Олимпиада школьников «Шаг в будущее» по профилю

«Инженерное дело».

Регистрационный номер 56116.

Секция: Информационные технологии.

**Прикладной проект «Создание онлайн-платформы для дистанционной подготовки к экзаменам по математике».**

Выполнил: Чугайнов Арсений Денисович,

ученик 10А класса МБОУ Гимназии №19

им. Н.З. Поповичевой г. Липецка.

Научный руководитель: Ложков Кирилл Германович,

педагог дополнительного образования,

место работы: ГБУ ДО ЦДО ЛО.

2026 год

**Аннотация**

Математика – это очень сложная, но важная наука. В школе она является основным и обязательным предметом для изучения. Для поступления в ВУЗ необходимо сдать ЕГЭ по базовой или профильной математике, для поступления на СПО необходимо сдать ОГЭ по математике. Одним словом, без математики в наши дни - никуда. Для большинства школьников её изучение составляет огромную проблему. Многим даже недостаточно школьных уроков для того, чтобы полностью понять материал, поэтому они нанимают репетиторов, но не всегда удается найти хорошего преподавателя, который жил бы в шаговой доступности от дома. Те, кто занимается очно, тратят очень много времени и денег на дорогу до своих репетиторов.

Моя работа направлена на решение проблемы недостатка доступных и эффективных инструментов для осуществления дистанционного обучения математике с преподавателем. В ходе проекта разработана веб-платформа, которая позволяет ученикам решать математические задания на различные темы, представленные на сайте, с автоматической их проверкой. Также есть возможность решать домашние задания, которые в виде файлов прикрепляют ученикам учителя; а преподавателям - контролировать и анализировать учебный процесс, проверять и оценивать решения, которые им в ответ отправляют ученики. Система может генерировать задачи в реальном времени, прилагать к ним подробные решения и начислять за них баллы, которые отображаются в таблице успеваемости группы (это создает эффект соревнования между учениками).

Платформа реализована с использованием технологий Python, Flask, SQLite, SQL Alchemy, HTML и CSS.

Результатом работы является полностью функционирующая образовательная платформа, готовая к внедрению в учебный процесс, имеющая потенциал на дальнейшее развитие.

**Содержание**

Словарь терминов и сокращений…………………………………….4

Введение……………………………………………………………….5

1. Цели и задачи………………………………………………………….6
2. Обзор существующих решений………………………………………7
3. Технологический стек и обоснование выбора инструментов………8
4. Архитектура системы………………………………………………9-10
5. Процесс создания основных модулей………………………………..11
   1. Модуль аутентификации и управления пользователями………..11-12
   2. Модуль генерации и проверки встроенных математических задач...13
      1. Комплекс заданий………………………………………….13
      2. Логика математических заданий………………………14-16
      3. Разборы заданий………………………………………...16-17
      4. Возможные расширения модуля…………………………..17
   3. Модуль домашних заданий……………………………………………17
      1. Техническая реализация модуля………………………18-20
      2. Возможные расширения модуля………………………….20
   4. Модуль системы мониторинга решений………………………….20-21
      1. Техническая реализация модуля………………………21-23
      2. Возможные расширения модуля………………………….23
   5. Интерфейс пользователя…………………………………………..23-24
6. Преимущества и практическая значимость………………………….25
7. Заключение и дальнейшие перспективы……………………………..26

Список использованных источников…………………………………27

Приложение №1………………………………………………………..28

Приложение №2…………………………………………………….29-35

**Словарь терминов и сокращений.**

**БД** – база данных

**ФИПИ** – Федеральный институт педагогических измерений

**Фреймворк** – программная платформа, определяющая структуру программной системы и предоставляющая набор компонентов и инструментов для решения типовых задач разработки. Проще готовя, каркас проекта.

**ЕГЭ** – единый государственный экзамен

**ОГЭ** – основной государственный экзамен

**СПО** – среднее профессиональное образование

**КИМ** – контрольно–измерительные материалы (экзаменационные бланки с заданиями)

**ООП** – объекто-ориентированное программирование.

**ORM** – технология программирования, которая обеспечивает взаимодействия между ООП и реляционными БД

**Лэндинг-страница** – (главная, гостевая) автономная веб-страница, созданная специально для ознакомления пользователя с платформой.

