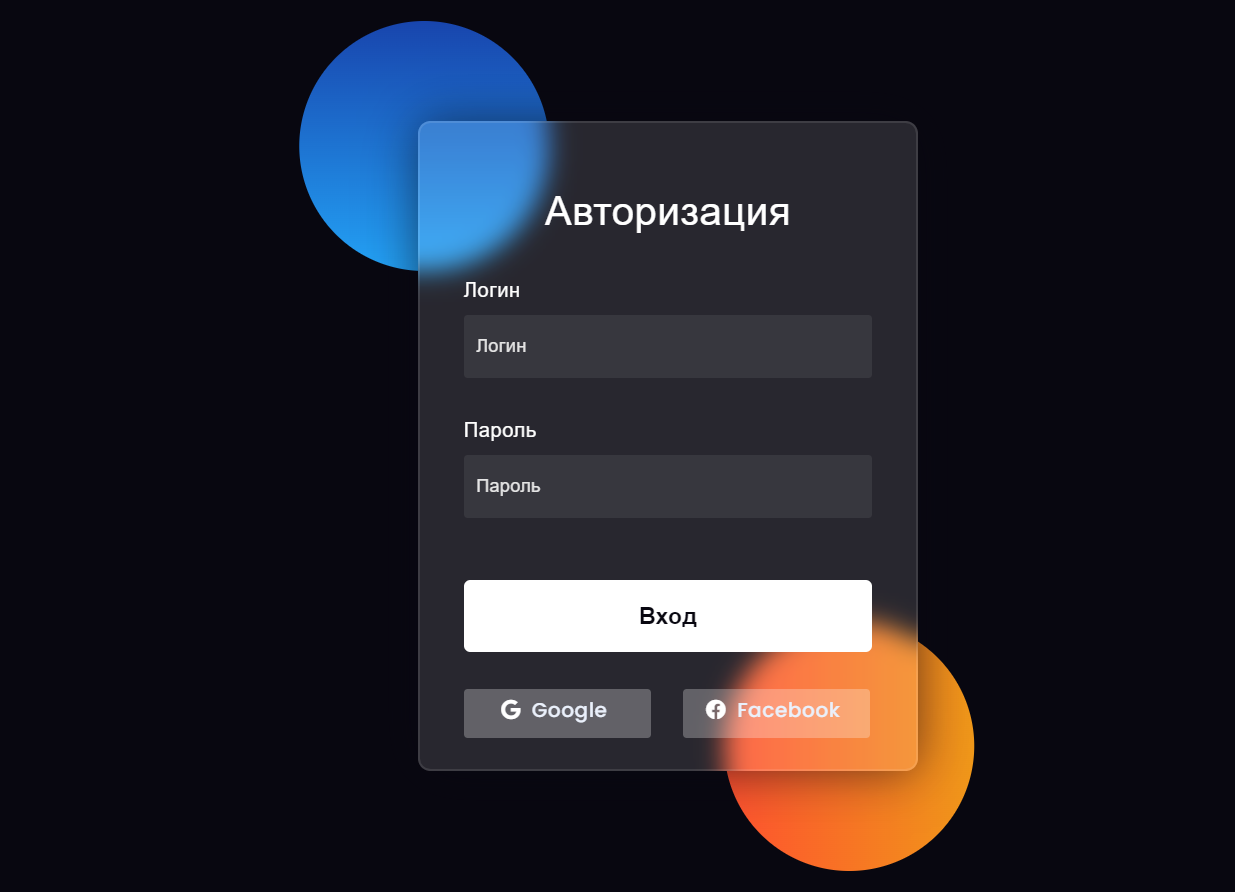


Рисунок 8. Начальная страничка «Планировщика задач»

# **ЗАКЛЮЧЕНИЕ**

В результате выполнения индивидуального проекта по разработке информационной системы «Планировщик задач»на основе фреймворка Django и языка программирования Python достигнуто успешное выполнение поставленных задач и целей. Проект направлен на создание эффективного инструмента для организации рабочего процесса, управления временными ресурсами и повышения личной продуктивности.

В ходе работы над проектом решены следующие задачи:

Разработка пользовательского интерфейса с использованием технологии Django. Проектирование и реализация привлекательного и интуитивно понятного интерфейса системы, обеспечивающего удобство использования для пользователей.

Написание эффективного программного кода. Создание логики работы информационной системы, включая функции планирования и отслеживания задач, с использованием языка программирования Python.

Механизм сохранения и загрузки данных. Разработка механизма сохранения и загрузки данных о состоянии системы, включая сохранение планов, задач и другой важной информации для последующего восстановления.

Отображение результатов задач. Реализация механизма отображения результатов задач в отдельном окне, обеспечивающего удобство взаимодействия и анализа для пользователей.

