**Государственное бюджетное общеобразовательное учреждение города Москвы «Лицей «Вторая школа» имени В.Ф. Овчинникова»**

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ**

**ОЛИМПИАДА ШКОЛЬНИКОВ**

**Профиль «Информационные технологии»**

**Командный кейс №6 «Образовательный портал»**

Никитин Тимофей Андреевич 11А

Степанова Елена Александровна 11Б

Кравцов Николай Константинович 11В

Андреев Глеб Андреевич 11Г

Еронин Максим Владимирович 11Д

Руководитель:

Федоров Кирилл Евгеньевич,

Учитель информатики

Лицей «Вторая школа» имени В.Ф. Овчинникова

**Москва, 2025**

Оглавление

[**Обоснование выбора языка программирования и используемых программных средств** 2](#_Toc190095340)

[**Структурная схема проекта** 4](#_Toc190095341)

# **Обоснование выбора языка программирования и используемых программных средств**

**Язык программирования: Python**

Мы выбрали Python в качестве основного языка программирования для нашего проекта по следующим причинам:

1. **Простота и читаемость кода**: Python известен своей простотой и читаемостью, что позволяет быстро разрабатывать и поддерживать код. Это особенно важно для образовательных проектов, где код может быть использован и модифицирован различными разработчиками.

2. **Богатая экосистема библиотек и фреймворков**: Python имеет обширную экосистему библиотек и фреймворков, которые упрощают разработку веб-приложений. В частности, мы использовали Flask, легковесный веб-фреймворк, который предоставляет все необходимые инструменты для создания веб-приложений.

3. **Сообщество и поддержка**: Python имеет большое и активное сообщество разработчиков, что обеспечивает доступ к множеству ресурсов, документации и примеров кода. Это облегчает решение возникающих проблем и ускоряет процесс разработки.

**Веб-фреймворк: Flask**

Мы выбрали Flask в качестве веб-фреймворка для нашего проекта по следующим причинам:

1. **Легковесность и гибкость**: Flask является микрофреймворком, что означает, что он предоставляет только базовые функции, необходимые для создания веб-приложений. Это позволяет нам добавлять только те компоненты, которые действительно нужны, и сохранять приложение легковесным и быстрым.

2. **Простота интеграции**: Flask легко интегрируется с различными библиотеками и инструментами, такими как SQLAlchemy для работы с базами данных, Jinja2 для шаблонизации и WTForms для обработки форм. Это позволяет нам использовать лучшие инструменты для каждой задачи.

3. **Документация и сообщество**: Flask имеет отличную документацию и большое сообщество пользователей, что облегчает обучение и решение проблем, возникающих в процессе разработки.

**База данных: SQLite**

Мы выбрали SQLite в качестве базы данных для нашего проекта по следующим причинам:

1. **Простота использования**: SQLite не требует настройки сервера базы данных, что упрощает процесс разработки и развертывания. Это особенно важно для небольших проектов и образовательных целей.

2. **Легковесность**: SQLite является легковесной базой данных, которая хранит все данные в одном файле. Это делает её идеальной для небольших и средних проектов, где не требуется высокая производительность и масштабируемость.

3. **Поддержка Python**: SQLite поддерживается стандартной библиотекой Python, что упрощает интеграцию и использование в нашем проекте.

**Шаблонизатор: Jinja2**

Мы выбрали Jinja2 в качестве шаблонизатора для нашего проекта по следующим причинам:

1. **Простота и мощность**: Jinja2 предоставляет простой и интуитивно понятный синтаксис для создания HTML-шаблонов, а также поддерживает мощные функции, такие как наследование шаблонов и макросы.

2. **Интеграция с Flask**: Jinja2 является стандартным шаблонизатором для Flask, что обеспечивает бесшовную интеграцию и упрощает процесс разработки.

**Фронтенд: Bootstrap**

Мы выбрали Bootstrap в качестве CSS-фреймворка для нашего проекта по следующим причинам:

1. **Быстрая разработка**: Bootstrap предоставляет готовые компоненты и стили, которые позволяют быстро создавать красивые и отзывчивые пользовательские интерфейсы.

2. **Кроссбраузерная совместимость**: Bootstrap обеспечивает кроссбраузерную совместимость, что гарантирует корректное отображение нашего сайта на различных устройствах и браузерах.

3. **Документация и сообщество**: Bootstrap имеет отличную документацию и большое сообщество пользователей, что облегчает обучение и решение проблем, возникающих в процессе разработки.

**Заключение**

Выбор Python и Flask в качестве основных инструментов для разработки нашего проекта был обусловлен их простотой, гибкостью и мощной экосистемой. SQLite, Jinja2 и Bootstrap дополняют этот выбор, обеспечивая простоту использования, мощные возможности и быструю разработку. Эти инструменты позволяют нам создавать эффективное и удобное веб-приложение для подготовки к ЕГЭ.

