

УДК 004.588

**МИКРОСЕРВИСНАЯ АРХИТЕКТУРА НА ПРИМЕРЕ
АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ ПРОВЕРКИ
ПРАКТИЧЕСКИХ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ ПО РАЗДЕЛУ
«КИНЕМАТИКА»**

Шерматов А.А., студент гр. 4411, IV курс

Научный руководитель: Кремлева Э. Ш., доцент кафедры прикладной
математики и информатики

Казанский Национальный Исследовательский Технический Университет им.
А. Н. Туполева – КАИ

г. Казань

Аннотация: в статье рассматривается подход к разработке автоматизированной системы для оценки практических компетенций студентов в разделе физики "Кинематика" с использованием микросервисной архитектуры. Описываются основные преимущества микросервисов — гибкость, масштабируемость и отказоустойчивость — в контексте образовательных технологий. Приводятся теоретические основы кинематики и специфика формирования практических навыков у обучающихся. Предложена архитектура системы, включающая независимые сервисы для генерации задач, проверки решений и анализа результатов, а также рассмотрены этапы ее реализации и тестирования.

Ключевые слова: автоматизация, кинематика, микросервисная архитектура, образовательные технологии, FastAPI, MySQL, React, TypeScript, оценка знаний, индивидуализация обучения, масштабируемость

1. Введение

Современное образование сталкивается с задачей повышения эффективности оценки знаний и навыков учащихся в условиях стремительного развития цифровых технологий [9]. Традиционные методы проверки, такие как письменные задания или устные опросы, часто оказываются трудоемкими и субъективными, что затрудняет объективную оценку практических компетенций учащихся. Раздел физики "Кинематика", изучающий законы движения тел [1], играет важную роль в формировании аналитического мышления и способности применять теоретические знания на практике [4]. Однако ручная проверка решений задач по кинематике требует значительных

временных ресурсов и не всегда позволяет обеспечить индивидуальный подход к каждому учащемуся.

В этом контексте автоматизация процесса оценки практических компетенций становится актуальным решением, позволяющим повысить скорость, точность и масштабируемость проверки [6]. Разработка автоматизированной системы с использованием микросервисной архитектуры открывает новые возможности для создания гибких и устойчивых образовательных инструментов [5]. Микросервисы, благодаря своей модульности и независимости, идеально подходят для реализации сложных систем, где требуется интеграция различных функций — от управления задачами до анализа результатов. Цель данной статьи — рассмотреть подход к проектированию и реализации такой системы, способной эффективно оценивать практические компетенции учащихся по кинематике. В работе будут представлены теоретические основы, архитектурные решения, этапы разработки и перспективы применения системы в образовательной практике.

2. Теоретические основы

2.1. Кинематика как раздел физики

Кинематика — это раздел механики, изучающий движение тел без учета причин, его вызывающих [2]. Она сосредотачивается на описании траекторий, скоростей и ускорений, опираясь на такие фундаментальные понятия, как перемещение, время и координаты. Основной задачей кинематики является математическое моделирование движения, будь то равномерное прямолинейное движение, равноускоренное падение тела или сложные траектории, такие как параболическое движение снаряда [1]. В отличие от динамики, которая анализирует силы и их влияние на движение, кинематика ограничивается геометрическим и временным аспектами.

2.2. Практические компетенции обучающихся

Практические компетенции в контексте изучения физики представляют собой способность учащихся применять теоретические знания для решения конкретных задач, анализа явлений и интерпретации результатов [3]. В отличие от простого запоминания формул или законов, такие компетенции включают навыки моделирования, вычислений и критического мышления, что делает их ключевым элементом современного образования. В разделе "Кинематика" практические компетенции проявляются в умении учащихся анализировать условия движения, выбирать подходящие уравнения, выполнять расчеты и проверять физическую корректность полученных ответов.

2.3. Автоматизация проверки компетенций

Автоматизация проверки практических компетенций учащихся представляет собой процесс использования программных средств для оценки знаний и навыков, минимизируя участие человека [9]. В контексте изучения кинематики это означает создание систем, способных принимать ответы студентов, проверять их корректность и предоставлять обратную связь [6].

Такой подход становится все более востребованным в условиях роста числа учащихся, необходимости индивидуализации образования и ограниченности временных ресурсов преподавателей.

3. Микросервисная архитектура: концепция и преимущества

3.1. Определение микросервисной архитектуры

Микросервисная архитектура — это подход к разработке программного обеспечения, при котором приложение разделяется на набор небольших, независимых сервисов, каждый из которых выполняет конкретную функцию и взаимодействует с другими через четко определенные интерфейсы [5]. В отличие от традиционной монолитной архитектуры, где все компоненты системы (логика, интерфейс, база данных) тесно связаны и работают как единое целое, микросервисы представляют собой автономные модули, которые можно разрабатывать, развертывать и масштабировать отдельно друг от друга.

