

ПРОГРАММНОЕ СРЕДСТВО АВТОМОТИЗАЦИИ ОБУЧЕНИЯ ПРИ ПОДГОТОВКЕ К ЦЕНТРАЛИЗОВАННОМУ ТЕСТИРОВАНИЮ

Тарасевич К. В.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Косарева Е.М. – ассистент кафедры ПИКС

Аннотация. В данной статье рассматривается архитектурный анализ и проектирование программного средства, предназначенного для автоматизации подготовки к централизованному тестированию. Система разработана с учётом современных требований к масштабируемости, отказоустойчивости и высокой доступности.

Ключевые слова: автоматизация обучения, централизованное тестирование, микросервисная архитектура, *Java, Spring, PostgreSQL, Redis, Kafka*.

Введение. Современная образовательная среда активно использует цифровые технологии для повышения эффективности учебного процесса. Особенно актуальным становится применение программных решений в подготовке к централизованному тестированию, которое играет важную роль в системе оценки знаний и является одним из ключевых этапов поступления в высшие учебные заведения.

Традиционные методы подготовки к тестированию зачастую требуют значительных временных затрат и не всегда предоставляют студентам доступ к качественным образовательным материалам. В условиях растущей конкуренции за поступление в вузы необходимость эффективных инструментов для подготовки к экзаменам становится очевидной. Современные образовательные технологии позволяют предоставить студентам доступ к интерактивным материалам, адаптивным тестам и аналитике успеваемости, что значительно улучшает процесс обучения.

Разработка программного средства автоматизации обучения направлена на решение ключевых проблем подготовки к тестированию:

- создание удобной цифровой среды для изучения учебных материалов;
- формирование адаптивной системы тестирования, позволяющей студенту выявить пробелы в знаниях и целенаправленно работать над их устранением.

Основная часть. Разработка образовательного программного обеспечения требует выбора архитектурного подхода, который обеспечит высокую доступность, отказоустойчивость и возможность масштабирования системы в зависимости от роста нагрузки. Основными подходами, которые можно рассматривать при проектировании подобного решения, являются монолитная и микросервисная архитектуры.

Монолитная архитектура предполагает объединение всех функциональных компонентов системы в единое приложение. Такой подход может быть оправдан на начальном этапе разработки, так как он упрощает процесс развертывания и управления кодом. Однако с увеличением функциональности и нагрузки монолитное приложение становится сложным в поддержке, а его масштабирование требует значительных ресурсов.

Микросервисная архитектура, напротив, предполагает разделение системы на отдельные сервисы, каждый из которых отвечает за конкретную функциональность. Такой подход обладает рядом преимуществ:

Для реализации программного средства была выбрана микросервисная архитектура, обеспечивающая гибкость и масштабируемость системы. Основные компоненты включают:

- гибкость – возможность независимо развивать и обновлять отдельные компоненты системы;

– масштабируемость – возможность масштабирования отдельных сервисов в зависимости от нагрузки;

– отказоустойчивость – выход из строя одного микросервиса не приводит к полной недоступности системы.

С учетом требований к надежности и масштабируемости в проекте была выбрана микросервисная архитектура, обеспечивающая гибкость в разработке и адаптацию к изменяющимся потребностям пользователей.

Система автоматизации обучения при подготовке к централизованному тестированию включает в себя несколько ключевых компонентов, обеспечивающих её функциональность:

1 Управление образовательным контентом: Этот модуль отвечает за хранение и предоставление пользователям учебных материалов. Он реализован с использованием реляционной базы данных (*PostgreSQL*) для хранения структурированной информации и поискового движка *Elasticsearch* для быстрого поиска нужных данных.

2 Система адаптивного тестирования: Для оценки знаний пользователей и формирования персонализированной стратегии обучения разработан модуль тестирования. Он использует аналитическую базу данных, которая позволяет анализировать результаты тестов в режиме реального времени.

3 Обработка событий и взаимодействие сервисов: Взаимодействие между микросервисами организовано с помощью *Apache Kafka*, обеспечивающей асинхронную передачу сообщений и обработку событий. Такой подход позволяет системе эффективно масштабироваться и минимизировать задержки в обработке данных.

4 Кэширование и ускорение работы: Для оптимизации работы системы используется *Redis*, который позволяет кэшировать часто запрашиваемые данные, снижая нагрузку на основные базы данных и увеличивая скорость отклика системы.

