

ОСОБЕННОСТИ ПРИМЕНЕНИЯ LOW-CODE ПЛАТФОРМЫ В ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПОЛЬЗОВАТЕЛЯМИ С РАЗНЫМ УРОВНЕМ ПОДГОТОВКИ

PECULIARITIES OF LOW-CODE PLATFORM APPLICATION IN THE TECHNOLOGY OF INFORMATION SYSTEM DESIGN BY USERS WITH DIFFERENT LEVEL OF TRAINING

**B. Goryachkin
O. Parsheva
A. Kanaev**

Summary. Problem statement. Traditional information systems development requires in-depth knowledge of programming, databases, and application architecture, which limits the accessibility of the process to non-professionals and creates a barrier for many organisations. Low-code platforms offer design with minimal coding, but the question remains as to how effective it will be for different users to build information systems on their own and what factors may influence this.

Goal. To analyze the peculiarities of low-code platform application in the technology of information system design by users with different level of training and to offer recommendations for optimization of system development process.

Results. The peculiarities of use and architecture of the domestic low-code platform, advantages, and disadvantages of designing on it are defined. The importance of the system approach to process modelling in the context of low-code development is substantiated. The key aspects of modelling and basic notations are considered, it is shown how they influence the efficiency and flexibility of information systems. The synthesised technology of information system design on low-code platform is presented, based on the stages of which the efficiency of the process is evaluated by the method of calculating labour costs.

Practical significance. The results of the study make it possible to identify the real potential of low-code platforms to reduce the dependence of organisations on professional developers and reduce the time and cost of development. The identified barriers and success factors for users with different levels of training will allow platform owners and educational institutions to create more effective functionality and training materials. The paper also provides a comparative analysis of existing process modelling notations, which identified BPMN 2.0 as the most suitable for low-code development.

Keywords: Low-code, platform, technology, design, information system, process, modeling, BPMN, evaluation, efficiency.

Горячкин Борис Сергеевич

кандидат технических наук, доцент,
Московский государственный технический
университет им. Н.Э. Баумана
bsgor@mail.ru

Паршева Олеся Александровна

Московский государственный технический
университет им. Н.Э. Баумана
parsheva.alesya@mail.ru

Канаев Алексей Александрович

Московский государственный технический
университет им. Н.Э. Баумана
alekseykanaev@mail.ru

Аннотация. Постановка проблемы. Традиционная разработка информационных систем требует глубоких знаний в программировании, базах данных и архитектуре приложений, что ограничивает доступность процесса непрофессионалам и порождает барьер для многих организаций. Low-code платформы предлагают проектирование с минимальным кодированием, однако остаётся актуальным вопрос о том, насколько эффективным будет самостоятельное создание информационных систем различными пользователями и какие факторы на это могут повлиять.

Цель. Проанализировать особенности применения low-code платформ в технологии проектирования информационной системы пользователями с разным уровнем подготовки и предложить рекомендации для оптимизации процесса разработки систем.

Результаты. Определены особенности использования и архитектура отечественной low-code платформы, преимущества и недостатки проектирования на ней. Дано обоснование важности системного подхода к моделированию процессов в контексте малокодовой разработки. Рассмотрены ключевые аспекты моделирования и основные нотации, показано, как они влияют на эффективность и гибкость информационных систем. Представлена синтезированная технология проектирования информационной системы на low-code платформе, на основании этапов которой осуществлена оценка эффективности процесса методом расчета трудозатрат.

Практическая значимость. Результаты исследования дают возможность определить реальный потенциал low-code платформ для снижения зависимости организаций от профессиональных разработчиков и сокращения сроков и стоимости разработки. Выявленные препятствия и факторы успеха для пользователей с разным уровнем подготовки позволят владельцам платформ и образовательным учреждениям создавать более эффективный функционал и обучающие материалы. В работе также приведён сравнительный анализ существующих нотаций моделирования процессов, в результате которого BPMN 2.0 определена как наиболее подходящая при малокодовой разработке.

Ключевые слова: Low-code, платформа, технология, проектирование, информационная система, процесс, моделирование, BPMN, оценка, эффективность.

Введение

Мир технологий стремительно развивается, а вместе с ним и потребности в управленческих решениях, которые позволяют быстро и эффективно адаптироваться к изменениям. Low-code платформы стали ответом на этот вызов, предлагая инструменты для создания информационных систем без глубоких знаний в программировании. Однако, несмотря на свою простоту, метод требует особого подхода к проектированию.

