

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ОБУЧЕНИЕМ НА БАЗЕ
ВЕБ-ПРИЛОЖЕНИЯ DJANGO СО ВСТРОЕННЫМ
КОМПИЛЯТОРОМ PYTHON**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Коноваловой Валерии Сергеевны

Научный руководитель
зав. каф. техн. прог. , к. ф.-м. н.,
доцент

И. А. Батраева

Заведующий кафедрой
к. ф.-м. н., доцент

С. В. Миронов

Саратов 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Обзор образовательных технологий.....	4
1.1 Цифровое обучение	4
1.2 Веб–платформы для организации онлайн–обучения.....	4
1.3 Фреймворк Django для веб–разработки	4
2 Разработка веб–приложения для обучения программированию	5
2.1 Стек технологий	5
2.2 База данных	5
2.3 Обеспечение безопасности и разграничение прав доступа.....	7
2.4 Компоненты приложения.....	8
2.4.1 Разработка приложения ruTestingApp	8
2.4.2 Разработка приложения Main	8
2.4.3 Разработка приложения Users	9
2.4.4 Разработка приложения Courses	9
ЗАКЛЮЧЕНИЕ	12
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	13

ВВЕДЕНИЕ

Обучение — это непрерывный процесс, который длится на протяжении всей жизни человека, дающий людям возможность приспособиться к своему окружению и добиться личного развития и продвижения.

С появлением современных технологий и Интернета, часть обучения постепенно перешла в онлайн-формат, который обеспечивается за счёт созданных для этого специализированных веб-платформ, предоставляющих преподавателям и учащимся широкий спектр возможности обучения и взаимодействия друг с другом. Однако, имеющиеся веб-платформы обладают ограниченной кастомизацией и персонализацией, а так же не имеют встроенных средств компиляции пользовательского кода, что ограничивает возможности пользователя и вынуждает его обращаться к сторонним сервисам.

Актуальность данной работы состоит в том, что разработанное в рамках бакалаврской работы приложения предоставляет пользователю возможность создавать собственные курсы с уроками, посвящённые программированию на языке Python, загружать необходимый материал и практические задания, и в этом же приложении запускать проверку введённого пользователем кода.

Цель настоящей работы — разработка системы управления учебным процессом на базе веб-приложения со встроенным компилятором кода Python.

В связи с поставленной целью, основными **задачами** являются:

- Изучить виды современных платформ, созданных для организации онлайн-обучения;
- Проанализировать положительные и негативные стороны данного формата обучения;
- На основе полученных результатов продумать систему, удовлетворяющую основным потребностям различных пользователей в организации процесса обучения;
- Разработать веб-приложение, соответствующее данной системе.

Структура и объём работы. Бакалаврская работа состоит из введения, двух разделов, заключения, списка использованных источников и восьми приложений. Общий объём работы — 69 страниц, из них 51 страница — основное содержание, включая 20 рисунков, список использованных источников информации — 20 наименований.

1 Обзор образовательных технологий

В данном разделе рассматриваются образовательные технологии, предназначенные для организации цифрового обучения, их особенности, а также их преимущества и недостатки.

1.1 Цифровое обучение

Цифровое обучение представляет собой форму получения знаний с использованием цифровых технологий, предоставляющие образовательный контент. Цифровое обучение может принимать разнообразные формы обучения, такие как: виртуальные классы, онлайн–курсы, вебинары и различные мультимедийные ресурсы, включающие в себя видеоролики, анимацию, симуляции и интерактивные упражнения, которые могут помочь учащимся усвоить материал и развить более глубокое понимание предмета [1, 2].

1.2 Веб–платформы для организации онлайн–обучения

Веб–платформы являются одним из инструментов в организации онлайн–обучения. Они предоставляют учащимся централизованное место для доступа к материалам курса, а также обеспечивают возможность общения с преподавателями, совместной работы с другими студентами и отправку заданий.

