

Болодурина И. П., Пояркова Е. Н., Неволлина В. В.

Оренбургский государственный университет имени В. А. Бондаренко (Оренбург, Россия)

E-mail: liza.yarko.e@mail.ru

ПРОЕКТИРОВАНИЕ ВНУТРИКОРПОРАТИВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ПОДГОТОВКИ СИСТЕМНЫХ АНАЛИТИКОВ В КРУПНОЙ ИТ-КОМПАНИИ

В условиях устойчивого дефицита квалифицированных специалистов в области системного и бизнес-анализа крупной ИТ-компании необходимы внутрикорпоративные образовательные программы подготовки системных аналитиков. Эффективные корпоративные механизмы профессиональной подготовки, позволяют в сжатые сроки формировать у сотрудников комплекс компетенций, соответствующих современным требованиям цифровой экономики, отраслевым стандартам и практическим задачам ИТ-проектов. Цель исследования заключается в анализе структуры, содержания и методических принципов построения образовательной программы, ориентированной на развитие профессиональных компетенций системных аналитиков и обеспечение их готовности к выполнению производственных задач. Эмпирической основой исследования послужил реестр учебных артефактов программы, представляющий собой структурированный рабочий документ, содержащий сведения об образовательных результатах, тематическом содержании модулей, методиках обучения, форматах проведения занятий, цифровых образовательных средах, используемых программных инструментах и распределении ответственности между преподавателями-экспертами. В работе применены методы анализа научной и нормативной литературы, контент-анализа структуры программы и сопоставления ее компонентов с требованиями профессионального стандарта «Системный аналитик», свода знаний BABOK Guide и модели педагогического дизайна ADDIE. Показано, что программа включает входное диагностическое тестирование, пять тематических модулей, командную и индивидуальную практику, а также итоговую оценку результатов обучения. Особое внимание уделено реализации компетентностного подхода и принципов андрагогики посредством практико-ориентированного и смешанного обучения, использования сквозных кейсов и современных инструментов системного анализа, моделирования бизнес-процессов, проектирования архитектуры информационных систем и разработки API.

Ключевые слова: корпоративное обучение, системный аналитик, бизнес-анализ, компетентностный подход, андрагогика, модель ADDIE, BABOK, смешанное обучение, педагогический дизайн, оценка эффективности обучения.

Bolodurina I. P., Poyarkova E. N., Nevolina V. V.

Orenburg State University named after V. A. Bondarenko (Orenburg, Russia)

E-mail: liza.yarko.e@mail.ru

DESIGNING AN IN-HOUSE CORPORATE TRAINING PROGRAM FOR SYSTEM ANALYSTS IN A LARGE IT COMPANY

This paper presents the methodological experience of designing an in-house corporate training program for system analysts developed for a large IT company facing a persistent shortage of qualified specialists in systems and business analysis. The relevance of the study is determined by the need to establish effective corporate training mechanisms capable of rapidly developing employees' professional competencies in accordance with the current demands of the digital economy, industry standards, and the practical requirements of IT projects. The purpose of the study is to analyze the structure, content, and methodological principles underlying the design of a corporate training program aimed at developing the professional competencies of system analysts and preparing them for real-world professional activities. The empirical basis of the study is a registry of educational artifacts serving as a structured working document that contains information on learning outcomes, module content, teaching methods, instructional formats, digital learning environments, software tools, and the distribution of responsibilities among subject-matter experts. The research employed methods of scientific and regulatory literature analysis, content analysis of the training program structure, and comparison of its components with the requirements of the Russian Professional Standard "System Analyst," the Business Analysis Body of Knowledge (BABOK® Guide), and the ADDIE instructional design model. The findings demonstrate that the program consists of an entry diagnostic assessment, five thematic modules, team-based and individual practical assignments, and a final assessment of learning outcomes. Particular attention is paid to implementing the competency-based approach and the principles of andragogy through practice-oriented and blended learning, the use of integrated case studies, and modern professional tools for systems analysis, business process modeling, information systems architecture design, and API development. The paper substantiates the advantages of the proposed corporate training model and identifies directions for its further improvement, including the implementation of Kirkpatrick's four-level training evaluation model and the development of post-training learner support mechanisms.

