

МИКРОСЕРВИСНАЯ АРХИТЕКТУРА: РЕШЕНИЕ ДЛЯ УПРАВЛЕНИЯ ЦИФРОВЫМИ ЭКОСИСТЕМАМИ В УНИВЕРСИТЕТАХ

MICROSERVICE ARCHITECTURE: A SOLUTION FOR MANAGING DIGITAL ECOSYSTEMS AT UNIVERSITIES

**D. Ogorodov
N. Ivanov**

Summary. This scientific article examines the importance and advantages of implementing microservice architecture in the context of higher education institutions. The relevance of the topic is due to the need to create a unified digital ecosystem capable of coping with growing amounts of data and ensuring high flexibility and scalability of university information systems in the context of digital transformation. The purpose of the work is to substantiate the transition from monolithic to microservice architecture, identify the key principles of its implementation and evaluate the expected results. The study examined the main limitations of monolithic systems, developed strategies for the transition to a microservice architecture, and provided examples of real-world cases demonstrating the successful application of this technology. The results show that the microservice architecture improves operational efficiency, simplifies application development and testing, and enables individual deployment of each service. The materials of the article will be useful for universities seeking to modernize their digital systems, and can be used to create an educational digital environment at the national level.

Keywords: microservice architecture, monolithic architecture, digital ecosystem, universities, higher education, scalability, flexibility.

Огородов Дмитрий Владимирович
Аспирант, ФГБОУ ВО «РГУ им. А.Н. Косыгина»
ogorodov-dv@rguk.ru

Иванов Никита Геннадьевич
ФГБОУ ВО «РГУ им. А.Н. Косыгина»
ivanov-ng@rguk.ru

Аннотация. В статье исследуется необходимость и преимущества внедрения микросервисной архитектуры в контексте высших учебных заведений. Актуальность темы обусловлена необходимостью создания единой цифровой экосистемы, способной справляться с растущими объемами данных и обеспечивать высокую гибкость и масштабируемость университетских информационных систем в условиях цифровой трансформации. Цель работы — обосновать переход от монолитной к микросервисной архитектуре, выявить ключевые принципы ее внедрения и оценить ожидаемые результаты. В ходе исследования были изучены основные ограничения монолитных систем, разработаны стратегии перехода на микросервисную архитектуру, а также приведены примеры реальных кейсов, демонстрирующих успешное применение данной технологии. Полученные результаты показывают, что микросервисная архитектура способствует повышению операционной эффективности, упрощает разработку и тестирование приложений, а также обеспечивает возможность индивидуального развертывания каждого сервиса. Материалы статьи будут полезны для университетов, стремящихся к модернизации своих цифровых систем, и могут быть использованы для формирования образовательной цифровой среды на национальном уровне.

Ключевые слова: микросервисная архитектура, монолитная архитектура, цифровая экосистема, университеты, высшее образование, масштабируемость, гибкость.

Введение

В последние годы, высшие учебные заведения по всей России активно внедряют цифровые инструменты в образовательные и административные процессы. Тенденции к цифровизации были сформированы на совете Безопасности 25 июля 2007 года, где была утверждена «Стратегия развития информационного общества России», а также «Концепция системной интеграции информационных технологий (ИТ) в высшей школе», «Концепция развития сети телекоммуникаций в системе высшего образования Российской Федерации». [1]

Однако, несмотря на очевидные преимущества цифровизации, многие высшие учебные заведения сталкиваются с проблемами управления цифровыми экосистемами. Одной из ключевых проблем является использование устаревшей монолитной архитектуры

информационных систем, которая ограничивает масштабируемость, гибкость и оперативность в адаптации новых технологий. Такая архитектура становится препятствием для полноценного функционирования университетских цифровых экосистем, особенно в условиях роста объемов данных и расширения спектра цифровых сервисов. В настоящей статье рассматривается возможность решения указанных проблем посредством перехода на микросервисную архитектуру. Автор анализирует преимущества и принципы внедрения микросервисов, а также показывает, как они могут революционизировать подход к управлению цифровыми экосистемами в университетах. Материалы исследования позволят образовательным учреждениям не только повысить эффективность работы, но и сформировать единую образовательную среду, способствующую реализации национальных целей цифровизации.

