

MANAGEMENT

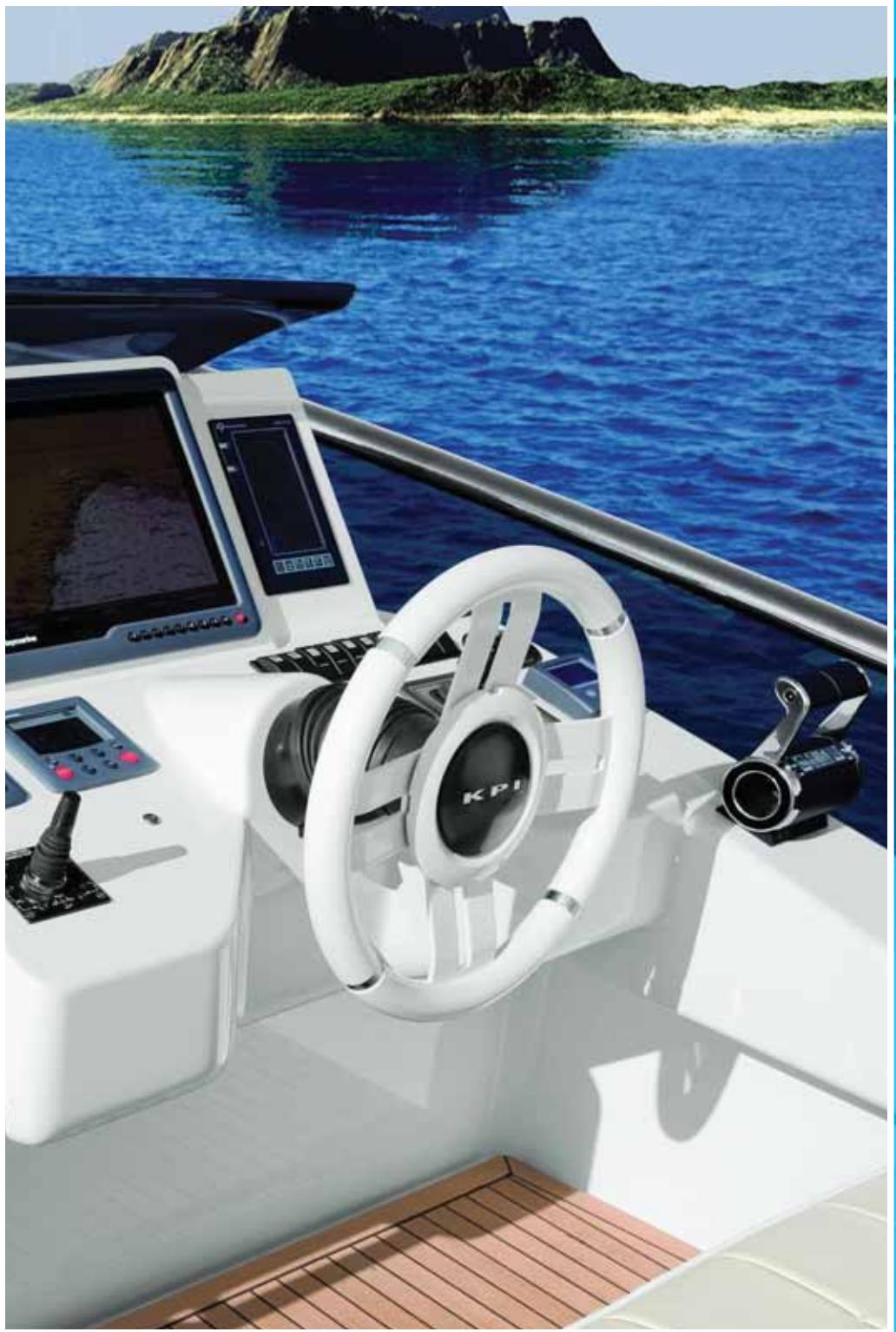
№ 2 [42] • 2017



**КАК УСИЛИТЬ
ПРОДАЖИ?**

**KPI ДЛЯ ВЫСШЕГО
РУКОВОДСТВА**

**AGILE ИЛИ
WATERFALL?**



MANAGEMENT

- 4** М. Вишнякова. Система мотивирования для генерального директора
- 10** С. Андреев. Бессмысленные KPI: почему эффективность людей иногда лучше не измерять
- 13** Л. Дубовицкая. Совещания – действенный рабочий инструмент или пустое времяпрепровождение?
- 16** О. Мамажонов. Какая методология разработки лучше: Agile или Waterfall?
- 20** С. Карпов. Как и когда применять метод Agile?
- 22** И. Мясникова. Кайдзен – систематизированный здравый смысл
- 23** Ю. Бударов. В Кайдзен – за... улыбкой
- 24** И. Инелеев. Почему клиенты выбирают сертифицированных поставщиков
- 26** 9 бизнес-стратегий, которые вы должны обязательно изучить
- 29** А. Мокеев. Реализация стратегии развития предприятия: пошаговый алгоритм
- 32** А. Мокеев. Как сделать сервис ключевой составляющей бизнеса?



- 36** В. Качалов. Надо ли сообщать потребителю об отправке несоответствующей продукции?
- 40** В. Чичаев. Практические аспекты внедрения энергоменеджмента: энергоанализ

А. Туйтебаева. Система энергоменеджмента как инструмент исполнения Закона об энергосбережении и повышения энергоэффективности

- 45** Д. Гришин. Оклад или процент от продаж: как найти компромисс
- 47** Р. Хонниэз. Как усилить продажи в имеющемся отделе продаж?
- 54** С. Макеева. Лови рыбу там, где она водится, или 10 примеров, как с помощью кросс-маркетинга увеличить продажи на 35%
- 57** А. Иванин. Комplименты клиентам – инструмент, который увеличивает продажи в 2 раза
- 61** 35 ошибок менеджера по продажам
