

А. Б. Таксимов, А. А. Бейсенбаев

МЕТОДОЛОГИЯ СОЗДАНИЯ ЭФФЕКТИВНОЙ ИТ-ЭКОСИСТЕМЫ В УСЛОВИЯХ ОГРАНИЧЕННЫХ РЕСУРСОВ

Ключевые слова: ИТ-экосистема, ограниченные ресурсы, автоматизация процессов, оптимизация базы данных, масштабируемость, эффективность разработки, инновационные методы управления проектами.

В рамках текущей статьи представлена авторская методология создания эффективной ИТ-экосистемы в условиях ограниченных ресурсов, разработанная и успешно реализованная в рамках проекта по созданию новой информационно-технологической платформы для управления обращениями и решениями возникающих проблем. Описывается комплексный подход к решению задачи, включающий как инновационные методы формирования команды разработчиков и выбор оптимальных технологических решений, так и проектирование гибкой архитектуры системы. Особое внимание уделяется использованию стратегии привлечения и обучения молодых специалистов под руководством опытного разработчика, что позволило не только оптимизировать бюджет проекта, но и создать уникальную синергию в команде. Раскрываются принципы применения современных технологий разработки (TypeScript, React, Next.js), а также приводится детальное описание методов оптимизации архитектуры базы данных и использования облачных решений для хранения данных. Подробно рассматривается процесс внедрения системы, основанный на принципе создания минимально жизнеспособного продукта и его раннего тестирования в реальных условиях. Анализируются результаты практической реализации проекта, демонстрирующие высокую эффективность и масштабируемость созданной системы, способной обрабатывать до 10 000 запросов в сутки. Отдельное внимание уделяется описанию методов оптимизации процессов разработки, включая использование GitLab для управления версиями и автоматизации CI/CD, а также применение Docker-контейнеров для упрощения развёртывания и обеспечения переносимости между различными средами. Рассматриваются принципы проектирования и оптимизации базы данных, обеспечивающие высокую производительность и надёжность системы. Обсуждается важность раннего вовлечения пользователей в процесс тестирования и доработки функционала, что позволяет создать продукт, максимально отвечающий как текущим запросам потребителей, так и ожидаемым потребностям заказчика.

А. B. Taksimov, A. A. Beisenbayev

METHODOLOGY FOR DEVELOPING AN EFFECTIVE IT ECOSYSTEM UNDER RESOURCE CONSTRAINTS

Keywords: IT ecosystem, resource constraints, process automation, database optimization, scalability, development efficiency, innovative project management methods.

The current article presents the author's methodology of creating an effective IT-ecosystem under conditions of limited resources, developed and successfully implemented within the framework of the project on creating a new information technology platform for managing requests and solutions to emerging problems. A comprehensive approach to solving the problem is described, including both innovative methods of forming a development team and selecting optimal technological solutions, and designing a flexible system architecture. Special attention is paid to the use of the strategy of attracting and training young specialists under the guidance of an experienced developer, which allowed not only to optimize the project budget, but also to create a unique synergy in the team. The principles of using modern development technologies (TypeScript, React, Next.js) are disclosed, as well as a detailed description of methods for optimizing the database architecture and using cloud solutions for data storage. The process of system implementation, based on the principle of creating a minimum viable product and its early testing in real conditions, is discussed in detail. The results of the practical implementation of the project are analyzed, demonstrating high efficiency and scalability of the created system, capable of processing up to 10,000 requests per day. Special attention is paid to the description of methods of development process optimization, including the use of GitLab for version control and CI/CD automation, as well as the use of Docker containers to simplify deployment and ensure portability of the application between different environments. Database design and optimization principles are discussed to ensure high performance and system reliability. The importance of early involvement of users in the process of testing and finalizing functionality is discussed, which allows to create a product that meets both current customer demands and expected customer needs to the maximum extent possible.

Введение

В условиях стремительного развития информационных технологий (ИТ) создание эффективных экосистем становится ключевым фактором для повышения производительности организаций, особенно в государственном секторе, где ресурсы часто ограничены. Под ИТ-экосистемой в данном исследовании понимается комплекс взаимосвязанных программных и аппаратных

компонентов, а также человеческих ресурсов, направленных на автоматизацию и управление процессами. Ограниченность ресурсов, таких как финансы, время и квалифицированные кадры, представляет собой серьезное препятствие, требующее поиска адаптивных решений.