Материалы и методы

В данной статье использованы различные научные методы исследования, которые обеспечивают глубину анализа и объективность выводов. Основными методами стали анализ, синтез и обобщение. Метод анализа позволил рассмотреть особенности микросервисной архитектуры, ее преимущества и недостатки в контексте цифровизации высших учебных заведений. Метод синтеза способствовал интеграции информации из различных источников для формирования целостного представления о проблеме. Обобщение использовано для выделения ключевых закономерностей и тенденций в управлении цифровыми экосистемами университетов. Для систематизации и обобщения информации применялись результаты научных исследований отечественных авторов, что обеспечило достоверность и актуальность рассматриваемых данных. В основу настоящей работы легли труды таких исследователей, как Н.А. Бородин, И.Б. Богданов, А.Г. Кирилов, О.В. Нестерук и других, которые внесли значительный вклад в изучение информационных технологий и их применения в образовательной сфере. Также в исследовании учитывались данные национальных стратегий и проектов, таких как «Стратегия развития информационного общества России» и национальный проект «Образование», а также практические примеры внедрения микросервисной архитектуры в образовательных учреждениях. Такой подход позволил не только рассмотреть проблему в теоретическом аспекте, но и представить ее практическую значимость.

Результаты и обсуждение

В настоящее время стратегия цифровизации реализуется на основе национального проекта «Образование» утвержденного президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (2019–2024 годы) [2]. Цифровизация значительно оптимизирует работу университетов и повышает уровень образования, но, к сожалению, многие высшие учебные заведения сталкиваются с проблемами в управлении своей цифровой экосистемой из-за отсутствия единого подхода в разработке и внедрении продуктов.

Традиционная монолитная архитектура, используемая в университетах, приводит к созданию системы, которую трудно поддерживать. Микросервисная архитектура же предлагает гибкое и масштабируемое решение, которое может революционизировать то, как университеты управляют своей цифровой экосистемой.

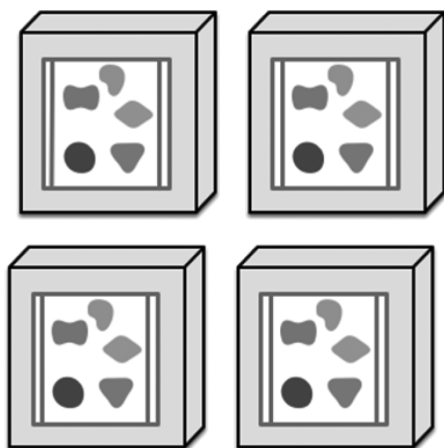
Микросервисная архитектура — это подход к разработке программного обеспечения, который предполагает разбиение приложения на более мелкие автономные сервисы. Каждый микросервис отвечает за определенную задачу и может взаимодействовать с другими продуктами через API. [3-5]

Основным недостатком монолитной архитектуры, повсеместно используемой в университетах, является ее ограниченность в масштабируемости. Когда цифро-

A monolithic application puts all its functionality into a single process...



... and scales by replicating the monolith on multiple servers



A microservices architecture puts each element of functionality into a separate service...



... and scales by distributing these services across servers, replicating as needed.

