МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Кафедра информационных технологий**

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ №7**

**по дисциплине**

**«ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

Выполнила студентка группы 35/2 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. С. Паничева

Отчет принял доц. каф. ИТ А. Н. Полетайкин

Краснодар

2024

Лабораторная работа №7

Тема: Разработка программного обеспечения ПС

Цель: Приобретение навыков проектирования и разработки программной

системы при помощи современных инструментальных средств.

1. В соответствии с требованиями технического задания, разработанного при

выполнении лабораторной работы №3, провести обоснованный выбор

средства разработки специального ПО. Разработать схему общесистемного

ПО на подобие схемы, показанной на рис. 14.

2. В соответствии с требованиями технического задания, разработанного при

выполнении лабораторной работы №3, а также проектными решениями,

разработанными при выполнении лабораторных работ №4 и №5, разработать

специальное программное обеспечение ПС.

3. Выполнить описание разработанных компонентов приложения в виде табл. 14.

Типы компонентов указать согласно табл. 12. Имена компонентов-файлов

привести с указанием расширения.

Таблица 14. Перечень разработанных компонентов приложения

№ Имя Тип Описание

4. Построить структурную схему разработанного приложения в виде диаграммы

компонентов UML, выражающую взаимодействие его компонентов с

компонентами БД в процессе функционирования приложения.

5. Запустить приложение на выполнение. Убедиться в соответствии результатов

выполнения приложения требованиям, установленным в техническом задании.

При обнаружении логических ошибок задокументировать их и устранить.

6. Представить экранные формы компонентов приложения, в том числе отчетов.

7. Проанализировать код приложения по критерию сложности. В качестве

критерия сложности использовать:

− число модулей (классов) приложения;

− суммарное число переменных подпрограмм (методов классов), включая их

формальные параметры;

− суммарное количество операторов подпрограмм (методов классов);

− глубину вложенности структурных операторов ветвления и повторения;

− глубину наследования классов.

8. Выполнить описание физических элементов ПС в виде табл. 15. Типы

элементов указать согласно табл. 13.

Таблица 15. Перечень узлов программной системы

№ Имя Тип Описание

9. Построить диаграмму развертывания UML, выражающую зависимости между

узлами ПС и развернутыми на них компонентами из табл. 14.

Тема: Цифровой помощник учителя математики

Ход работы:

В соответствии с требованиями технического задания, разработанного при выполнении лабораторной работы №3, а также проектными решениями, разработанными при выполнении лабораторных работ №4 и №5, разработано специальное программное обеспечение.

Программа представлена в виде веб-приложения. Код программы написан на JavaScript, а также на языке программирования Python. Эти языки были выбраны среди множества других, так как именно они поддерживаются всеми современными веб-браузерами и позволяют разрабатывать приложения быстро и эффективно. Python используется с фреймворком Flask, который позволяет создавать веб-приложения с минимальным количеством кода и упрощает процесс разработки.

Для создания структуры и содержимого веб-страницы используется язык гипертекстовой разметки HTML. Для внешнего оформления веб-страниц было применено несколько инструментов: CSS - для удобного оформления и придания стилей, а также JavaScript - для добавления взаимодействия с различными компонентами.

Для визуализации веб-графики используется элемент HTML5 - Canvas, который позволяет программно рисовать на веб-странице с помощью JavaScript.

Кроме того, для хранения и обработки данных в приложении используется система управления базами данных SQLite, которая интегрирована в веб-приложение с помощью Python и Flask.

В соответствии с требованиями технического задания была разработана структурная схема программного обеспечения, представленная ниже на рисунке 1.