Keywords: corporate training; system analyst; business analysis; competency-based approach; andragogy; ADDIE model; BABOK Guide; blended learning; instructional design; training effectiveness evaluation.

Дефицит квалифицированных системных и бизнес-аналитиков остаётся одной из наиболее острых кадровых проблем российской ИТ-отрасли: спрос на специалистов, способных формализовать бизнес-требования, спроектировать архитектуру информационной системы и организовать её интеграцию с окружением, растёт быстрее, чем предложение выпускников профильных образовательных программ. Ситуация усугубляется тем, что содержание профессии системного аналитика постоянно расширяется от классического анализа требований до владения архитектурными нотациями, знания баз данных, протоколов интеграции и практик развёртывания приложений, что отражено в актуальной редакции профессионального стандарта «Системный аналитик» [1].

В этих условиях крупные ИТ-компании всё чаще инвестируют в создание внутренних («корпоративных») школ и программ переподготовки, которые дополняют или частично замещают классическое высшее образование, обеспечивая целевую и быструю подготовку сотрудников под задачи конкретных проектов и продуктовых команд. Однако систематизированного описания методических принципов проектирования подобных программ в отечественной литературе немного: большинство публикаций посвящено либо теоретическим моделям педагогического дизайна безотносительно к ИТ-контексту [2]–[4], либо техническому описанию отдельных инструментов и нотаций без учёта образовательной логики их изложения.

Цель настоящей статьи описать и систематизировать принципы проектирования такой программы: её структуру, теоретико-методологические основания, форматы обучения и подходы к оценке результатов на примере реестра учебных артефактов внутрикорпоративной программы подготовки системных аналитиков в крупной ИТ-компании. Источником эмпирического материала послужил структурированный документ (реестр программы), в котором для каждого учебного занятия зафиксированы этапы и компоненты программы, артефакты, сроки готовности, ссылки на материалы, цели, образовательные результаты, темы и подтемы, дополнительные материалы, тайминг, используемая образовательная среда, методики и форматы обучения, а также ответственный преподаватель-эксперт.

В качестве материала исследования использован реестр (карта) образовательной программы, представляющий собой таблично организованный документ, применяемый разработчиками программы как единый источник правды на всех этапах проектирования обучения. Для каждой единицы содержания в реестре зафиксированы этап и компонент программы (вебинар, практика по командам, индивидуальная практика, тест), артефакт, который должен быть создан к учебному занятию, дедлайн его готовности, ссылка на материалы, формулировка цели занятия, образовательные результаты в форме измеримых учебных действий, содержание (темы и подтемы), дополнительные материалы для самостоятельного изучения, тайминг, используемая образовательная среда и необходимое программное обеспечение, методики и форматы обучения (онлайн/офлайн), а также имя преподавателя-эксперта.

Такая структура фиксации данных по своей логике соответствует первым трём этапам широко применяемой в педагогическом дизайне модели ADDIE — Analysis (Анализ потребностей и целевой аудитории), Design (Проектирование содержания и образовательных результатов) и Development (Разработка материалов), а зафиксированные в реестре входное и итоговое тестирование реализуют этапы Implementation (Внедрение) и Evaluation (Оценка) этой модели [2]. Анализ реестра проводился методом контент-анализа: содержательные единицы группировались по модулям и темам, форматы обучения по типам активности (лекция/вебинар, командная практика, индивидуальная практика, тестирование), после чего сопоставлялись с профессиональными компетенциями, зафиксированными в профессиональном стандарте «Системный аналитик» [1] и в своде знаний по бизнес-анализу BABOK Guide [5].