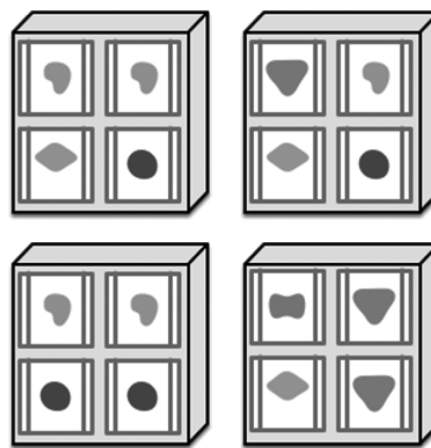


Рис. 1. Пример монолитной и микросервисной архитектуры [3]

вая экосистема ВУЗа растет, появляется все больше сервисов, увеличивается кодовая база, монолит становится слишком большим и сложным для управления и разработки, что приводит к деградации производительности и отказам в работе системы.

В свою очередь микросервисная архитектура может предоставить цифровым экосистемам университетов ряд преимуществ, таких как: масштабируемость и гибкость цифровой экосистемы; возможность развертывания каждого сервиса отдельно; упрощение разработки и тестирования приложения в целом, благодаря независимости микросервисов друг от друга.

В настоящее время на рынке информационных технологий России наблюдается активное внедрение микросервисной архитектуры. Одна из ведущих транснациональных компаний в этой области в июле 2022 года представила открытый исходный код фреймворка «userver», предназначенного для создания высоконагруженных приложений. Этот инновационный фреймворк обеспечивает быстрое создание эффективных микросервисов на языке C++ и уже успешно применяется в различных сервисах, включая службы доставки, торговые площадки и другие. [6]

Компания начала свой путь в области разработки программного обеспечения с использованием монолитной архитектуры, где все компоненты системы были интегрированы в одно приложение. Однако, с ростом бизнеса и расширением спектра сервисов, монолитная архитектура стала препятствием для дальнейшего развития.

В ответ на эти вызовы была разработана стратегия перехода с монолитной системы на микросервисную архитектуру. В рамках этой стратегии были выделены следующие основные принципы внедрения:

- Переход должен быть максимально простым для разработчиков;
- В новой системе необходимо иметь возможность использовать части старого кода и действующих разработчиков;
- Необходимо использовать язык программирования, не зависящий от одного вендора/компании, являющийся типизированным. [7]

Хоть на российском рынке уже и существуют успешные примеры использования микросервисной архитектуры, на данный момент это касается только бизнес решений. В сфере высшего образования приоритетом остается монолитная система разработки с децентрализованным использованием вендорных сервисов.

Большинство университетов в России успешно применяют готовые программные продукты, предназна-

ченные для автоматизации работы ВУЗа. Это включает управление студенческим контингентом, мониторинг учебного процесса, организацию и ведение приемной кампании и т.д. Однако основная проблема таких продуктов заключается в том, что они охватывают только образовательную часть жизнедеятельности университета. [8]

В связи с этим, разработчики стремятся реализовать на своих платформах и другие более узко специализированные решения, такие как системы для документооборота, бухгалтерию, отдел кадров и т. д. Хотя эти платформы позиционируют себя как немонолитное решение и набор механизмов, которые могут применяться как по отдельности, так и совместно, в любых сочетаниях, на практике они воспринимаются пользователями (университетами) иначе.

При внедрении этих продуктов университеты проходят множество этапов доработки под свои особенности образовательного процесса. Это связано с тем, что эти продукты являются комплексными и создаются для общего использования, тем самым пренебрегая вопросом использования возможностей масштабируемости системы.

Внося собственные правки в существующие системы, эти программные продукты становятся монолитными, поскольку дальнейшие общие обновления полностью аннулируют внесенные изменения.

Еще одной важной особенностью работы университетов является использование оптимального решения в конкретной задаче. Это приводит к повышению количества вендорных и опенсорс сервисов, таких как различные программные комплексы, системы управления контентом, сервисы для проведения онлайн-экзаменов, системы управления обучением и т.д.

Таким образом, в настоящий момент российские университеты можно разделить на два типа. К первому типу можно отнести университеты, использующие собственные или готовые монолитные решения. Ко второму, наиболее распространенному типу, относятся университеты, использующие монолитную систему в качестве основной и сторонние сервисы, не связанные с общей архитектурой (рис. 2).