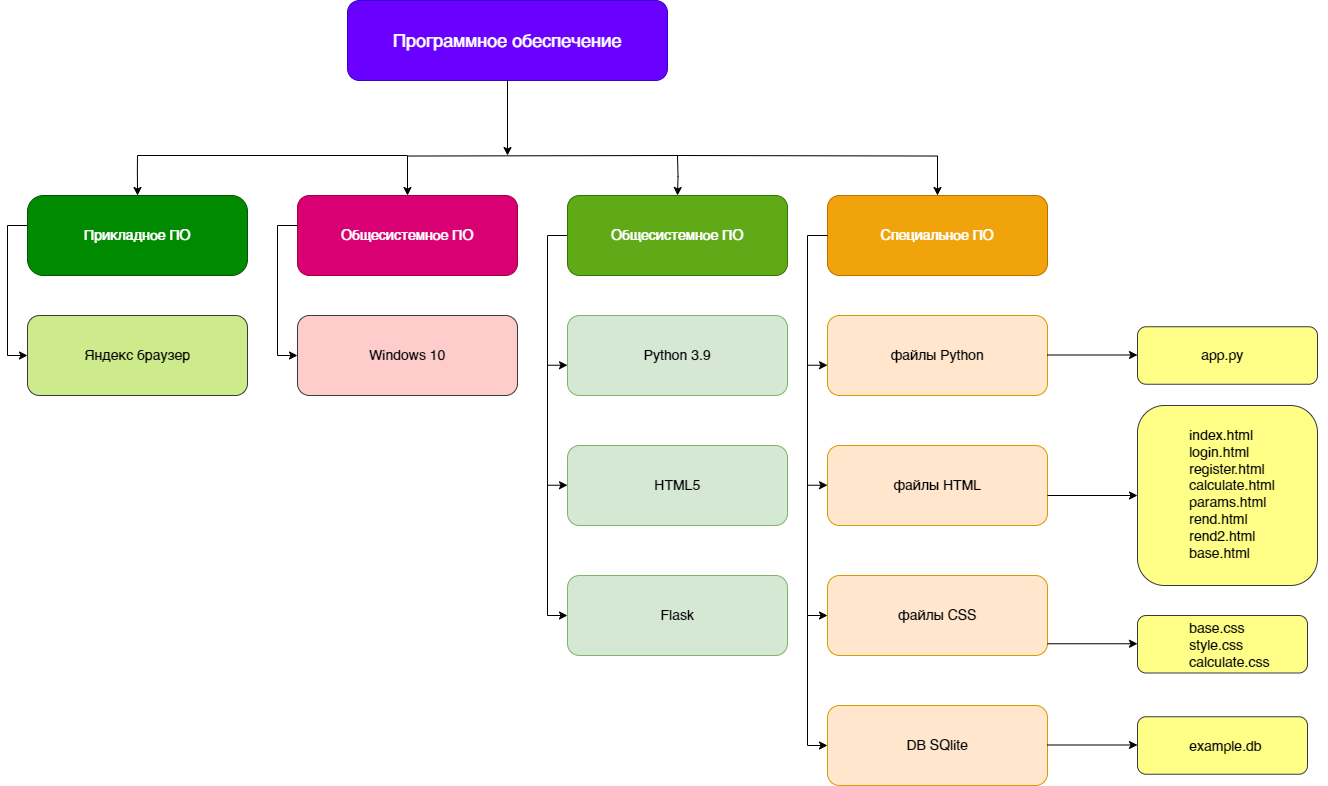


Рисунок 1 – Структурная схема программного обеспечения

Интерфейс веб-приложения цифрового помощника учителя математики очень прост в использовании. Его основная цель - упростить процесс обучения и решения математических задач для учителей и учеников. Интерфейс программы представлен на рисунке 2.

В верхней части сайта расположена шапка, которая содержит три основные вкладки:

1. Главная страница: здесь пользователи могут видеть описание данного сервиса.
2. Графический калькулятор: этот элемент позволяет визуализировать математические функции, введенные пользователем.
3. Готовые задачи: эта функция предоставляет различные математические задачи по разным темам, таким как линейные уравнения, квадратные уравнения и тригонометрические.

