DOI 10.37539/2949-1991.2026.45.10.022
УДК 004.05

Корольков Александр Дмитриевич, аспирант 2 курса,
АНО ВО «Российский новый университет»
Korolkov Aleksandr Dmitrievich,
Russian New University (RosNOU)

Хакимова Аида Хатифовна,
ведущий научный сотрудник,
кандидат биологических наук, доцент,
АНО ВО «Российский новый университет»
Khakimova Aida Khatifovna,
Russian New University (RosNOU)

РАСПРЕДЕЛЕНИЕ ОТВЕТСТВЕННОСТИ И ПОЛНОМОЧИЙ ПРИ ПРИНЯТИИ РЕШЕНИЙ О ТЕХНИЧЕСКОМ КАЧЕСТВЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ALLOCATION OF RESPONSIBILITIES AND AUTHORITY IN DECISION-MAKING ON SOFTWARE TECHNICAL QUALITY

Аннотация. На основе нарративного обзора сопоставлены практики принятия решений о техническом качестве программного обеспечения. Выделены четыре связи: передача сведений, согласование действия, передача решения исполнителю и возврат результатов проверки. Предложен порядок анализа ответственности и полномочий участников внутри программного проекта.

Abstract. A narrative review compares decision-making practices concerning software technical quality. Four links are identified: information sharing, agreement on an action, assignment of the agreed action for implementation, and feedback on verification results. A procedure is proposed for analysing participants responsibilities and authority within a software project.

Ключевые слова: Техническое качество программного обеспечения; проектная команда; ответственность; полномочия; принятие решений; проверка кода; обратная связь.

Keywords: Software technical quality; project team; responsibility; authority; decision-making; feedback.

Управление техническим качеством программного обеспечения включает работу над продуктом и принятие решений об условиях этой работы. A. Alami и O. Krancher связывают внешнее качество с соответствием потребностям и отсутствием дефектов. Внутреннее качество они определяют через свойства кода и конструкции, от которых зависят поддержка и изменение продукта [1].

Вопрос о распределении ответственности возникает, когда после технической оценки необходимо выбрать дальнейшее действие. W. Behutiye и соавторы выделяют неясную ответственность, ограничения времени и недостатки тестирования [2]. Обнаружение дефекта, согласование исправления и проверка результата требуют от участников разных обязательств и полномочий.

Цель статьи состоит в раскрытии связи между ответственностью, полномочиями и обратной связью при принятии решений о техническом качестве внутри команды и программного проекта. Исследовательский вопрос: как распределяются эти обязанности и права в практиках, описанных в исследованиях.

Для ответа на этот вопрос выполнен нарративный обзор. При тематическом отборе учитывалось организационное содержание работ: роли участников разработки, требования к качеству, проверка изменений и согласование действий по дефектам. Источники подбирались по русским и английским сочетаниям терминов: качество программного обеспечения, ответственность, командные роли и принятие решений. Из исследований продуктовых линий отбирались описания процедур, влияющих на решения в отдельном проекте.

Единицей сопоставления выбрана практика принятия решения о конкретном объекте: изменении кода, сценарии проверки, требовании, дефекте или этапе выпуска. Для каждой практики определялись участники и используемые ими сведения, способ выбора действия, исполнение и пересмотр решения.

При анализе ответственности учитываются обязательства по выполнению и проверке действий, а также представления участников об их ответственности за результат. Полномочие означает право выбрать или согласовать действие в отношении конкретного объекта. Учитываются должность участника и фактически выполняемая функция.

При обобщении материала отдельно рассматривались эмпирические описания процессов, предлагаемые организационные схемы и обзорные обобщения. Практики сопоставлялись по связям между действиями участников. Анализ носит качественный характер и описывает организацию решений в рассмотренных условиях, преимущественно в гибкой разработке. Количественное ранжирование практик по эффективности не проводилось.

При проверке кода участники совместно оценивают подготовленное изменение. A. Alami, V. V. Jensen и N. A. Ernst относят к индивидуальным основаниям ответственности личные стандарты, профессиональную добросовестность, гордость за работу и репутацию [3]. Автор кода обсуждает изменение с проверяющим. Принятие обратной связи исследователи интерпретируют как переход к разделяемой ответственности за качество [3].

Взаимодействие автора и проверяющего представлено на рисунке 1.



Рисунок 1. Совместная ответственность при проверке кода

A. Alami и O. Krancher указывают на связь добровольной проверки с прозрачностью работы. Обсуждение решений они соотносят с обменом знаниями, а сообщение о проблемах с психологической безопасностью [1].