Основные принципы микросервисной архитектуры включают:

- *Декомпозицию*: разделение системы на небольшие, управляемые части.
- *Независимость*: каждый сервис имеет собственную базу данных и может обновляться без влияния на остальные.
- *Автономность*: сервисы разрабатываются и поддерживаются отдельными командами.
- *Интероперабельность*: взаимодействие через стандартизованные API.

3.2. Преимущества для образовательных систем

Микросервисная архитектура предоставляет значительные преимущества для образовательных систем, особенно в контексте автоматизации проверки практических компетенций. Ее модульный и гибкий подход позволяет создавать надежные, масштабируемые и адаптируемые решения, отвечающие современным требованиям цифровизации образования.

4. Проектирование автоматизированной системы

4.1. Цели и задачи системы

Разработка автоматизированной системы для проверки практических компетенций учащихся по разделу "Кинематика" с применением микросервисной архитектуры направлена на решение ключевых проблем традиционных методов оценки и повышение качества образовательного процесса. Основная цель — создание инструмента, который минимизирует участие учителя в рутинной проверке решений, обеспечивая объективность и индивидуализацию обучения [3].

Цели и задачи:

1. *Автоматизация оценки компетенций* - создание инструмента, который минимизирует участие учителя в рутинной проверке решений, позволяя сосредоточить его усилия на методической работе и взаимодействии с учащимися.
2. *Обеспечение объективности* - исключение субъективных факторов при оценке знаний и навыков, таких как различия в подходах преподавателей

или их усталость, за счет использования единых алгоритмических критерииев.

3. *Повышение доступности обучения* - предоставление учащимся возможности проходить проверку знаний в удобное время и получать немедленную обратную связь, что особенно актуально для дистанционного и массового образования.
4. *Индивидуализация образовательного процесса* - адаптация заданий и обратной связи к уровню подготовки каждого учащегося для формирования устойчивых практических компетенций в кинематике.

4.2. Архитектура системы

Архитектура автоматизированной системы для проверки практических компетенций обучающихся по разделу "Кинематика" основана на принципах микросервисной архитектуры, что обеспечивает ее модульность, масштабируемость и устойчивость. Система состоит из независимых сервисов: управление задачами, проверка решений и пользовательский интерфейс, взаимодействующих через REST API [7]. Такое разделение позволяет гибко адаптировать систему к различным образовательным сценариям и упрощает ее дальнейшее развитие.

Обзор компонентов (микросервисов):

1. Сервис управления задачами и темами:
 - *функциональность* – система должна поддерживать различные типы заданий, такие как числовые расчеты (время падения), аналитические (вывод формул), которые составляет преподаватель;
 - *технологии* - для вычислений используется Python.
2. Сервис проверки решений:
 - *функциональность* - проверяет ответы учащихся, сравнивая их с правильными решениями. Поддерживает разные форматы: числа (например, $s=40\text{ м}$), текстовые пояснения.
 - *технологии* - алгоритмы обработки данных.
3. Сервис пользовательского интерфейса:
 - *функциональность* - обеспечивает взаимодействие учащихся и преподавателей с системой через веб-приложение, отображает задачи, принимает ответы, показывает результаты.
 - *технологии* – используются JavaScript (React), REST API для связи с другими сервисами.
4. Взаимодействие между сервисами:
 - *протоколы связи* - основной способ взаимодействия — REST API с использованием HTTP-запросов (GET - для получения задачи, POST - для отправки ответа).

4.3. Технологический стек

Технологический стек автоматизированной системы для проверки практических компетенций по кинематике основан на современных инструментах, обеспечивающих высокую производительность, безопасность и

удобство разработки в рамках микросервисной архитектуры. В рамках данного примера используются Python с FastAPI для бэкенда, React с TypeScript для фронтента [10] и MySQL для хранения данных [8]. База данных MySQL обеспечивает структурированное хранение задач и ответов учащихся, что соответствует принципам проектирования баз данных [8]. Ниже, более подробно, представлены описания стеков и его применения.

Языки программирования и фреймворки:

- **Python с FastAPI:** FastAPI — это асинхронный веб-фреймворк, используемый для разработки микросервисов, таких как сервис управления задачами и сервис проверки решений. Благодаря поддержке асинхронного программирования (async/await) и автоматической генерации документации API (на основе OpenAPI), FastAPI идеально подходит для создания высокопроизводительных REST API.

- **TypeScript** используется для написания фронтента и серверной логики пользовательского интерфейса. TypeScript добавляет статическую типизацию к JavaScript, что повышает надежность кода и упрощает его поддержку, особенно при работе с динамическими данными, такими как ответы учащихся или графики движения.