Этот сдвиг парадигмы в разработке предоставляет организациям новые возможности, и осмысление эффективности применения low-code пользователями с разным уровнем подготовки становится ключевым фактором для принятия обоснованных решений в выборе стратегии цифровой трансформации. Анализ основных особенностей тогда необходим для понимания перспектив и ограничений нового подхода.

Специфика low-code платформы в работах исследователей

Существует определенное количество методологий разработки информационных систем и инструментов их проектирования. Наиболее привлекательными являются платформы, которые позволяют проектировать, проводить модификацию системы с минимальным кодированием, такие как low-code [1].

При классической разработке архитектура усложняется из-за использования отдельных инструментов. Малокодовая же разработка интегрирует все эти функции в одну платформу, сокращая временные затраты и усиливая на проектирование [2].

К тому же, low-code платформы упрощают создание программного продукта, позволяя пользователям без глубоких знаний программирования разрабатывать информационные системы, но на больших и сложных проектах обратиться к специалисту всё же придётся [3].

Благодаря этому становится возможным построение полноценных междисциплинарных команд аналитиков и разработчиков, обеспечивающих активную интеграцию формируемой инфраструктуры не только в оборудование и технические системы организации, но также в его программные системы экономического профиля и общую структуру процессов [4].

Необходимо отметить, что по причине высокой динамичности изменений процессов, вызванных сокращением или расширением организации, изменением номенклатуры предоставляемых услуг или выпускаемой продукцией, изменением законодательства и пр., применяются BPM-системы. Они позволяют оптимизировать

повторный анализ, построение и внедрение новых моделей процессов [5].

Нередко создатель low-code платформы добавляет собственные языки программирования, думая, что это упрощает жизнь конечного разработчика. В итоге это не всегда верно и становится крайне затратно потом найти специалиста под эту платформу или обучить его самостоятельно без привлечения ресурсов создателя. Неприятным моментом является привязанность многих платформ к доступу в интернет. Немаловажным фактом является относительно высокая стоимость самих платформ [6].

Основные характеристики отечественной low-code платформы

Отечественные low-code платформы предназначены для решения следующего блока задач:

- Разработка информационных систем любой сложности с минимальным привлечением программистов.
- Автоматизация специфических процессов, не охваченных функционалом insource и outsource решений.
- Обеспечение интеграции бизнес-систем с внутренними и внешними цифровыми ресурсами и сервисами.
- Импортозамещение цифровых инструментов.

В современной платформе реализованы принципы малокодовой (low-code) разработки программного обеспечения и технологии RAD-проектирования, ориентированные на максимально быстрое получение результата в условиях сильных ограничений по срокам и бюджету и нечетко определенных требований к продукту.

Применение платформы для разработки собственных информационных систем имеет ряд ключевых особенностей. Она дает возможность создавать любые учетные разделы и регистры под имеющиеся задачи, а не использовать готовые блоки, подобно стандартному конструктору. Это значит, что система будет полностью соответствовать специфике отрасли, а не подстраиваться под ограничения типового решения.

Высокая скорость разработки обеспечивается тем, что практически готовый макет системы создает аналитик, владеющий соответствующей компетенцией. На этом этапе отсутствует необходимость привлечения программистов, что значительно сокращает время и стоимость разработки, позволяя быстро прототипировать и корректировать требования.

В платформе можно вносить изменения в проект поэтапно на любой стадии, что делает процесс разработки

более гибким. Это позволяет оперативно реагировать на меняющиеся бизнес-требования и избежать дорогостоящих переделок в конце проекта.

Она также позволяет при создании решений не разрабатывать общий системный функционал, предназначенный для администрирования, контроля данных, управления ролями и пользователями. Такая особенность экономит ресурсы разработчиков, поскольку основные административные функции уже встроены в платформу, позволяя сосредоточиться на бизнес-логике приложения.

Использование платформы имеет следующие важные преимущества:

- Снижение рисков на начальных стадиях проектирования и создания информационных систем.
- Возможность постоянного тестирования функционала и своевременного его изменения.
- Реальная оценка текущего состояния проекта за счет возможности получения промежуточных результатов.
- Гибкие механизмы для многопользовательской разработки.