Несмотря на их положительные стороны, такие как: независимость от платформы, простота доступа и автоматическое обновления контента, данный инструмент создаёт для пользователей некоторые ограничения, выраженные в высоких технических требованиях, расходах, связанных с приобретением продукта, и в ограниченной кастомизации.

1.3 Фреймворк Django для веб–разработки

В качестве фреймворка для разработки итогового приложения был использован веб–фреймворк Django [3–5].

Django представляет собой высокоуровневый фреймворк языка программирования Python, позволяющий создавать веб–сайты и веб–приложения, отличающиеся быстротой реализации и лёгкостью поддержки и администрирования. Являясь одной из самых популярных бесплатных платформ для веб–разработки с открытым исходным кодом, Django имеет активное сообщество и русскоязычную документацию, что говорит о поддержке актуальности программного обеспечения и перспективах развития данного фреймворка [6–10].

2 Разработка веб–приложения для обучения программированию

В данном разделе представлено описание используемого стека технологий и инструментов, которые применялись в процессе разработки веб–приложения, разбор логики работы приложения, а также примеры программного кода, которые реализует необходимый функционал системы.

2.1 Стек технологий

При разработке приложения было решено использовать следующий стек технологий:

1. Язык программирования Python (версия 3.8) [11–15];
2. PostgreSQL — система объектно-реляционных баз данных с открытым исходным кодом, которая ценится своей надёжностью и производительностью; используется в качестве базы данных приложения [16];
3. Django (версия 4.2.1) — фреймворк для создания веб–приложений на языке программирования Python, который использует шаблон проектирования MVC (Model–View–Controller);
4. Pytest — среда тестирования языка Python, созданная в рамках проекта PyPy; используется для написания различных типов тестов программного обеспечения; используется в реализации компилятора для выполнения автоматических тестов пользовательского кода;
5. Oracle VM VirtualBox — виртуальная машина, в которой запускается компиляция пользовательского кода.

2.2 База данных

Как было сказано выше, в качестве базы данных приложения была выбрана PostgreSQL, обеспечивающая высокий уровень целостности и безопасности данных. Также она отличается хорошей производительностью и расширяемостью [17–20].

Структура базы данных представлена на рис. 1.