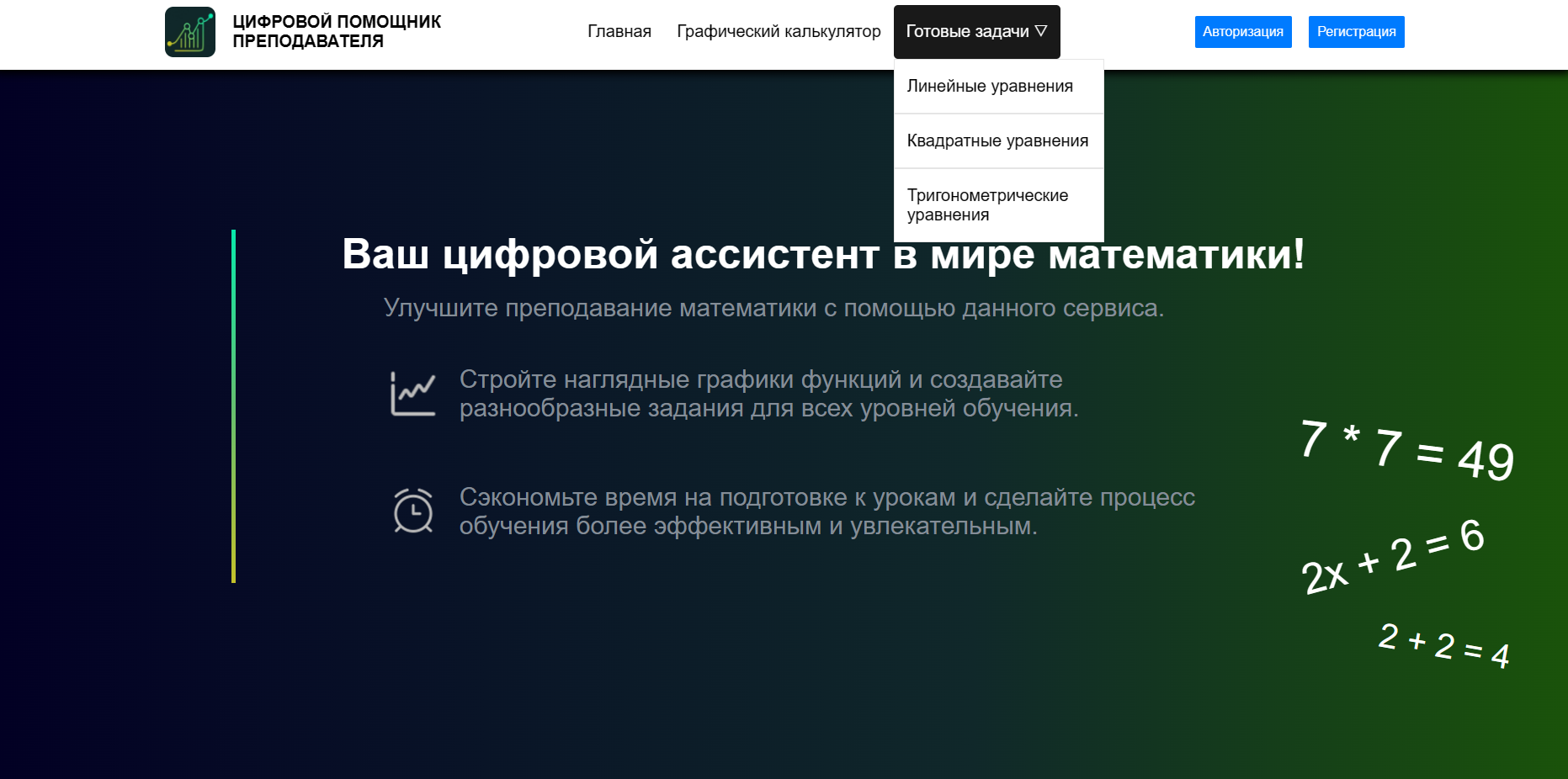


Рисунок 2 – Интерфейс программы при загрузке веб-страницы

После того, как пользователь выбирает тему, в сервисе отображаются параметры для выбора задачи из сохраненных задач в базе данных или настройки генерации задачи.

На рисунке 3 показан блок настройки параметров для темы линейных уравнений.