С помощью платформы можно создавать сложные функциональные корпоративные системы либо локальные информационные сервисы. При этом системы будут обладать следующими характеристиками:

- Полное отсутствие лицензионной нагрузки.
- Открытые программные коды.
- Отсутствие любых ограничений по срокам использования или количеству пользователей.
- Отсутствие ограничений по вертикальному и горизонтальному масштабированию.

В ходе исследования существующих решений авторы пришли к выводу, что платформа и создаваемые на её основе системы должны иметь трёхзвенную архитектуру (рис. 1).



Рис. 1. Архитектура современной low-code платформы

Для работы на стороне клиента используются любые интернет-браузеры. Приложение функционирует на базе языка программирования Java и свободно расширяемых фреймворков. На уровне базы данных чаще всего используется СУБД PostgreSQL любой версии.

Современная платформа представляет собой студию для разработки функционала и содержит следующие базовые модули:

1. Модуль администратор — для администрирования создаваемой системы и выполнения системных настроек.
2. Модуль для хранения и настройки нормативно-справочной информации.
3. Модуль для настройки интеграции с внешними информационными ресурсами.
4. Основной модуль — конструктор конфигурации, в котором и осуществляется создание функционала будущей системы.

Спроектированная в платформе информационная система также имеет модульную архитектуру. В её состав по умолчанию включаются модули: администратор, нормативно-справочная информация, интеграция и все функциональные модули, разработанные с помощью конструктора конфигураций.

Нотации моделирования процессов

Ключевой элемент эффективной low-code разработки — моделирование процессов (рис. 2). Процесс определяется как комплекс мероприятий, направленный на воспроизведение некоторого продукта или услуги. Под моделированием процессов понимается создание абстрактного представления реальных процессов с использованием графических схем, диаграмм и других инструментов, позволяющее визуализировать и анализировать последовательность действий, взаимодействие участников и ресурсов.

Модель процесса разрабатывается визуальными средствами при помощи графического представления объектов — Business Process Modeling Notation (BPMN). На данный момент идет активное создание платформ BPMS, отвечающих требованиям современных динамических бизнес-процессов [7].

Системный подход к моделированию процессов предоставляет ясный образ работы организации и позволяет определить необходимые функции системы для достижения желаемых результатов.

Однако часто моделирование процессов остается незадействованным этапом, что может привести к нарушениям работы системы и неудовлетворенности пользователей. Вместо того, чтобы стать инструментом



Рис. 2. Основные компоненты процесса

успеха, low-code может превратиться в опасность, если не применяется грамотно. Рассмотрим далее основные нотации моделирования процессов (рис. 3).

IDEFO является методом функционального моделирования, ориентированным на описание функций организации и их взаимосвязей. IDEFO использует диаграммы с четырьмя квадрантами, представляющими входные данные, выходные данные, управление и ресурсы. Эта нотация идеально подходит для анализа процессов в целом, а также для разработки системных требований к программному обеспечению. IDEFO помогает понять, как функции взаимодействуют друг с другом, какие ресурсы необходимы для выполнения каждой функции и какие ограничения накладываются на эти функции. При проектировании информационных систем IDEFO позволяет структурировать процессы, определить ключевые функции системы и затем разработать функциональные требования к информационной системе.

Базовая диаграмма процесса (Basic Flowchart) — это простая и наглядная нотация, используемая для визуализации последовательности шагов в процессе. Она использует графические символы (прямоугольники, ромбы, стрелки) для представления действий, условий и направлений потока процесса. Базовые диаграммы процессов особенно полезны при проектировании информационных систем для представления алгоритмов, логических операций и последовательности действий в системе. К примеру, можно использовать диаграмму процесса для визуализации алгоритма обработки заказа, ввода данных или автоматизации определенного действия в системе.

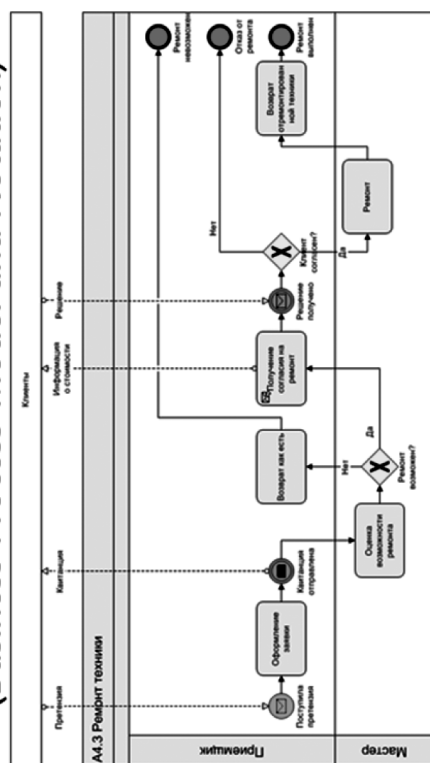
Процедура (Cross Functional Flowchart) — есть визуальный язык, помогающий понять, как различные отде-