Отправной точкой проектирования программы является не перечень тем, а перечень образовательных результатов, то есть конкретных профессиональных действий, которые участник должен уметь выполнять по завершении обучения (например, «формулировать бизнес-требования», «моделировать бизнес-процессы в нотации BPMN», «проектировать REST API»). Такая логика соответствует компетентностному подходу, в рамках которого целью обучения яв-

ляется формирование операционализируемых профессиональных компетенций, соотносимых с требованиями профессиональных стандартов и рынка труда. Перечень трудовых функций системного аналитика, зафиксированный в профессиональном стандарте «Системный аналитик» [1], и области знаний бизнес-анализа, описанные в BABOK Guide [5] (выявление и согласование требований, их анализ и определение решения, управление жизненным циклом требований), образуют внешнюю рамку, относительно которой выстраивается содержание программы: от определения границ проекта и бизнес-требований (BABOK-подобные задачи бизнес-анализа) до проектирования архитектуры и интеграционных решений (задачи системного анализа и проектирования).

Поскольку участниками программы являются взрослые работающие сотрудники компании, а не студенты классических образовательных организаций, при проектировании учитывались принципы андрагогики, сформулированные М. Ноулзом: опора на жизненный и профессиональный опыт обучающегося, ориентация на решение практических задач, а не абстрактное знание, внутренняя мотивация к обучению и потребность понимать, зачем изучается та или иная тема [6]. Практическим выражением этих принципов в анализируемой программе стало то, что почти каждое теоретическое занятие («вебинар») сопровождается практикой по командам или индивидуальной практикой на материале сквозных учебных кейсов, а формулировка цели каждого занятия явно отвечает на вопрос «зачем мы учим этому», что напрямую соотносится с андрагогическим требованием осмысленности обучения [6], [7].

Пятиэтапная модель ADDIE (Analysis, Design, Development, Implementation, Evaluation) остаётся одной из наиболее универсальных моделей проектирования обучения, применимых как в высшем образовании, так и в корпоративном секторе [2]. Эмпирические исследования показывают, что последовательное прохождение этапов ADDIE, начиная от анализа компетенций и потребностей аудитории до итоговой оценки результатов, позволяет повышать удовлетворённость обучающихся содержанием и результатами программы [2]. В анализируемой программе этап Implementation (Внедрение) реализован через

технологии смешанного обучения, характеризующегося сочетанием синхронных занятий (онлайн-вебинары в среде видеоконференцсвязи и интерактивных досок, а также очные офлайн-семинары) с асинхронной работой в системе управления обучением (LMS) и самостоятельным изучением дополнительных материалов [3], [4].

Наличие в программе входного и итогового тестирования, сопоставляющего результаты «было / стало», соответствует логике первых двух уровней широко используемой в корпоративном обучении четырёхуровневой модели оценки эффективности Д. Киркпатрика, показывающей реакции участников и усвоения знаний (Reaction, Learning) [8]. Как отмечается в литературе, посвящённой оценке эффективности дистанционного и смешанного обучения, для получения полной картины эффективности программы целесообразно дополнительно измерять изменение рабочего поведения сотрудников и бизнес-результаты через 3–6 месяцев после завершения обучения (уровни Behavior и Results) [9].

Проанализированная программа имеет линейную архитектуру «входная диагностика → пять содержательных модулей (24 занятия) → итоговая диагностика», что отражает применение модели ADDIE на уровне всей программы в целом.

Входное тестирование выполняет две функции: во-первых, отбор и сегментацию целевой аудитории обучения по исходному уровню владения компетенциями в области бизнес-анализа, системного анализа и проектирования информационных систем; во-вторых, фиксацию точки отсчёта для последующего измерения прироста знаний. Банк вопросов формировался таким образом, чтобы на каждый образовательный результат программы приходился минимум один проверочный вопрос, а типология заданий включала вопросы с одиночным и множественным выбором, задания на сопоставление и на восстановление последовательности, а также открытые вопросы, что позволяет проверять как декларативное, так и процедурное знание.