В отличие от бизнеса и компаний, университетам гораздо проще осуществить переход с монолитной системы на микросервисную архитектуру. Это связано с относительно малым числом пользователей, существующими готовыми решениями на рынке, а также наличием квалифицированных специалистов. Немаловажной особенностью внедрения микросервисной архитектуры является ее обособленность на первоначальных этапах разработки, что позволяет создавать продукты с парал-

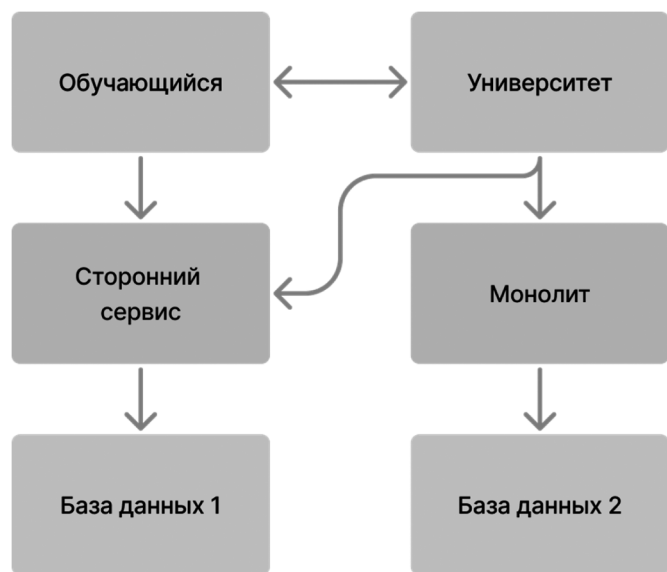


Рис. 2. Смешанная архитектура

тельно существующей монолитной системой, а в дальнейшем и параллельная работа до момента полного замещения.

При переходе на микросервисную архитектуру стоит учитывать, что деятельность университета можно разделить на несколько направлений: [9]

- образовательная;
- воспитательная (социальная);
- научная;
- имиджевая;
- дополнительная учебная;
- внеучебная;
- производственная;
- методическая;
- административная.

В первую очередь, потребуется определить приоритетные направления, сервисы которых необходимы для создания единой цифровой экосистемы, такие как: еди-

