

УДК 004.415.2

DOI: [10.26102/2310-6018/2026.53.2.006](https://doi.org/10.26102/2310-6018/2026.53.2.006)

Механизм управления требованиями и качеством IT-продуктов

М.Р. Котова^{1✉}, В.Е. Большев²

¹Инвестиционный банк Барклайс, Прага, Чешская Республика

²Институт психологии Российской академии наук (ИП РАН), Москва,
Российская Федерация

Резюме. Статья посвящена исследованию взаимосвязи качества IT-продуктов и процессов управления требованиями, влияющими на предсказуемость разработки и результативность функционирования программных систем. Актуальность исследования обусловлена тем, что именно качество требований во многом определяет уровень дефектности программных продуктов, трудоемкость переработок и риски проектной реализации. В работе раскрыты теоретические подходы к пониманию качества IT-продукта и уточнено его трактование как функции соответствия требованиям, ожиданиям пользователей и ограничениям разработки. На основе анализа литературы определены группы основных требований (бизнес-, пользовательские, функциональные, нефункциональные и системные требования к качеству IT-продукта), а также выявлены типовые проблемы их формализации, влияющие на качество продукта. Методологическую основу исследования составили теоретический анализ и обобщение научных источников, сравнительный анализ подходов к инженерии требований, систематизация и теоретическое моделирование процессов управления качеством IT-продуктов. Установлены закономерности, связывающие качество требований с уровнем дефектности, трудоемкостью переработок и рисками проектной реализации. Предложены и описаны структурные элементы механизма управления требованиями и качеством IT-продуктов (подсистемы сбора, анализа, согласования требований и контроля качества, а также вспомогательный инструмент – структурированное описание продукта). Разработан механизм, обеспечивающий сквозную прослеживаемость требований и формирование качества на всех стадиях жизненного цикла разработки. Сделаны выводы о научной и практической значимости механизма, его универсальности и потенциале повышения эффективности процессов разработки программного обеспечения.

Ключевые слова: качество IT-продукта, инженерия требований, критерии качества, управление требованиями, подсистема контроля качества, механизм управления качеством, жизненный цикл разработки программного обеспечения.

Для цитирования: Котова М.Р., Большев В.Е. Механизм управления требованиями и качеством IT-продуктов. *Моделирование, оптимизация и информационные технологии*. 2026;14(2). URL: <https://moitvvt.ru/ru/journal/article?id=2146> DOI: 10.26102/2310-6018/2026.53.2.006

Mechanism for managing requirements and quality of IT products

M.R. Kotova^{1✉}, V.E. Bolshev²

¹Barclays Investment Bank, Prague, Czech Republic

²Institute of Psychology Russian Academy of Sciences, Moscow, the Russian Federation

Abstract. The article examines the relationship between the quality of IT products and requirements management processes that influence development predictability and the performance of software systems. The relevance of the study is determined by the fact that the quality of requirements largely defines the defect rate of software products, the effort required for rework, and the risks associated with project implementation. The study outlines theoretical approaches to understanding IT product quality

and clarifies its interpretation as a function of compliance with requirements, user expectations, and development constraints. Based on a literature review, the main groups of requirements (business, user, functional, non-functional, and system requirements related to IT product quality) are identified, and typical issues in their formalization that affect product quality are revealed. The methodological framework of the study is based on theoretical analysis and synthesis of scientific sources, comparative analysis of approaches to requirements engineering, systematization, and theoretical modeling of IT product quality management processes. The research establishes patterns linking the quality of requirements with defect levels, rework intensity, and project implementation risks. The structural components of a mechanism for managing requirements and IT product quality are proposed and described, including subsystems for requirements elicitation, analysis, alignment, and quality control, as well as a supporting tool – a structured product description. A mechanism ensuring end-to-end traceability of requirements and the formation of quality at all stages of the software development lifecycle is developed. The paper concludes with insights into the scientific and practical significance of the mechanism, its universality, and its potential to enhance the efficiency of software development processes.

Keywords: IT product quality, requirements engineering, quality criteria, requirements management, quality control subsystem, quality management mechanism, software development lifecycle.

For citation: Kotova M.R., Bolshev V.E. Mechanism for managing requirements and quality of IT products. *Modeling, Optimization and Information Technology*. 2026;14(2). (In Russ.). URL: <https://moitvvt.ru/journal/article?id=2146> DOI: 10.26102/2310-6018/2026.53.2.006

Введение

Интенсивные темпы развития мирового ИТ-сектора, сопровождающиеся усиливающейся конкуренцией и ускоряющимся инновационно-технологическим циклом, определяют возрастающие требования к качеству создаваемых ИТ-продуктов. Как отмечается в современной научной литературе, между конкуренцией и инновациями в ИТ-секторе существует каузальная связь, представленная тем, что конкурентное давление становится стимулирующим инновации фактором. В частности, согласно существующим оценкам, расширение конкуренции увеличивает количество патентов на 27,3 % и количество ненаследуемых цитирований на 38,2 %; причем чем конкурентнее рыночный сегмент, тем более выраженным будет подобный эффект [1]. Закономерно, конкурентоспособность ИТ-компаний формируется не только исходя из их технологического лидерства, но и способности поддерживать качество продуктов, что неразрывно связано с готовностью учитывать требования пользователей и эффективно управлять изменениями.

