

открываться список ингредиентов блюда и пошаговый рецепт блюда с текстовыми пояснениями. При прохождении первого блюда и его мини-игр будет доступно следующее блюдо. Открытие следующей страны будет доступно с накоплением определенного количества монет.

В результате для разрабатываемого приложения «*CulinaryTravel*» был выбран клиент-серверный архитектурный подход, такие технологии для клиентской части, как *HTML*, *CSS*, *JavaScript*, и язык программирования *C#* для серверной части приложения, в качестве способа хранения данных – СУБД *MSSQL*, а также разработаны структура базы данных и макет пользовательского интерфейса.

#### Литература

1. Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер. – М. : Вильямс, 2006. – 544 с.
2. Введение в Ajax. – Режим доступа: <http://javascript.ru/ajax/intro>. – Дата доступа: 29.03.2024.
3. Petkovic, D. Microsoft SQL Server 2019: A Beginner's Guide 7th Edition / D. Petrovic. – McGraw-Hill, 2020. – 1643 с.

## WEB-ПРИЛОЖЕНИЕ ПРЕДОСТАВЛЕНИЯ ОБРАЗОВАТЕЛЬНЫХ УСЛУГ ДЛЯ УЧЕНИКОВ СРЕДНЕЙ ШКОЛЫ ПО ХИМИИ И БИОЛОГИИ

А. А. Глушенок

*Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Республика Беларусь*

Научный руководитель В. Н. Шибeko

*Представлены основные проблемы автоматизации учебного процесса между учителями и ученикам, определены функциональные требования, обоснован выбор технологии реализации и архитектура проектируемого web-приложения.*

**Ключевые слова:** сервис, образовательные услуги, автоматизация процесса обучения.

Во все времена родители очень трепетно относятся к образованию своих детей, отдавая их в школы, а некоторые даже в несколько, например, музыкальные или художественные, ведь важно вложить в ребёнка и культурные знания, любовь к окружающему миру, умение видеть красоту и позитивные стороны каких-либо вещей и ситуаций. Однако естественные науки такие как физика, химия, биология, астрономия, география, геология, экология, также важны в жизни каждого человека, иначе, не зная причину естественных процессов, мы бы жили в мире магии.

Для самого учителя иметь подобный интернет-ресурс является хорошей практикой. Для учеников такой учитель всегда будет иметь положительные отзывы, так как он идёт в ногу со временем и находится с учениками «на одной волне», возможность хранить важные материалы для себя, например, планы занятий, методическая литература и учеников, к примеру, задания, полезные статьи и ссылки, материалы для внеклассных и дополнительных занятий, домашние задания, тесты и др.

Современное развитие технологий предоставляет огромное множество средств по реализации *web*-приложений, использующих абсолютно разные языки программирования.

Первым и особо важным для крупных и полнофункциональных сайтов является уровень сервера. Полностью невидимый для конечного пользователя данный уровень представляет собой ядро всего сайта.

Для выполнения сервисной части приложения был выбран строго типизированный объектно-ориентированный язык программирования *Java*. Для реализации поставленной задачи использован *Spring*, или *SpringFramework* – один из самых популярных фреймворков для создания веб-приложений на *Java* в соответствии с [1].

Для *front-end* части выбран фреймворк *Next.js*, обладающий множеством функций, которые делают его привлекательным для разработчиков. Являясь библиотекой для создания пользовательских интерфейсов согласно [2].

В данной работе будет разработано *web*-приложение для предоставления образовательных услуг по предметам «химия» и «биология». В ходе сбора и анализа данных о требованиях к подобным приложениям был выделен следующий необходимый функционал:

– справочники: ученики, классы, модули, предметы, контрольные, тесты, задания, параграфы;

– ведение справочников.

Опираясь на полученные требования к функционалу, для успешного выполнения поставленных задач необходимо обеспечить наличие в приложении следующих пунктов:

– *web*-приложение, основанное на *REST* архитектуре;

– микросервисная архитектура;

– реляционная база данных;

– классы и интерфейсы для доступа и структуризации к базе данных;

– интерфейсы для выполнения *CRUD* операций;

– классы-модели для сущностей баз данных;

– классы-контроллеры для обработки запросов.

