

DOI: 10.37791/2687-0657-2024-18-5-111-123

# Компаративный анализ гибких методологий разработки программного обеспечения в условиях современных высококонкурентных рынков

А.Ю. Анисимов<sup>1\*</sup>, Е.Н. Токмакова<sup>1</sup>, А.Е. Трубин<sup>1</sup>

<sup>1</sup>Университет «Синергия», Москва, Россия

\**anisimov\_au@mail.ru*

**Аннотация.** Проблема выбора эффективной методологии управления проектами является одной из ключевых задач в области разработки информационных систем. В современном мире разработка программного обеспечения представляет собой один из наиболее востребованных видов проектной деятельности, и практически в каждой сфере производства используется специализированное программное обеспечение, которое играет центральную роль в функционировании предприятий. В связи с этим создание эффективного процесса разработки становится важнейшим условием успешной реализации проекта. Несмотря на существование большого количества современных методологий управления, выбор наиболее подходящей остается сложной задачей, требующей детального анализа целей и критериев. Успешность проекта как для исполнителя, так и для заказчика во многом зависит от корректного выбора модели управления. Целью исследования является анализ ключевых аспектов методологий управления разработкой программного обеспечения в условиях высококонкурентных современных рынков. Основная задача исследования включает сравнительный анализ гибких методологий управления разработкой ПО. В статье представлены результаты сравнительного анализа существующих гибких методологий. Обоснована целесообразность применения методологии Kanban в качестве инструмента управления разработкой ПО, а также выявлены потенциальные риски, связанные с выбором оптимальной методологии. Полученные результаты имеют как теоретическое, так и практическое значение, содействуя обоснованному выбору наиболее подходящей методологии управления проектами в сфере разработки ИС в условиях современных высококонкурентных рынков.

**Ключевые слова:** управление проектами, управление разработкой проектов ПО, гибкие методологии, традиционные методологии, риски, Kanban, Agile

**Для цитирования:** Анисимов А.Ю., Токмакова Е.Н., Трубин А.Е. Компаративный анализ гибких методологий разработки программного обеспечения в условиях современных высококонкурентных рынков // Современная конкуренция. 2024. Т. 18. № 5. С. 111–123. DOI: 10.37791/2687-0657-2024-18-5-111-123

# Comparative Analysis of Flexible Software Development Methodologies in Modern Highly Competitive Markets

A. Anisimov<sup>1\*</sup>, E. Tokmakova<sup>1</sup>, A. Trubin<sup>1</sup>

<sup>1</sup>*Synergy University, Moscow, Russia*

*\*anisimov\_au@mail.ru*

**Abstract.** The problem of choosing an effective project management methodology is one of the key tasks in the field of information systems development. In the modern world, software development is one of the most demanded types of project activities, and in almost every field of production specialized software (software) is used, which plays a central role in the functioning of enterprises. In this regard, the creation of an effective development process becomes the most important condition for the successful implementation of the project. Despite the existence of a large number of modern management methodologies, choosing the most appropriate one remains a difficult task that requires a detailed analysis of goals and criteria. The success of the project, both for the contractor and for the customer, largely depends on the correct choice of the management model. The purpose of the study is to analyze key aspects of software development management methodologies in highly competitive modern markets. The main objective of the study includes a comparative analysis of flexible software development management methodologies. The article presents the results of a comparative analysis of existing flexible methodologies. The expediency of using the Kanban methodology as a software development management tool is substantiated, and potential risks associated with choosing the optimal methodology are identified. The results obtained have both theoretical and practical significance, contributing to the informed choice of the most appropriate project management methodology in the field of IP development in modern highly competitive markets.

**Keywords:** project management, software project development management, flexible methodologies, traditional methodologies, risks, Kanban, Agile

**For citation:** Anisimov A., Tokmakova E., Trubin A. Comparative Analysis of Flexible Software Development Methodologies in Modern Highly Competitive Markets. *Sovremennaya konkurentsia*=Journal of Modern Competition, 2024, vol.18, no.5, pp.111-123 (in Russian). DOI: 10.37791/2687-0657-2024-18-5-111-123

## Введение

Растущий спрос на цифровые решения со стороны бизнеса, государства и общества стимулирует инновации в ИТ-сфере, способствуя ускоренному росту данного сектора. Разработка высокотехнологичных продуктов стала одним из приоритетов как внутри страны, так и во всем мире. Высокая потребность в ИТ-продуктах заставляет их производителей искать новые решения и подходы, способствующие сокращению времени

на разработку без негативного влияния на качество конечного продукта. Изменение процессов разработки ПО затрагивает как его техническую и бизнес-сторону, так и систему управления. Новшества в одной из этих составляющих оказывает воздействие и на другие две области разработки.

Любой проект направлен на получение прибыли от производимого продукта, а заказчик, в свою очередь, заинтересован в качественном продукте, пользующемся спросом у потребителя. Именно баланси-

рование интересов исполнителя и заказчика в рамках реализации любого проекта, в том числе ИТ-проекта, является ключевым при решении о переходе от традиционных методологий в управлении проектами к гибким.