# **Структурная схема проекта**

**1. Пользовательский интерфейс (UI)**

* **HTML-шаблоны (Jinja2):**
  + Используются для динамической генерации HTML-страниц на стороне сервера.
  + Взаимодействуют с Flask для передачи данных из базы данных или логики приложения.
* **CSS (Bootstrap):**
  + Обеспечивает стилизацию и адаптивный дизайн интерфейса.
  + Подключается как статический файл.
* **JavaScript:**
  + Добавляет интерактивность на стороне клиента (например, анимации, AJAX-запросы).
  + Может взаимодействовать с Flask через API.

**2. Веб-сервер (Flask)**

* **Основные функции:**
  + Обрабатывает HTTP-запросы (GET, POST и т.д.).
  + Выполняет бизнес-логику приложения.
  + Взаимодействует с базой данных и файловой системой.
* **Маршруты (Routes):**
  + Определяют, какие функции Flask вызываются при обращении к определённым URL.
  + Пример: /login, /upload, /dashboard.
* **Шаблонизация (Jinja2):**
  + Flask использует Jinja2 для рендеринга HTML-страниц с динамическими данными.

**3. База данных (SQLite)**

* **Хранение данных:**
  + Таблицы для хранения структурированных данных (например, пользователи, задания, результаты).
* **Взаимодействие с Flask:**
  + Мы сделали файл, который напрямую обращается к базе данных, и уже основное приложение взаимодействует с данными полученными с помощью файла db\_functions.py

**4. Файловая система**

* **Статические файлы:**
  + CSS, JavaScript, изображения и другие ресурсы, которые не изменяются динамически.
  + Хранятся в папке static/ и обслуживаются Flask.
* **Загружаемые файлы:**
  + Файлы, загружаемые пользователями (например, задания в формате Excel).
  + Обрабатываются Flask (загрузка, чтение, удаление).

Блок-схема работы основного алгоритма:

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Контент, сгенерированный ИИ, может содержать ошибки.

Визуализация структурной схемы:

Веб-сервер (Flask):

* Обработка запросов
* Взаимодействие с базой данных

Пользовательский интерфейс (UI):

* HTML (Jinja2)
* CSS (Bootstrap)
* JavaScript

База данных (SQLite):

* Хранение данных пользователей, заданий, групп и курсов

Файловая система:

* Статические файлы
* Загруженные файлы

**Схема базы данных**:

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Контент, сгенерированный ИИ, может содержать ошибки.

**Функциональная схема проекта**

Управление заданиями:

* Просмотр задания
* Фильтрация
* Решение и проверка

Регистрация и авторизация:

* Регистрация
* Вход
* Выход

Управление группами:

* Создание групп
* Управление группами
* Присоединение

Курсы и материалов:

* Просмотр курсов
* Загрузка файлов

Форум:

* Общение
* Обсуждение

[Репозиторий с программным кодом проекта](https://github.com/AnderPlay1/sdai_ege_rep/)

**1) Главная страница:**

**Изображение выглядит как текст, снимок экрана, Человеческое лицо, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.**

Главная страница отвечает за навигации на сайте, является стартовой и основной. Доступна всем пользователям и позволяет получить доступ к следующим страницам:

* Форум
* Банк заданий
* Варианты
* Список курсов
* Добавление задачи
* Авторизация / Группы пользователя
* Регистрация / Личный кабинет

**2) Варианты:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Контент, сгенерированный ИИ, может содержать ошибки.**

**3) Список курсов:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.**

**4) Добавление задачи:**

**Изображение выглядит как снимок экрана, текст, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**5) Авторизация:**

**Изображение выглядит как снимок экрана, текст, программное обеспечение, Операционная система

Контент, сгенерированный ИИ, может содержать ошибки.**

**6) Регистрация:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**7) Личный кабинет:**

**Изображение выглядит как текст, программное обеспечение, Значок на компьютере, веб-страница

Контент, сгенерированный ИИ, может содержать ошибки.**

**8) Страница задания:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**9) Мои группы:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**10) Войти в группу по коду:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**11) Страница варианта:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Контент, сгенерированный ИИ, может содержать ошибки.**

**12) Форум:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**13) Создание обсуждения:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**14) Обсуждение на форуме:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Контент, сгенерированный ИИ, может содержать ошибки.**

**15) Отображение группы:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Контент, сгенерированный ИИ, может содержать ошибки.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Контент, сгенерированный ИИ, может содержать ошибки.**

**16) Видео урок:**

**Изображение выглядит как текст, человек, снимок экрана, Человеческое лицо

Контент, сгенерированный ИИ, может содержать ошибки.**

**17) Теория к курсу:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.**