**Аутентификация** – процесс проверки подлинности пользователя или системы, направленный на подтверждение их права доступа к определенным ресурсам или данным.

**Репозиторий** – место, где хранятся некие цифровые данные.

**AI** – искусственный интеллект.

**Дедлайн** – срок выполнения задачи

**Id** – уникальный идентификационный номер некого объекта.

**Введение**

Современное образование активно переходит в цифровую среду, что особенно актуально в условиях дистанционного обучения. Подготовка к экзаменам по математике требует регулярной практики решения задач и контроля со стороны преподавателя. Существующие образовательные платформы часто ограничены в функционале, не предоставляют подробных разборов решений или не обеспечивают постоянную связь между учениками и учителями.

Актуальность данной работы обусловлена необходимостью создания специализированного инструмента, который объединяет в себе положительные стороны своих аналогов воедино. Предлагаемая платформа позволяет ученикам решать, как встроенные на сайте математические задачи с автоматической генерацией и проверкой, так и дополнительные задания от преподавателя; получать мгновенную обратную связь и видеть свой прогресс относительно одногруппников. Учителя получают возможность отслеживать успеваемость и активность каждого ученика, создавать домашние задания и анализировать историю решений группы.

1. **Цель и задачи**

**Цель проекта:** Создание веб-платформы для осуществления дистанционного обучения математике под контролем преподавателя со следующими возможностями: регистрация с выбором роли, генерация, проверка и оценка решения задач, управление группами, технология домашних заданий и аналитика учителей через историю решений и таблицу успеваемости.

**Задачи проекта:**

1. Проанализировать существующие аналоги и определить их недостатки и положительные моменты.
2. Разработать архитектуру системы с разделением ролей.
3. Реализовать модуль генерации математических задач различных типов.
4. Создать систему автоматической проверки ответов и начисления баллов в случае правильного ответа.
5. Разработать модуль управления группами с возможностью приглашения и удаления участников.
6. Реализовать систему домашних заданий с загрузкой файлов.
7. Создать интерфейс аналитики для учителей с фильтрацией решений по фамилии, имени, отчеству, дате.
8. Сформулировать преимущества и практическую значимость платформы.
9. **Обзор существующих решений**

Ниже приведены примеры образовательных платформ, а также их плюсы и недостатки.

1. **Яндекс Репетитор.** Включает бесплатные краткие учебные курсы для учащихся старшей школы, помогает освоить основные школьные дисциплины и подготовиться к итоговым экзаменам. Плюсы: бесплатное обучение, возможность проведения дистанционных занятий. Недостатки: отсутствие прямого контакта с преподавателем, и как следствие, необходимость высокой самоорганизации.
2. **Сайт ФИПИ.** Содержит открытый банк заданий ЕГЭ, открытые варианты КИМ, демоверсии, спецификации, кодификаторы. Плюсы: помогает получить практику решения задач, так как они там представлены в огромных количествах на все возможные темы. Недостатки: на платформе нет абсолютно никакой системы взаимодействия учеников и преподавателей, ресурс не подразумевает подготовку к экзаменам под контролем и поддержкой учителя, изучение материала исключительно самостоятельное, требует высокой самоорганизации.
3. **Платформа «Учи.ру».** Образовательная платформа для школьников с 1 по 11 класс. Плюсы: система анализирует результаты и показывает, по какой теме у учеников возникают сложности, дает дополнительные задания по этим темам. Минусы: доступ не является полностью бесплатным (в день дается 20 бесплатных попыток на решение задач, не по всем предметам), для того чтобы снять лимиты и ограничения родитель должен зарегистрироваться на платформе и купить своему ребенку не дешевую платную подписку.

Как итог стоит отметить, что у существующих образовательных ресурсов есть свои положительные и отрицательные стороны. В своем проекте я попытался объединить отдельные плюсы этих платформ и решить их недостатки.