Традиционные методологии разработки программного обеспечения, такие как Waterfall, основываются на последовательном выполнении фаз проекта, начиная от сбора требований и заканчивая внедрением и сопровождением системы. Данный подход характеризуется высоким уровнем предсказуемости и контроля, что делает его привлекательным для проектов с четко определенными требованиями и временными рамками.

Каждый этап в каскадной модели является автономным. Для перехода на следующий этап в разработке необходимо, чтобы предыдущие этапы были полностью закончены. Вносить корректировки в середине процесса в каскадной модели не представляется возможным. Именно поэтому основным требованием к применению каскадной модели является представление процесса разработки перед началом самого процесса [14].

Каскадная модель разработки является легко адаптируемой для смены команд. Waterfall-модель не подразумевает жестких временных ограничений на разработку, каждый этап прорабатывается детально. В условиях такой работы у любого участника команды, независимо от уровня его компетенций, есть возможность избежать проблем с пониманием целей проекта при знакомстве с ним и продуктом, который является результатом работы в проекте.

Нехватка наиболее подходящих моделей и инструментов является основной проблемой гибкого управления процессом разработки ИС. В большинстве случаев не только глобальные изменения в ИТ-сегменте влияют на эффективность модели управления, но и корректировки, происходящие внутри проекта (изменение

требований, сроков поставки продукта, финансирования и т. п.).

В современных компаниях, создающих ИС, чаще всего для управления разработкой проекта используются Agile. Руководители не всегда выбирают одну методологию, а чаще всего используют смешанный или гибридный подход. Такой выбор чаще всего обусловлен тем, что одна методология не может полностью покрыть все проектные аспекты и требования [13]. Комбинирование двух или даже трех методик позволяет создать эффективную модель контроля процесса разработки ИС его управления.

Гибридная методология представляет собой интеграцию элементов различных подходов к разработке программного обеспечения, что позволяет адаптировать процессы под конкретные условия проекта и потребности клиентов.

Преимущества гибридной методологии, такие как улучшенное управление рисками, повышенная вовлеченность команды, делают ее привлекательной для различных типов проектов и организаций. Однако ее успешное применение требует тщательной интеграции, четкого определения процессов и ролей, а также наличия высококвалифицированных специалистов, что позволяет минимизировать риски и обеспечить достижение высоких результатов в разработке программного обеспечения [10].

Основу гибридной модели управления разработкой ИТ-проекта должен составлять закон синергии, так как только на данной основе будет производиться совместный эффект, который будет больше, чем разрозненное функционирование каждой из моделей управления по отдельности. Проекты, использующие гибридное управление, берут адаптивность и гибкость Agile-методик и комбинируют их с традиционными методами управления разработки проектов [7].

Гибкие методологии позволяют адаптироваться к изменениям и непрерывно улуч-

шать процесс разработки, что приводит к созданию высококачественного продукта, удовлетворяющего потребности пользователей. Их преимущества при разработке программного обеспечения, такие как гибкость и адаптивность, улучшенное взаимодействие с клиентом, постоянное улучшение процессов, повышенная мотивация и вовлеченность команды, а также сокращение времени выхода на рынок, делают их незаменимыми в условиях современных динамичных и высококонкурентных рынков.

Agile-методологии позволяют также наладить процессы в проектах, разработка в которых изначально была построена на основе каскадной модели.

Организация процесса на основе философии Agile также положительно влияет на мотивацию участников команды. Распределение задач и их ежедневное обсуждение делает процесс прозрачным. Каждый член команды ежедневно видит результаты достижений. Кроме того, задачи более понятны участникам, а значит, увеличивается уровень ответственности в работе. Достичь такого эффекта довольно проблематично, следуя традиционным методологиям. Команда меньше общается, нет культуры ежедневных митингов и обсуждений задач, результат проделанной работы чаще всего виден только в конце этапа.

### **Сравнительный анализ гибких методологий**

Гибкие методологии предлагают иной подход к разработке программного обеспечения, основанный на итеративной разработке, что позволяет командам быстро адаптироваться к изменениям требований и максимально использовать возможности для улучшения продукта.

Несмотря на большое разнообразие Agile, представленных на сегодняшний день, большинство проектов, занимающихся разработкой ПО, не могут применять

их в чистом виде [4]. Такая ситуация чаще всего складывается вследствие нескольких факторов:

1. Изначально проект был построен с использованием стандартных методологий управления процессом разработки.

2. Ландшафт проекта включает в себя множество систем.

3. Большинство команд работают с использованием традиционных методологий управления разработкой ИС.

4. Команды не всегда являются full-stack. Существуют команды, занимающиеся только тестированием.

Выбор подходящей методологии зависит от множества факторов, включая тип проекта, размер команды, организационную культуру и требования клиентов, что требует тщательного анализа, сравнения и адаптации для достижения наилучших результатов в разработке программного обеспечения.