Разработана следующая схема базы данных (рис. 1).

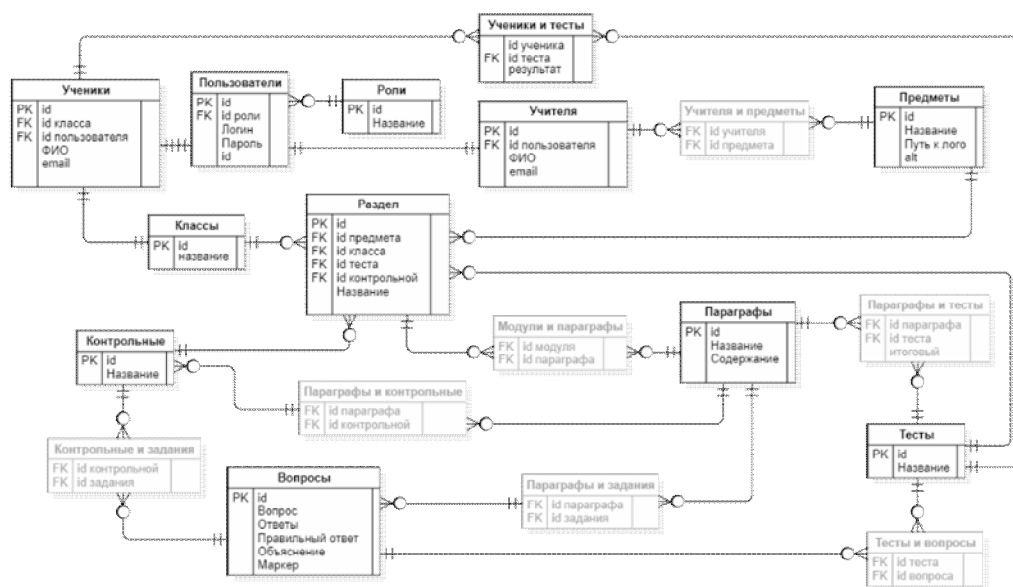


Рис. 1. Реляционная схема базы данных

Для поставленной задачи была выбрана микросервисная архитектура приложения. Микросервисная архитектура предполагает разработку и поддержку приложений с использованием небольших модульных сервисов, а не создание программного обеспечения в виде одного большого унифицированного блока кода (монолита). Ос-

новая концепция архитектуры в том, чтобы разделить сложное приложение на несколько небольших автономных и управляемых компонентов. Это позволяет повысить гибкость разработки, улучшить отказоустойчивость и облегчить поддержку приложения (рис. 2).

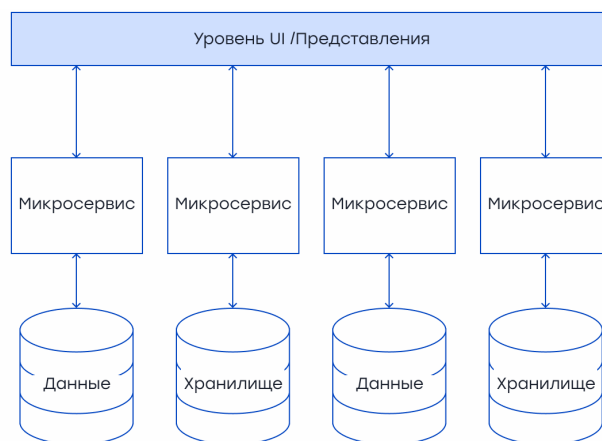


Рис. 2. Архитектура приложения

Для каждого микросервиса выбрана луковичная архитектура, которая является одной из самых популярных и распространённых архитектурных моделей для разработки приложений. Она предоставляет структуру и организацию для построения гибких, масштабируемых и легко поддерживаемых систем. Луковичная архитектура предполагает разделение кода приложения на «слои» с чётко разграниченными обязанностями, представленная на рис. 3.