лы компании взаимодействуют при выполнении задач. Представим себе таблицу, где столбцы — это отделы (например: продажи, маркетинг, производство), а строки — этапы процесса. Используя символы для действий, решений и передачи информации, мы можем наглядно показать, как информация и задачи перемещаются между отделами. Это позволяет увидеть, где возникают задержки, дублирование усилий, а также найти возможности для оптимизации.

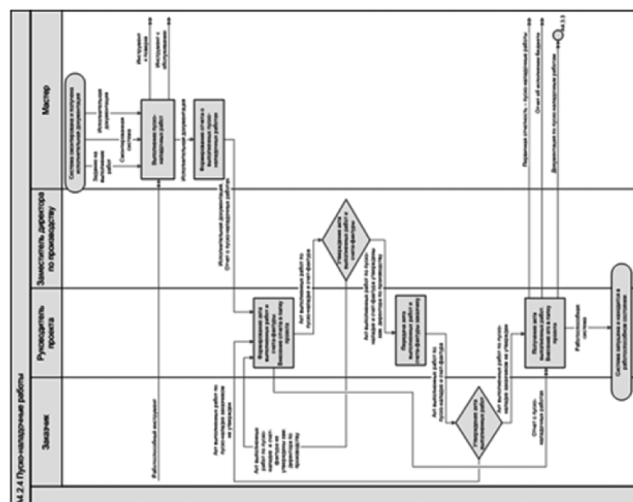
BPMN 2.0 (Business Process Model and Notation) позиционируется как стандартная нотация для моделирования бизнес-процессов, предназначенная для описания сложных процессов с разными участниками и взаимодействиями. Она предполагает использование широкого набора символов для представления различных типов действий, условий, ролей, пулов, подпроцессов, событий и т.д. BPMN 2.0 является одной из самых популярных нотаций для моделирования процессов в контексте проектирования информационных систем, потому что она позволяет создать детальную и понятную модель процессов, которая может быть использована как для разработки системы, так и для документации и обучения пользователей.

EPC (Event-Driven Process Chain) выступает нотацией для моделирования процессов, фокусирующейся на взаимодействиях между разными функциями или организационными единицами. Она использует символы для представления функций, событий и связей между ними. EPC позволяет продемонстрировать последовательность событий и действий в процессе, а также показать взаимодействие между разными подсистемами и системами. Эта нотация особенно полезна при проектировании информационных систем, включающих взаи-

4. BPMN (Business Process Model and Notation)



3. Процедура (Cross Functional Flowchart)



5. EPC (Event-Driven Process Chain)

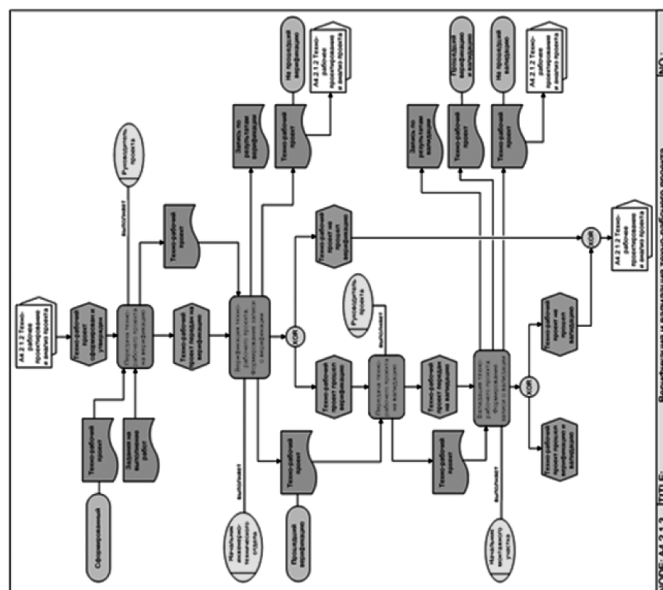


Рис. 3. Нотации моделирования процессов

модействие с внешними системами или учет различных событий. Например, EPC можно использовать для моделирования процесса обработки заказа, где участвуют внешние поставщики, система управления запасами и система доставки.