Совместная ответственность проявляется и в тестировании. T. Natarajan и соавторы в двухлетнем исследовании непрерывного тестирования описывают, как участники вместе готовили сценарии приёмки при уточнении и планировании работы. Разработчики и инженеры тестирования участвовали в реализации исполняемых спецификаций и автоматизации. Тестировщики вносили специальный вклад в формулирование сценариев и выявление граничных случаев [4]. Совместная работа сохраняла различия в компетенциях и действиях участников.

В ходе исполнения участники получали сведения от сборок и тестов [4]. Поэтому обсуждать изменение можно было как на основании экспертного замечания, так и по результатам проверки ожидаемого поведения продукта.

С. В. Широкова и Т. А. Тарасова предлагают закрепить за инженером по качеству проверку требований, работу с метриками и регламентами, а также анализ сборки. Релиз-менеджеру отводятся планирование и контроль выпуска [5].

Распределение функций между участниками может меняться в зависимости от проблемы. R. Hoda, J. Noble и S. Marshall описывают неформальные роли в самоорганизующихся командах, возникающие в дополнение к организационным [6].

Проверка кода и тестирование относятся к оценке изменений. При включении требования в работу проекта сведения о качестве используются для выбора содержания продукта. T. Olsson, K. Wnuk и T. Gorschek, исследуя выбор требований к качеству в продуктовой линии, описывают полномочия менеджера программного продукта. Он решает, включать ли функцию в продукт и когда её выпускать. Решения принимаются непрерывно и при необходимости согласуются между менеджерами продуктов [7]. В обзоре рассматривается процедура выбора содержания работы отдельного проекта.

При выборе учитываются предложения участников и сведения о пользователях [7]. Однако получение таких сведений не всегда приводит к действию. В описанном авторами случае проблемы работы устройств от батареи было сложно выделить группу для решения вопроса [7].

В модели T. Olsson, K. Wnuk и S. Jansen менеджеры продукта задавали направление реализации, а уточнением занимались они сами или участники с другими ролями. При несоответствии пользовательским ожиданиям вопрос возвращался к разработке и при необходимости передавался выше по цепочке уточнения [8]. Эта процедура предусматривала пересмотр ранее выбранного содержания работы.

При разборе дефекта участники сопоставляют техническую оценку с ожидаемым поведением продукта. M. Begum и Y. Dittrich описывают этот процесс в команде сопровождения Microsoft Dynamics. Инженеры поддержки уточняют проблему, старшие инженеры готовят технические сведения, а менеджеры программы объясняют ожидаемое поведение. Затем участники совместно выбирают действие: исправить дефект, отказаться от исправления, продолжить исследование или признать поведение соответствующим замыслу [9].

Порядок выбора действия по дефекту и пересмотра решения представлен на рисунке 2.