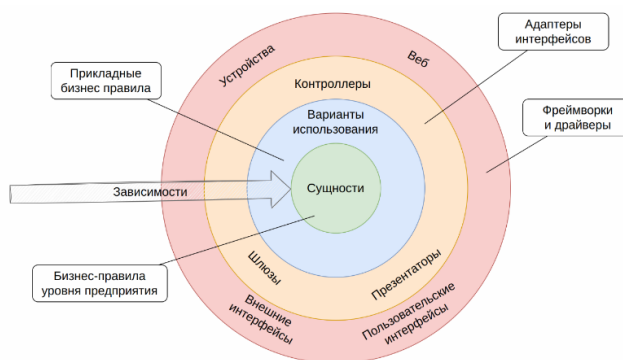


Рис. 3. Архитектура приложения

В результате всех вышеперечисленных факторов был выбран стек технологий, разработана трёхслойная архитектура приложения, схема базы данных.

Основываясь на выполненном анализе, разработана следующая структура приложения (рис. 4).

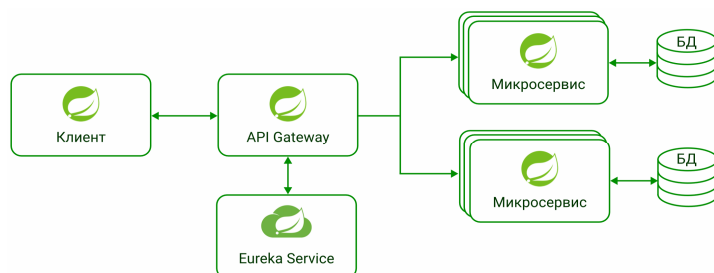


Рис. 4. Структура приложения

Ни одно приложение не может обойтись без авторизации и аутентификации, по этой причине обеспечивается безопасность с использованием *SpringSecurity*. *SpringSecurity* – это фреймворк, предоставляющий механизмы построения систем аутентификации и авторизации, а также другие возможности обеспечения безопасности для корпоративных приложений, созданных с помощью *SpringFramework* согласно [3]. Также добавлен *JWT*. *JWT* – это *JSONWebTokens*, простой и безопасный способ передачи информации между клиентом и сервером с помощью шифрования. *JWT* состоит из трех отдельных частей, разделенных точкой (рис. 5).

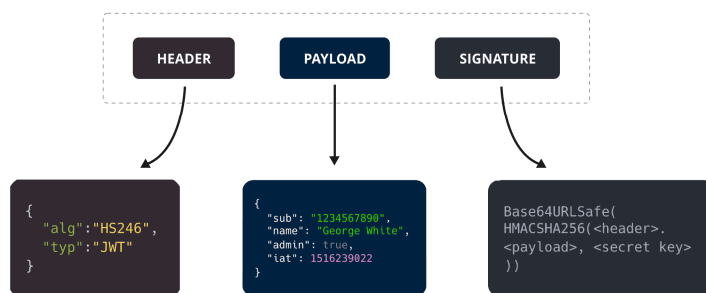


Рис. 5. Структура JWT

В данной работе была поставлена задача по разработке *Web*-приложения для предоставления образовательных услуг для учеников средней школы по классу химии и биологии. Данное приложение позволяет облегчить образовательный процесс как для учеников, так и для учителя. Учителю данное приложение позволит не носить с собой учебники с теоретическими и практическими заданиями, также необходимую методическую литературу или учебные планы. Ученикам позволит не носить с собой и без того много книг, а держать всё в своём смартфоне.

Внедрение данного *web*-приложения позволит облегчить нагрузку на учеников и учителей, автоматизирует процесс проверки знаний, позволит формировать статистику успеваемости учеников для учителя.

#### Литература

1. *Spring*. – Режим доступа: <https://javarush.com/groups/posts/spring-framework-java-1>. – Дата доступа: 11.03.2024.
2. *Next.js*. – Режим доступа: <https://habr.com/ru/companies/auriga/articles/786912/>. – Дата доступа: 25.03.2024.
3. *SpringSecurity*. – Режим доступа: <https://habr.com/ru/articles/203318/>. – Дата доступа 02.04.2023.