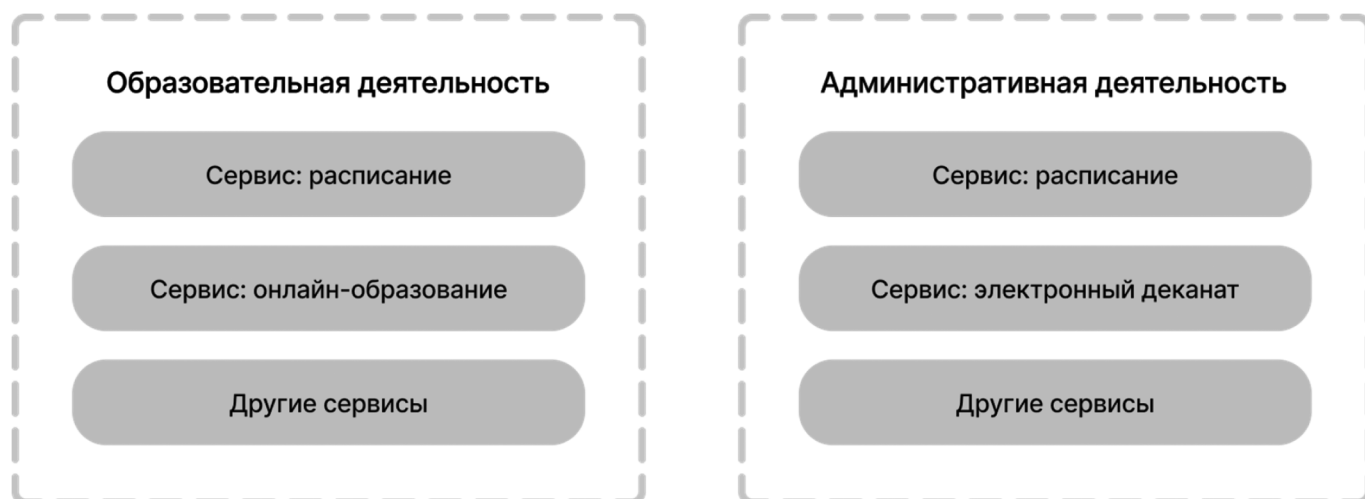


Рис. 3. Первичные сервисы направлений

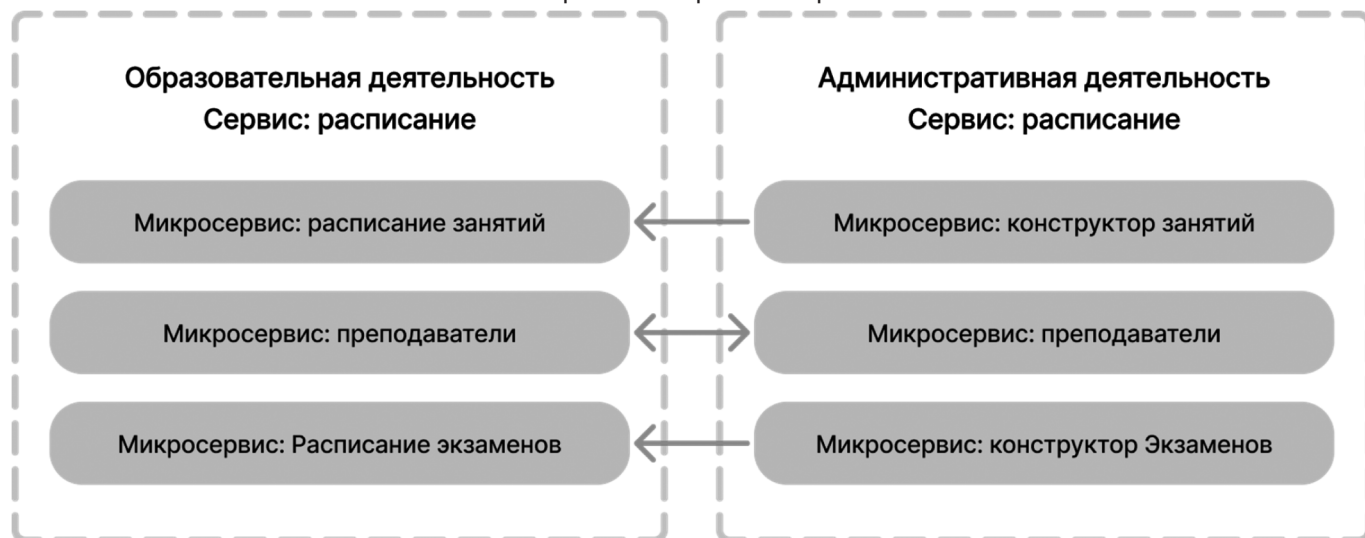


Рис. 4. Пример проектирования микросервисов

ный сервис авторизации, система управления обучением, информационные системы для обучающихся и др. В связи с гибкостью разработки и масштабируемости микросервисной архитектуры, нет нужды разработать сервисы, охватывающие все направления, на первом этапе достаточно сделать акцент на нескольких, которые станут основой экосистемы (рис. 3).

На рис. 3 приведен пример выбора первичных сервисов по направлениям образовательной и научной деятельности. Важно отметить, что сервисы могут дублироваться, например, «Сервис: Расписание». Студенты, преподаватели и административный персонал взаимодействуют с данным сервисом, но со специализированным интерфейсом, на основе ролей и уровня доступа. Как только определены основные сервисы, их необходимо спроектировать на основе микросервисов, каждый из которых должен отвечать за определенную задачу и взаимодействовать друг с другом через API (рис. 4).