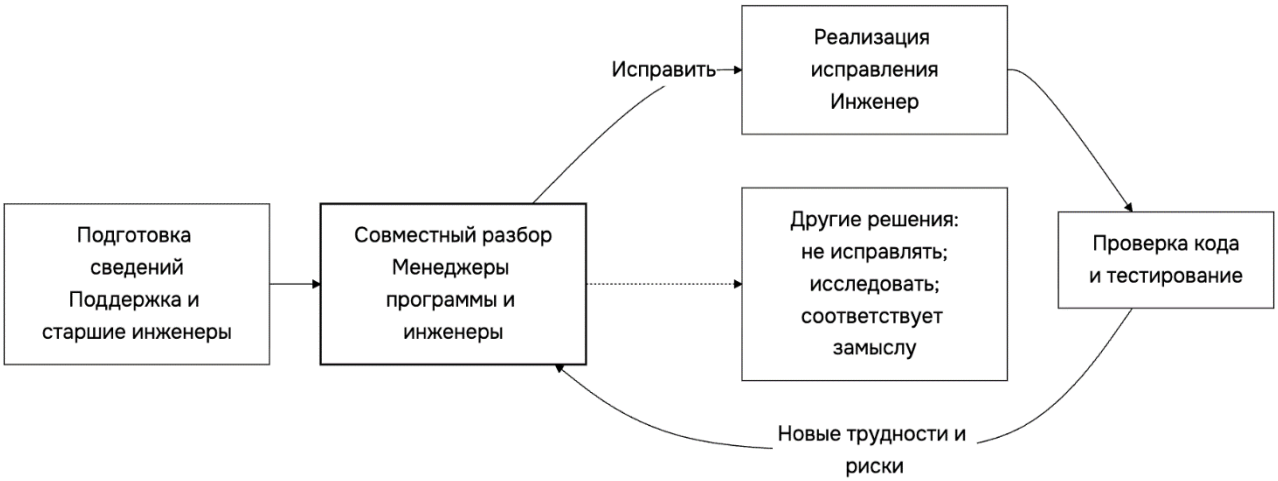


Рисунок 2. Решение по дефекту: исправление и альтернативные действия

В описанном M. Begum и Y. Dittrich случае сложного дефекта участники пересмотрели первоначальный вариант исправления, обнаружив угрозу повреждения данных и побочных изменений. Отсутствие тестов для затрагиваемой функциональности затрудняло безопасное

исправление. Инженеры и менеджер программы уточнили связанные сценарии и временный обходной вариант. Детали поведения потребовали дополнительного согласования, после которого участники вернулись к реализации и проверке [9].

Для обоснования выбора действия используется также оценка риска. С. А. Федоров сопоставляет подходы к оценке риска критичных дефектов [10]. Ю. Б. Филонова и Л. В. Макарова рассматривают угрозы срокам, согласование требований и изменений [11].

Различия предметов решений, участников и форм согласования обобщены в таблице 1.

Таблица 1

Организационные механизмы принятия решений о техническом качестве

Практика и основание описания	**Объект решения**	**Участники и форма выбора**	**Основание выбора**	**Проверка и пересмотр**
Проверка кода; фокус-группы	Изменение кода	Автор и проверяющий обсуждают изменение; проверяющий выражает одобрение	Представленный код, замечания, профессиональная оценка	Автор дорабатывает изменение по обратной связи [3]
Совместное тестирование; исследование практики Scrum	Сценарии и ожидаемое поведение	Разработчики и тестировщики совместно уточняют сценарии, сохраняя специализацию	Ожидаемое поведение и граничные случаи	Сборки и тесты возвращают сведения разработчикам [4]
Выбор требований; исследование продуктовой линии	Включение функции и время выпуска	Менеджер продукта выбирает содержание; при необходимости согласует его с другими менеджерами	Предложения участников, сведения о пользователях	Проблемы качества становятся основанием для пересмотра содержания [7]
Разбор дефекта; технический отчёт о команде сопровождения	Дальнейшее действие по дефекту	Менеджеры программы и инженеры совместно разбирают проблему и выбирают действие	Отчёт, техническое исследование, сценарии использования	Реализация, проверка кода, тестирование; новые риски возвращают вопрос на обсуждение [9]
Контроль выпуска; предлагаемая организационная схема	Прохождение этапов выпуска	Релиз-менеджер планирует и контролирует этапы; инженер по качеству проверяет требования и результаты	Требования, метрики, результаты сборки	Разработчик исправляет ошибки, результаты сборки анализируются [5]

По результатам сопоставления выделены четыре связи управления качеством. Первая состоит в передаче сведений тем участникам, которые готовят решение, как это происходит при уточнении отчёта о дефекте [9]. При анализе проекта устанавливаются адресат сообщения о проблеме и состав сведений.

Вторая связь заключается в согласовании действия. Одобрение кода, включение функции в продукт и выбор действия по дефекту имеют разные предметы решения [3, 7, 9]. Чтобы определить полномочия участника, необходимо указать, что именно он вправе выбрать и как согласуется этот выбор.

Третья связь предполагает передачу решения исполнителю. Согласованное исправление становится основанием для реализации и последующих проверок [9]. Указываются исполнитель и ожидаемый результат работы.

Четвёртая связь возвращает результаты проверки к выбору действия. Обнаружение нового риска может привести к пересмотру исправления [9]. Несоответствие пользовательским ожиданиям может потребовать уточнения требований [8]. Выделенные связи обобщены на рисунке 3.



Рисунок 3. Аналитическая схема связей управления качеством

Схема обобщает рассмотренные практики, однако влияние распределения ролей на результат требует отдельной эмпирической оценки.

С учётом этого ограничения схему можно применять для разбора работы над конкретным проектом. Сначала выбирается эпизод: изменение кода, включение требования или обработка дефекта. Для него описываются объект решения, исходные сведения, участники обсуждения и согласованное действие. Затем устанавливается, кто отвечает за исполнение и какой результат позволяет завершить работу либо вернуться к обсуждению.