1. **Технологический стек и обоснование выбора инструментов**

Изображение выглядит как текст, снимок экрана, логотип, Бренд

Содержимое, созданное искусственным интеллектом, может быть неверным.

Для технической реализации платформы выбран следующий стек.

1. Python – высокоуровневый язык программирования с богатыми набором различных библиотек, инструментов и фреймворков.
2. Flask – легковесный веб-фреймворк, позволяющий быстро разрабатывать функциональные веб-приложения. Выбран за свою простоту и гибкость.
3. SQLite – встраиваемая БД, не требующая отдельного сервера, идеальна для прототипирования и небольших проектов.
4. SQL Alchemy – ORM для работы с БД на уровне объектов Python в ООП. Выбран, так как позволяет удобно работать с данными.
5. WTF-Forms – библиотека для создания и валидации форм в Flask.
6. HTML/CSS – для создания пользовательского интерфейса.
7. Встроенные модули Python.
8. Среда разработки – IDE PyCharm 2025.
9. Контроль версий – GIT hub.
10. **Архитектура системы**

Система построена по классической трехзвенной архитектуре, называемой MVC.

M – model. Модель – это основная логика приложения, отвечает за данные, методы работы с ними и структуру программы. Она служит связующим звеном между представлением и контроллером.

V – view. Представление – визуализация информации, которая поступает от модели и контроллера. Он отображает данные на уровне пользовательского интерфейса.

C – controller. Контроллер обеспечивает взаимодействие с системой: обрабатывает запросы пользователя, проверяет полученную информацию и передает ее модели.

Данная архитектура идеально подходит для образовательного веб-приложения среднего масштаба, балансируя между простотой разработки, производительностью и возможностями расширения.

Далее представлена статистика (рис.1) и визуализация (рис.2) архитектуры моего проекта.

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 1. Статистика объема строк кода в проекте

Изображение выглядит как текст, снимок экрана, Самоклеющийся листок, диаграмма

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 2. Визуализация архитектуры проекта.

1. **Процесс создания основных модулей**

**5.1 Модуль аутентификации и управления пользователями.**

Как и на большинстве веб-сайтов, первым делом пользователь попадает на лэндинг-страницу (приложение №2 рис. 1), на которой представлены все преимущества платформы и ее предлагаемые сервисы, а также есть возможность войти в учетную запись или создать ее.

В моем проекте создана следующая система регистрации. Пользователь заполняет форму (приложение №2 рис. 2): ФИО, email, пароль; а также отмечает или пропускает галочку («Вы учитель?»). Если он нажимает эту галочку, то пользователь получает роль Учитель на платформе, если пропускает – роль Ученик.

В зависимости от выбранной роли пользователь получает доступ к разным возможностям на платформе.

**Учитель:**

1. Создает группы
2. Приглашает учеников по уникальной ссылке (модуль uuid)
3. Прикрепляет и проверяет домашние задания
4. Анализирует успеваемость при помощи таблицы и истории решений группы (приложение №2 рис. 6 и 14).

**Ученик:**

1. Вступает в группы
2. Решает встроенный задания и получает за них баллы (приложение №2 рис. 7)
3. Прикрепляет в виде файлов свои решения
4. Видит свои результаты в таблице успеваемости, сравнивает их с результатами своих одногруппников (приложение №2 рис. 5).

Для работы с данными были созданы две таблицы БД первая - информация о пользователях (users), вторая - о группах (groups). Для того чтобы каждый пользователь мог состоять в нескольких группах сразу, была сформирована третья таблица (group\_members), в которой при помощи внешних ключей (group\_id, user\_id) была установлена связь между id пользователя и id группы.

Ниже показана схема визуализация параметров, свойств и взаимодействий этих трех таблиц.

Изображение выглядит как текст, диаграмма, снимок экрана, число

Содержимое, созданное искусственным интеллектом, может быть неверным.

UQ – уникальный

NOT\_NULL–

не пустой

FK – внешний ключ

PK – первичный ключ

Работа с БД была реализована при помощи библиотеки SQL Alchemy и ORM-моделей. Она позволила работать с данными как с объектами Python, это очень удобно и сильно упрощает данную задачу.