Сравнение Agile-методологий было произведено, основываясь на следующих критериях (табл. 1):

1. Развитие – это критерий, определяющий, насколько хорошо проходит процесс разработки ПО на том или ином этапе.

2. Первоначальный план – определяет последовательность действий, которые необходимо произвести для исполнения бизнес-процессов.

3. Документация – характеристика, определяющая принцип ведения документации в проекте.

4. Тестирование – критерий, описывающий подход к проверке функционирования ПО.

5. Команда – критерий, определяющий участников команды и их роли и обязанности.

6. Инспекция кода – критерий, определяющий подход к проверке исходного кода программы с целью избегания ошибок, допущенных в процессе разработки, и их управления до представления продукта заказчику.

**Таблица 1.** Сравнительный анализ гибких методологий  
 Table 1. Benchmarking Agile methodologies

Методология <i>Methodology</i>	Scrum	Kanban	Extreme Programming (XP)	Feature Driven Development (FDD)	Model Driven Development (MDD)	Domain Driven Development (DDD)
Критерий <i>Criterion</i>						
Развитие	Каждый этап разработки (итерация) включает в себя написание кода, инспекцию кода, тестирование	Не включает в себя инструменты для регулирования процесса разработки ПО. Дает возможность отслеживать процесс разработки	Автотестирование кода (юнит и функциональное). Непрерывная интеграция. Ревью кода. Постоянно обновляющийся план работы	Декомпозиция функций на подобласти. Основное внимание уделяется дизайну и написанию кода. Всегда есть продукт, который можно показать заказчику	Не включает в себя инструменты для регулирования процесса разработки ПО. Предлагает инструменты для написания кода (концепция модельно-ориентированного подхода)	Не включает в себя инструменты для регулирования процесса разработки ПО
Первоначальный план	Точное планирование каждой итерации	Не включает в себя инструменты для планирования	Приблизительное планирование. Постоянное обновление плана по мере конкретизации требований	Не включает в себя инструменты для планирования	Не включает в себя инструменты для планирования	Не включает в себя инструменты для планирования
Документация	Нестрогое ведение документации	Не включает в себя инструменты для регулирования ведения документации	Нестрогое ведение документации	Нестрогое ведение документации. Основной источник документации – написанный код	Нестрогое ведение документации. Основной источник документации – UML-диаграммы, модели	Нестрогое ведение документации. Основной источник документации – схемы с описанием верхнеуровневой архитектуры
Тестирование	В конце каждой итерации	Не включает в себя инструменты для регулирования процесса тестирования	Автотестирование	Не включает в себя инструменты для регулирования процесса тестирования	Не включает в себя инструменты для регулирования процесса тестирования	Не включает в себя инструменты для регулирования процесса тестирования
Инспекция кода	Опционально	Не включает в себя инструменты для регулирования процесса инспекции кода	Обязательный этап	Не включает в себя инструменты для регулирования процесса инспекции кода	Не включает в себя инструменты для регулирования процесса инспекции кода	Не включает в себя инструменты для регулирования процесса инспекции кода

Источник: составлено авторами по материалам [2, 5–8, 11].

Результаты сравнительного анализа Agile, представленные в таблице 1, показали, что Scrum является универсальной методологией, отвечающей всем заявленным критериям. Узконаправленными методологиями можно назвать Model Driven Development и Feature Driven Development, так как эти методологии соответствуют только одному из шести заявленных критериев сравнения и больше ориентированы на отлаживание процессов, в которые вовлечены только разработчики, но не целая команда.

Все рассмотренные методологии в той или иной степени предлагают инструменты для регулирования процесса разработки ПО. Критерию «Развитие» не соответствует только модель Domain Driven Development. Сквозное регулирование процесса предлагает Scrum. Инструменты данной методологии позволяют отслеживать и отлаживать процессы на всех этапах – планирования, разработки, тестирования, а также технической поддержки. В Scrum есть четко регулированные сроки поставки ИТ-продукта, кроме того, обозначены роли для каждого участника команды, что позволяет разделить ответственность и функции в команде. У всех остальных рассмотренных моделей нет такого обширного инструментария. Так, например, в Kanban нет обязательных ролей и регулярных спринтов. Весь процесс работы представлен как непрерывное производство. Основная задача Kanban – это визуализация процесса и ограничение количества незавершенной работы (WIP) [1].

Extreme Programming, в свою очередь, подразумевает практику тесного взаимодействия с заказчиком. Представитель заказчика является частью команды и в зону его ответственности входит написание пользовательских историй (User Story), а также решение по срокам и порядку реализации этих историй. Разработка в XP осуществляется итерациями длительностью 2–3 недели [1]. Экстремальность данной

методологии заключается в поверхностном подходе к решению задач. В качестве решения принимается первое, самое простое решение, которое будет расширяться в последующих итерациях [11].