Модуль «Введение в профессию» формирует базовое понимание роли системного аналитика в жизненном цикле ИТ-проекта: типов информационных и автоматизированных систем, моделей организации проектных работ (каскадная, V-образная, спиральная, гибкая), а также

состава артефактов, создаваемых аналитиком. Далее рассматривается определение границ и содержания проекта, техники выявления и подтверждения требований, различие бизнес-потребностей и бизнес-требований и критерии качества требований — тематика, содержательно пересекающаяся с ключевыми понятиями и техниками BABOK Guide [5] и с классическими принципами инженерии требований [10], [11].

Модуль «Моделирование бизнес-процессов и бизнес-правил» посвящён описанию текущей («as is») бизнес-ситуации средствами формальных нотаций VACD и BPMN [12], моделированию бизнес-правил в нотации DMN, причинно-следственному анализу (диаграмма Исикавы, техника «5 почему») и проектированию целевой («to be») ситуации с помощью канвы возможностей (Opportunity Canvas) и карты влияния (Impact Map). Отдельное занятие посвящено выявлению и описанию требований стейкхолдеров в форме пользовательских историй (user story) с разработкой критериев приёмки — практика, типичная для гибких методологий разработки требований.

Модуль «Разработка системных требований. Управление требованиями» формирует навыки формулирования требований к данным (концептуальная модель данных, словарь данных, диаграммы состояний жизненного цикла сущностей), функциональных требований (реестр требований, варианты использования) и нефункциональных требований по девяти аспектам качества системы (производительность, безопасность, удобство использования и др.), а также макетирования экранных форм. Завершает модуль тема управления требованиями: их трассировка, приоритизация и работа с изменениями в ходе жизненного цикла проекта — темы, соответствующие как BABOK Guide [5], так и международному стандарту инженерии требований ISO/IEC/IEEE 29148 [11] и профильной литературе по инженерии требований [10].

Модуль «Архитектура и технологии интеграции информационных систем. Проектирование и эксплуатация баз данных» — самый объёмный содержательно модуль программы. Он включает основы архитектуры информационных систем и артефакты ИТ-архитектора (Tech Radar, Architecture Decision Record), архитектурные подходы и паттерны (слоистая и монолитная архитектура, клиент-серверная

архитектура, SOA, микросервисы) с проектированием диаграмм в нотации C4 [13] и учётом принципов декомпозиции монолита на микросервисы [14]; реляционные и нереляционные СУБД, включая внутреннее устройство СУБД, нормализацию и денормализацию, транзакции и SQL-запросы (DDL и DML), документо-ориентированные и графовые NoSQL-хранилища [15]; технологии интеграции информационных систем — от архитектурного стиля REST [16] и спецификации OpenAPI до GraphQL, SOAP, gRPC, вебхуков, протокола WebSocket и брокеров сообщений Apache Kafka и RabbitMQ, а также подходы к развёртыванию приложений (контейнеризация, Kubernetes, IaaS, CI/CD, стратегии Blue-Green и Canary) и организации мониторинга и наблюдаемости приложений.

Модуль «Проектирование веб-приложений. UML для проектирования структуры и поведения информационных систем» закрепляет практику проектирования API (REST, GraphQL, SOAP, gRPC) и переходит к объектно-ориентированному и доменно-ориентированному проектированию: технике Event Storming, принципам предметно-ориентированного проектирования (Domain-Driven Design) [17], видам и последовательности применения UML-диаграмм [7], [13] вариантов использования, деятельности, классов и объектов, состояний, последовательности, пакетов, компонентов и развёртывания вплоть до реализации спроектированных классов в виде кода на языке Python. Программу завершает итоговая лекция-дискуссия «Анализ, проектирование и ИИ: риски и возможности», отражающая актуальную для профессии тенденцию интеграции инструментов искусственного интеллекта в практику аналитика.