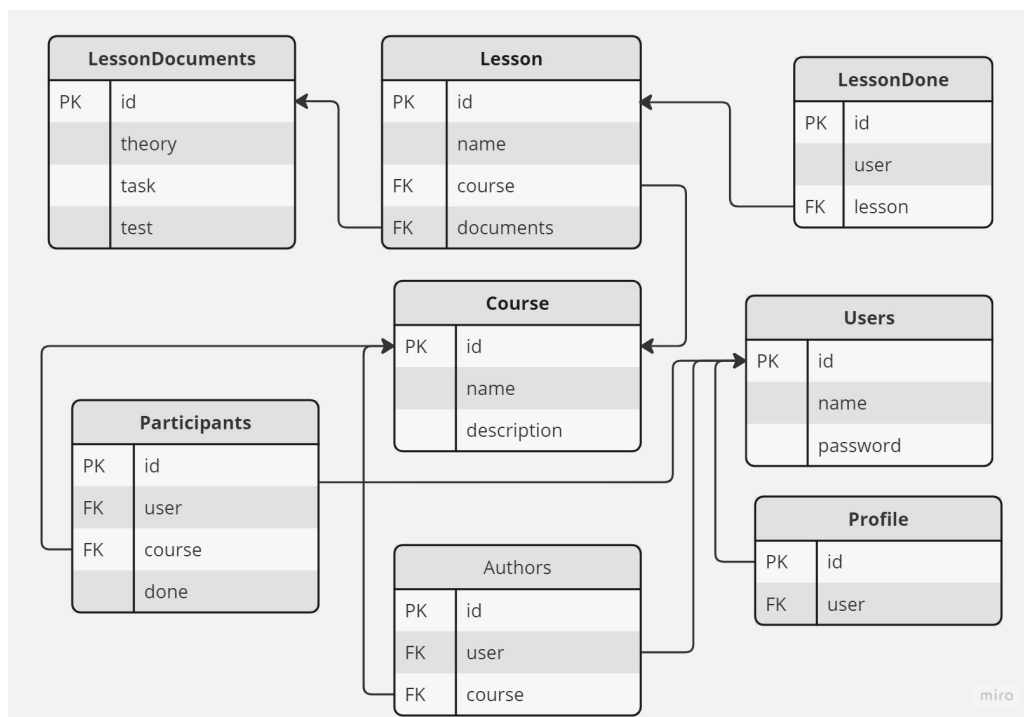


Рисунок 1 – Структура базы данных веб-приложения

Запросы пользователя, связанные с любыми изменениями полей базы данных (например, добавление нового курса или изменение данных об уроке курса), вызывают соответствующую функцию из view-файла, в которой выполняется запрос к необходимой таблице базы данных, удовлетворяющей запросу Пользователя.

При регистрации Пользователя, информация о его имени и пароле сохраняется в таблицу Users, а также создаётся запись в таблице Profile, где хранятся данные, отображаемые на странице профиля Пользователя системы. Все активные курсы в системе сохранены в таблице Course. Все дальнейшие взаимодействия с курсом также включают в себя обращения к данной таблице базы данных.

Модель Course имеет связь с такими таблицами, как:

- Participants — хранит данные о пользователях и курсах, на которые они записаны, а также колонку «done» логического типа Boolean, True-значение которой указывает на завершение Пользователем данного курса (по умолчанию имеет значение False);
- Authors — хранит данные о создателе курса;
- Lesson — хранит информацию о уроках и курсах, которым они принадлежат.

Информация, необходимая для урока, включающая в себя текст теоре-

тического материала, текст практического задания, выводится на соответствующих вкладках урока. Сами текстовые файлы хранятся в директории проекта (media/), пути к которым прописаны в таблице Documents. Хранящаяся в Lesson информация об ID записи в Documents даёт доступ к документам, необходимым для конкретного урока. После успешного выполнения практического задания, в таблицу LessonsDone добавляется запись с информацией о пользователе и уроке, которые он выполнил, тем самым обеспечивая сохранения прогресса прохождения курса для каждого его участника.

Таким образом, подобная структура данных позволяет хранить в себя всю информацию, необходимую для удовлетворения пользовательских запросов и быстрой работы веб-приложения.

2.3 Обеспечение безопасности и разграничение прав доступа

В Django для управления доступом пользователей к определенным частям веб-приложения используется встроенная система аутентификации и авторизации, что обеспечивает гибкий и настраиваемый подход. Для аутентификации пользователя проверяется имя и пароль, при этом все пароли пользователей хешируются.

Основываясь на том, что Django включает в себя встроенную модель пользователя, которая содержит поля для определения ролей и разрешений пользователей, в приложении были созданы четыре роли:

- Суперпользователь — тот, кто имеет доступ непосредственно к самому исполняемому коду и консоли; по умолчанию — владелец сервера;
- Администратор — имеет возможность создавать и редактировать курсы и уроки, а также редактировать группы пользователей; необходимо назначение конкретного пользователя Администратором со стороны Суперпользователя;
- Преподаватель — имеет возможность создавать и редактировать курсы и уроки, просматривать список участников курса; необходимо назначение конкретного пользователя Преподавателем со стороны Администратора или Суперпользователя;
- Обычный пользователь — имеет возможность присоединиться к курсу, а так же доступ к его содержимому и отправке заданий на проверку;

Также стоит обратить внимание, что в веб-приложении присутствует функционал, который отвечает за компиляцию кода, введенного пользователем в

специальное поле. Для того, чтобы предотвратить возможное вмешательство в структуру системы, веб–приложение запускается на виртуальной машине, что позволяет изолировать пользователя от компонентов системы. Если злоумышленник попытается вмешаться в работу системы, ему не удастся добраться до кода и данных, которые хранятся в приложении.