На реализацию данного модуля у меня ушло около 2 недель.

**5.2 Модуль генерации и проверки встроенных математических задач.**

Для создания данного модуля все технологии, реализующие встроенные задания, были помещены в отдельный скрипт maths. Данный скрипт составляет класс MyMath, его методы отвечают за генерацию, решение и проверку всех типов заданий. При инициализации сервер создает экземпляр этого класса и дальше использует его методы до тех пор, пока сайт не закроется. Это более оптимально, чем использовать набор отдельных функций, так как в таком случае приходилось бы каждый раз заново инициализировать каждую вызванную функцию. Логика, отвечающая за генерацию разборов заданий, реализована в скрипте configs.

На создание данного модуля ушло около 3 недель, ниже предлагаю подробнее рассмотреть процесс его реализации.

* + 1. **Комплекс заданий**

На сайте представлен набор заданий по математике, предназначенных для учеников как младшей, так и старшей школы.

**Задания:**

1. Линейные уравнения
2. Квадратные уравнения
3. Биквадратные уравнения
4. Иррациональные уравнения
5. Уравнения с модулем
6. Линейные неравенства на метод интервалов
7. Примеры на арифметические операции (3 уровней сложности)

Данный комплекс задач дает умение решать уравнения разных типов, объясняет непростые математические тонкости и принципы решения на простых примерах, предоставляет фундаментальные знания школьной алгебры.

* + 1. **Логика математических заданий**

Архитектура класса математических задач построена таким образом, чтобы каждый тип задания имел единую структуру методов, которая отвечает за его реализацию, это: метод генерации, решения, проверки, выявления коэффициентов, индивидуальный набор вспомогательных функций, исходя из этого, предлагаю разобрать все технологии на примере одного типа задания – Квадратное уравнение.

Замечание. На сервере для хранения задания, которое пользователь решает на данный момент, задействованы временные файлы cookie (модуль requests). Это нужно для того, чтобы корректно обрабатывать обновления страницы с заданием.

**Метод генерации.**

Для генерации любого уравнения в системе есть заранее заложенный шаблон (см изображение ниже).

Изображение выглядит как текст, снимок экрана, Шрифт, алгебра

Содержимое, созданное искусственным интеллектом, может быть неверным.

Разберем данную технологию. Квадратное уравнение - ax² + bx + c = 0. Генерируются случайные целые числа при помощи библиотеки random, но они могут быть отрицательными или равными 1, если в чистом виде их подставить в шаблон, то получится что-то на подобии: 1x² + -3x + 5 = 0. Эта запись некорректна, так как не принято писать минус после плюса и единицу перед x. Поэтому каждый коэффициент проходит этап форматирования, где он сравнивается с 0 и 1, после чего в зависимости от результатов сравнения встраивается в общую конструкцию уже корректно (1 – перед x не пишется, -1 – перед x остается только минус, к отрицательным числам прибавляется конструкция: <пробел>-<пробел>, к положительным: <пробел>+<пробел>). После такого форматирования получается корректная привычная запись: x² - 3x + 5 = 0, которую и видит пользователь на экране (приложение №2 рис. 8).

**Метод решения.**

Так как каждое задание имеет фиксированный шаблон, к каждому из них можно написать свой алгоритм решения, что и было реализовано:

1. Линейное уравнение – перенос коэффициентов с x в одну сторону, без – в другую.
2. Квадратное уравнение – дискриминант.
3. Биквадратное уравнение – замена x² и решение квадратного уравнения.
4. Иррациональное уравнение и уравнение с модулем – ограничение на правую часть и возведение обеих частей в квадрат.
5. Линейное неравенство – метод интервалов.

Разберем данную технологию на Квадратном уравнении. Принцип решения квадратного уравнения через дискриминант:

Изображение выглядит как текст, Шрифт, рукописный текст, число

Содержимое, созданное искусственным интеллектом, может быть неверным.