Итоговое тестирование, аналогичное входному по структуре и типологии заданий, выполняет функцию суммативной оценки и позволяет сопоставить профессиональные компетенции участника «до» и «после» прохождения программы, а также сформировать индивидуальные рекомендации по дальнейшему развитию.

С точки зрения формата обучения программа последовательно реализует принцип смешанного обучения, теоретический материал подаётся в форме онлайн-вебинаров с использованием сред удалённого интерактивного взаимодействия и интерактивных досок, тогда как ряд занятий (управление требованиями, основы архитекту-

ры, развёртывание приложений, проектирование поведения объектов системы) проводится очно в формате офлайн-семинаров и воркшопов, как правило, для тем, требующих интенсивной групповой работы с физическими артефактами (стикеры, маркеры, бумага). После каждого теоретического блока предусмотрена командная или индивидуальная практика на сквозных учебных кейсах с использованием профессиональных инструментов, начиная от специализированных редакторов BPMN- и UML-диаграмм (Camunda Modeler, PlantUML) и сервисов проектирования реляционных схем (DrawSQL) до инструментов тестирования API (Postman) и облачных платформ для работы с GraphQL (Hasura Cloud), что обеспечивает перенос теоретических знаний в практический навык непосредственно в учебной среде.

Проведённый анализ показывает, что рассмотренная программа обладает рядом методических сильных сторон. Во-первых, она системно опирается на признанные профессиональные рамки — профессиональный стандарт «Системный аналитик» [1] и BABOK Guide [5], что обеспечивает соответствие результатов обучения рыночным требованиям к квалификации. Во-вторых, последовательная связка «теория — командная/индивидуальная практика» после каждого занятия реализует андрагогический принцип обучения через опыт [6] и соответствует этапам Development (Разработка) и Implementation (Внедрение) модели ADDIE [2]. В-третьих, использование входного и итогового тестирования с сопоставимой структурой заданий закладывает основу для измерения прироста компетенций (уровни Reaction и Learning модели Киркпатрика [8]), а привлечение действующих экспертов-практиков в качестве преподавателей повышает актуальность и прикладной характер содержания.

Вместе с тем можно указать на ряд направлений для дальнейшего совершенствования программы. Содержательный объём 24 занятия, объединённых в пять модулей, с общей продолжительностью, исчисляемой десятками академических часов, создаёт риск когнитивной перегрузки обучающихся, особенно в модулях, где несколько крупных технических тем (напри-

мер, архитектура, интеграция и базы данных) изучаются в сжатые сроки. Смягчить этот риск может введение элементов распределённого повторения и поддерживающего сообщества практики после завершения программы. Кроме того, оценка эффективности программы в её нынешнем виде ограничивается входным и итоговым тестированием, то есть первыми двумя уровнями модели Киркпатрика; для получения полной картины целесообразно дополнительно оценивать изменение рабочего поведения участников на реальных проектах и вклад обучения в бизнес-показатели команд через 3–6 месяцев после завершения программы (уровни Behavior и Results) [8], [9]. Наконец, поскольку андрагогическая модель предполагает высокую внутреннюю мотивацию обучающихся [6], для сотрудников с разным уровнем исходной вовлечённости может потребоваться дифференциация форматов поддержки — от менторства до дополнительных консультаций с преподавателем.

Рассмотренный опыт проектирования внутрикорпоративной школы системных аналитиков демонстрирует, что эффективная программа корпоративного обучения ИТ-специалистов может быть построена на пересечении трёх методологических оснований: компетентностного подхода, опирающегося на профессиональные стандарты и отраслевые своды знаний [2], [4]; андрагогических принципов обучения взрослых, реализуемых через практико-ориентированный, кейсовый формат [6]; и модели педагогического дизайна ADDIE, задающей технологическую последовательность анализа, проектирования, разработки, внедрения и оценки программы [2]. Ключевым инструментом проектирования выступает единый реестр учебных артефактов, фиксирующий образовательные результаты, содержание, форматы и ответственность за каждый элемент программы, что обеспечивает управляемость и воспроизводимость процесса разработки обучения. Дальнейшие исследования могут быть направлены на эмпирическую оценку эффективности подобных программ по полной четырёх-уровневой модели Д. Киркпатрика [8], а также на сравнительный анализ аналогичных корпоративных программ в разных ИТ-компаниях.