При анализе рискованного исправления следует проследить, как участники уточняют поведение продукта, оценивают последствия и повторно согласуют действие. Такой порядок наблюдался в описанном случае сложного дефекта [9]. Фиксируются результат согласования, исполнитель и условия пересмотра. Если решение касается выпуска продукта или ресурсов проекта, соответствующие полномочия следует описать отдельно от права одобрить техническое изменение.

Применение схемы зависит от условий взаимодействия участников. A. Alami и O. Krancher описывают препятствия, вызванные напряжённостью общения и недоступностью знаний о бизнесе [1]. Поэтому при анализе обязанностей следует оценивать, доступны ли участникам необходимые знания и могут ли они обсуждать возникающие проблемы.

Общая ответственность команды сочетается с функциональной специализацией её участников. Полномочия каждого из них определяются предметом решения и процедурой согласования. Результат обзора включает сравнительную характеристику практик и схему четырёх связей, по которым можно проследить переход от сведений о качестве к действию и пересмотру решения. Схема может использоваться для уточнения обязанностей участников и процедур обратной связи внутри отдельного проекта.

Список литературы:

1. Alami A., Krancher O. How Scrum Adds Value to Achieving Software Quality?: авторская рукопись // IT University of Copenhagen: исследовательский репозиторий. – 2022. – URL: https://researcher.itu.dk/p/en/research-outputs/how-scrum-adds-value-to-achieving-software-quality.

2. Management of quality requirements in agile and rapid software development: A systematic mapping study / W. Behutiye, P. Karhapää, L. López [et al.]: препринт. – 2020. – arXiv:2002.02303v1. – DOI: https://doi.org/10.48550/arXiv.2002.02303. – URL: https://arxiv.org/abs/2002.02303v1.

3. Alami A., Jensen V. V., Ernst N. A. Accountability in Code Review: The Role of Intrinsic Drivers and the Impact of LLMs: препринт. – 2025. – arXiv:2502.15963v1. – DOI: https://doi.org/10.48550/arXiv.2502.15963. – URL: https://arxiv.org/abs/2502.15963v1.

4. The continuous testing practice: A two-year action research study to streamline software quality in scrum teams / T. Natarajan, S. Pichai, K. Perumal, Anitha TR // Journal of Systems and Software. – 2026. – Vol. 236. – Art. 112829. – DOI: https://doi.org/10.1016/j.jss.2026.112829. – Репозиторий использованной копии: https://hdl.handle.net/10344/31650.

5. Широкова С. В., Тарасова Т. А. Формирование и адаптация структуры и функциональности команды проекта при внедрении методологии CI/CD // Фундаментальные и прикладные исследования в области управления, экономики и торговли: сборник трудов Всероссийской научно-практической и учебно-методической конференции: в 8 ч. – Санкт-Петербург: Политех-Пресс, 2023. – Ч. 3. – С. 195–205.

6. Hoda R., Noble J., Marshall S. Self-Organizing Roles on Agile Software Development Teams // IEEE Transactions on Software Engineering. – 2013. – Vol. 39, no. 3. – P. 422–444. – DOI: https://doi.org/10.1109/TSE.2012.30.

7. Olsson T., Wnuk K., Gorschek T. An Empirical Study on Decision making for Quality Requirements: препринт. – 2018. – arXiv:1812.04884v1. – DOI: https://doi.org/10.48550/arXiv.1812.04884. – URL: https://arxiv.org/abs/1812.04884v1.

8. Olsson T., Wnuk K., Jansen S. A validated model for the scoping process of quality requirements: a multi-case study // Empirical Software Engineering. – 2021. – Vol. 26. – Art. 26. – DOI: https://doi.org/10.1007/s10664-020-09896-7.

9. Begum M., Dittrich Y. What happens in Triage? An Empirical Study of Bug Triage in Software Product Evolution: technical report. – Copenhagen: IT University of Copenhagen, 2024. – (IT University Technical Report Series; TR-2024-210). – URL: https://en.itu.dk/Research/Research-resources/Technical-Reports/Technical-Reports-Archive/2024/TR-2024-210.

10. Федоров С. А. Сравнение подходов для оценки риска возникновения критичных дефектов в релизах программного обеспечения // Информационные технологии в моделировании и управлении: подходы, методы, решения: сборник материалов VIII Всероссийской научной конференции с международным участием. – Тольятти: Тольяттинский государственный университет, 2025. – С. 141–148.

11. Филонова Ю. Б., Макарова Л. В. Анализ рисков при разработке программных продуктов // Дневник науки. – 2024. – № 1 (85). – URL: https://elibrary.ru/item.asp?id=63436511.