Учитывая вышеприведенное, как никогда актуальными становятся вопросы управления требованиями и качеством ИТ-продуктов, сопряженные с влиянием качества на рыночный успех. По мнению A. Nazhifan и Y. Lia, качество продукта в совокупности с инновационностью прямо формирует конкурентное преимущество и определяет состояние бренда, доверие потребителей к нему. При этом в сфере программной разработки качество рассматривается и как совокупность технических характеристик, и как степень соответствия продукта заявленным требованиям, ожиданиям пользователей и организационным ограничениям процесса разработки [2]. Соответственно, качество ИТ-продуктов невозможно обеспечить в отсутствие развитой системы и механизмов управления требованиями. Так, именно требования формируют основу проектной деятельности, задают рамки проектируемой системы, служат критерием приемки и основным источником оценки степени соответствия продукта ожиданиям заказчика. Кроме того, проблемы качества в большинстве случаев являются следствием несогласованности требований, их размытости, неполноты, отсутствия формализации или постоянных изменений без соблюдения процедур управления. Поэтому связь «качество и требования» является прямой и функциональной, поскольку нарушение

цепочки управления требованиями приводит к дефектам программного продукта, переработкам, задержкам выпуска и снижению потребительской ценности конечного ИТ-продукта.

Опираясь на данные представления о процессах управления требованиями и качеством ИТ-продуктов, целесообразность приобретает разработка и формализация процессов, образующих систему менеджмента качества (далее – СМК); основу подобной формализации составляет разработка механизма управления, который позволит структурировать работу над требованиями, обеспечить согласованность действий всех участников разработки и выстроить последовательный процесс формирования качества на всех стадиях жизненного цикла продукта, что определило цель и границы настоящего исследования.

Цель исследования – разработать механизм управления требованиями и качеством ИТ-продуктов.

Методы исследования

Теоретико-методологический базис исследования составили труды ученых, посвященные вопросам качества ИТ-продукта, требований к качеству, классификации и формализации требований и критериев качества; отдельное внимание уделено современным стандартам СМК, применимым к деятельности ИТ-компаний, а также положениям инженерии требований. В части методов исследования применялись теоретический анализ, синтез, сравнительный анализ, сопоставление, обобщение, систематизация, а также теоретическое моделирование.

Проблематика качества программных продуктов занимает центральное место в современной инженерии программного обеспечения. В международной практике качество определяется как степень соответствия программной системы установленным требованиям, ожиданиям пользователей и ограничительным условиям процесса разработки. Подобное определение качества заложено в стандартах серии ISO/IEC 25000 (SQuaRE), в частности, ISO/IEC 25010, в котором качество программного продукта понимается как совокупность внутренних, внешних и эксплуатационных характеристик, направленных на удовлетворение потребностей заинтересованных сторон. Таким образом, качество ИТ-продукта выражается в сохранении баланса между тремя аспектами: требования, ожидания, и ограничения разработки, что позволяет рассматривать качество не как абстрактную категорию, а как функциональный результат управления требованиями.

Согласно тезисам исследования А.А. Жуйковой, в инженерии требований их развитие традиционно рассматривается как итерационный процесс исследования проблемы, формализации, документирования и проверки корректности понимания целей продукта, что непосредственно влияет на качество итоговой системы. В работе автора указывается, что требования задают рамки будущей системы, а их спецификация является не столько описанием функций, сколько и определением характеристик, напрямую связанных с качеством продукта. Автор разделяет требования на две крупные группы – требования заказчика и требования разработчика. Соответственно, качественный продукт создается только при условии согласования данных групп требований и устранения разрывов в коммуникации, направленной на сообщение и учет требований [3].

Примечательными в контексте рассуждений о теоретической сущности качества видятся тезисы исследования С.В. Зубковой и Р.А. Ястребова, которыми выделяется, что качество ИТ-продукта формируется уже на стадии постановки и анализа требований. В свою очередь, нарушения в управлении требованиями нередко становятся основным источником дефектов, переработок и отклонений в сроках разработки. Авторы

отмечают, что основными факторами снижения предсказуемости качества (и, закономерно, роста рисков качества) выступают: субъективность требований, отсутствие формализованной ответственности и несогласованность действий между заказчиком и разработчиком [4].

Опираясь на представления авторов, отметим, что качество не существует отдельно от процесса управления требованиями, поскольку оно является производной его зрелости, т. е. результатом. Также стоит указать, что в соответствии с ГОСТ Р ИСО/МЭК 25010-2015, параметры качества программного обеспечения структурируются по восьми основным характеристикам (Рисунок 1):