2.4 Компоненты приложения

Было решено разделить итоговое приложение на отдельные составляющие, чтобы обеспечить простоту разработки и дальнейшего сопровождения продукта. Далее в работе подробнее рассматривается каждое из составляющих проекта.

2.4.1 Разработка приложения `ruTestingApp`

Данное приложения является базовым, в котором прописываются все настройки проекта, пути к файлам с шаблонами, стилями и медиа (изображения, различные документы, музыка и прочее), а также взаимодействия с базой данных. Помимо этого в `ruTestingApp` находятся все необходимые утилиты для работы всего проекта.

Отличительная черта данного приложения заключается в том, что в нём отсутствуют части `Models` и `Views`, так как они прописываются в вспомогательных приложениях. Однако, в нём есть часть `Urls`, в которой подключаются `urls`–файлы других составляющих итогового веб–приложения.

Далее идёт рассмотрение каждого вспомогательного приложения проекта.

2.4.2 Разработка приложения `Main`

Данное приложение является основой для других приложений. В нём хранятся базовые шаблоны страниц и прописаны основные переходы между URL–адресами. Именно здесь назначается статус главной страницы веб–приложения и описываются функции отображения базовых HTML–страниц, переход на которые доступен Пользователю из любого места приложения.

Приложение `Main` не взаимодействует с базой данных и весь его функционал заключается в отображении страниц на основе прописанных шаблонов и предоставлении Пользователю возможности обращения к другим частям приложения посредством предоставляемого ему интерфейса.

2.4.3 Разработка приложения Users

Каждому новому пользователю предлагается произвести регистрацию или вход в систему. До тех пор, пока пользователь не авторизуется, у него не будет доступа к курсам, предлагаемых веб–приложением. При попытке перехода к списку курсов, Пользователь будет перенаправлен на страницу входа в систему.

Переходя на страницу регистрации, Пользователю предоставляется форма для заполнения информации, необходимой для создания пользовательского аккаунта. По умолчанию, новые Пользователи добавляются в группу «Students». Добавить Пользователя в группу «Admins» или «Teachers» может только Суперпользователь, который на странице администрирования может менять роли и либо расширять, либо ограничивать права доступа зарегистрированных в системе Пользователей.

2.4.4 Разработка приложения Courses

Учитывая тот факт, что итоговое приложение создано с целью предоставления возможности пользователям создавать курсы и проходить обучение, данный раздел является наиболее важным с точки зрения прописанного в нём функционала. В данной части описан функционал, дающий Пользователям следующие возможности:

- создание новых курсов, редактирование и удаление существующих курсов;
- отображение списка курсов;
- создание, редактирование и удаление уроков внутри курса;
- взаимодействие с уроком, включающее в себя просмотр теоретического материала урока и его практического задания, а также выполнение данного задания посредством считывания из поля пользовательского программного кода и его проверка.

Стоит отметить, что Пользователи с ролью «Students» не имеют права доступа к созданию и редактированию курсов и уроков. Эта возможность есть только Суперпользователя, Администраторов и Преподавателей.

Чтобы лучше понять логику работы данного приложения, рассмотрим процесс создания нового курса. Он начинается с отправки Пользователем запроса о необходимости выполнения данного действия. Получив этот сигнал, веб–приложение определяет по пути, прописанному в urls–файле, функцию, ко-

торию необходимо вызвать для удовлетворения полученного запроса. Так как курсы могут создавать только Администраторы и Преподаватели, в функции выполняется проверка Пользователя на принадлежность к данным группам. Пользовательский запрос выполняется только в том случае, если ему присвоена соответствующая роль. В противном случае действие выполняться не будет и Пользователь будет перенаправлен на страницу с сообщением об отсутствии прав доступа к данному функционалу.

После проверки, в соответствии полученной из запроса информации о курсе, а именно его название и краткое описание, происходит создание экземпляра курса, информация о котором записывается в соответствующую таблицу в базе данных. Выполнив эти действия, приложение перенаправляет пользователя на страницу со списком имеющихся в системе курсов.