Feature Driven Development, Model Driven Development и Domain Driven Development, согласно результатам сравнительного анализа, предлагают наименьшее количество инструментов для управления процессом разработки. Эти три методологии больше ориентированы на бизнес-процессы, чем на процессы взаимодействия внутри команды. FDD, MDD и DDD не предлагают разделение на роли, в них нет ограничения по срокам выпуска функциональности. Они не способны удовлетворить потребность в регулировании процесса разработки, но они могут быть использованы как дополнительный инструмент или фреймворк в совокупности с методологиями Scrum или XP.

Данные, представленные далее в таблице 2, позволяют заключить, что только две методологии соответствуют критерию «Первоначальный план». Четко структурированный и оформленный план проекта предлагает методология Scrum. XP, в свою очередь, также предлагает план процесса разработки, но только в верхнеуровневом виде. Ввиду того, что XP больше ориентирован на проекты, в которых изначально нет хорошо сформулированных требований, эта методология построена таким образом, что способствует адаптации процесса разработки к постоянно меняющемуся плану и требованиям.

Критерий «Документация» можно назвать опциональным в каждой из представленных методологий. В Scrum и XP есть оформленные требования в виде пользовательских историй, но нет подробно описанных тест-кейсов. В XP источник тест-кейсов – это автоматизированные тесты, то есть код, написанный для проверки функциональности кода самого приложения, а в Scrum тест-кейсы чаще всего оформ-

лены в чек-листы. FDD в качестве инструмента для документации предлагает написанный код. Выбор такой модели разработки ПО накладывает ответственность на разработчиков, чей код должен быть понятным. В MDD и DDD вся основная документация оформлена в UML-диаграммы и модели.

Последними критериями, которые были взяты в качестве критериев сравнительного анализа, были «Тестирование» и «Инспекция кода». Целью обоих критериев является проверка функциональности приложения с целью предотвращения возникновения ошибок. Отличием является то, что инспекция кода – это тоже тестирование, только статическое, то есть без запуска кода. В Scrum, например, не выделяют инспекцию кода как отдельный этап в процессе разработки. Несмотря на это, многие команды проводят инспекцию каждый раз перед выдачей кода заказчику.

В отличие от Scrum, Extreme Programming выделяет инспекцию кода как отдельный этап в разработке. Учитывая специфику направленности этой методологии, можно сказать, что для XP этот этап обязательный, так как методология в целом ориентирована на регулирование процессов, связанных с написанием кода.

В FDD, DDD, MDD и Kanban нет четких инструкций по проведению тестирования и выделению его в качестве отдельного этапа в процессе разработки. Тем не менее это не означает, что данные методологии полностью исключают этап тестирования. Отсутствие тестирования как этапа разработки ПО, вероятно, обусловлено тем, что эти методологии применимы как дополнительный инструмент, но не способны в одиночку удовлетворить потребности в регулировании процесса разработки.

Проведя сравнительный анализ Agile в разработке ПО, можно сделать ряд выводов касательно их применения в качестве самостоятельного инструмента для регулирования процесса.

Как показал анализ, наиболее универсальной методологией является Scrum, так как она покрывает все потребности проекта, а именно четкое распределение ролей и обязанностей, сроки, которые команда выделяет на процесс разработки, планирование и написание документации, анализ проделанной работы и работу над ошибками. Построение процесса разработки с применением методологии Scrum является более эффективным в случае, если команда высокомотивирована и дисциплинирована, а также имеет опыт работы в сфере разработки программного обеспечения.

Больше всего инструментов после методологии Scrum предлагает методология Extreme Programming. В ней нет разделения на роли, но осуществляется планирование, инспекция кода, тестирование, что больше подходит проектам, в которых требования не ясны или быстро меняются. В общем и целом процесс разработки, согласно методологии XP, является более неформальным, но одновременно требует от команды высокого уровня самодисциплины. Данная методология не предполагает написания большого количества отчетов или построения множества моделей.

Kanban-методология больше ориентирована на ограничение количества незавершенной работы и достижение наибольшей эффективности в проекте. Основным инструментом, который предлагает Kanban, является визуализация процесса. Использование Kanban-доски позволяет следить за процессом развития проекта, контролировать количество невыполненных или начатых задач. Тем не менее Kanban как методология управления разработкой проекта обретает наибольшую эффективность в совокупности с другими методологиями, например Scrum. Иными словами, Kanban хорошо использовать как дополнительный набор инструментов для управления процессом. Кроме того,

Kanban удобен для тех команд, которые ограничены в человеческих ресурсах, так как эта методология не подразумевает четкого распределения ролей.

Model Driven Development, Feature Driven Development и Domain Driven Development, так же как и Kanban, довольно проблематично использовать в чистом виде, так как эти методологии ориентированы на бизнес-процессы. Такие методологии больше подходят для многоуровневых проектов, в которых очень большое количество реализаций. Одновременно с этим они допускают риск возникновения неудачного результата, причиной которого может являться непонимание концепции проекта и его бизнес-целей. Кроме того, использование этих методологий усложняется тем, что в них нет разделения на роли.