Объектом исследования стала информационная система "Планировщик задач", а предметом исследования — средства разработки, включая Django и Python. Исследование проведено с использованием методов анализа литературных источников, изучения образовательных методик и сравнительного анализа аналогичных информационных систем.

Актуальность проекта обоснована не только растущей потребностью в эффективных инструментах управления задачами, но и важностью создания средств, способствующих повышению продуктивности и оптимизации рабочего процесса.

Таким образом, дипломный проект по созданию информационной системы на Django и Python успешно реализован, предоставляя пользователям инновационный инструмент для эффективного управления задачами и содействуя развитию стремления к знаниям и достижению высоких результатов в учебе и работе.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ИНТЕРНЕТ-РЕСУРСОВ**

*Законодательные и нормативные акты:*

1. ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления – 92 с.
2. ГОСТ 7.32-2017 Отчет о научно-исследовательской работе. Структура и правила оформления – 47 с.
3. ГОСТ 7.82-2001 Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления – 39 с.
4. ГОСТ Р 7.0.100-2018 Библиографическая запись. Библиографическое описание. Общие требования и правила составления – 122 с.
5. ГОСТ Р 7.0.12-2011 Библиографическая запись. Сокращение слов и словосочетаний на русском языке. Общие требования и правила составления. – М.: ИПК Издательство стандартов, 2004. – 169 с.
6. ГОСТ Р 7.0.5-2008 Библиографическая ссылка. Общие требования и правила составления – 32 с.
7. Единая система программной документации. – М.: Стандартинформ, 2005. – 128 с.

*Учебная и научная литература:*

1. Агальцов, В.П. Математические методы в программировании: учебник / В.П. Агальцов. – 2-е изд., перераб. и доп. – Москва: ИД «ФОРУМ»: ИНФРА-М, 2023. – 240 с.
2. Мюллер Д. П. Си Шарп для чайников – Москва, Санкт-Петербург 2019 ООО "Диалектика" – Текст: электронный. – Режим доступа: <https://vk.com/doc161644846_634870821?hash=y8mbUO5UhKRlCR4tJkTnQsTg7BV4cQQhbNC6FkByWTP&dl=f4zdcFgzbAIpG5Qn0BCz1QINJJ49cQORMlAhF4bX8js> (дата обращения: 24.03.2024).

*Интернет документы:*

1. Аксенова Т.Г. Онлайн-курс по технологии разработки программного обеспечения [Электронный ресурс]. – Режим доступа:<https://classroom.google.com/u/2/w/NTc2MzExNTI0MTY0/t/all> (дата обращения: 24.03.2024).
2. Полное руководство по языку программирования Python и фреймворку Django. – [Электронный ресурс]. – Режим доступа: http://example.com/python-django-tutorial/ (дата обращения: 23.03.2024).
3. Ресурс для сравнительного анализа задачных менеджеров Any.do. [Электронный ресурс]. – Режим доступа: https://www.any.do/ (дата обращения: 23.03.2024).
4. Ресурс для сравнительного анализа задачных менеджеров Microsoft To-Do. [Электронный ресурс]. – Режим доступа: https://to-do.microsoft.com/ (дата обращения: 22.03.2024).
5. Ресурс для сравнительного анализа задачных менеджеров Todoist. [Электронный ресурс]. – Режим доступа: https://todoist.com/ (дата обращения: 21.03.2024).
6. Ресурс для сравнительного анализа задачных менеджеров Trello. [Электронный ресурс]. – Режим доступа: https://trello.com/ (дата обращения: 21.03.2024).
7. Руководство по работе в среде Visual Studio. – [Электронный ресурс]. – Режим доступа: https://docs.microsoft.com/ru-ru/visualstudio/ (дата обращения: 20.03.2024).
8. Форум программистов и сисадминов - [Электронный ресурс]. – Режим доступа: https://www.cyberforum.ru/unity/thread1230990.html (дата обращения: 21.03.2024).
9. Django Python Professional Modern Chat App UI Tutorial. [Электронный ресурс]. – Режим доступа: https://youtu.be/V9DkvcT27WI?si=IZqGPf56ZE-hPwxy (дата обращения: 21.03.2024).
10. Python Django UI | How to Design Chat Messaging App in Django. [Электронный ресурс]. – Режим доступа: https://youtu.be/tUSCm\_t6Ypw?si=6kq\_yLs2WYn6MAt (дата обращения: 21.03.2024).
11. Python Django UI Tutorials: 21 - Custom Dialog System Message Box Popup. [Электронный ресурс]. – Режим доступа: https://youtu.be/jrgT-fbV2tM?si=rpP3KNq5OeZyygz4 (дата обращения: 22.03.2024).