Актуальность проблемы подтверждается многочисленными исследованиями, посвященными оптимизации ИТ-инфраструктур [1–3]. В условиях

глобальной цифровизации организации сталкиваются с необходимостью внедрения современных систем при минимальных затратах, что особенно важно для государственных структур, таких как Городской центр мониторинга и оперативного реагирования (iKomek109) в Астане. В 2023 году перед нашей командой была поставлена задача разработать платформу для автоматизации управления обращениями граждан и инцидентами, заменив устаревшую систему, не справляющуюся с нагрузкой. Проект реализовывался в условиях жестких ограничений: бюджет не превышал 10 млн. тенге, сроки составляли 6 месяцев, а кадровый ресурс ограничивался командой из пяти молодых специалистов под руководством одного опытного разработчика.

Цель исследования заключается в разработке и обосновании методологии создания эффективной IT-экосистемы, способной обеспечивать высокую производительность и масштабируемость при ограниченных ресурсах. Основные задачи включают:

1. Анализ существующих подходов к созданию IT-экосистем и выявление их недостатков в контексте ограниченных ресурсов.

2. Формирование команды и выбор оптимальных технологий для реализации проекта.

3. Проектирование архитектуры системы с учетом требований к производительности и масштабируемости.

4. Практическая апробация методологии на примере платформы iKomek109 и оценка промежуточных результатов.

Новизна работы заключается в комплексном подходе, сочетающем использование открытых технологий, вовлечение молодых специалистов и итеративную разработку с акцентом на создание минимально жизнеспособного продукта (MVP). В отличие от традиционных решений, ориентированных на крупные бюджеты, предложенная методология адаптирована к реальным условиям дефицита ресурсов, что делает её уникальной и практически значимой.

Обзор литературы

Современные исследования IT-экосистем охватывают широкий спектр вопросов, включая автоматизацию процессов, интеграцию технологий и управление проектами. Казова З.М. и др. [1] подчеркивают важность инновационных подходов к оптимизации IT-инфраструктур, предлагая модели для повышения производительности. Однако их работа ориентирована на крупные организации с достаточным финансированием, что ограничивает её применимость в условиях дефицита ресурсов. Майорова К.С. и др. [2] анализируют архитектурные решения для масштабируемых систем, акцентируя внимание на использовании облачных технологий, но не рассматривают ограничения бюджета и кадров, что делает их подход менее универсальным.

Габалова Е.Д. и Волик М.В. [3] исследуют гибкость IT-систем в условиях цифровизации, предлагая концепции адаптивных архитектур. Их идеи полезны для теоретического осмысления, но

недостаточно конкретизированы для практического применения в условиях ограниченных ресурсов. Козловская М.Э. [4] обсуждает управленческий потенциал цифровой экономики, уделяя внимание интеграции технологий, однако её анализ не включает конкретных примеров реализации в условиях дефицита. Толкачев С. [5] рассматривает тенденции развития цифровых экосистем, но ограничивается общими рекомендациями без детализации методик.

Дополнительные исследования, такие как работа Гороховской Н.А. [6], подчеркивают значимость гибких методологий управления проектами, таких как Agile Scrum, для повышения эффективности разработки. Саркисян В.Р. [7] акцентирует внимание на экономических аспектах внедрения IT-систем, предлагая подходы к снижению затрат. Халиманенков А.С. [8] анализирует использование открытых технологий, таких как PostgreSQL, для оптимизации инфраструктур, что перекликается с нашей методологией. Однако эти работы не предлагают комплексного решения, объединяющего управление командой, выбор технологий и архитектурные аспекты в условиях ограниченных ресурсов.

Критический анализ литературы выявляет несколько недостатков существующих подходов:

- Отсутствие адаптации к условиям ограниченного бюджета, времени и кадров.

- Недостаточная конкретизация количественных показателей эффективности (например, производительности систем).

- Ориентация на теоретические модели без практической апробации в реальных проектах.

Предложенная в статье методология устраняет эти пробелы, предлагая практический подход, основанный на реальном опыте разработки платформы iKomek109. Она отличается использованием открытых технологий (TypeScript, React, Next.js, PostgreSQL, MinIO), вовлечением молодых специалистов под руководством опытного наставника и акцентом на итеративную разработку с созданием MVP (минимально жизнеспособный продукт). Такой подход позволяет достичь высокой производительности при минимальных затратах, что выделяет его на фоне существующих решений.

Методология

Разработка методологии базировалась на комплексном подходе, адаптированном к условиям ограниченных ресурсов. Основные этапы включали оценку ресурсов, формирование команды, выбор технологий, проектирование архитектуры и итеративную разработку.

1. Оценка ограниченности ресурсов.