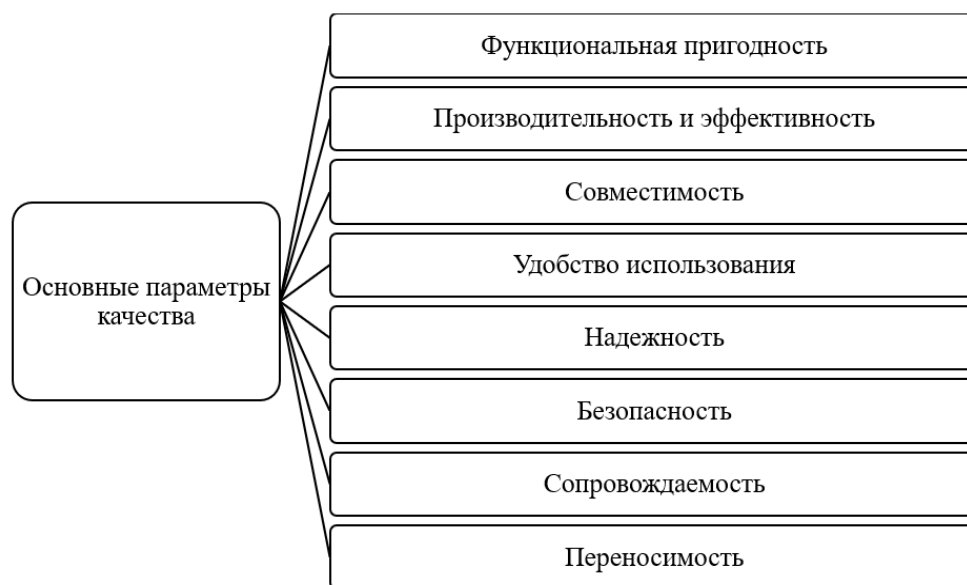


Рисунок 1 – Основные параметры качества программного обеспечения в соответствии с ГОСТ Р ИСО/МЭК 25010-2015

Figure 1 – Key software quality characteristics in accordance with GOST R ISO/IEC 25010-2015

Обращаясь к Рисунку 1, стоит подчеркнуть, что каждая из представленных характеристик раскрывается через набор подхарактеристик и конкретных метрик, которые служат основой для оценки качества продукта на различных стадиях жизненного цикла. Например, удобство использования определяется метриками понятности интерфейса, доступности, простоты освоения; надежность определяется частотой отказов, устойчивостью к ошибкам и восстанавливаемостью; функциональная пригодность – полнотой, целесообразностью и точностью реализации требований и т. д.

Исходя из наличия параметров качества, которые могут формироваться постепенно по мере разработки ИТ-продукта, качество стоит отождествлять с системой свойств, которые отражают корректность и полноту требований, конкретные характеристики архитектуры ИТ-продукта, состояние кода, сервисов и т. п., корректность тестирования, а также соблюдение критериев приемки. Тем самым, переходя от теоретического понимания качества к его прикладному смыслу, под качеством стоит понимать системную категорию, опирающуюся на процессы инженерии требований, архитектурного проектирования, разработки и тестирования.

Кроме того, на практике важным элементом обеспечения качества ИТ-продукта становятся входные требования; именно требования являются базисом проектирования, разработки и последующего контроля качества ИТ-продуктов. Как справедливо указывает В.К. Батоврин, требования определяют функциональные границы системы, параметры ее работы, ограничения среды и критерии успешности, и рассматриваются

как формализованная модель ожиданий и потребностей заинтересованных сторон [5]. Поэтому, опираясь на практический опыт, допущенные ошибки при работе с требованиями (начальный этап разработки ИТ-продукта) нередко впоследствии становятся причиной дефектов, которые возникают в процессе эксплуатации.

При этом данную точку зрения разделяют и зарубежные авторы, которые также указывают на факт зависимости качества программного продукта от качества процесса управления требованиями. Так, J. Frattini и соавторы отмечают, что требования являются важной переменной, определяющей предсказуемость разработки. Нарушения в структуре требований по их оценкам формируют значительную часть дефектов системы. Авторы предлагают гармонизированную теорию качества требований с позиции их полноты, непротиворечивости, проверяемости, трассируемости и однозначности [6].

Опираясь на систематический обзор S. Khalid и соавторов, стоит также указать, что наиболее частой причиной неудачи ИТ-продуктов становятся именно неоднозначные требования, которые вкупе с неразвитой коммуникацией приводят к ряду ошибок. В числе последних выделяются: несоответствие техник сбора контексту проекта, неопределенная проектная видимость, пропуск стейкхолдеров и противоречия в требованиях [7].

Результаты

Учитывая рассмотренные исследования, требования к ИТ-продукту можно представить в виде пяти формализованных групп:

- во-первых, бизнес-требования, которые отражают организации, ценности продукта, экономическую целесообразность и стратегические ожидания заказчика. Тем самым данная группа требований формирует контекст разработки и определяет, «зачем» создается продукт;
- во-вторых, пользовательские требования, описывающие потребности конечных пользователей, ожидаемые сценарии взаимодействия, задачи, которые продукт помогает решать. Основное внимание при рассмотрении данных требований уделяется вопросам удобства, функциональности, эффективности использования;
- в-третьих, функциональные требования, которые определяют назначение и то, что должна «делать» система (ИТ-продукт и его поведение): сервисы, действия, реакции на события, бизнес-правила;
- в-четвертых, нефункциональные требования, связанные с принципами работы системы, которые можно свести к надежности, производительности, безопасности, масштабируемости, удобству использования;
- в-пятых, системные и архитектурные требования, которые определяют техническую среду, ограничения среды исполнения, используемые технологические стеки, интеграционные интерфейсы, требования к безопасности и поддержке и т. п.

Кроме того, в соответствии с группами требований выделяются и конкретные типовые проблемы – неполнота, неясность, неоднозначность, несогласованность бизнес-требований с системными, архитектурными и функциональными [7, 8]; избыточная детализация или обобщенность требований [9]; отсутствие прототипов, примеров или спецификаций требований; отказ от документирования требований [10]; и т. п.