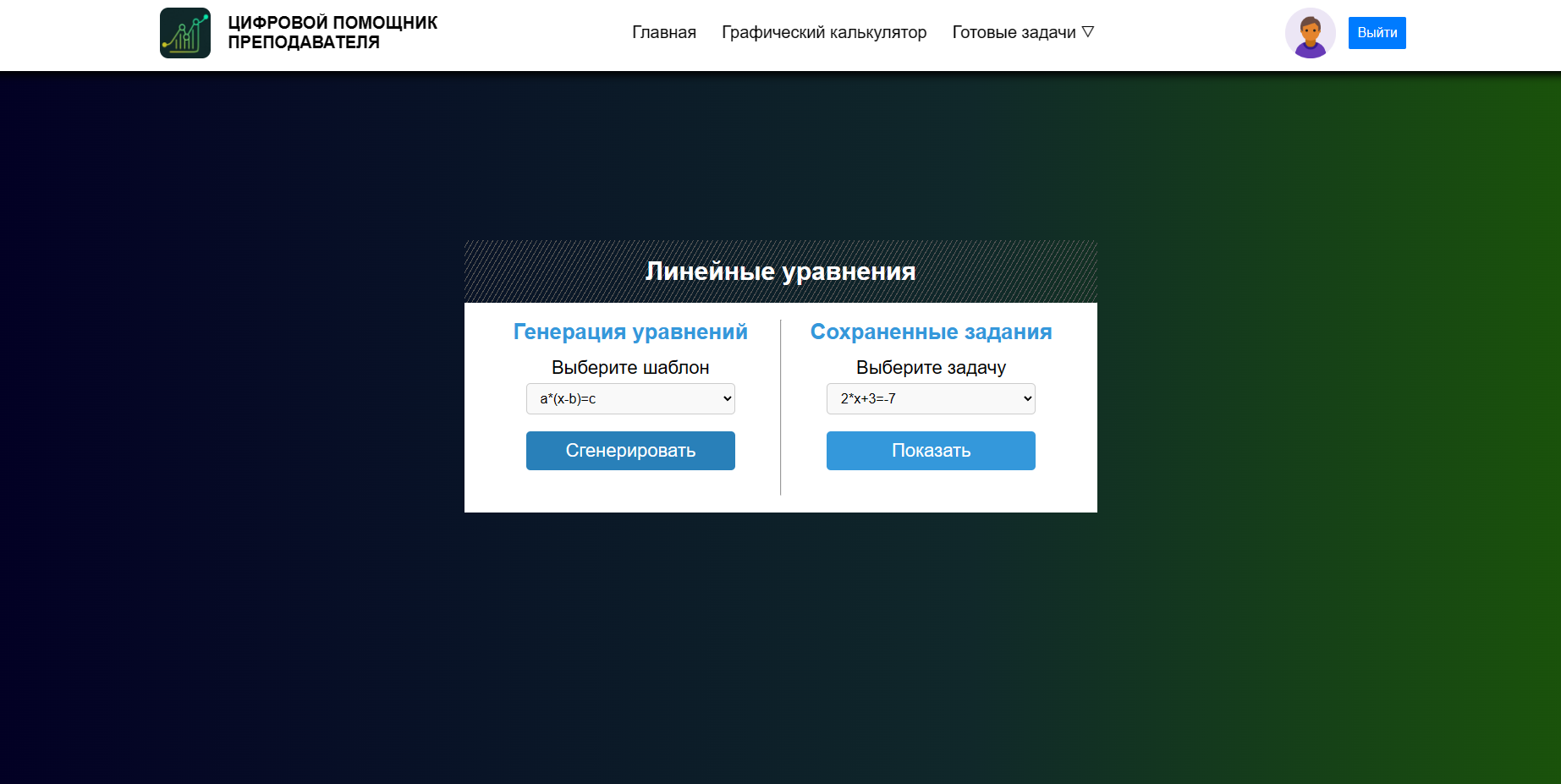


Рисунок 3 – Настройка параметров для темы линейных уравнений

После нажатия кнопки «Сгенерировать» в разделе настройки параметров генерации, на странице отображается название и условие задачи. Примеры отображения сгенерированной задачи представлены ниже на рисунках 4 и 5.

Когда ученики решат задачу аналитическим способом, преподаватель сможет нажать кнопку «Показать график» или «Показать ответ». В первом случае отобразится график, на котором можно увидеть координаты, что поможет лучше понять суть задачи. Во втором случае появится ответ на задачу, так как в приложение добавлены функции для разбора уравнения и его решения.