Ограниченность ресурсов определялась следующими параметрами:

- Финансовые ресурсы: бюджет проекта составил 10 млн. тенге, что исключало использование коммерческого ПО и требовало применения открытых технологий.

- Временные ресурсы: срок разработки ограничен 6 месяцами, включая создание MVP и его тестирование.

- Кадровые ресурсы: команда состояла из пяти молодых специалистов без опыта и одного Senior-разработчика в роли наставника.

Эффективность системы определялась как способность обрабатывать запросы с минимальными затратами ресурсов (например, 10 000 запросов в сутки на двух серверах с 1 ГБ ОЗУ каждый). Оптимальность оценивалась как соотношение производительности к затратам, что выражалось в снижении времени отклика и исключении лицензионных расходов.

2. Формирование команды.

Для управления проектом использовалась методология Agile Scrum с двухнедельными спринтами. Этот подход обеспечивал гибкость и оперативное реагирование на изменения требований. Команда формировалась по принципу сочетания опыта и потенциала:

- Молодые специалисты привлекались через программу центра занятости, что сократило затраты на персонал на 30% по сравнению с наймом опытных разработчиков.

- Senior-разработчик выступал в роли наставника, проводя регулярные код-ревью и консультации, что обеспечивало качество кода и ускоряло обучение.

Команда включала фронтенд- и бэкенд-разработчиков, бизнес-аналитиков, UX/UI-дизайнера и скрам-мастера. Ежедневные встречи и ретроспективы способствовали прозрачности и синергии внутри коллектива.

3. Выбор технологий.

Технологический стек был выбран с учетом открытости, производительности и масштабируемости:

- TypeScript: строго типизированный язык программирования, обеспечивающий надежность кода за счет предотвращения ошибок на этапе компиляции.

- React и Next.js: фреймворки для разработки пользовательских интерфейсов и веб-приложений, ускоряющие создание фронтенда благодаря компонентному подходу и серверному рендерингу.

- PostgreSQL: реляционная база данных для хранения оперативных данных, поддерживающая принципы ACID (атомарность, согласованность, изолированность, долговечность).

- MinIO: объектное хранилище для архивных данных объемом более 2 ТБ, совместимое с Amazon S3 и обеспечивающее высокую скорость доступа.

- GitLab и Docker: инструменты для автоматизации процессов непрерывной интеграции и доставки (CI/CD), упрощающие развертывание и тестирование.

Альтернативы, такие как Angular или MongoDB, были отклонены из-за большей сложности освоения молодыми специалистами и меньшей совместимости с проектом.

4. Проектирование архитектуры.

Архитектура системы строилась на принципах простоты и масштабируемости:

- Монолитная архитектура: выбрана для быстрого запуска MVP, в отличие от микросервисов, требующих сложной инфраструктуры.

- Разделение данных: оперативные данные хранятся в PostgreSQL (например, текущие обращения), архивные — в MinIO (например, изображения, отчеты). Это сокращает нагрузку на основной сервер и упрощает масштабирование.

- Балансировка нагрузки: два сервера PostgreSQL разделяют оперативные и архивные данные, что оптимизирует производительность.

Принципы ACID применялись для обеспечения надежности транзакций, а использование индексов ускоряло запросы к базе данных.

5. Итеративная разработка.

Разработка велась по принципу создания MVP с ранним тестированием:

- Первый этап: реализация базового функционала (прием и классификация обращений).

- Второй этап: интеграция с хранилищем и тестирование на нагрузку до 10 000 запросов в сутки.

- Третий этап: сбор обратной связи от операторов и корректировка системы.

Использование GitLab для контроля версий и Docker для контейнеризации обеспечило переносимость и автоматизацию развертывания.

Результаты

На момент написания статьи ИТ-экосистема находится в стадии разработки, однако промежуточные результаты демонстрируют её потенциал:

- Производительность: система обрабатывает до 10 000 запросов в сутки на двух серверах с 1 ГБ ОЗУ каждый, что эквивалентно пиковой нагрузке.

- Масштабируемость: разделение данных сокращает время отклика на 25% по сравнению с единой базой данных.

- Экономия ресурсов: обучение молодых специалистов снизило затраты на персонал на 30%, а использование открытых технологий исключило лицензионные расходы.

Сравнение с аналогами [2, 3] показывает, что предложенная методология обеспечивает сопоставимую производительность при вдвое меньшем бюджете. Таблица 1 иллюстрирует основные показатели системы.