Таким образом, формируется явная закономерность во влиянии требований на качество ИТ-продукта; последнее определяется как соотношение качества требований с качеством и степенью их реализации. Исходя из этого, формируется система предпосылок, обосновывающих необходимость разработки механизма управления требованиями и качеством ИТ-продукта как взаимосвязанными подсистемами. Требования при этом являются отдельной системой такого механизма (Рисунок 2):

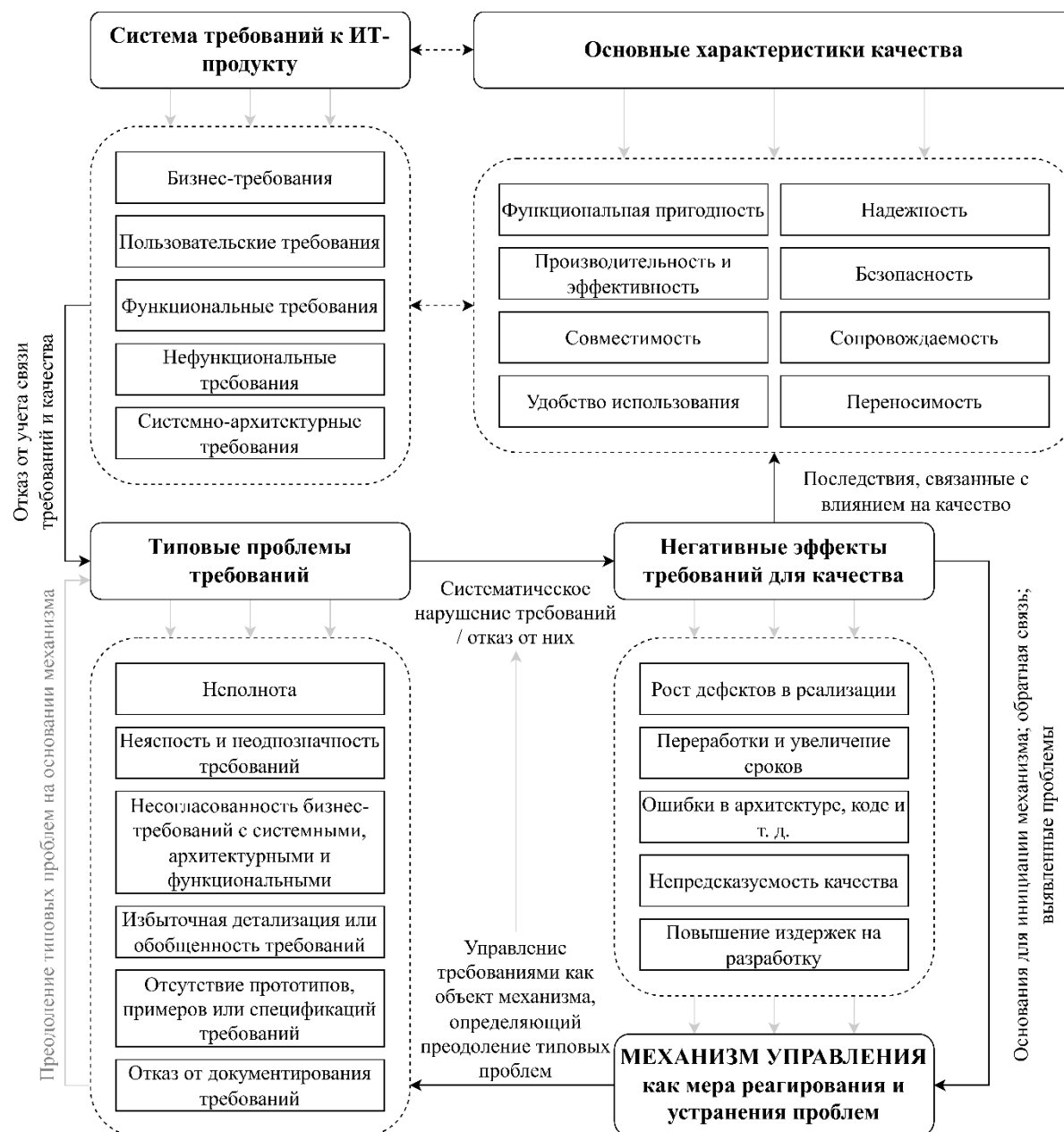


Рисунок 2 – Требования как фактор обеспечения качества ИТ-продукта
Figure 2 – Requirements as a factor in ensuring IT product quality

Опираясь на Рисунок 2, отметим, что необходимость создания механизма управления требованиями и качеством обусловлена связью между зрелостью управления требованиями и качеством ИТ-продуктов. Исходя из анализа литературы, механизм управления требованиями и качеством ИТ-продукта можно определить как совокупность взаимосвязанных подсистем, которые обеспечивают полный цикл работы с требованиями. Механизм функционирует как система процессов, в которой требования выступают входом, а качественный ИТ-продукт – выходом, т. е. результатом успешного действия механизма. Структурно механизм можно представить в виде 4 основных подсистем и дополнительного инструмента управления качеством (Таблица 1).