### **Применение Kanban-методологии в качестве инструмента управления разработкой ИС**

Разработка современного ПО – это в первую очередь кросс-функциональная и командная деятельность, для осуществления которой необходим правильно подобранный инструмент управления. Ключевую роль играет система управления и контроля, так как техническая модернизация сама по себе не имеет столь высокого положительного влияния, если она происходит обособлено. Задача управления в данном случае – это оптимизация процессов путем внедрения технологических новшеств. Правильно построенная модель управления процессом разработки ПО является первым основным шагом к успеху команды.

Помимо того, что Kanban – это отдельная методология, она также представляет собой инструмент для построения процесса создания продукта. Руководители проекта сами определяют, как они будут использовать эти инструменты.

Стоит отметить, что в Kanban присутствует своя философия, которая основана на принципе непрерывного совершенствования, осуществляемого за счет выбора задач из списка невыполненной работы и перемещения этих задач в основной и постоянный поток работы [8].

Внедрение Kanban-методики в проект, построенный на базе каскадной модели, позволяет не только визуализировать процесс работы, но также и упростить его. Сам процесс внедрения Kanban-методологии не является затратным как с финансовой, так и с организационной точки зрения [8].

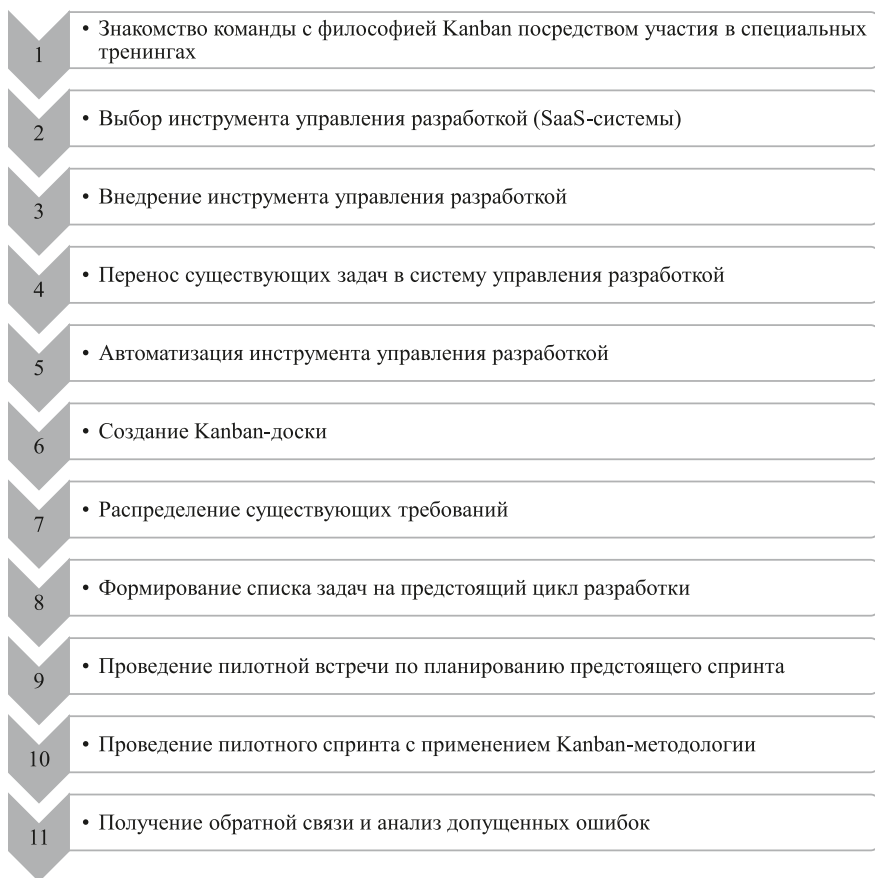
Основным этапом внедрения Kanban-методологии является знакомство всех участников команды с философией Kanban. Все члены команды должны понимать, как работают процессы и какова цель методики. Чаще всего компании нанимают специалистов для проведения специальных тренингов с целью ознакомления команды с базовыми знаниями о методике Kanban. После этого все участники команды должны настроить процесс в инструменте для управления процессом разработки – создать соответствующие сущности, сформировать список задач на ближайший спринт, настроить Kanban-доску.

Начиная с момента планирования и заканчивая релизом задачи, весь процесс разработки является понятным и прозрачным для ее участников. Кроме этого, такой подход в управлении процессом разработки позволяет улучшить коммуникацию внутри команды и повышает мотивацию среди ее участников путем визуализации процесса.

Таким образом, внедрение Kanban-методологии заключается в следующих этапах (рис. 1).

Использование инструмента Kanban делает процесс более эффективным, визуализируя состояние процесса разработки на момент поступления задачи в тестирование. Можно наглядно увидеть, позволяют ли ресурсы команды взять в работу дополнительную задачу.





Источник: составлено авторами по материалам [2, 3, 5, 8].

**Рис. 1.** Этапы внедрения Kanban-методологии

Fig. 1. Stages of implementing Kanban methodology

### Риски выбора эффективной методологии управления разработкой проекта

В существующих реалиях выбор эффективной методологии управления процессом – это основа успешной реализации проекта. В связи с этим необходимо проводить тщательный анализ и оценку рисков, связанных с каждой методологией, и выбирать ту, которая наилучшим образом соответствует уникальным характеристикам и требованиям конкретного ИТ-проекта.