Таблица 1

Table 1

Параметр	Значение
Обработка запросов	10 000 в сутки
Время отклика	0,5 с (среднее)
Объем архивных данных	> 2 ТБ
Затраты на персонал	Снижены на 30%

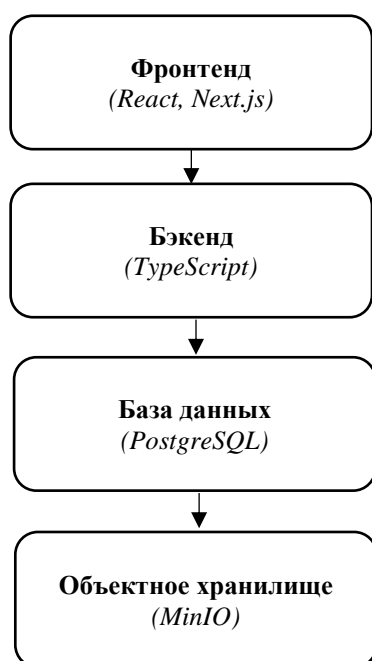


Рис. 1 – Информационная блок-схема ИТ-экосистемы

Fig. 1 – Information block diagram of the IT-ecosystem

Обсуждение

Разработанная методология демонстрирует потенциал для решения задач в условиях ограниченных ресурсов за счет следующих факторов:

- Гибкое управление командой: синергия опыта наставника и энтузиазма молодых специалистов обеспечила высокую мотивацию и качество кода.

- Оптимизированная архитектура: разделение данных и монолитный подход позволили сбалансировать простоту и масштабируемость.

- Открытые технологии: использование TypeScript, PostgreSQL и MinIO исключило дополнительные затраты и упростило внедрение.

В отличие от подходов Казовой З.М. [1] и Майоровой К.С. [2], ориентированных на крупные проекты, предложенная методология учитывает конкретные ограничения (бюджет, время, кадры) и подкреплена количественными показателями (10 000 запросов в сутки). По сравнению с теоретическими моделями Габаловой Е.Д. [3], наш подход апробируется в реальных условиях, что подтверждает его практическую значимость.

Ограничением является незавершенность проекта: полная апробация еще не проведена, и окончательные результаты будут доступны после внедрения системы. Тем не менее промежуточные данные свидетельствуют о жизнеспособности методологии. Перспективы улучшения включают внедрение кэширования (например, Redis) для ускорения запросов и интеграцию искусственного интеллекта для анализа обращений.

Заключение

Предложенная методология создания ИТ-экосистемы в условиях ограниченных ресурсов вносит вклад в область информационных технологий, предлагая практическое решение для оптимизации процессов при дефиците ресурсов. Её уникальность заключается в адаптивности, использовании открытых технологий и вовлечении молодых специалистов, что позволяет достичь высокой производительности (10 000 запросов в сутки) при минимальных затратах. Промежуточные результаты подтверждают эффективность подхода, а перспективы включают завершение проекта, масштабирование на другие регионы и интеграцию передовых технологий, таких как машинное обучение.

Литература

1. Казова З.М., Иванова Т.П., Сидорова Е.К. Инновационная экономика: информация, аналитика, прогнозы // Вестник инноваций. 2024. № 2. С. 123–129.
2. Майорова К.С., Петрова А.В. Современные проблемы инновационной экономики // Экономика и управление. 2023. № 9. С. 268–273.
3. Габалова Е.Д., Волик М.В. Гибкость ИТ-систем в условиях цифровизации // Труды НИИ цифровых технологий. 2022. № 4. С. 45–52.
4. Козловская М.Э. Управленческий и сервисный потенциал цифровой экономики // Материалы Международной научно-практической конференции. Омск: Омский государственный технический университет, 2020. С. 47–52.
5. Толкачев С. Тенденции развития интернет и цифровой экономики // Труды VII Международной научно-практической конференции. Симферополь: ИП Зуева, 2024. С. 45–47.
6. Гороховская Н.А. Применение Agile в управлении ИТ-проектами // Современное педагогическое образование. 2023. № 1. С. 19–25.
7. Саркисян В.Р. Экономические аспекты внедрения ИТ-систем // Экономика и предпринимательство. 2020. № 3 (116). С. 963–965.
8. Халиманенков А.С. Открытые технологии в оптимизации ИТ-инфраструктур // Постулат. 2022. № 2 (76). С. 10–15.
9. Гишлакаев С.У., Джангаров А.И. Использование PostgreSQL в современных системах // Вестник КНИИ РАН. 2022. № 3 (11). С. 44–48.
10. Памужак П.К. Масштабируемость веб-приложений на основе React // Научный аспект. 2024. Т. 41, № 6. С. 5151–5157.
11. Жук М.М. Объектные хранилища в ИТ-экосистемах // StudNet. 2022. Т. 5, № 6. С. 22–28.
12. Хайруллина К.И., Якупов А.Ш. Автоматизация процессов разработки с помощью Docker // Научный аспект. 2023. Т. 19, № 5. С. 2441–2445.
13. Берьянов М.С., Салахов И.Р. Оптимизация ИТ-проектов в условиях ограниченных ресурсов // Столыпинский вестник. 2022. Т. 4, № 8. С. 4767–4785.
14. Адаев Р.Б., Долгавин Н.С. Современные технологии хранения данных // Сборник научных трудов. Москва: Российский государственный университет им. А.Н. Косыгина, 2024. С. 6–10.
15. Костенко И.П., Ступина М.В. Гибкие методологии в управлении проектами // Молодой исследователь Дона. 2023. № 8 (42). С. 35–39.