Фронтенд

- **React (с TypeScript)** - библиотека для создания интерактивного пользовательского интерфейса. React используется для построения веб-приложения, где учащиеся получают задачи, вводят ответы (числовые, текстовые или графические) и видят обратную связь. TypeScript в связке с React обеспечивает строгую типизацию компонентов, что минимизирует ошибки при работе с данными от бэкенда. Например, учащийся может построить график зависимости скорости от времени с помощью библиотеки вроде Chart.js, интегрированной в React.

База данных

- **MySQL** — реляционная база данных, выбранная для хранения структурированных данных системы. MySQL используется для записи задач, ответов учащихся, результатов проверки и статистики успеваемости. Например, таблица tasks может содержать поля id, description, author_id, theme_id, а таблица submissions — user_id, task_id, answer, is_correct. Высокая производительность и поддержка транзакций делают MySQL подходящим выбором для образовательной системы с большим объемом данных.

5. Заключение

Разработка автоматизированной системы для проверки практических компетенций обучающихся по разделу "Кинематика" с применением микросервисной архитектуры демонстрирует значительный потенциал для модернизации образовательного процесса [5]. Автоматизация не только снижает нагрузку на учителей, но и предоставляет учащимся немедленную обратную связь, способствуя более глубокому освоению кинематики через практику [4]. Перспективы включают использование технологий Интернета вещей для оптимизации образовательных ресурсов [7].

Использование технологического стека, включающего FastAPI, MySQL, React и TypeScript, обеспечивает высокую производительность, надежность и удобство взаимодействия с системой. Разделение функций на независимые сервисы — генерацию задач, проверку решений, анализ результатов и интерфейс — позволяет оптимизировать разработку и эксплуатацию, а также открывает возможности для дальнейшего расширения системы на другие разделы физики или дисциплины. Автоматизация не только снижает нагрузку на учителей, но и предоставляет учащимся немедленную обратную связь, способствуя более глубокому освоению кинематики через практику.

Вместе с тем внедрение подобной системы требует тщательной проработки алгоритмов проверки, интеграции компонентов и учета технических ресурсов образовательных учреждений. Рекомендуется начинать с пилотного внедрения, чтобы оценить эффективность системы на практике и скорректировать ее функционал. Перспективы дальнейших исследований включают интеграцию искусственного интеллекта для анализа сложных решений и изучение влияния системы на успеваемость учащихся. Таким образом, предложенный подход демонстрирует значительный потенциал для трансформации образовательных технологий, делая процесс обучения более доступным, объективным и ориентированным на практические результаты.

Список литературы

1. Савельев И. В. Курс общей физики. Т. 1: Механика. — Москва: Лань, 2019. — 432 с.
2. Ландау Л. Д., Лифшиц Е. М. Теоретическая физика. Т. 1: Механика. — Москва: Физматлит, 2020. — 224 с.
3. Кремлева Э. Ш., Валитова Н. Л., Новикова С. В. Современные e-learning системы обучения математике студентов естественно-научных направлений // Образовательные технологии и общество. — 2018. — Т. 21, № 3. — С. 349–371.
4. Кремлева Э. Ш., Новикова С. В. Программные средства поддержки самостоятельной работы студентов в рамках курса "Компьютерное моделирование процессов и систем" для студентов технических вузов // Образовательные технологии и общество. — 2018. — Т. 21, № 1. — С. 363–387.
5. Валитова Н. Л., Кремлева Э. Ш. Методология создания программно-методических комплексов дистанционного обучения с мультиязычной поддержкой преподавания для магистров с не-инженерным базовым образованием (на примере курса "Методы оптимизации") // Образовательные технологии и общество. — 2019. — Т. 22, № 1. — С. 175–184.
6. Снегуренко А. П., Сосновский С. А., Новикова С. В., Яхина Р. Р., Валитова Н. Л., Кремлева Э. Ш. Применение инструментов электронного обучения для международной академической мобильности неанглоязычных студентов-математиков // Интеграция образования. — 2019. — Т. 23, № 1 (94). — С. 8–22.
7. Каблуков В. Р., Валитова Н. Л., Кремлева Э. Ш. Подготовка базы данных использования Интернета вещей для регулирования потребления энергии // Научный аспект. — 2023. — Т. 13, № 4. — С. 1587–1592.
8. Келлер А. Р., Кремлева Э. Ш. Проектирование базы данных для кинотеатра // Научный аспект. — 2024. — Т. 28, № 3. — С. 3541–3546.
9. Иванов А. А., Петрова Е. В. Автоматизация образовательных процессов: современные подходы и технологии // Вестник образования. — 2022. — № 3. — С. 45–52.
10. Чинн П. React быстро: веб-приложения на React, JSX, Redux и GraphQL. — Санкт-Петербург: Питер, 2019. — 480 с.