Авторами был проведен сравнительный анализ для определения наиболее подходящей при проектировании нотации. IDEFO, фокусируясь на функциональных декомпозициях, отлично подходит для структурирования сложных систем, но не так эффективна для описания динамических процессов взаимодействия. Базовая диаграмма процесса, сосредоточенная на последовательности действий, проста и понятна, но недостаточно гибка для сложных процессов с параллельными путями и условиями ветвления. Процедура, хотя и специализируется на визуализации взаимодействия между отделами, но не обеспечивает такого уровня структурирования и формализации, как BPMN 2.0. Наконец, EPC, ориентированная на события, не всегда удобна для конкретной реализации и может быть слишком абстрактной для практического применения.

BPMN 2.0, в отличие от других нотаций, отлично подходит для системного подхода к моделированию процессов, потому что позволяет визуализировать все процессы организации как единое целое, разбивать их на подпроцессы, указывать роли и ответственных, что способствует комплексному анализу и повышению эффективности работы. Кроме того, нотация обеспечивает возможность перехода от визуальной схемы к XML-коду, автоматизируя создание базовых элементов системы и упрощая проектирование на low-code платформе.

Технология проектирования на low-code платформе

Определив основные особенности и наиболее подходящую нотацию моделирования бизнес-процессов, авторы проработали синтезированную технологию проектирования. Её важность заключается в том, что она позволяет создавать и развертывать приложения значительно быстрее и с меньшими затратами, чем традиционные методы разработки, делая цифровые решения доступными для более широкого круга пользователей и организаций. Авторами предлагается разделить проектирование информационных систем с помощью платформы на пять этапов.

На первом этапе с помощью интерфейса платформы настраиваются таблицы базы данных, которые будут использоваться в учетных разделах системы. В разделе, условно именуемом «Таблицы базы данных», добавляются соответствующие таблицы и настраиваются необходимые параметры.

На втором этапе по созданным таблицам настраиваются разделы и экранные формы. Соответствующий интерфейс платформы генерируется автоматически по указанным при настройке параметрам. Здесь же подбираются пункты контекстного меню, связанные со специальными действиями или печатью отчетов из раздела.

На третьем этапе осуществляется формирование функционального модуля, настройка его главного меню и включение в модуль созданных разделов, действий и отчетов.

На четвертом этапе осуществляется разработка и привязка к разделам специальных действий, процедур и отчетов. Это единственный этап, где могут быть задействованы программисты SQL. С их помощью разрабатываются по запросу аналитика соответствующие хранимые процедуры, которые потом привязываются к действиям в разделах.

На последнем этапе осуществляется тестирование создаваемой системы. В первую очередь проводится функциональное тестирование. При обнаружении ошибок в функционале они исправляются в студии. В платформе также может быть заложен функционал нагрузочных испытаний.

После успешных проверок система выгружается из студии и может быть введена в эксплуатацию. Функционал, созданный с помощью отечественной платформы, может стать неким фреймворком для филиалов или смежных организаций, а также выступить фундаментом государственной информационной системы.

Оценка эффективности применения low-code платформы при проектировании информационной системы

Проведем сравнительный анализ и оценим эффективность проектирования информационной системы на low-code платформе и с использованием инструментария программирования. В основу расчета трудозатрат (человеко-часы) заложены описанные выше этапы, а также следующая классификация информационных систем:

- Простая система (начальная подготовка пользователей): несколько таблиц, базовый функционал, минимальная кастомизация.
- Средняя система (средняя подготовка пользователей): несколько десятков таблиц, более сложная бизнес-логика, средняя кастомизация.
- Сложная система (продвинутая подготовка пользователей): большое количество таблиц, комплексная бизнес-логика, высокая степень кастомизации, интеграции со сторонними системами.

Важно подчеркнуть, что универсальных стандартов расчета трудозатрат, применимых ко всем типам проек-

тов разработки, не существует. Поэтому оценки (рис. 4) являются результатами системного анализа проведенных ранее исследований [8, 9] и экспертных оценок. В реальной практике они осуществляются так же по задачам, существующим сценариям или планам спринтов.