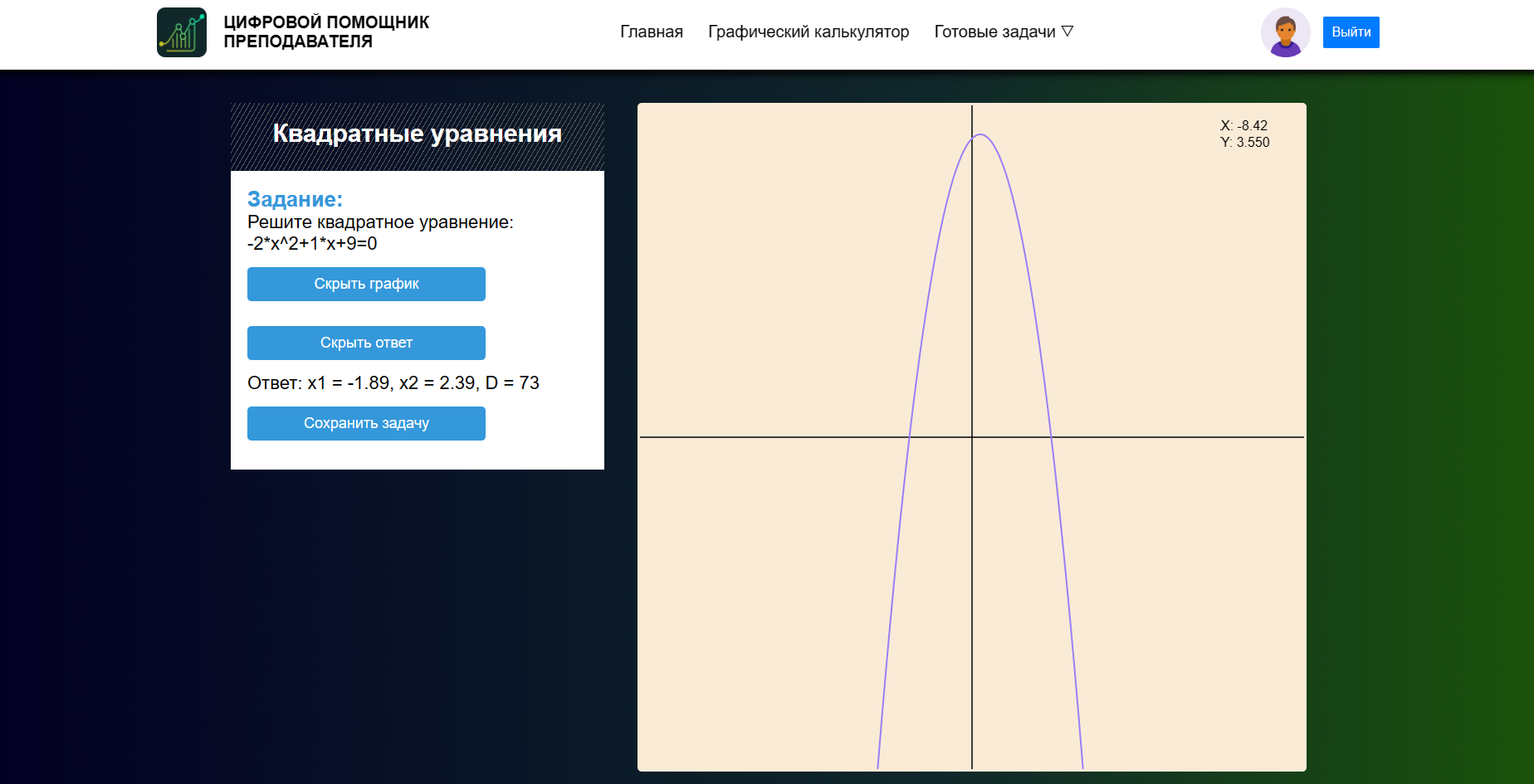


Рисунок 4 – Отображение сгенерированной задачи по теме квадратных уравнений

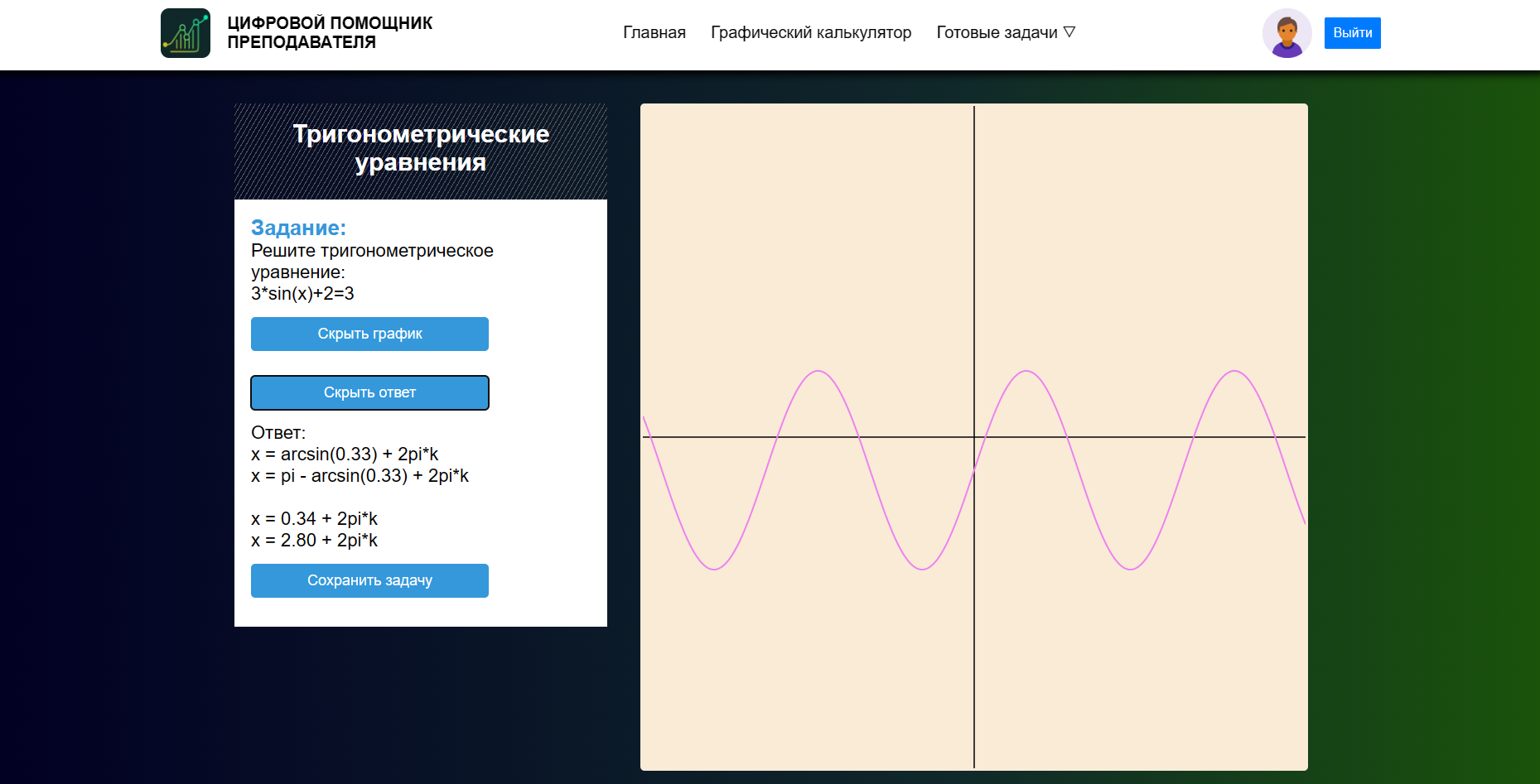


Рисунок 5 – Отображение сгенерированной задачи по теме тригонометрических уравнений

Каждый преподаватель может зарегистрироваться на данном сервисе, чтобы иметь возможность сохранять задачу. Поскольку у каждого зарегистрированного пользователя есть уникальный идентификатор, сохраненная задача будет привязана к этому идентификатору и доступна только авторизованному пользователю.

Приложение использует сессии, поэтому после закрытия браузера пользователь останется авторизованным до тех пор, пока не нажмет кнопку "Выйти из аккаунта". Форма регистрации в веб-приложении представлена на рисунке 6.

