

Филиал Московского Государственного Университета
имени М.В. Ломоносова в г. Ташкенте
Факультет прикладной математики и информатики
Кафедра прикладной математики и информатики

Фаттахова Ильнура Шавкатовна

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**НА ТЕМУ: « РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ
ВЗАИМОДЕЙСТВИЯ С УЧЕНИКАМИ »**

**« DEVELOPING A WEB APPLICATION FOR INTERACTING WITH
STUDENTS »**

**По направлению 01.03.02 «ПРИКЛАДНАЯ МАТЕМАТИКА И
ИНФОРМАТИКА»**

ВКР просмотрена и рекомендована к защите
зав. кафедрой «ПМиИ», к.ф.-м.н., доцент _____ Строгалов А.С.

Научный руководитель
кан.физ-мат наук, и.о.доц _____ Альхамов Р.Р

Ташкент 2021

Содержание

| | |
|---|----|
| Аннотация | 3 |
| Введение | 4 |
| Постановка задачи | 5 |
| Разработка эскизного и технического проекта | 6 |
| Серверная часть | 9 |
| Заключение | 11 |
| Список литературы | 12 |

Аннотация

В настоящее время, особенно в период пандемии, учителя различных категорий столкнулись с проблемой проверки домашних работ, а также с отбором заданий для проверки в целом.

Создание уникальных проверочных работ, которые в дальнейшем не будут осложнены проверкой — довольно часто повторяющаяся работа, способная весьма быстро наскучить. В данной дипломной работе было рассмотрено текущее обстоятельство и предложено решение выявленной проблемы.

Структура представляет из себя веб-приложение, готовое к работе на любом устройстве – будь-то телефон, или компьютер.

Ключевые слова: веб-приложение, разработка, сайт, генератор.

Currently, especially during a pandemic, teachers of various categories are faced with the problem of checking homework, as well as the selection of assignments for checking in general.

Creation of unique verification work that will not be further complicated by verification is a fairly frequent repetitive work that can get bored very quickly. In this thesis, the current circumstance was considered and a solution to the identified problem was proposed.

The structure is a web application ready to work on any device - be it a phone or a computer.

Keywords: web application, development, site, generator.

Введение

В курсовой работе было создано веб-приложение, которое генерирует СЛАУ, квадратные и кубические уравнения. Однако существенным минусом было отсутствие серверной части, которая позволила бы ученикам и учителям заходить на платформу и взаимодействовать с ней. Также из минусов стоит обозначить не проработанный дизайн и слабую мобильную адаптацию.

В данной дипломной работе вышеперечисленные минусы были учтены и исправлены. Также была добавлена автоматическая проверка задач, был создан личный кабинет учителя и ученика.

Помимо этого сервис стал включать в себя больше подвидов задач, что позволит учителям генерировать работы разной сложности.

Постановка задачи

За основу приложения, как уже было указано выше, были взяты разработки из курсовой работы. В ходе разработки курсовой были рассмотрены следующие сервисы: DrFrost [1] и KutaSoftware [2]

Однако, как показало исследование, у данных сервисов есть существенные минусы. Они являются хорошими помощниками для учителей в англо-язычных странах. Сервисы позволяют создавать различного рода листинги из легких математических задач для начальной школы, а так же более сложных заданий. Проблема, к примеру, приложения KutaSoftware, в сложности установки и приобретения, так как версия с бесплатным доступом рассчитана лишь на короткий период, а полная версия стоит не малых денег. В случае же с DrFrost необходима регистрация. Помимо этого платформа обладает небольшим функционалом для учителей, наоборот, она больше рассчитана на учеников, которые будут решать данную задачу. [3]

Так же одним из известных источников стали сервисы, действующие на территории Российской Федерации Учи.ру [4] и Я.Класс [5].

Данные сервисы позволяют контролировать работу учеников начальных классов (uchi.ru) и средних классов 6-8 (Я.класс). Однако, в связи с тем, что сервисы рассчитаны на РФ, то и соответственно регистрация и пользование ими может принадлежать только лицензированным образовательным учреждениям, расположенным в РФ. Также, пользование данными сервисами подразумевает в себе ежемесячную оплату подписки на данный сервис, что является неудобным минусом для пользования зарубежом.

В ходе разработки данного веб-приложения были учены все плюсы и минусы вышеперечисленных сервисов, проведен мониторинг среди учеников, на мотивированность использования данного приложения. Мониторинг показал, что в условиях пандемии им было бы проще и удобнее выполнять задания на компьютере, имея возможность проработать пробелы в своих знаниях.

Также, при создании данного сервиса была предложена идея автоматической рассылки задания для группы учеников, что существенно ускорит работу учителя.