Так как программа практически любого курса разделена на тематические блоки, был реализован необходимый функционал для создания уроков внутри курса. Таким образом, переходя на страницу курса, у Администраторов и Преподавателей в системе есть возможность добавлять в курс новые уроки и редактировать уже имеющиеся.

Процесс создания урока имеет некоторые отличия от процесса создания курса: помимо названия урока необходимо прикрепить к нему нужные текстовые файлы и Python–скрипт с функциями для проведения автоматических тестов практического задания. Все файлы сохраняются в директорию Как уже было сказано ранее, файлы сохраняются в директории «media».

Для удобства Пользователя осваивать материалы урока, был создан удобный интерфейс переключения между теоретическим материалом, текстом задания и полем для ввода решения задания. Практическое задание предполагает под собой написание небольшого Python–скрипта, который удовлетворяет решению поставленной задачи. Написав свой код, Пользователь отправляет запрос веб–приложению на его проверку, после чего запускает процесс компиляции считанного текста.

Процесс обработки и проверки скрипта можно разделить на 3 этапа:

1. Считывание текста из POST–запроса Пользователя с HTML–страницы;
2. Преобразование текста в Python–скрипт и его создание в файл;
3. Проверка Python–скрипт с помощью автоматических тестов с помощью среды тестирования pytest и анализ результатов.

Когда Пользователь отправляет запрос о проверке введенного в поле кода, вызывается соответствующая функция из views-файла, которая запускает процесс обработки полученного текста. Скрипта записывается в файл, после чего в функции запускается подпроцесс с флагом «pytest» и указанием пути расположения файла с автоматическими тестами для данного задания. По завершению процесса начинается обработка его результатов. Здесь возможно 3 варианта развития событий:

1. в коде была допущена ошибка и он не скомпилировался;
2. код корректный, но были пройдены не все тесты;
3. код корректный, все тесты пройдены.

После анализа процесса компиляции и прохождения тестов, функция возвращает Пользователю соответствующее сообщение о результате проверки его кода, тем самым обеспечивая обратную связь с Пользователем. Пример успешного выполнения задания представлен на рис. 2.

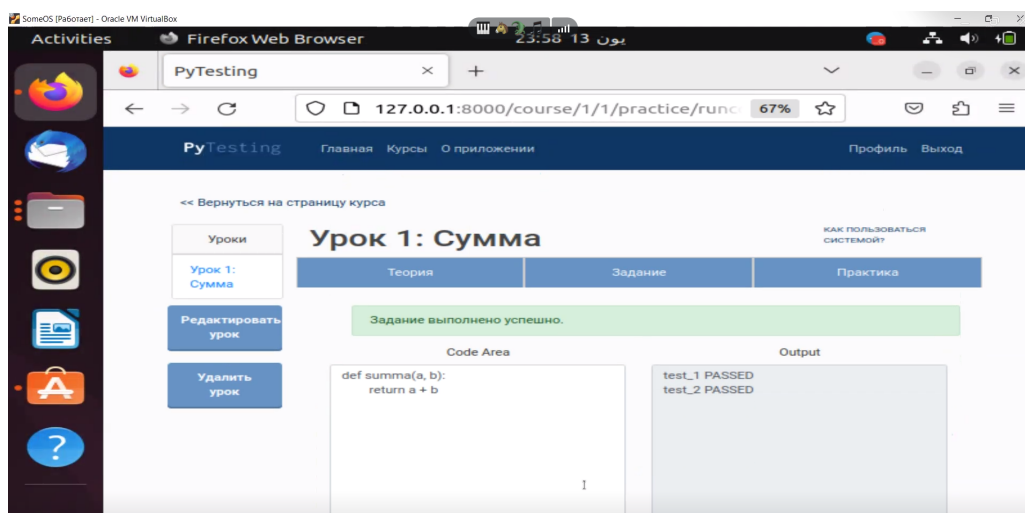


Рисунок 2 – Демонстрация вывода сообщения об успешной сдаче задания

Стоит обратить внимание на то, что по такому же принципу помимо Python владелец приложения может добавить поддержку других языков программирования.