Основные потенциальные риски были классифицированы на внешние и внутренние (табл. 2).

Для определения наиболее эффективного варианта необходимо выделить ряд критериев, например финансовая составляющая проекта, возможности команды и т. д. Выделение критериев методологии управления разработкой ИТ-проектов предоставляет возможность команде выбрать модель управления, которая идеально соответствует не только целям проекта, но и возможностям команды и требованиям заказчика. Отсюда следует вывод, что одним из возможных рисков могут быть неправильно выделенные критерии и, как следствие, неверно подобранная концепция управления разработкой. Данный риск может возникнуть еще до нача-

Таблица 2. Классификация потенциальных рисков

Table 2. Classification of potential risks

Виды рисков <i>Types of risks</i>	Область возникновения <i>Area of origin</i>
Внутренние	<b>Организационная.</b> Неполное понимание необходимости перехода на новую модель разработки
	<b>Проект.</b> Неправильная оценка особенностей конкретного проекта
	<b>Команда.</b> Неправильная оценка возможностей команды, недостаточный опыт и подготовка команды в выбранной методологии могут снизить эффективность ее применения
	<b>Спротивление изменениям.</b> Внедрение новой методологии может столкнуться с сопротивлением сотрудников и недостаточной поддержкой руководства
	<b>Недостаток ресурсов.</b> Внедрение новой методологии может требовать дополнительных ресурсов (время, финансы, технологии), которые не всегда доступны
Внешние	
• конкурентные	<b>Конкуренция.</b> Смежные команды могут создать наиболее эффективную систему управления процессом
• рыночные	<b>Внешняя среда.</b> Возможно появление непредвиденных обстоятельств, способных повлиять на финансирование внедрения новой методики

Источник: составлено авторами по материалам [1, 4, 7, 9, 12].

ла первого этапа внедрения новой методологии.

Главным риском при внедрении новой методологии в проекте является неполное понимание необходимости перехода на новую модель разработки. В таком случае снижается эффективность внедрения новой методологии, так как участники команды не способны следовать всем принципам внедряемой методологии. Данный риск существует на всех этапах во время пилотного спринта.

Другим, не менее существенным риском является риск неправильной оценки возможностей команды. Данному риску больше всего подвержен этап разработки, так как из-за неправильной оценки возможностей команды возникает вероятность, что команда не успеет выполнить задачи, взятые в спринт.

Все риски, связанные с внедрением Kanban-методологии в проекте, могут проявить себя в течение первого спринта,

по итогам которого у команды будет возможность проанализировать их и обдумать решение.

## Заключение

В современной индустрии разработки программного обеспечения существует множество методологий, каждая из которых предлагает уникальный подход к управлению проектами, обеспечивая различные уровни гибкости, контроля и предсказуемости.

Различия методологий при разработке ПО отражают разнообразие подходов к управлению проектами и стремление к оптимизации процессов для достижения наилучших результатов. Проведенный сравнительный анализ современных гибких методологий управления разработкой программного обеспечения позволяет глубже понять их преимущества и недостатки в условиях высококонкурентных рынков.

В статье обоснована целесообразность внедрения Kanban-методологии в качестве эффективного инструмента управления разработкой программного обеспечения. Методология Kanban ориентирована на управление процессами разработки проекта, сосредотачивая внимание на ограничении количества одновременно выполняемых задач и достижении оптимальной эффективности. Основным средством, предлагаемым Kanban, является визуализация процесса работы. Кроме того, Kanban удобен для команд с ограниченными человеческими ресурсами, поскольку не требует строгого разделения ролей. Применение Kanban повышает эффективность процесса разработки проекта ИС.

Выбор эффективной методологии управления проектом является многогранной задачей, требующей тщательного анализа и учета множества факторов, таких как специфика проекта, характеристики команды, уровень подготовки и т. д. Недооценка рисков, связанных с неправильным выбором методологии, может привести к значительным организационным, временным и финансовым потерям, что в конечном счете негативно скажется на успешности реализации проекта. Поэтому для минимизации этих рисков рекомендуется проводить предварительный анализ и пилотное внедрение выбранной методологии, а также обеспечивать обучение и поддержку команды на всех этапах проекта.