Таблица 1 – Структурные элементы механизма и их описание
Table 1 – Structural elements of the mechanism and their description

№	Структурный элемент	Описание
1	Подсистема сбора требований	Обеспечивает формирование исходного набора требований, идентификацию заинтересованных сторон, выявление проблем, ожиданий, ограничений, сценариев использования и пользовательских путей. На этапе действия подсистемы формируется первичная модель продукта, которая содержит в себе элементы бизнес-ценностей, пользовательских историй, целей и контекста
2	Подсистема анализа и формализации требований	Выполняется структурирование требований, их декомпозиция, классификация (функциональные, нефункциональные, бизнес-, системные и др.), проверка на корректность, полноту и непротиворечивость. Создается спецификация требований как документ-руководство для архитекторов, разработчиков и тестировщиков
3	Подсистема согласования требований с командой	Состоит из процессов визуализации требований, разработки критериев приемки, ревью требований в команде, планирования реализации в рамках спринтов или этапов, управления изменениями и версиями продукта. Подсистема необходима для того, чтобы сформировать единое понимание продукта всеми участниками разработки
4	Подсистема контроля качества (СМК)	Рассматривает качество как интегральный результат реализации требований, коммуникаций и иных элементов качества. Подсистема состоит из контрольных мероприятий, анализа качества кода, функционального и нефункционального тестирования, проверки соответствия критериям приемки, выявления отклонений и их устранения до поставки продукта
5	Структурированное описание продукта как инструмент управления качеством	Отражает типовую структуру описания продукта, а именно его: цель и проблему, целевую аудиторию, сценарии использования, основные ограничения, критерии приемки, визуальный прототип

Общая система функционирования механизма выстраивается следующим образом (Рисунок 3):

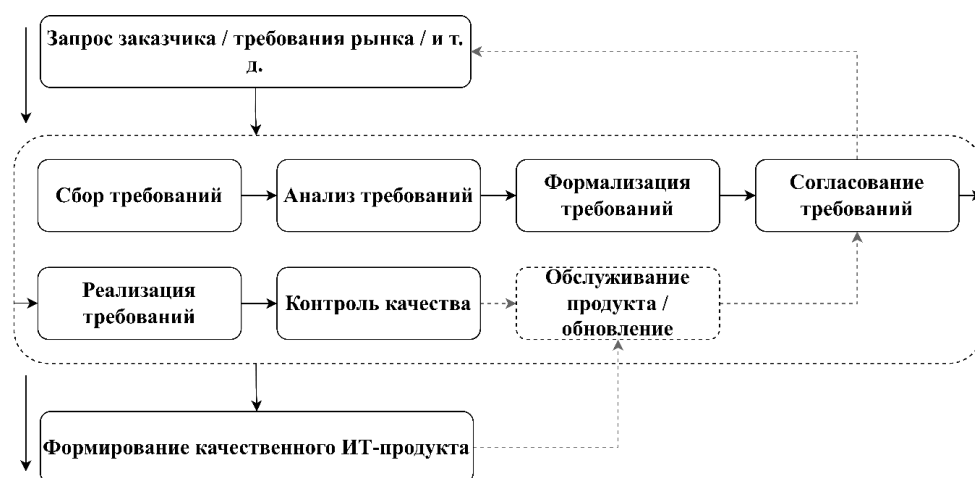


Рисунок 3 – Принцип функционирования механизма управления требованиями и качеством ИТ-продуктов

Figure 3 – Operating principle of the mechanism for managing requirements and IT product quality

Итак, очевидно, что действие механизма представляется в качестве сквозного цикла преобразования требований в качество, который обеспечивает прослеживаемость требований на всех стадиях создания продукта, предотвращает возникновение дефектов из-за неполных или неясных требований, определяет предсказуемость результата за счет применения критериев качества и приемки, устанавливает взаимосвязанность действий всех участников разработки (заказчика, аналитиков, архитекторов, разработчиков, тестировщиков и др.). При этом расширенное действие механизма можно представить в виде следующих подсистем и их формализации (Рисунок 4):