5 Пользовательский интерфейс: Фронтенд-система разработана с использованием современных веб-технологий, что позволяет пользователям взаимодействовать с образовательным контентом через удобный интерфейс.

Архитектурная схема системы представлена на рисунке 1.

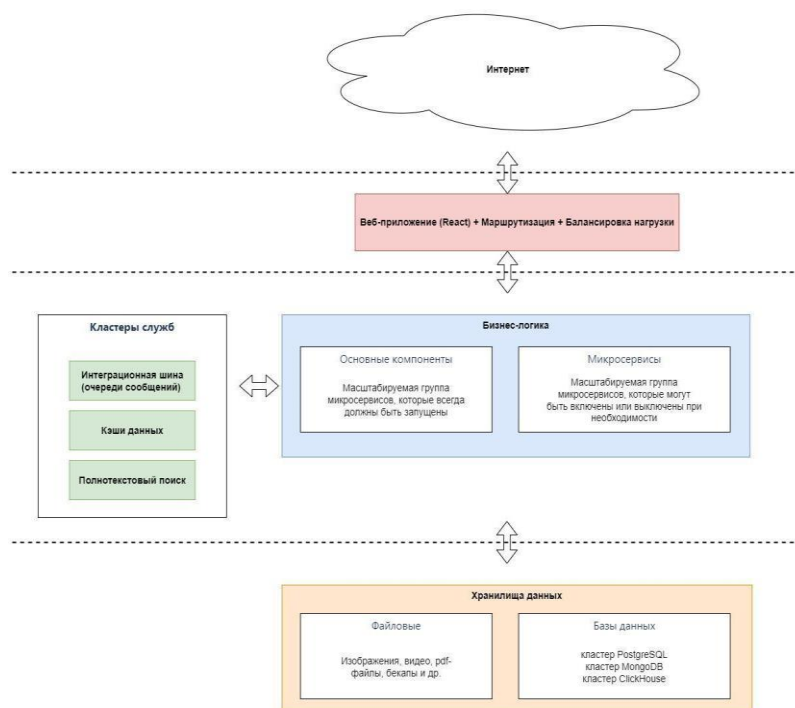


Рисунок 1 – Архитектурная схема системы

Заключение. Разработка программного средства автоматизации обучения при подготовке к централизованному тестированию представляет собой важный шаг в цифровизации образовательного процесса. Микросервисная архитектура, использованная в проекте, обеспечивает гибкость, масштабируемость и отказоустойчивость системы, позволяя эффективно обрабатывать большие объемы данных и адаптироваться к различным образовательным сценариям.

Применение современных технологий, таких как *Apache Kafka*, *Elasticsearch* и прочих перечисленных ранее технологий, позволяет оптимизировать процессы анализа данных, улучшая персонализированный подход к обучению. В результате пользователи получают доступ к удобной цифровой среде, в которой подготовка к тестированию становится более структурированной, эффективной и интерактивной.

Предложенная архитектура и технологические решения обеспечивают возможность дальнейшего развития системы, расширения ее функциональности и интеграции с другими образовательными платформами, что делает ее перспективным инструментом для цифрового обучения.

Список литературы

1. Fowler M. *Microservices: a Definition of This New Architectural Term*. [Электронный ресурс] – Режим доступа: <https://martinfowler.com/articles/microservices.html>
2. *Apache Kafka Documentation*. [Электронный ресурс] – Режим доступа: <https://kafka.apache.org/documentation/>
3. *ClickHouse Documentation*. [Электронный ресурс] – Режим доступа: <https://clickhouse.com/docs/en/>

УДК 004.415.2

SOFTWARE TOOL FOR AUTOMATION OF TRAINING IN PREPARATION FOR CENTRALIZED TESTING

Tarasevich K. V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Kosareva E.M. – assistant of the department of ICSD

Annotation. This article discusses the architectural analysis and design of a software tool designed to automate preparation for centralized testing. The system is developed taking into account modern requirements for scalability, fault tolerance and high availability. Various architectural approaches are considered, including monolithic and microservice architecture. The key components of the system, their interaction and design principles that ensure efficient data processing and support for a large number of users are also described.

Keywords: automation of training, centralized testing, microservice architecture, Java, Spring, PostgreSQL, Redis, Kafka.