A, b, c – коэффициенты уравнения

Чтобы воспользоваться одной из этих формул необходимо найти коэффициенты и тогда задача будет решена.

**Метод выявления коэффициентов.**

Шаблонность заданий позволяет также написать для каждого типа уравнения паттерн регулярных выражений и при помощи библиотеки re вытаскивать из уравнений коэффициенты и подставлять их в формулы и алгоритмы решений. Эти паттерны хранятся в словаре, находящемся в конструкторе математического класса.

**Метод проверки.**

После того как пользователь решил предложенное ему уравнение, ввел ответ в текстовое поле и нажал кнопку «Ответить», сервер вызывает метод проверки. Сначала система вызывает метод решения соответствующего задания, тот при помощи метода выявления коэффициентов решает переданное ему уравнение, возвращает правильный ответ, и затем метод проверки сравнивает его с ответом пользователя (примечание: т.к. ответы пользователя и системы сравниваются как строки, оформление ответа **важно**, на сайте по этому поводу представлены подробные инструкции). Далее в зависимости от правильности ответа, пользователю либо начисляется, либо не начисляется балл. В случае ошибки предлагается проверить расчёты и попробовать снова.

* + 1. **Разборы заданий**

Я также предусмотрел возможность посмотреть подробный разбор любого уравнения (приложение №2 рис. 9). Стоит отметить, что разборы, представленные на сайте, для некоторых типов заданий, показывают нестандартные пути решения, которые позволяют выполнить их быстрее и с меньшим шансом допустить ошибку. В пример можно привести разбор уравнения с модулем.

**Стандартный метод. Метод, представленный на сайте.**

Изображение выглядит как текст, Шрифт, рукописный текст, белый

Содержимое, созданное искусственным интеллектом, может быть неверным.Изображение выглядит как Шрифт, белый, дизайн, типография

Содержимое, созданное искусственным интеллектом, может быть неверным.

Еще один важный момент – как проходит сам процесс генерации решения. В системе заложены заранее продуманные шаблоны, по которым будет строится решение, для каждого типа задания отдельно. Когда на сервер поступает запрос на получение разбора задачи, программа при помощи словаря конфигураций подбирает подходящий шаблон и просто подставляет в него числа для конкретного уравнения.

**5.2.4 Возможные расширения**

В будущем для расширения потенциала платформы можно будет добавить контрольные тесты с таймером, включить новые типы уравнений в комплекс заданий.

**5.3 Модуль домашних заданий.**

Комплекс заданий, представленный на сайте, охватывает достаточно узкий спектр тем и дает понимание лишь базовых фундаментальных вещей. Поэтому я решил добавить пользователям с ролью Учитель возможность прикреплять собственные задания (приложение №2 рис. 11) в виде файлов (работа с файлами реализована при помощи модуля os), а Ученикам – возможность отправлять в ответ свои решения этих заданий (приложение №2 рис. 10). Преподаватель может просматривать их (приложение №2 рис. 12), писать комментарии, оценивать баллами вручную (приложение №2 рис. 13). Созданный модуль дает возможность не просто самостоятельно получать базовые знания по предмету, а полноценно готовиться к любым экзаменам вместе со своим школьным учителем или репетитором, а также постоянно быть с ним на связи, не выходя из дома. На разработку и внедрение в платформу по данному направлению ушло около 3 недель.

**5.3.1 Техническая реализация модуля.**

Для того, чтобы реализовать такую технологию, были добавлены еще три таблицы в БД с помощью ORM-моделей: (homeworks – домашнее задание, homework\_solutions – список решений, связующая таблица, homework\_solutions\_files – файлы с решениями). Далее были установлены связи между таблицами groups и users (смотри раздел 5.1) с таблицей homeworks (по внешним ключам group\_id, user\_id), чтобы домашнее задание просматривалось только в той группе, в которой состоит учитель, выложивший его. Далее установлена связь между homeworks и homework\_solutions (по внешнему ключу homework\_id), а также между homework\_solutions и homework\_solutions\_files (по внешнему ключу solution\_id). Таким образом, в каждой группе может быть список домашних заданий, к каждому из них может быть прикреплен список решений (приложение №2 рис. 12), который отправили ученики, а к каждому решению – файл.