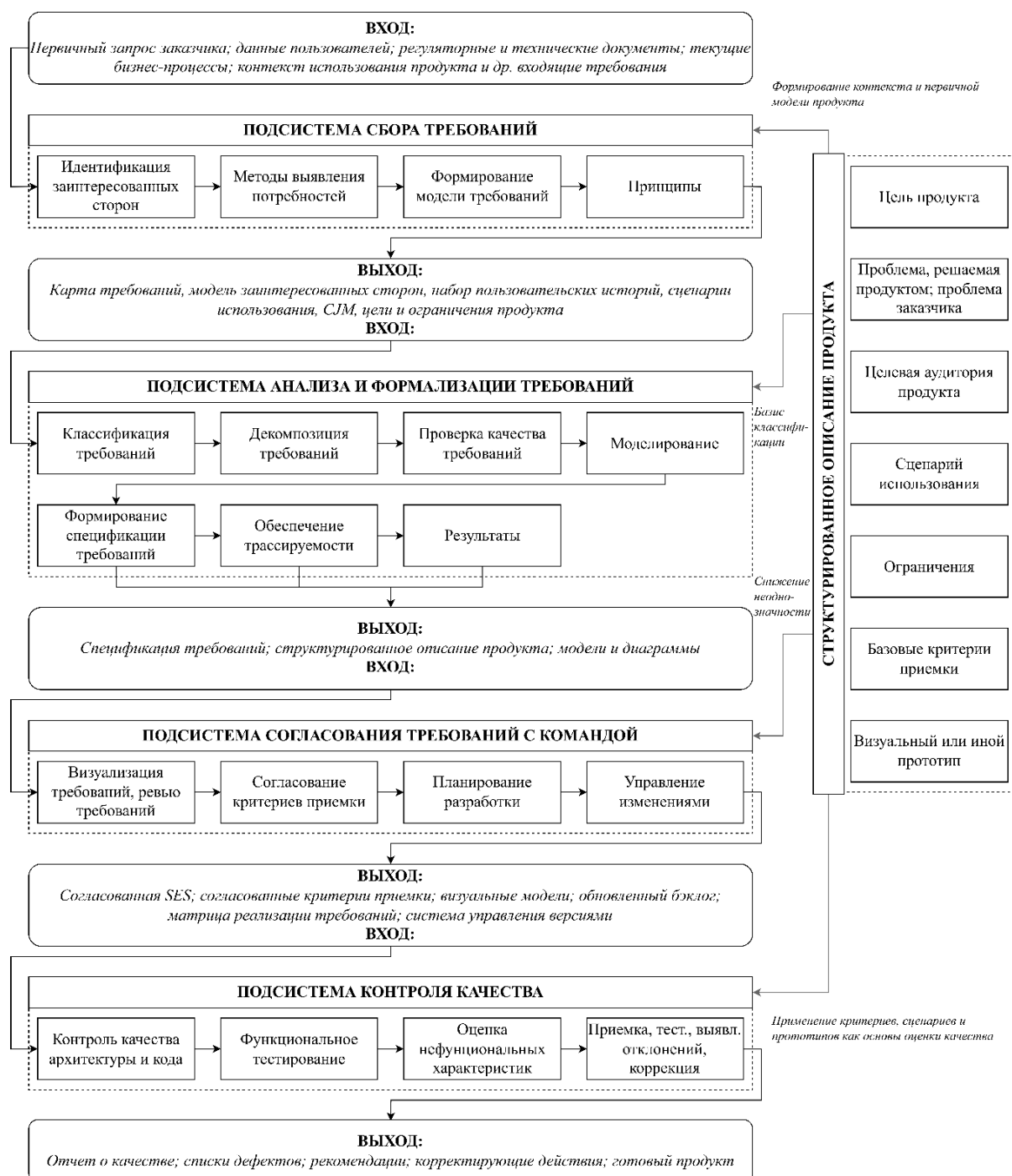


Рисунок 4 – Подсистемы, образующие механизм управления требованиями и качеством ИТ-продуктов

Figure 4 – Subsystems forming the mechanism for managing requirements and IT product quality

Обсуждение

Так, представленный механизм предполагает, что на каждом из этапов в ходе инженерии требований будут применяться соответствующие инструменты, каждый из которых формируется на базе этапов, входящих в подсистемы механизма. Так, например, на первом этапе формируется первичная модель продукта, выявляются заинтересованные стороны, фиксируются потребности, контекст использования и проблемы, что позволяет создать фундамент для дальнейшей их детализации; за каждым действием скрывается система характерных методов. Аналогичные требования предъявляются ко всем подсистемам механизма, направленным на обеспечение высокого качества IT-продукта. При этом механизм может реализовываться циклично, итеративно, а также в соответствии с Agile-методологиями управления, что предполагает повторную реализацию механизма после завершающей подсистемы.

Практическая значимость предложенного механизма проявляется в том, что он задает воспроизводимые основания для внедрения управления требованиями и качеством, фиксации минимально необходимого набора инструментов управления, ролей и метрик. В прикладном плане механизм может быть реализован как регламент сквозного прохождения требований, с переходом от первичного запроса до приемки; соответственно, каждая подсистема формирует измеримый результат (выход), который становится входом для следующей подсистемы, что снижает долю неформализованных решений и повышает управляемость качества (Таблица 2).

Таблица 2 – Практическая реализация механизма управления требованиями и качеством IT-продуктов

Table 2 – Practical implementation of the mechanism for managing requirements and IT product quality

Подсистема механизма	Практические процедуры	Фиксируемые результаты	Контрольная точка
Подсистема сбора требований	Идентификация заинтересованных сторон и их ожиданий. Сбор исходных требований через интервью, опросы, воркшопы и анализ существующих материалов. Фиксация проблемы, цели продукта и границ решения. Формирование первичных пользовательских сценариев и ценности продукта	Реестр заинтересованных сторон. Формулирование проблемы и цели продукта. Первичный реестр требований (эпики, пользовательские истории). Описание контекста использования и сценариев	Утверждение проблемы, цели продукта и границ решения. Согласование первичного реестра требований и состава стейкхолдеров
Подсистема анализа и формализации требований	Декомпозиция и уточнение требований. Классификация требований (бизнес, пользовательские, функциональные, нефункциональные, системные и архитектурные). Проверка требований на полноту, однозначность, проверяемость и непротиворечивость. Формирование критериев приемки и требований к тестированию	Спецификация требований (PRD, SRS или аналогичный документ). Реестр и формулировки критериев приемки. Матрица трассируемости требований. Реестр допущений, ограничений и зависимостей	Проведение ревью требований с фиксацией замечаний и решений. Утверждение спецификации требований и критериев приемки как основы для разработки и тестирования