На рис. 4 видно, как работают и взаимодействуют микросервисы, некоторые уникальные для каждого сер-

виса, разработанные для решения конкретной задачи, а другие обращаются друг к другу, чтобы решать смежные задачи. Такой метод использования микросервисов значительно облегчает разработку продуктов, повышает ее отказоустойчивость и производительность. Следующим этапом является непосредственная разработка, где каждый микросервис должен разрабатываться независимо и должен быть тщательно протестирован, для предотвращения нежелательных сценариев в будущем. В заключение необходимо провести развертывание микросервисов и управления ими. Данный этап включает в себя настройку системы для мониторинга микросервисов и обеспечения их бесперебойной работы.

На рис. 5 представлен пример реализации сервиса расписания для трех типов ролей, с использованием микросервисной архитектуры. Данный пример показывает, как можно комбинировать различные микросервисы для решения общей задачи, а также демонстрирует простоту внедрения новых микросервисов для безграничного расширения цифровой экосистемы университета (Рис. 6).

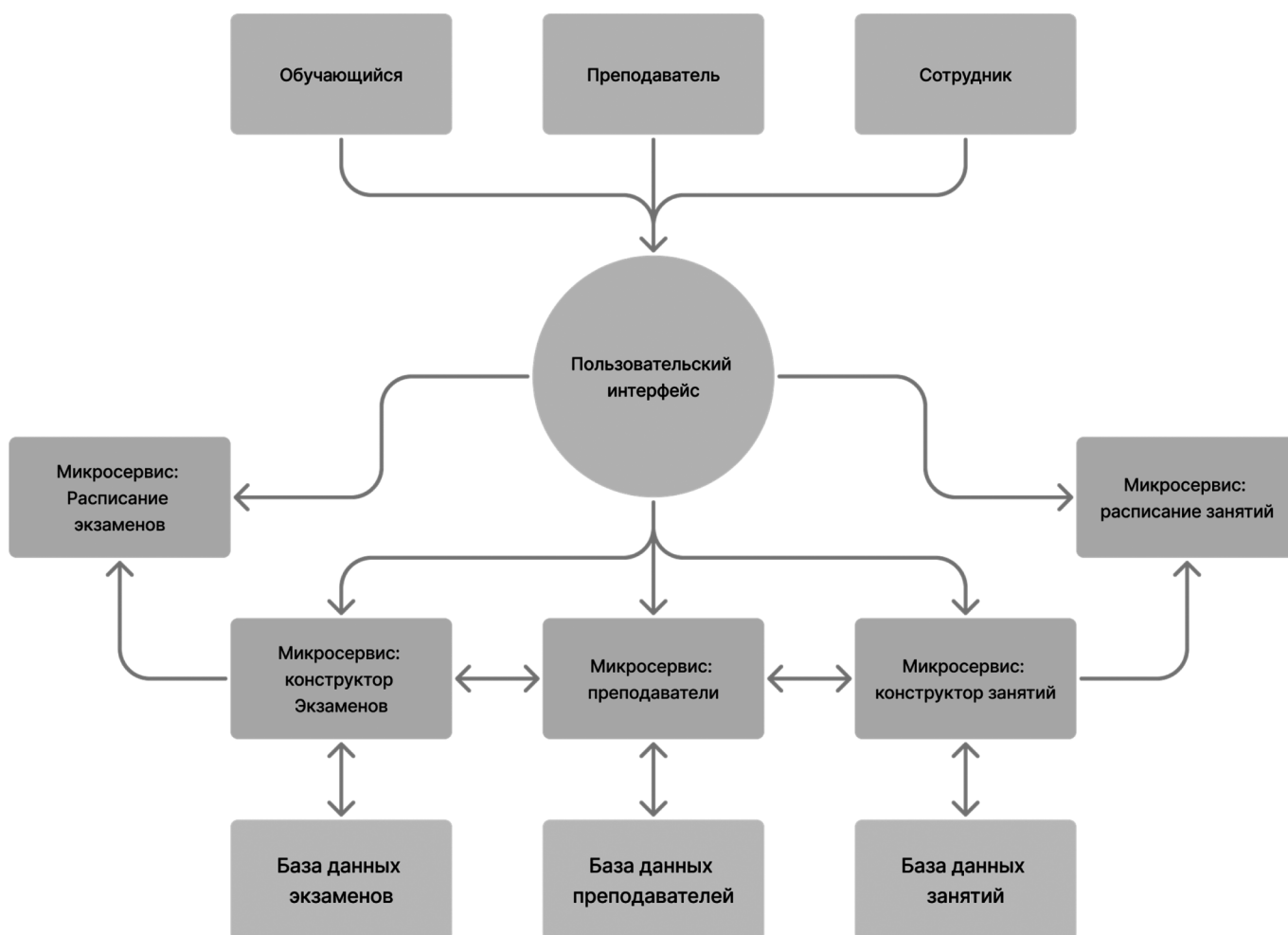


Рис. 5. Пример Сервиса расписания на основе микросервисной архитектуры

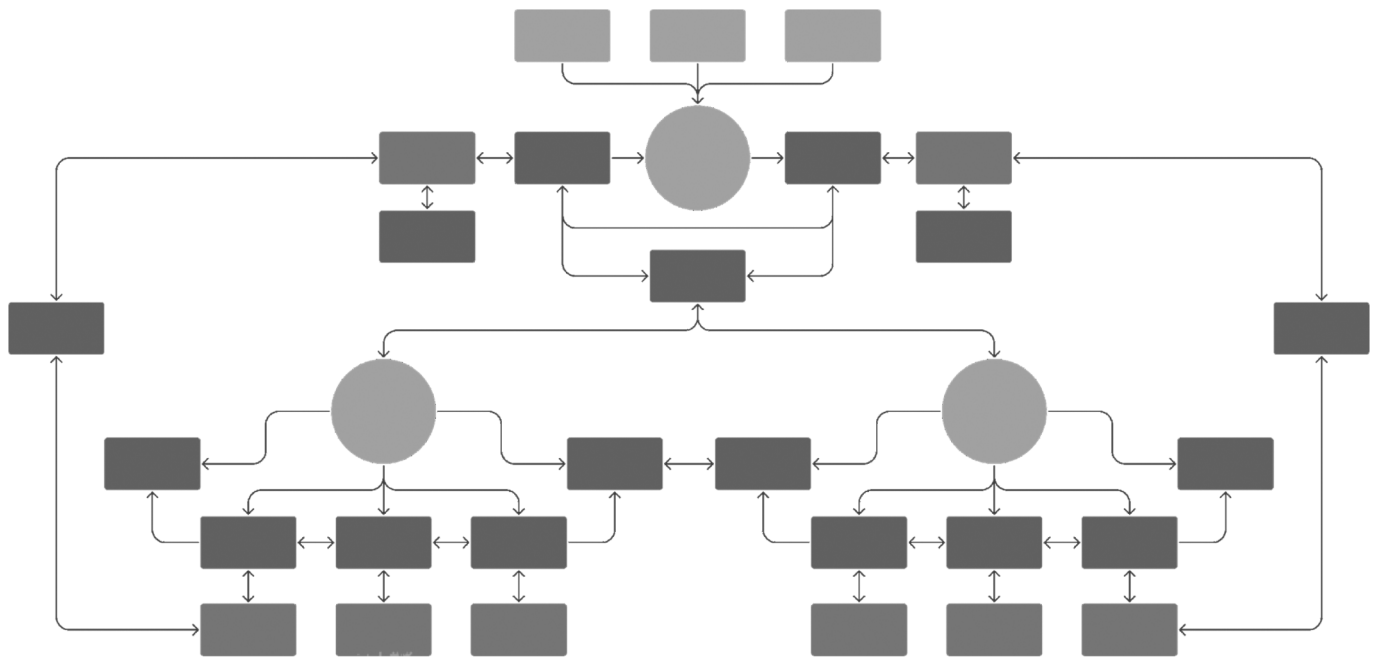


Рис. 6. Пример развития цифровой экосистемы на основе микросервисной архитектуры

Заключение