Разработка эскизного и технического проекта

Для разработки дизайна интерфейса был использован онлайн-сервис Figma, на котором были подобраны цвета и размечен примерный корсет будущего проекта. На (рис. 1) изображена мобильная версия будущего проекта.

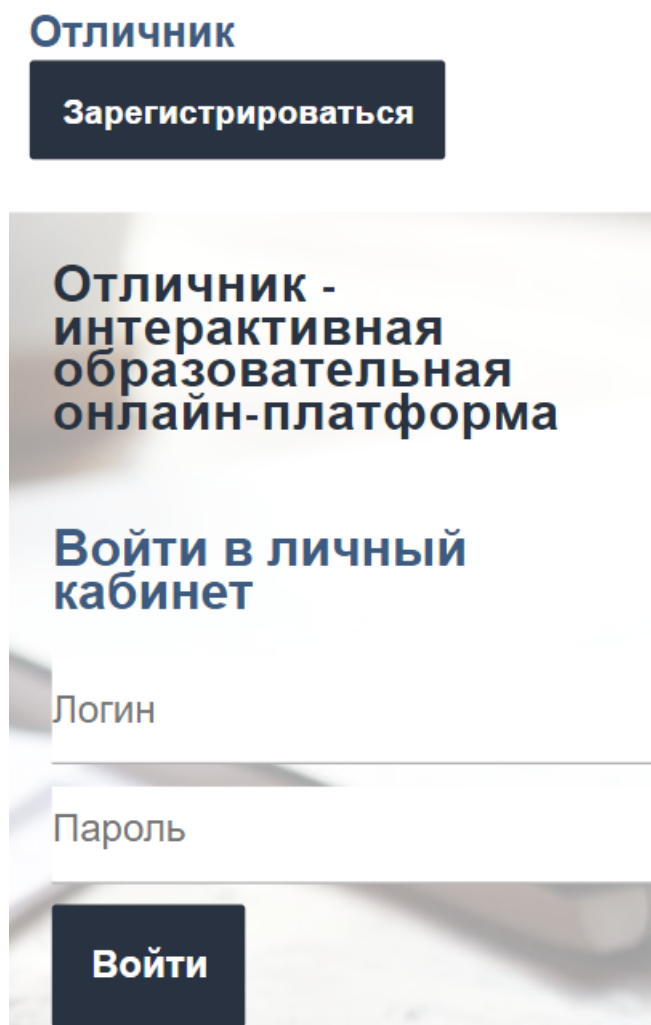


Рис. 1: Мобильная версия будущего приложения

Для разработки алгоритма было принято использовать следующие языки программирования: Python (Django, Django Rest Framework), JavaScript (ECMAScript6), непосредственно задания описаны на языке \LaTeX [6]. 2 Структуру приложения можно описать следующим образом:

1) на главной странице доступна информация о сервисе, регистрация нового пользователя и кнопка «вход» для уже существующих пользователей

Различия для ученика и учителя состоят лишь в личном кабинете.

Отличник - интерактивная образовательная онлайн-платформа

Войти в личный кабинет

Логин

Пароль

Войти

Что такое отличник?

Отличник это образовательная онлайн-платформа, где ученики смогут выполнять домашние задания, а учителя без труда проверять их работу.

Почему мы?

Мы ценим ваше время. Мы знаем как тяжело учителям проверять домашние работы, а ученикам трудно сосредоточиться. Поэтому формат интерактива поможет ученикам справиться с заданиями различной сложности, а учителю не придется проверять гору тетрадей.



Рис. 2: Декстоп версия будущего приложения

2) в ходе выполнения задания учеником автоматическая проверка выявляет верные и неверные решения. Неверные решения можно исправить. Если решение было верно кнопка «сохранить» становится не активной.

3) учитель может создавать собственные задачи. Для этого существует панель ввода условия и шаблон как должна быть записана задача на языке \LaTeX . Описаны команды для корректного отображения задач, доступен предварительный просмотр. (рис. 3)

4) для учителя также доступна массовая рассылка задания. Для этого необходимо выбрать задания, группу учеников для которых они предназначены, обозначить название (к примеру: домашняя работа на 20 апреля) и обозначить сроки (опционально).

5) для ученика в личном кабинете доступны решения задач, направленных учителем и проработка собственных навыков. Функционал при решении задач от учителя и по выбранным темам остается неизменным. Сама работа включает в себя две составляющие: серверную и клиентскую.

Отличник

Создание нового примера

Введите название или номер задания

Введите тему, к которой относится задание

Для верного отображения формул записывайте их в следующем формате:

Для отображения степени используйте символ $^$: так x^2 будет записываться как x^2 .