Далее представлена подробный визуализация связей и сигнатур описанных таблиц.

Изображение выглядит как диаграмма, текст, План, Технический чертеж

Содержимое, созданное искусственным интеллектом, может быть неверным.

Для того чтобы освобождать память сервера, от не нужных файлов была внедрена технология каскадного удаления:

homework\_solution\_files (файлы с решениями) удаляются при удалении homework\_solutions (списка решений),

homework\_solutions (список решений) удаляется при удалении homeworks (домашнего задания).

* + 1. **Возможные расширения модуля.**

В будущем для расширения потенциала платформы можно будет добавить дедлайны выполнения домашних заданий и уведомления об отправке пользователями платформы файлов (заданий, ответов).

**5.4 Модуль системы мониторинга решений.**

Встроенные задания проверяются системой автоматически, вернее сказать: система не проверяет решение, а только окончательный ответ. Этот подход не совсем корректен, ведь большинство заданий, например линейное неравенство или иррациональное уравнение – это задачи, требующие на экзаменах развернутого ответа и оформления. Поэтому я решил добавить возможность Ученикам: прикреплять к сгенерированным заданиям файлы с решениями, а Учителям просматривать их. Для этого создана отдельная вкладка «История решений» (приложение №2 рис. 14). Благодаря этому учителя могут получать более полную информацию об активности своих учащихся.

В «Истории решений» Учитель может искать решения при помощи умной системы поиска (приложение №2 рис. 15) по следующим параметрам: Имя, Фамилия, Отчество, дата отправки (работа с датами реализована при помощи библиотеки datetime). Нет необходимости заполнять все данные, поисковая система будет выдавать список решений, основываясь на том, что введет пользователь в форме (только фамилию, или только имя, или отчество, или дату отправки; а возможно их комбинации). Данный модуль был разработан мною в течение двух недель.

* + 1. **Техническая реализация.**

Для создания данной системы было добавлено еще две таблицы БД (solutions, solutions\_files) и несколько взаимосвязей между имеющимися и вновь созданными таблицами (Смотри приложение №1). Данная структура работает по похожему принципу, что и модуль Домашних заданий, описанный в разделе 5.3. Список решений (solutions, связующая таблица) привязан к группе (по внешнему ключу group\_id) и к пользователю (по внешнему ключу user\_id), который его отправил, а к каждому отдельному решению прикреплены файлы (solution\_files; по внешнему ключу solution\_id).

Далее представлена более подробная визуализация описанной работы с данными.

Изображение выглядит как текст, диаграмма, линия, План

Содержимое, созданное искусственным интеллектом, может быть неверным.

Примечание. Полную визуализацию структуры БД смотри в Приложении №1.

Для того, чтобы освободить память сервера от отработанной информации реализовано каскадное удаление: solution\_files (файлы с решениями) удаляются при удалении solutions (списка решений).

* + 1. **Возможные расширения модуля.**

В будущем для расширения потенциала платформы можно будет добавить уведомления для Учителя об отправке Учениками новых решений.

* 1. **Интерфейс пользователя.**

Создавая ранее описанные модули также составлялись html-страницы и wtf-формы, таким образом создание интерфейса пользователя продолжалось в течение всего периода создания платформы.

Платформа использует минималистичный и функциональный дизайн, ориентированный на удобство восприятия образовательного контента.

**Основные преимущества:**

чистота и простота – нет отвлекающих цветов и эффектов, фокус на содержании;

навигационная иерархия – интуитивно понятные пути для разных ролей;

подтверждение опасных действий – модальные окна при удалении пользователя из группы или очистки истории решений;

скрытие навигации при решении задания – чтобы не отвлекать ученика лишним интерфейсом.

Для отображения страниц было написано более 20 HTML-шаблонов.