Таблица 2 (продолжение)
Table 2 (continued)

Подсистема согласования требований с командой	Визуализация основных сценариев и интерфейсных решений. Проведение командного обсуждения требований и критериев приемки. Планирование реализации в рамках релиза или спринта с определением объема поставки. Формализация процедуры изменений, версионности требований и правил приоритизации	Прототип (макет) пользовательских сценариев. План релиза или план спринта с привязкой требований. Журнал изменений требований с версиями. Протокол командного согласования требований и критериев приемки	Подтверждение готовности требований к реализации. Согласование объема поставки и правил управления изменениями перед началом разработки
Подсистема контроля качества (или СМК)	Подготовка тестового покрытия на основе критериев приемки. Функциональное тестирование, проверка нефункциональных требований, регрессионное тестирование. Анализ дефектов с классификацией причин и оценкой их влияния. Подтверждение соответствия результата критериям приемки перед выпуском	Набор тестовых случаев и чек-листов по критериям приемки. Реестр дефектов с классификацией причин и статусами устранения. Отчет о качестве релиза (результаты тестирования, риски, ограничения). Материалы приемки и релиз-ноты	Проверка готовности к релизу по критериям приемки и результатам тестирования. Подтверждение отсутствия критических дефектов и согласование выпуска
Структурированное описание продукта	Формирование описания продукта для унификации понимания всеми участниками разработки. Фиксация целевой аудитории, сценариев использования, ограничений и критериев приемки в едином формате. Сопоставление описания продукта с требованиями и планом тестирования для исключения расхождений	Паспорт продукта (цель, проблема, целевая аудитория). Описание ключевых сценариев использования. Реестр ограничений и допущений. Свод критериев приемки и согласованный прототип	Проверка согласованности паспорта продукта, требований, критериев приемки и прототипа. Подтверждение единого понимания продукта командой и заказчиком

Исходя из описанных в Таблице 2 положений, рассмотрим практическую реализацию механизма управления требованиями и качеством ИТ-продуктов более детально. Во-первых, механизм переводится в практический инструмент (в целях его применимости и практической значимости) через стандартизированный комплект инструментов и процедур. Для подсистемы сбора требований обязательными выходами выступают:

- 1) реестр стейкхолдеров и их целей;
- 2) формулировка проблемы и цели продукта;
- 3) первичный backlog (эпики и истории);

4) карта пользовательских сценариев (user journey).

Для подсистемы анализа и формализации выделяются (в качестве продолжения предыдущих выходов):

5) спецификация требований с классификацией (бизнес-требования, пользовательские, функциональные, нефункциональные, архитектурные);

6) матрица трассируемости требований (требование, связанный с ним тест, релиз);

7) критерии приемки.

Для подсистемы согласования выделяются:

8) прототип и/или макет ключевых пользовательских потоков;

9) протокол ревью требований и решение о включении в релиз (change decision).

Для подсистемы контроля качества выделяются:

10) план тестирования и набор проверок по критериям приемки;

11) отчет о дефектах с классификацией причин (в том числе с использованием категории «дефект требования»);

12) итоговый акт приемки и релиз-ноты.

Во-вторых, практическая применимость обеспечивается назначением ответственности и точек контроля. В типовой команде ответственность может быть распределена следующим образом: владелец продукта отвечает за формулировку целей и бизнес-приоритетов; аналитик за формализацию и проверку требований; архитектор за архитектурные ограничения и нефункциональные требования; разработчик за реализацию; тестировщик или QA за верификацию критериев приемки; руководитель разработки за утверждение готовности и управление изменениями. Наличие фиксированных контрольных точек (ревью требований, ревью критериев приемки, readiness-check перед разработкой, quality-gate перед релизом) сводит к минимуму риски скрытой неоднозначности требований и снижает вероятность появления дефектов на более поздних стадиях.

В-третьих, механизм допускает проверку практического эффекта за счет метрик, которые прямо связываются с требованиями и качеством. К числу минимальных метрик относятся доля требований с критериями приемки; доля требований, прошедших ревью (requirements review coverage); количество изменений требований после начала разработки (change rate); доля дефектов, причиной которых является некорректность и неполнота требований (requirements-origin defects); процент успешной приемки без доработок; а также показатели качества эксплуатации (например, частота инцидентов, связанных с регрессией сценариев). Использование данных метрик делает результат, как минимум, измеримым в контексте внедрения механизма и позволяет интерпретировать отклонения как разрывы в конкретной подсистеме (сбор, формализация, согласование или контроль качества).

Заключение

Итак, разработка и внедрение механизма управления требованиями и качеством ИТ-продуктов становятся важными направлениями повышения эффективности процессов их разработки. Механизм стоит считать универсальным и инвариантно применимым для обширного спектра ИТ-продуктов. Основными принципами механизма являются ориентация на раннее выявление требований, минимизацию их размытости, несогласованности и отсутствия формализованных подсистем управления. Предложенный механизм направлен на устранение ряда основополагающих проблем отрасли:

- неполноты и неоднозначности требований;
- несогласованности между бизнесом и разработчиками;

- отсутствия критериев приемки и визуального представления продукта;
- слабой коммуникации между участниками разработки;
- недостаточного внимания к нефункциональным требованиям;
- разрыва между требованиями, архитектурой, тестированием и итоговым качеством.