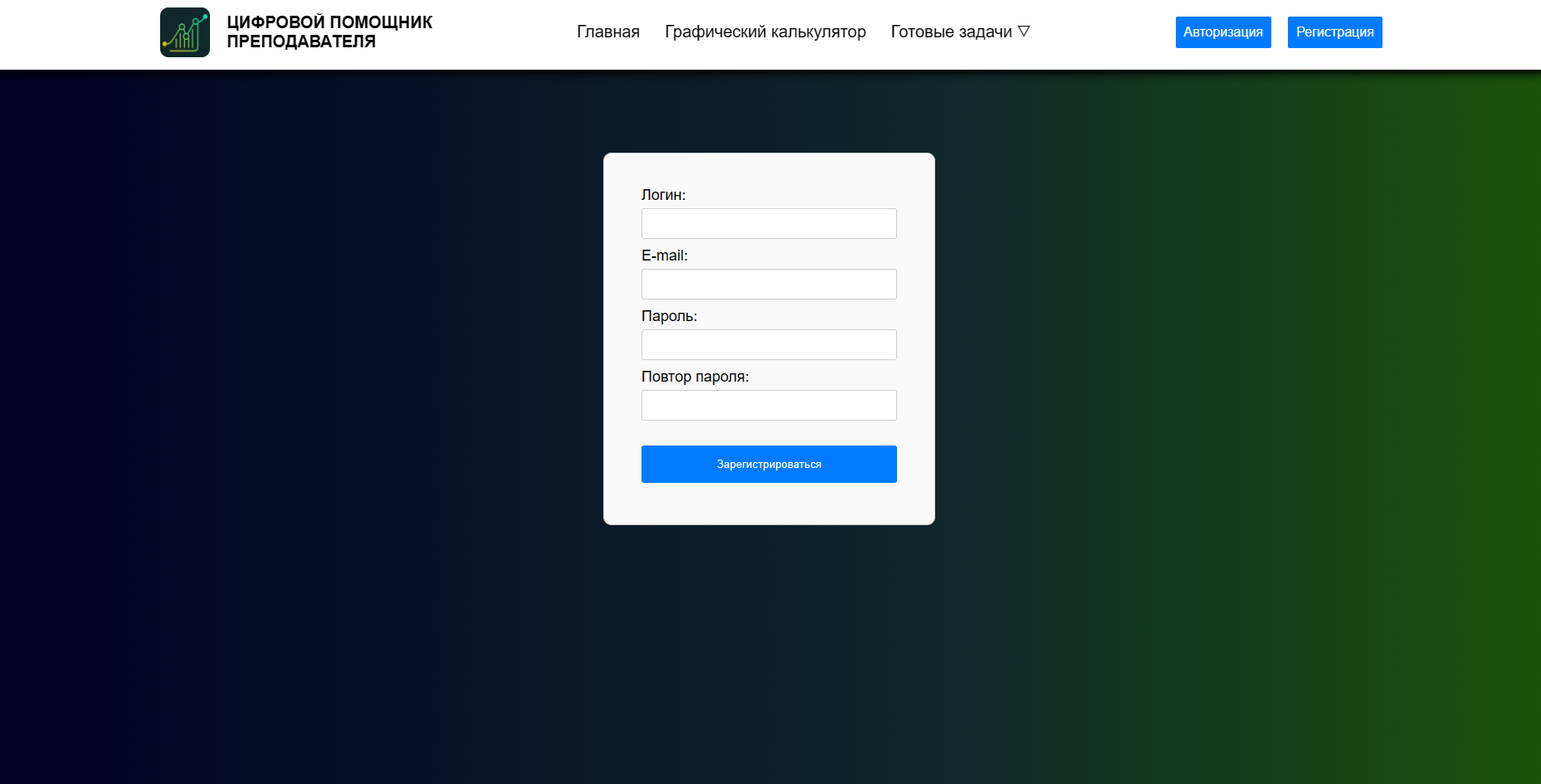


Рисунок 6 – Форма регистрации в цифровом помощнике учителя математики

Кроме генерации различных типов задач, пользователь может воспользоваться еще одной функцией сервиса – графическим калькулятором. Где пользователь сначала выбирает тип функции из списка, затем заполняет все необходимые поля и нажимает кнопку «Отобразить». После этого на экране отображается график функции.

Программа позволяет отображать графики различных типов функций, включая линейные, тригонометрические, экспоненциальные, степенные, гиперболические, параметрические, а также графики эллипсов и окружностей. Пример сгенерированных графиков функции от одной переменной показан на рисунке 7.

# 

Рисунок 7 – Пример сгенерированных графиков

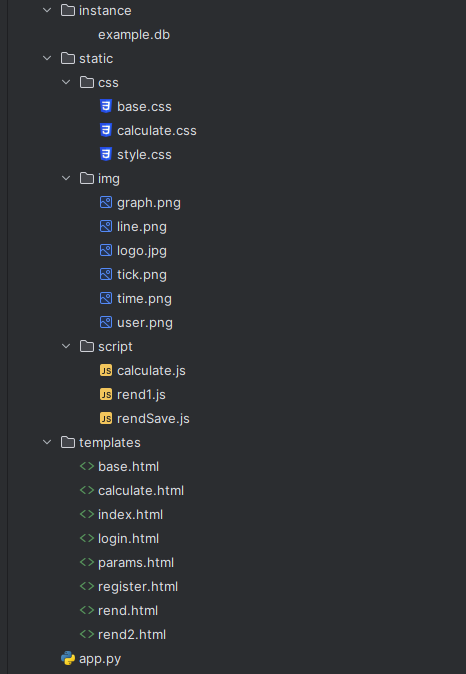


Рисунок 8 – Список разработанных модулей ПС

Таблица 1. Перечень разработанных компонентов приложения

|  |  |  |
| --- | --- | --- |
| № | Имя компонента | Описание |
| 1 | index.html | Главная страница со ссылками на все формы и на другие компоненты |
| 2 | login.html | Форма для авторизации пользователя |
| 3 | register.html | Форма для регистрации пользователя |
| 4 | calculate.html | Графический калькулятор, который позволяет пользователям строить графики математических функций |
| 5 | params.html | Страница, которая позволяет пользователям настроить параметры для генерации различных задач |
| 6 | rend.html | Страница, на которой отображается сгенерированная задача, ее график и ответ |
| 7 | rend2.html | Страница, на которой отображается сохраненная пользователем задача, ее график и ответ |
| 8 | base.html | Базовая страница для отображения меню на всех других страницах |
| 8 | calculate.js | Скрипт, с помощью которого работает графический калькулятор |
| 9 | rend1.js | Скрипт, с помощью которого отображается сгенерированная задача, ее график и ответ |
| 10 | rendSave.js | Скрипт, с помощью которого отображается сгенерированная задача, ее график и ответ |
| 11 | base.css | Файл, который отвечает за стилизацию главного меню |
| 12 | calculate.css | Файл, который отвечает за стилизацию графического калькулятора |
| 13 | style.css | Файл, который отвечает за стилизацию страниц с формами регистрации и авторизации, а также настройки параметров и отображения задач |
| 14 | example.db | Файл с базой данных |

После компиляции приложение было запущено, синтаксических ошибок не обнаружено, результаты соответствуют заявленным и не имеют логических ошибок.

Ниже на рисунке 9 представлена диаграмма компонентов во взаимодействии с объектами базы данных.