Подводя итоги исследования, можно сделать вывод, что микросервисная архитектура революционизирует то, как университеты управляют своими цифровыми экосистемами. Ее преимущества многочисленны и помогут университетам оставаться конкурентоспособными в эпоху цифровых технологий. Безусловно, многие ВУЗы столкнутся с трудностями внедрения и смены уже устоявшихся монокристаллических систем, но в итоге именно этот переход сможет сформировать единую цифровую образовательную среду всей страны, объединяя в себе лучшие практики во всех сферах образования.

В рамках данной работы были представлены примеры проектирования микросервисов, которые наглядно демонстрируют подход к архитектурной разработке.

Рассмотрен пример сервиса управления расписанием, реализованного на основе микросервисной архитектуры, а также пример построения и развития цифровой экосистемы университета. Эти примеры показывают, как универсальные принципы микросервисной архитектуры могут быть адаптированы к специфике образовательных процессов. Материалы статьи могут быть полезны для разработчиков, IT-специалистов и руководителей высших учебных заведений, которые занимаются вопросами цифровизации и автоматизации. Представленные методики проектирования и описания ключевых преимуществ микросервисного подхода могут служить основой для создания новых цифровых решений в образовательной сфере. Кроме того, полученные результаты могут быть использованы в научных исследованиях, посвященных разработке и внедрению современных архитектур в управлении образовательными экосистемами.

ЛИТЕРАТУРА

1. Бородин Н.А., Богданова И.Б. Особенности осуществления государственной политики в области информатизации образования в современной России // Инженерный вестник Дона. 2012. №1. URL: ivdon.ru/magazine/archive/n1y2012/635.