**Ключевые страницы:**

Base.html – базовый каркас, подключение CSS

Landing.html - главная

Login.html, register.html - аутентификация

Profile.html, edit\_profile.html - профиль

Groups.html, group\_details.html – управление группами

Task\_window.html, task\_opened.html – решение задач

Group\_homeworks.html – список домашних заданий группы

Group\_solutions.html – история решений группы

**Интересные технологии:**

анимации при наведении на кнопку.

1. **Преимущества и практическая значимость**

**Преимущества платформы:**

1. Постоянная обратная связь с преподавателем
2. Геймификация обучения – система баллов и таблица успеваемости
3. Удобный мониторинг успеваемости – учитель отслеживает активность учеников через историю решений, таблицу и решения домашних заданий
4. Подробные разборы заданий с нестандартными, более оптимальными путями решения, нежели стандартные методы.

**Практическая значимость:**

Платформа может быть внедрена в школах для дистанционной подготовки к ОГЭ и ЕГЭ по математике (и не только), а также для дополнительных занятий с репетиторами. Она сокращает время проверки шаблонный заданий учителем и концентрирует его внимание на дополнительных собственных заданиях. Создает между учениками соревновательный эффект, ведь каждый хочет стать обладателем самого высокого балла в группе, тем самым мотивирует их к продолжению обучения.

1. **Заключение и дальнейшие перспективы**

В ходе работы была успешно создана онлайн-платформа для дистанционной подготовки к экзаменам по математике. Система включает все заявленные функции: регистрацию с ролями, генерацию и проверку задач, управление группами, систему баллов, технологию домашних заданий и аналитику учителей через историю решений и таблицу успеваемости.

**Дальнейшие перспективы развития:**

1. Расширение комплекса встроенных заданий (тригонометрия, логарифмы, задачи с параметрами, геометрия);
2. Разработка мобильного приложения;
3. Интеграция с системами школьного электронного журнала;
4. Добавление возможности проведения онлайн-тестирований с таймером;
5. Внедрение AI-ассистентов для учителей.

Итог: Платформа представляет собой готовое решение для внедрения в образовательный процесс и имеет потенциал для масштабирования и дальнейшего развития, а также для конкуренции с другими подобными проектами.

**Сам проект можно загрузить из публичного git-hub репозитория по ссылке:** [**https://github.com/Manex0990/profmat**](https://github.com/Manex0990/profmat)**. Инструкция по запуску и информация для тестирования находится в файле README.**

**Список использованных источников**

1. Документация Flask – URL: <https://flask.palletsprojects.com/>
2. Документация SQL Alchemy – URL:  <https://www.sqlalchemy.org/>
3. Официальный сайт PyCharm – URL: <https://www.jetbrains.com/pycharm/>
4. Сайт ФИПИ – URL: <https://fipi.ru/>
5. Сайт Учи ру – URL: <https://uchi.ru/>
6. Сайт Яндекс Репетитор – URL: <https://education.yandex.ru/ege/go/math>
7. Miro Интерактивная доска (для создания схем) – URL: <https://miro.com/app/dashboard/>
8. Официальный сайт олимпиады Шаг в Будущее – URL: <https://olymp.bmstu.ru/>

**Приложение №1. Схема сигнатур таблиц и зависимостей между ними представлена ниже.**

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Приложение №2. Демонстрация.**

**Рис 1. Лэндинг страница (фрагмент).**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Рис 2. Форма регистрации.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 3. Профиль.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 4. Список групп (учитель).**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 5. Таблица успеваемости глазами ученика.**

Изображение выглядит как программное обеспечение, снимок экрана, Операционная система, Мультимедийное программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Рис 6. Управление группой учителем.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис7. Выбор встроенных заданий.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 8. Страница задания на примере Иррационального уравнения.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 9. Подробный разбор на примере Квадратного уравнения.**

**Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 10. Список домашних заданий глазами ученика.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 11. Список домашних заданий глазами учителями.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 12. Список решений домашнего задания.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 13. Оценка решения домашнего задания.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Рис 14. История решений.**

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, Операционная система

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Рис 15. Поиск по решениям.**

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

Содержимое, созданное искусственным интеллектом, может быть неверным.**