По результатам оценки авторы отмечают, что во всех трёх вариантах сложности информационной системы проектирование на low-code платформе показывает значительное (в 2–2,5 раза) сокращение времени по сравнению с разработкой через код. При этом следует помнить, что осуществление четвёртого этапа невозможно без применения дополнительного кода и, соответственно, внешних инструментов.

Для наиболее сложных информационных систем, хотя low-code всё ещё выигрывает по времени, разница не столь критична, поскольку необходимы дополнительные настройки для сложной логики и интеграций. Безусловно, это требует высоких пользовательских навыков и увеличивает период работы на платформе. Также не стоит забывать, что на выбор подхода влияют не только временные затраты, но и другие факторы (бюджет, на-

личие квалифицированных специалистов, требования к кастомизации и т.д.).

Заключение

Использование low-code платформы открывает новые возможности для быстрой и гибкой разработки, но требует особого внимания к архитектуре и особенностям взаимодействия с ней. Синтезированная технология проектирования на low-code платформе подразумевает комплексный подход, включающий в себя возможность адаптировать нотацию, определение необходимого функционала платформы и системное моделирование бизнес-процессов. Такой подход позволяет создавать эффективные, гибкие и масштабируемые информационные системы, которые отвечают современным требованиям.

Будущие исследования в этой области должны сосредоточиться на изучении влияния различных типов low-code платформ и подходов к их применению на результаты проектирования. Крайне важно также проанализировать возможность использования мало-