### Список литературы

1. Абдалов М.С., Трубин А.Е., Нечаев А.М., Чаусов Д.Н. Оценка эффективности применения технологий моб-программирования в управлении IT-проектами // Современная конкуренция. 2023. Т. 17. № 4. С. 80–89. DOI: 10.37791/2687-0649-2023-17-4-80-89.
2. Богданов А.Е., Давлеткиреева Л.З. Применимость методологии Kanban для управления небольшими IT-проектами с мелкими командами разработчиков // Глобальная экономика в XXI веке: роль биотехнологий и цифровых технологий: сборник научных статей. – М., 2020. С. 24–28.
3. Васильева А.Д., Буторин А., Котегова Л. Технология управления проектами на основе методологии Agile // Вестник Алтайской академии экономики и права. 2021. № 1-2. С. 118–124.
4. Гришаева И.Н., Вайтекунайте П.Ю. Использование гибких моделей управления командами IT-проектов // Управление человеческими ресурсами – основа развития инновационной экономики: материалы X Международной научно-практической конференции. – Красноярск, 2021. С. 250–254. DOI: 10.53374/9785864338810\_25.
5. Кобзев А., Трифонов И.В. Анализ современных тенденций использования гибких методов в разработке проектов // Стратегии бизнеса. 2021. Т. 9. № 3. С. 69–73. DOI: 10.17747/2311-7184-2021-3-69-73.
6. Козлов С.В., Иванова М.В. Основные принципы применения технологии экстремального программирования при разработке программного обеспечения // Современное состояние и перспективы развития науки и образования: сборник научных трудов. – Анапа, 2021. С. 196–205.
7. Коломыцева А.О., Белоусов В. Современные технологии управления проектами информатизации на основе методологий Agile-Scrum и Waterfall // Новое в экономической кибернетике. 2017. № 4. С. 106–118.
8. Метельская Ю.Н., Шафранович П.С., Кашникова И.В. Использование Scrum, Kanban в проектах гибкой разработки программного обеспечения для различных сфер деятельности организаций // Роль и место инноваций в сфере агропромышленного комплекса: материалы Всероссийской (национальной) научно-практической конференции. – Курск, 2020. С. 419–423.
9. Онокой Л.С. Гибкие подходы к разработке программного обеспечения: эволюция и перспективы развития // Качество. Инновации. Образование. 2021. № 1 (171). С. 57–67. DOI: 10.31145/1999-513x-2021-1-56-66.

10. Раубецкий А.В. Технология управления проектами и проектными командами на основе методологии гибкого управления проектами Agile // Научные исследования в современном мире: опыт, проблемы и перспективы развития: сборник научных статей. – Уфа, 2023. С. 57–66.
11. Родина М. А. Экстремальное программирование (XP): подходы и опыт «гибкого» внедрения в современных компаниях // Взгляд молодых ученых на проблемы устойчивого развития: сборник научных статей. – М., 2019. Т. 8. С. 142–148.
12. Торосян Е.К., Тюлькина А.С. Критерии выбора методологии управления IT-проектами // Петербургский экономический журнал. 2020. № 1. С. 99–108. DOI: 10.25631/PEJ.2020.1.99.10.
13. Филатов Е.А., Курзаева Л.В. К вопросу о выборе Agile-технологий управления проектами // Управление проектами: сборник статей. – Магнитогорск, 2023. С. 48–57.
14. Cohen E. The Definitive Guide to Project Management Methodologies // Workamajig. URL: <https://www.workamajig.com/blog/project-management-methodologies> (дата обращения: 29.10.2024).

### Сведения об авторах

Анисимов Александр Юрьевич, ORCID 0000-0002-8113-4523, канд. экон. наук, доцент, заместитель директора по учебно-методической работе факультета информационных технологий, доцент кафедры информационного менеджмента им. профессора В.В. Дика, Университет «Синергия», Москва, Россия, anisimov\_au@mail.ru

Токмакова Елена Николаевна, ORCID 0000-0001-9963-2726, канд. экон. наук, доцент, доцент кафедры цифровой экономики, Университет «Синергия», Москва, Россия, e\_tokmakova@mail.ru

Трубин Александр Евгеньевич, ORCID 0000-0002-7189-5679, канд. экон. наук, доцент, заведующий кафедрой цифровой экономики, Университет «Синергия», Москва, Россия, niburt@yandex.ru

Статья поступила 19.07.2024, рассмотрена 05.08.2024, принята 20.08.2024

### References

1. Abdalov M., Trubin A., Nechaev A., Chausov D. Performance Evaluation of Application Mob Programming Technologies in IT Project Management. *Sovremennaya konkurentsya=Journal of Modern Competition*, 2023, vol.17, no.4, pp.80-89 (in Russian). DOI: 10.37791/2687-0649-2023-17-4-80-89.
2. Bogdanov A.E., Davletkireeva L.Z. *Primenimost' metodologii Kanban dlya upravleniya nebol'shimi IT-proektami s melkimi komandami razrabotchikov* [Applicability of the Kanban methodology for managing small IT projects with small development teams]. *Global'naya ekonomika v XXI veke: rol' biotekhnologii i tsifrovyykh tekhnologii: sbornik nauchnykh statei* [Global Economy in the 21st Century: The Role of Biotechnologies and Digital Technologies: Collection of Scientific Articles]. Moscow, 2020, pp.24-28.
3. Vasilyeva A.D., Butorin A., Kotegova L. Project management technology based on the Agile methodology. *Vestnik Altaiskoi akademii ekonomiki i prava*, 2021, no.1-2, pp.118-124 (in Russian).
4. Grishaeva I.N., Vaitekunaite P.Yu. Using flexible models of IT-project team management. *Upravlenie chelovecheskimi resursami – osnova razvitiya innovatsionnoi ekonomiki: materialy X Mezhdunarodnoi nauchno-prakticheskoi konferentsii* [Human Resource Management – The Basis for the Development of an Innovative Economy: Materials of the X International Scientific and Practical Conference]. Krasnoyarsk, 2021, pp.250-254 (in Russian). DOI: 10.53374/9785864338810\_25.
5. Kobzev A. Trifonov I.V. Analysis of current trends in the use of flexible methods in project development. *Strategii biznesa=Business Strategies*, 2021, vol.9, no.3, pp.69-73 (in Russian). DOI: 10.17747/2311-7184-2021-3-69-73.
6. Kozlov S.V., Ivanova M.V. *Osnovnye printsipy primeneniya tekhnologii ekstremal'nogo programmirovaniya pri razrabotke programmnogo obespecheniya* [Basic principles of using Extreme