01.07.2026

Список литературы:

1. Об утверждении профессионального стандарта «Системный аналитик»: приказ Министерства труда и социальной защиты Российской Федерации от 27.04.2023 № 367н (зарегистрирован в Минюсте России 25.05.2023, № 73453). — Текст : электронный.

2. Луканина, М. В. Модель ADDIE в педагогическом дизайне: практический опыт НИТУ МИСИС / М. В. Луканина, С. Г. Меркулова // Высшее образование в России. — 2023. — Т. 32, № 10. — С. 151–166. — DOI: 10.31992/0869-3617-2023-32-10-151-166.
3. Интернет-обучение: технологии педагогического дизайна / М. В. Моисеева, Е. С. Полат, М. Ю. Бухаркина, М. И. Нежурина. — М. : Издательский дом «Камерон», 2004. — 216 с. — ISBN 5-9594-0015-4.
4. Использование инструментов педагогического дизайна для обеспечения качества смешанного обучения / Е. А. Другова, С. Б. Велединская, И. И. Журавлева, М. Ю. Дорофеева. — Томск : Изд-во Томского гос. ун-та, 2021. — 64 с.
5. A Guide to the Business Analysis Body of Knowledge (BABOK Guide), version 3.0. — Toronto : International Institute of Business Analysis (IIBA), 2015. — 512 p.
6. Ноулз, М. Ш. Современная практика образования взрослых. Андрагогика против педагогики / М. Ш. Ноулз. — М. : [б. и.], 1980. — 234 с.
7. Fowler, M. UML Distilled: A Brief Guide to the Standard Object Modeling Language / M. Fowler. — 3rd ed. — Boston : Addison-Wesley, 2004. — 175 p. — ISBN 0-321-19368-7.
8. Kirkpatrick, D. L. Evaluating Training Programs: The Four Levels / D. L. Kirkpatrick, J. D. Kirkpatrick. — 3rd ed. — San Francisco : Berrett-Koehler Publishers, 2006. — 399 p. — ISBN 978-1-57675-348-4.
9. Ковалева, М. Н. Метрики эффективности дистанционного обучения в преподавании социально-гуманитарных дисциплин / М. Н. Ковалева, Д. В. Шибаев, Т. И. Сеницына // Международный научно-исследовательский журнал. — 2021. — № 3-3 (105). — С. 51–54. — DOI: 10.23670/IRJ.2021.105.3.068.
10. Wiegers, K. Software Requirements / K. Wiegers, J. Beatty. — 3rd ed. — Redmond : Microsoft Press, 2013. — 672 p. — ISBN 978-0-7356-7966-5.
11. ISO/IEC/IEEE 29148:2018. Systems and software engineering — Life cycle processes — Requirements engineering. — Geneva : ISO, 2018. — 104 p.
12. Business Process Model and Notation (BPMN), Version 2.0. — Needham, MA : Object Management Group, 2011. — 538 p.
13. OMG Unified Modeling Language (OMG UML), Version 2.5.1. — Needham, MA : Object Management Group, 2017. — 796 p.
14. Newman, S. Building Microservices: Designing Fine-Grained Systems / S. Newman. — 2nd ed. — Sebastopol : O'Reilly Media, 2021. — 610 p. — ISBN 978-1-492-03402-5.
15. Kleppmann, M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems / M. Kleppmann. — Sebastopol : O'Reilly Media, 2017. — 616 p. — ISBN 978-1-4493-7332-0.
16. Fielding, R. T. Architectural Styles and the Design of Network-based Software Architectures : dissertation for the degree of Doctor of Philosophy / R. T. Fielding. — Irvine : University of California, 2000. — 180 p.
17. Evans, E. Domain-Driven Design: Tackling Complexity in the Heart of Software / E. Evans. — Boston : Addison-Wesley, 2003. — 560 p. — ISBN 0-321-12521-5.