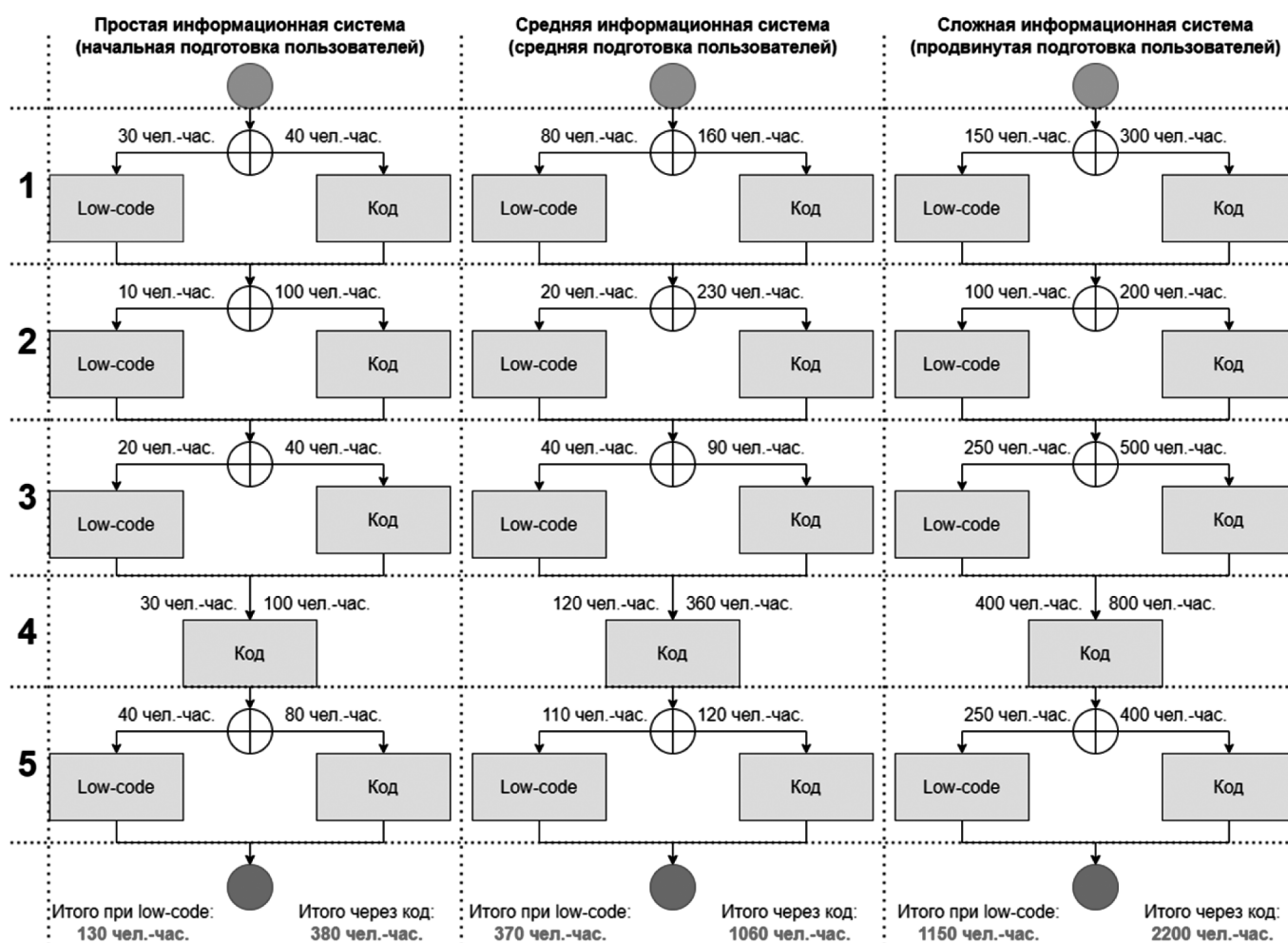


Рис. 4. Расчет трудозатрат при проектировании информационной системы

кодовой разработки в контексте государственных информационных систем, учитывая специфические требования к безопасности, масштабируемости, соответствию

нормативным актам и интеграции с существующей инфраструктурой.

ЛИТЕРАТУРА

1. Колдунова И.Д. Разработка автоматизированных информационных систем на базе low-code платформ / И.Д. Колдунова, В.В. Лихачев, А.А. Макаревич // Перспективы науки. — 2023. — № 3(162). — С. 30–33. — EDN ZDJMLG.
2. Кашеваров В.И. Применение концепций Low-code и BPM для создания новой методики обработки багажа авиапассажиров / В.И. Кашеваров // Информационные технологии в управлении и экономике. — 2022. — № 2(27). — С. 25–32. — EDN WHWTAQ.
3. Абдуллаев Э.А. Low-code и No-code платформы / Э.А. Абдуллаев. — Текст: непосредственный // Молодой ученый. — 2024. — № 32 (531). — С. 6–7. — URL: <https://moluch.ru/archive/531/117021/> (дата обращения: 15.12.2024).
4. Голов Р.С. Перспективы использования технологий «low-code» и «no-code» при проектировании цифровой инфраструктуры высокотехнологичного предприятия / Р.С. Голов, А.В. Мыльник // СТИН. — 2023. — № 2. — С. 33–35. — EDN QZQRIR.
5. Патрусова А.М. Системный анализ процессов организации: инструментальный исследования / А.М. Патрусова // Труды Братского государственного университета. Серия: Экономика и управление. — 2021. — Т. 1. — С. 117–123. — EDN TPOYIR.
6. Иванов Ф.Ф. Использование Low-code платформ при переходе на процессный подход в создании автоматизированных систем // Ф.Ф. Иванов, Г.С. Яковлев // Вестник КРАУНЦ. Физико-математические науки. — 2020. № 1 30. — С. 120–126. — <http://krassec.ru/yakovlev2020-1/> (дата обращения: 15.12.2024). doi: 10.26117/2079–6641-2020-30-1-120-126).
7. Петровская К.В. Системы BPM для управления бизнес-процессами / К.В. Петровская // Приоритетные направления развития науки и образования: сборник статей VII Международной научно-практической конференции: в 2 ч., Пенза, 20 июня 2019 года. Том Часть 1. — Пенза: «Наука и Просвещение» (ИП Гуляев Г.Ю.), 2019. — С. 80–81. — EDN UKBYPM.
8. Семенов В.П. Мягкая оценка эффективности разработки интеллектуальных информационных систем / В.П. Семенов, Р.В. Соколов, И.Л. Андреевский // Международная конференция по мягким вычислениям и измерениям. — 2024. — Т. 1. — С. 483–485. — EDN JXJUFZ.
9. Катасонова Н.С. RICEFS-классификация разработок и настроек для оценки трудозатрат // Корпоративные информационные системы. — 2023. — №4 (24) — С. 26–37. — URL: <https://corpinfosys.ru/archive/2023/issue-24/230-2023-24-ricefclassification> (дата обращения: 15.12.2024).

© Горячкин Борис Сергеевич (bsgor@mail.ru); Паршева Олеся Александровна (parsheva.alesya@mail.ru);

Канаев Алексей Александрович (alekseykanaev@mail.ru)

Журнал «Современная наука: актуальные проблемы теории и практики»