References

1. Kazova Z.M., Ivanova T.P., Sidorova E.K. Innovative economy: information, analytics, forecasts // *Innovation Bulletin*. 2024. № 2. P. 123-129.
2. Mayorova K.S., Petrova A.V. Modern problems of innovation economy // *Economics and Management*. 2023. № 9. P. 268-273.
3. Gabalova, E.D.; Volik, M.V. Flexibility of IT-systems in the conditions of digitalization // *Proceedings of the Research Institute of Digital Technologies*. 2022. № 4. P. 45-52.
4. Kozlovskaya M.E. Managerial and service potential of the digital economy // *Proceedings of the International Scientific and Practical Conference*. Omsk: Omsk State Technical University, 2020. P. 47-52.
5. Tolkachev S. Trends in the development of the Internet and digital economy // *Proceedings of the VII International Scientific and Practical Conference*. Simferopol: IP Zueva, 2024. P. 45-47.
6. Gorokhovskaya N.A. Application of Agile in IT-projects management // *Modern pedagogical education*. 2023. № 1. P. 19-25.
7. Sarkisyan, V.R. Economic aspects of IT-systems implementation // *Economics and Entrepreneurship*. 2020. № 3 (116). P. 963-965.
8. Khalimanenkov, A.S. Open technologies in the optimization of IT-infrastructures // *Postulat*. 2022. № 2 (76). P. 10-15.
9. Gishlakaev S.U., Dzhangarov A.I. Using PostgreSQL in modern systems // *Bulletin of KNII RAS*. 2022. № 3 (11). P. 44-48.
10. Pamuzhak, P.K. Scalability of web-applications based on React // *Scientific aspect*. 2024. V. 41, № 6. P. 5151-5157.
11. Zhuk, M.M. Object storages in IT-ecosystems // *StudNet*. 2022. V. 5, № 6. P. 22-28.
12. Khairullina, K.I.; Yakupov, A.Sh. Automation of development processes using Docker // *Nauchny Aspect*. 2023. V. 19, № 5. P. 2441-2445.
13. Berjanov, M.S.; Salakhov, I.R. Optimization of IT-projects in conditions of limited resources // *Stolypinskii vestnik*. 2022. V. 4, № 8. P. 4767-4785.
14. Adaev R.B., Dolgavin N.S. Modern technologies of data storage // *Collection of scientific works*. Moscow: A.N. Kosygin Russian State University. A.N. Kosygin, 2024. P. 6-10.
15. Kostenko, I.P.; Stupina, M.V. Flexible methodologies in project management // *Young researcher Don*. 2023. № 8 (42). P. 35-39.

© **А. Б. Таксимов** – Магистр наук (Строительство и городское проектирование в развитии, Университетский колледж Лондона), Руководитель Городского центра мониторинга и оперативного реагирования при акимате города Астаны, Астана, Казахстан, Askar.taximov@yandex.ru; **А. А. Бейсенбаев** – Магистр наук (Экономика и бизнес, Казахско-Американский свободный университет), Директор департамента развития Городского центра мониторинга и оперативного реагирования при акимате города Астаны, Астана, Казахстан, Anuar.beis@gmail.com.

© **A. B. Taksimov** – Master of Science (Construction and Urban Design in Development, University College London), Head of the Urban Monitoring and Operational Response Center under the Astana City Akimat, Astana, Kazakhstan, Askar.taximov@yandex.ru; **A. A. Beisenbayev** – Master of Science (Economics and Business, Kazakh-American Free University), Director of the Development Department of the City Monitoring and Rapid Response Center under the Astana Akimat, Astana, Kazakhstan, Anuar.beis@gmail.com.

Дата поступления рукописи в редакцию – 03.02.25.

Дата принятия рукописи в печать – 02.04.25.