- Programming technology in software development]. *Sovremennoe sostoyanie i perspektivy razvitiya nauki i obrazovaniya: sbornik nauchnykh trudov* [Current state and prospects for the development of science and education, collection of scientific papers]. Anapa, 2021, pp.196-205.
7. Kolomyshva A.O., Belousov V.V. Modern technologies and management of informatization projects based on methodologies agile-scrum and waterfall. *Novoe v ekonomicheskoi kibernetike*, 2017, no.4, pp.106-118 (in Russian).
  8. Metelskaya Yu.N., Shafranovich P.S., Kashnikova I.V. *Ispol'zovanie Scrum, Kanban v proektakh gibkoi razrabotki programmogo obespecheniya dlya razlichnykh sfer deyatel'nosti organizatsii* [The use of Scrum, Kanban in flexible software development projects for various areas of activity of organizations]. *Rol' i mesto innovatsii v sfere agropromyshlennogo kompleksa: materialy Vserossiiskoi (natsional'noi) nauchno-prakticheskoi konferentsii* [The role and Place of Innovation in the Field of Agro-industrial Complex: Materials of the All-Russian (National) Scientific and Practical Conference]. Kursk, 2020, pp.419-423.
  9. Onokoy L.S. Flexible approaches to the development of software: Evolution and prospects of development. *Kachestvo. Innovatsii. Obrazovanie*=Quality. Innovation. Education, 2021, no.1(171), pp.56-66 (in Russian). DOI: 10.31145/1999-513x-2021-1-56-66.
  10. Raubetsky A. V. Project and project team management technology based on agile project management methodology. *Nauchnye issledovaniya v sovremennom mire: opyt, problemy i perspektivy razvitiya: sbornik nauchnykh statei* [Scientific Research in the Modern World: Experience, Problems and Development Prospects: Collection of Scientific Articles]. Ufa, 2023, pp.57-66 (in Russian).
  11. Rodina M.A., Mukhin K.Yu. Extreme Programming (XP): Approaches and best practices of “agile” implementing to modern companies. View of Young Scientists on the Problems of Sustainable Development: Collection of Scientific Articles. Moscow, 2019, vol.8, pp.142-148 (in Russian).
  12. Torosyan E.K., Tyulkina A.S. Selection criteria for IT project management. *Peterburgskii ekonomicheskii zhurnal*=St. Petersburg Economic Journal, 2020, no.1, pp.99-108 (in Russian). DOI: 10.25631/PEJ.2020.1.99.10.
  13. Filatov E.A., Kurzaeva L.V. *K voprosu o vybore Agile-tekhnologii upravleniya proektami* [On the issue of choosing Agile project management technologies]. *Upravlenie proektami: sbornik statei* [Project Management: Collection of Articles]. Magnitogorsk, 2023, pp.48-57.
  14. Cohen E. The Definitive Guide to Project Management Methodologies. Workamajig. Available at: <https://www.workamajig.com/blog/project-management-methodologies> (accessed 29.10.2024).

### About the authors

*Alexander Yu. Anisimov*, ORCID 0000-0002-8113-4523, Cand. Sci. (Econ.), Assistant Professor, Deputy Director for Educational and Methodological Work of Information Technologies Faculty, Information Management Department named after Professor V.V. Dick, Synergy University, Moscow, Russia, [anisimov\\_au@mail.ru](mailto:anisimov_au@mail.ru)

*Elena N. Tokmakova*, ORCID 0000-0001-9963-2726, Cand. Sci. (Econ.), Associate Professor, Digital Economy Department, Synergy University, Moscow, Russia, [e\\_tokmakova@mail.ru](mailto:e_tokmakova@mail.ru)

*Alexander E. Trubin*, ORCID 0000-0002-7189-5679, Cand. Sci. (Econ.), Associate Professor, Director of Digital Economy Department, Synergy University, Moscow, Russia, [niburt@yandex.ru](mailto:niburt@yandex.ru)

Received 19.07.2024, reviewed 05.08.2024, accepted 20.08.2024