ЗАКЛЮЧЕНИЕ

В процессе написания работы были выполнены следующие задачи:

1. Изучены виды современных платформ, предназначенных для организации онлайн–обучения;
2. Проанализированы преимущества и недостатки цифрового формата обучения;
3. Была создана система организации образовательного процесса, на основе которой было разработано веб–приложение с использованием фреймворка Django, дающее возможность выполнять задания, связанные с программированием на языке Python.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 *Карпенко, М.* Цифровая трансформация образования: актуальные проблемы, опыт, решения / М. Карпенко. — Москва: АЭО, 2021.
- 2 *Шахбанов, Ш.* Технологии цифрового образования : курс лекций / Ш. Шахбанов. — Махачкала: АЛЕФ, 2022.
- 3 Python Documentation by Version [Электронный ресурс]. — URL: <https://docs.djangoproject.com/en/4.2/> (Дата обращения 27.04.2023). Заголовок с экрана. Яз. англ.
- 4 *Shaw, B.* Web Development with Django / B. Shaw, S. Badhwar, A. Bird, D. Chandra, C. Guest. — Birmingham: Packt Publishing, 2021.
- 5 *Дронов, В.* Django. Практика создания Web-сайтов на Python / В. Дронов. — Санкт-Петербург: БХВ-Петербург, 2019.
- 6 *Ахметшин, Д.* Разработка приложений на языке программирования Python с использованием Фреймворка Django / Д. Ахметшин. — Казань: Школа, 2019.
- 7 *Дронов, В.* Django 3.0. Практика создания Web-сайтов на Python / В. Дронов. — БХВ-Петербург, 2021.
- 8 *Vincent, W.* Django for Beginners: Build websites with Python and Django / W. Vincent. — WelcomeToCode, 2022.
- 9 *Melé, A.* Django 3 By Example: Build powerful and reliable Python web applications from scratch, 3rd Edition / A. Melé. — Birmingham: Packt Publishing, 2020.
- 10 *George, N.* Build a Website With Django 3: A complete introduction to Django 3 / N. George. — London: GNW Independent Publishing, 2020.
- 11 Django. The web framework for perfectionists with deadlines [Электронный ресурс]. — URL: <https://www.python.org/doc/versions/> (Дата обращения 18.04.2023). Заголовок с экрана. Яз. англ.
- 12 *Лутц, М.* Python : карманный справочник / М. Лутц. — Санкт-Петербург: Диалектика, 2019.
- 13 *Васильев, А.* Python на примерах : практический курс по программированию / А. Васильев. — Санкт-Петербург: Наука и Техника, 2018.

- 14 *Таннинг, Ж.* Основы программирования в Python: три в одном / Ж. Таннинг. — Владимир: ВлГУ, 2021.
- 15 *Jaworski, M.* Expert Python Programming: Master Python by learning the best coding practices and advanced programming concepts, 4th Edition / М. Jaworski, Т. Ziadé. — Birmingham: Packt Publishing, 2021.
- 16 *Новиков, Б.* Основы технологий баз данных / Б. Новиков, Е. Горшкова. — Москва: Издательский дом «МИФ», 2022.
- 17 *Гринченко, Н.* Базы данных. Программирование на SQL / Н. Гринченко. — Москва: Издательство КУРС, 2023.
- 18 *Токмаков, Г.* Базы данных: язык SQL, программирование баз данных / Г. Токмаков. — Ульяновск: Ульяновский государственный технический университет, 2022.
- 19 *Juba, S.* Learning PostgreSQL 11: A beginner's guide to building high-performance PostgreSQL database solutions, 3rd Edition / S. Juba, A. Volkov. — Birmingham: Packt Publishing, 2019.
- 20 *Schönig, H.* Mastering PostgreSQL 12: Advanced techniques to build and administer scalable and reliable PostgreSQL database applications, 3rd Edition / H. Schönig. — Birmingham: Packt Publishing, 2019.