2. Заседания президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам «паспорт национального проекта «ОБРАЗОВАНИЕ» от 24 декабря 2018 г. № 16 // министерство науки и высшего образования. 2018 г.
3. Microservices a definition of this new architectural term // martinfowler.com URL: martinfowler.com/articles/microservices.html (дата обращения: 26.06.2023).
4. Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture // ieeexplore.ieee.org URL: ieeexplore.ieee.org/document/7436659 (дата обращения: 26.12.2024).
5. Continuous Deployment: Strategies // www.javacodegeeks.com URL: www.javacodegeeks.com/2014/12/continuous-deployment-strategies.html (дата обращения: 26.12.2024).
6. userver — The C++ Framework // github.com URL: <https://github.com/userver-framework> (дата обращения: 26.06.2024).
7. habr.com URL: habr.com/ru/companies/yandex/articles/674902/ (дата обращения: 26.12.2024).
8. Кирилов А.Г. Направления автоматизации системы управления вузом // Вестник Красноярского государственного педагогического университета им. В.П. Астафьева. 2015.
9. Нестерук О.В. Деятельность высших учебных заведений как один из аспектов эффективного функционирования сферы сервиса // Теория и практика сервиса: экономика, социальная сфера, технологии. 2012.
10. Кравченко Д.А. Микро сервисная архитектура // Интерактивная наука. 2022. №4 (69). С. 43–44.

© Огородов Дмитрий Владимирович (ogorodov-dv@rguk.ru); Иванов Никита Геннадьевич (ivanov-ng@rguk.ru)

Журнал «Современная наука: актуальные проблемы теории и практики»