Структура механизма направлена на последовательную трансформацию потребностей заказчика в проверяемые, согласованные и реализуемые требования, а затем в их верификацию в ходе контроля качества. Важным в механизме выступает вспомогательный инструмент – структурированное описание продукта, которое усиливает качество требований и формирует основу для тестирования, архитектурных решений и оценки качества.

Научная новизна предложенного механизма заключается в интеграции процессов управления требованиями и качеством в единую модель, в которой каждая подсистема не только выполняет свои функции, но и формирует входы для следующей, тем самым создается замкнутый цикл управления качеством.

В части практической значимости, предполагается, что внедрение механизма позволит достичь следующих эффектов: повышение предсказуемости разработки, снижение числа дефектов, вызванных ошибками требований, сокращение трудоемкости переработок, повышение согласованности решений, улучшение пользовательского опыта, а также повышение объективности оценки качества продукта.

В дальнейшем перспективными видятся задачи разработки метрик и обоснования методов, интегрированных в предложенный механизм; также немаловажной является задача совершенствования механизма в соответствии с потребностями команд разработчиков, что станет возможным после его экспериментальной оценки.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Chen T., Cheng H.K., Jin Y., Li Sh., Qiu L. Impact of Competition on Innovations of IT Industry: An Empirical Investigation. *Journal of Management Information Systems*. 2021;38(3):647–666. <https://doi.org/10.1080/07421222.2021.1962590>
2. Adilfi N., Yuldinawati L. The Influence of Innovation and Product Quality on the Competitive Advantage of Compass Products. *Eduvest – Journal of Universal Studies*. 2024;4(9):7983–7995. <https://doi.org/10.59188/eduvest.v4i9.17486>
3. Жуйкова А.А. Инженерия требований. *Экономика и социум*. 2018;(6):1429–1432. Zhuykova A.A. Technical requirements. *Ekonomika i sotsium*. 2018;(6):1429–1432. (In Russ.).
4. Зубкова С.В., Ястребов Р.А. Управление требованиями в ИТ-проектах: проблемы и способы решения. *Финансовые рынки и банки*. 2020;(3):3–7. Zubkova S.V., Yastrebov R.A. Requirements Management in IT Projects: Problems and Solutions. *Financial Markets and Banks*. 2020;(3):3–7. (In Russ.).
5. Батоврин В.К. Инженерия требований на современном промышленном предприятии. *Программная инженерия*. 2019;10(3):114–124. <https://doi.org/10.17587/prin.10.114-124> Batovrin V.K. Requirements Engineering at a Modern Industrial Enterprise. *Software Engineering*. 2019;10(3):114–124. (In Russ.). <https://doi.org/10.17587/prin.10.114-124>
6. Frattini J., Montgomery L., Fischbach J., Mendez D., Fucci D., Unterkalmsteiner M. Requirements quality research: a harmonized theory, evaluation, and roadmap. *Requirements Engineering*. 2023;28(4):507–520. <https://doi.org/10.1007/s00766-023-00405-y>

7. Khalid S., Rasheed U., Khaleeq uz Zaman U., et al. Ensuring quality in software requirement engineering process: A comparative study. *Egyptian Informatics Journal*. 2025;31. <https://doi.org/10.1016/j.eij.2025.100754>
8. Gobov D., Zuieva O. Software Quality Attributes in Requirements Engineering. *International Journal of Information Technology and Computer Science*. 2025;17(4):38–48. <https://doi.org/10.5815/ijitcs.2025.04.04>
9. Rosado da Cruz A.M., Cruz E.F. Machine Learning Techniques for Requirements Engineering: A Comprehensive Literature Review. *Software*. 2025;4(3). <https://doi.org/10.3390/software4030014>
10. Bajraktari A., Binder M., Vogelsang A. Requirements Engineering for Research Software: A Vision. In: *2024 IEEE 32nd International Requirements Engineering Conference (RE)*, 24–28 June 2024, Reykjavik, Iceland. IEEE; 2024. P. 423–431. <https://doi.org/10.1109/RE59067.2024.00050>

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Котова Милана Римовна, заместитель вице-президента, старший ИТ-бизнес-аналитик, Инвестиционный банк Барклайс, Прага, Чешская Республика.

e-mail: milanark2015@gmail.com

ORCID: [0009-0005-9135-4916](https://orcid.org/0009-0005-9135-4916)

Milana R. Kotova, Assistant Vice President, Senior IT Business Analyst, Barclays Investment Bank, Prague, Czech Republic.

Большев Вадим Евгеньевич, кандидат технических наук, Лаборатория технологий ИИ в психологии, Институт психологии Российской академии наук (ИП РАН), Москва, Российская Федерация.

e-mail: vadim57ru@gmail.com

ORCID: [0000-0002-5787-8581](https://orcid.org/0000-0002-5787-8581)

Vadim E. Bolshev, Candidate of Engineering Sciences, Laboratory of AI Technologies in Psychology, Institute of Psychology Russian Academy of Sciences, Moscow, the Russian Federation.

Статья поступила в редакцию 25.12.2025; одобрена после рецензирования 09.02.2026; принята к публикации 16.02.2026.

The article was submitted 25.12.2025; approved after reviewing 09.02.2026; accepted for publication 16.02.2026.