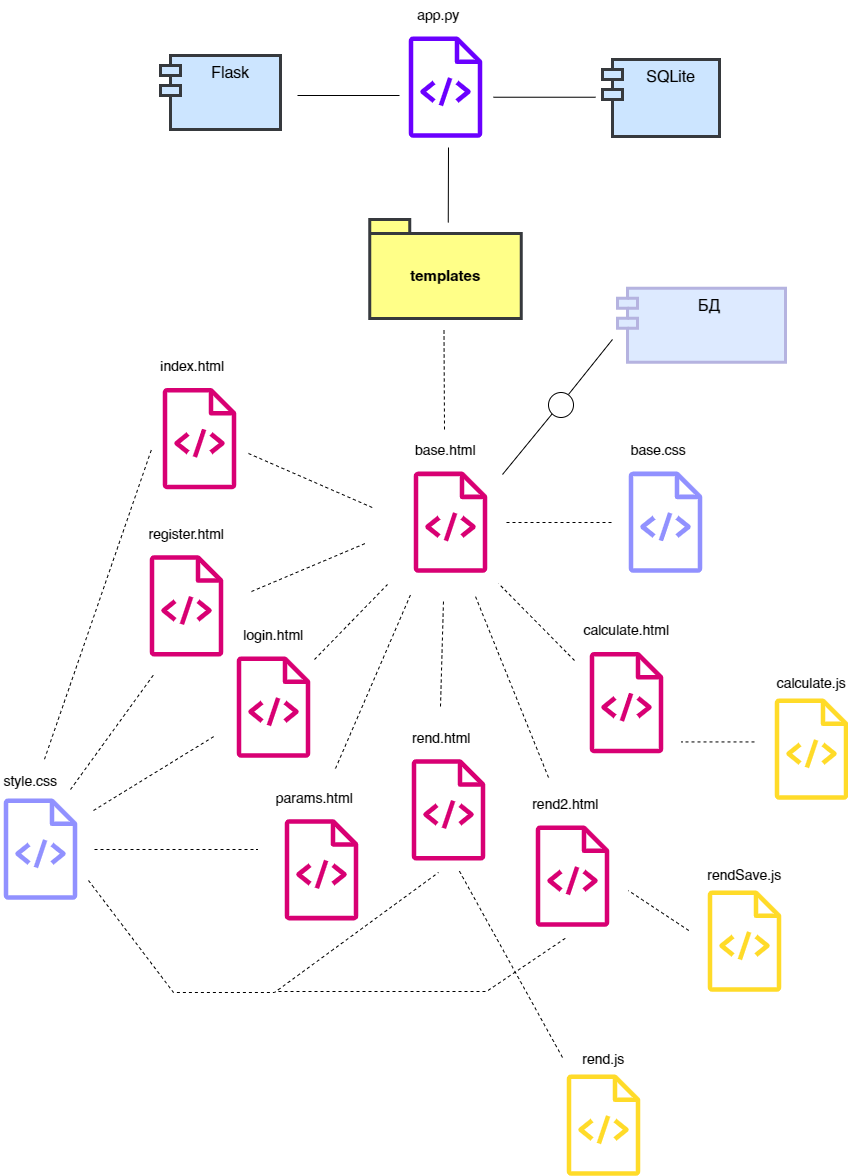


Рисунок 9 – Диаграмма компонентов

Программное обеспечение состоит из 14 модулей, включает в себя 95 переменных состояния, 41 вычисляемое свойство и 24 метода. Весь этот функционал обеспечивает работу программы и ее возможности для пользователя.

Перечень узлов программной системы представлен в таблице 2.

Таблица 2. Перечень узлов программы.

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Описание** |
| Web Server | «node» | Сервер, на котором развернуто веб-приложение (Github pages). |
| Application Server | «node» | Сервер, на котором работает backend приложения. |
| Database Server | «database» | Сервер базы данных PostgreSQL, хранящий данные приложения. |
| Client Devices | «device» | Устройства пользователей, например, компьютеры, планшеты или смартфоны, с которых осуществляется доступ к веб-приложению. |

Диаграмма развёртывания программной системы представлена на рисунке 10.

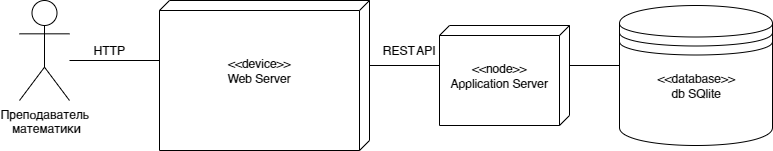


Рисунок 10 – диаграмма развёртывания

**Вывод:** я приобрела навыки разработки интерфейсной части программной системы при помощи современных средств разработки и создала приложение, а также проанализировала его код по сложности.