Для отображения объединяющей скобки необходимо использовать конструкцию

\begin{cases} первое уравнение $\backslash \backslash$ второе уравнение \end{cases} .

Для отображения дроби $\frac{\text{числитель}}{\text{знаменатель}}$.

Для отображения индекса использовать нижнее подчеркивание: так x_1 будет записан в виде x_1

Для отображения знака квадратного корня необходимо использовать $\sqrt{\text{выражение}}$.

Вся формула должна быть заключена в двойной символ $\$$ в начале и в конце.

Введите условие

Введите верное решение

Предварительный просмотр

Рис. 3: Создание нового примера в личном кабинете учителя

Серверная часть

Серверная часть представляет собой функционал, доступный только для администратора, к примеру, технического специалиста. Этот функционал недоступен ни для учеников, ни для учителей, а является лишь системой контроля для вышеперечисленных пользователей. Технический специалист может редактировать группы учеников, переопределяя принадлежность к той или иной группе, давать права доступа для учителя в личный кабинет. Для аутентификации и идентификации в клиент-серверном приложении используется JSON Web Token [7].

Токен – автогенерируемый ключ безопасности для передачи данных, который предоставляется открытым стандартом JWT [7], отвечающий всем необходимым критериям безопасности данных.

Как уже было сказано выше серверная часть написана на языке Python [8] с использованием фреймворка Django [9]. Клиент-серверная часть передает между собой информацию по технологии REST-api [10]. За работу с REST-api отвечает Django Rest Framework. Все данные с серверной части передаются в формате JSON - JavaScript Object Notation. [11]

Базы данных предоставлены в формате SQLite 3 [12]. Процесс взаимодействия базы данных происходит через интерфейс, предоставляемый фреймворком Django и носит название Django ORM - Object Relational Mapping. Этот инструмент позволяет взаимодействовать с базой данных, не делая SQL-запросы.

В SQL за каждую сущность отвечает таблица. В Django аналогом таблицы являются модели. Это позволяет использовать только язык программирования Python. Поля в модели равнозначны столбцам таблицы SQL.

У каждой сущности есть 3 основных файлов рабочего кода:

1) `urls.py` – описываются эндпоинты – относительные пути по которым клиент делает запрос

2) `serializer.py` – помогает переводить python-объекты в JSON. Данная часть отвечает именно за сериализацию. Сериализатор преобразует информацию, хранящуюся в базе данных и определенную с помощью моделей Django, в формат, который легко и эффективно передается через API.

3) `views.py` – в зависимости от своего функционала наследуется от встроенных Django классов. Получает запрос, обрабатывает его, вызывает сериалайзеры и отправляет пользователю некоторый ответ. Если

для обработки запроса необходимо обращение к модели и базе данных, то `views` взаимодействует с ними.

Клиент часть представляет собой внешнюю оболочку, так называемый интерфейс. Все происходящие события - открытие окон, реагирование на нажатие кнопок, регистрация, проверка, создание и рассылка работ – описаны на языке JavaScript. Далее мы рассмотрим клиент-часть более подробно.

Заключение

В данной работе было сконструировано приложение для работы с учениками, построенное на материалах курсовой работы. При построении приложения были использованы языки Python для серверной части и JavaScript для клиентской. Была проведена работа для доступности приложения в сети интернет.

Была скорректирована генерация примеров, таким образом получилось создать обширную базу, включающую в себя подвиды задач с разной степенью сложности.

Список литературы

- [1] Dr Frost. URL: <https://www.drfrstmths.com/index.php>.
- [2] Kuta Software LLC. URL: <https://www.kutasoftware.com/index.html>.
- [3] Фаттахова Ильнура Шавкатовна. «Курсовая работа - Создание приложения-генератора примеров». в: *Курсовая работа* (2020).
- [4] ООО Учи.ру. URL: <https://uchi.ru/>.
- [5] ООО ЯКласс. URL: <https://www.yaklass.ru/>.
- [6] Donald Ervin Knuth. *The TEXbook: ... describes TEX version 3.0 ...* Addison-Wesley, 2012.
- [7] JWT.io. URL: <https://jwt.io/>.
- [8] Guido Van Rossum. *Python Library Reference (Open Source Library)*. 12th Media Services, 1995.
- [9] Каплан-Мосс Дж Головатый А. *Django. Подробное руководство*. ISBN 978-5-93286-187-5, 2010.
- [10] Roy Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. 2000.
- [11] D. Crockford. *The application/json Media Type for JavaScript Object Notation (JSON)*. 2006.
- [12] Jay A. Kreibich. *Using SQLite*. 2010 г.