References:

1. On Approval of the Professional Standard “System Analyst”: Order of the Ministry of Labour and Social Protection of the Russian Federation No. 367n, April 27, 2023 (registered by the Ministry of Justice of the Russian Federation on May 25, 2023, No. 73453).
2. Lukanina M.V. and Merkulova, S.G. (2023). The ADDIE Model in Instructional Design: Practical Experience of NUST MISIS. *Higher Education in Russia*, 32(10), 151–166. URL: <https://doi.org/10.31992/0869-3617-2023-32-10-151-166>
3. Moiseeva M.V., Polat E.S., Bukharkina M.Yu. and Nezhurina M.I. (2004). *Internet-Based Learning: Instructional Design Technologies*. Moscow: Cameron Publishing House.
4. Drugova E.A., Veledinskaya S.B., Zhuravleva I.I. and Dorofeeva M.Yu. (2021). *Using Instructional Design Tools to Ensure the Quality of Blended Learning*. Tomsk: Tomsk State University Press.
5. A Guide to the Business Analysis Body of Knowledge (BABOK® Guide). Version 3.0. Toronto: International Institute of Business Analysis (IIBA), 2015.
6. Knowles M.S. (1980). *The Modern Practice of Adult Education: From Pedagogy to Andragogy*. Moscow. (Russian translation).
7. Fowler M. (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language* (3rd ed.). Boston: Addison-Wesley.
8. Kirkpatrick D.L. and Kirkpatrick J.D. (2006). *Evaluating Training Programs: The Four Levels* (3rd ed.). San Francisco: Berrett-Koehler Publishers.
9. Kovaleva M.N., Shibaev D.V. and Sinitsyna T.I. (2021). Metrics for Evaluating the Effectiveness of Distance Learning in Teaching Social Sciences and Humanities. *International Research Journal*, 3-3(105), 51–54. <https://doi.org/10.23670/IRJ.2021.105.3.068>
10. Wiegers K. and Beatty J. (2013). *Software Requirements* (3rd ed.). Redmond, WA: Microsoft Press.
11. ISO/IEC/IEEE 29148:2018. Systems and Software Engineering — Life Cycle Processes — Requirements Engineering. Geneva: International Organization for Standardization, 2018.
12. Object Management Group. (2011). Business Process Model and Notation (BPMN), Version 2.0. Needham, MA: OMG.
13. Object Management Group. (2017). OMG Unified Modeling Language (OMG UML), Version 2.5.1. Needham, MA: OMG.
14. Newman S. (2021) *Building Microservices: Designing Fine-Grained Systems* (2nd ed.). Sebastopol, CA: O'Reilly Media.
15. Kleppmann M. (2017). *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*. Sebastopol, CA: O'Reilly Media.
16. Fielding R.T. (2000). *Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation. University of California, Irvine.
17. Evans E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Boston: Addison-Wesley.

Сведения об авторах:

Болодурина Ирина Павловна, заведующий кафедрой прикладной математики

Оренбургского государственного университета имени В. А. Бондаренко, доктор технических наук, профессор
ORCID 0000-0003-0096-2587, E-mail: prmat@mail.osu.ru

Пояркова Елизавета Никитична, аспирант кафедры прикладной математики Оренбургского государственного университета имени В. А. Бондаренко
ORCID 0009-0008-4880-2711, E-mail: liza.yarko.e@mail.ru

Неволина Виктория Васильевна, заведующий кафедрой общей и профессиональной педагогики
Оренбургского государственного университета имени В. А. Бондаренко, доктор педагогических наук, доцент
ORCID 0000-0002-3506-1335, E-mail: nevolina-v@yandex.ru