- 63** Д. Савельев. 10 фактов от ваших клиентов
- 64** Т. Беляева. 4 правила Гарвардского метода переговоров без поражений
- 67** Ю. Музыра, А. Петиченко. Подготовка руководителей аптек по правилам надлежащей аптечной практики
- 70** Ю. Адлер. Куда идёт менеджмент?
- 77** В. Качалов. Как правильно управлять документированной информацией?

Главный редактор: Р. Ибрагимов

Экспертный совет: Абдуллаева Д., Волынская Н., Воробьева О., Илюхин В., Истомина Н., Кудяков И., Куйлибаев Э., Мухамедзянова Р., Пешков И., Пятков О., Раджабова Н., Расулова М., Цой Н.

Учредитель: ООО «Серт Интернейшнл»

Адрес редакции: 105082, г. Москва, ул. Рубцовская наб., д. 3, стр. 3, пом. I, комн. 25. Тел./факс: +7 (499) 346-77-14.

E-mail: info@iso-management.com, веб-сайт: www.iso-management.com.

Печать: ЧП «GRAND SIRIUS».

Тираж: 3000 экз. Заказ № XXX.

Дата подписания в печать: 15.05.2017 г.

Журнал зарегистрирован Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций (Россия), Управлением внутренней политики (Казахстан), Агентством по печати и информации (Узбекистан). Регистрационное свидетельство: №ФС77-59021 (Россия), №39 (Казахстан), №0539 (Узбекистан).

Подписной индекс:

в объединенном каталоге «Пресса России» – 93532 (Россия), в каталоге АО «Казпочта» – 11151 (Казахстан), в каталоге АК «Матбуот тарқатувчи» – 1151 (Узбекистан), в каталоге ГП «Кыргыз Почтасы» – 77369 (Кыргызстан).

Реализуется по свободной цене.

Журнал «Management» распространяется в странах СНГ с 2007 года.

КАКАЯ МЕТОДОЛОГИЯ РАЗРАБОТКИ ЛУЧШЕ: AGILE ИЛИ WATERFALL?



Отабек Мамажонов,
старший бизнес-консультант
по развитию бизнеса ООО
«TUNE Consulting»
E-mail: omamajonov@tuneconsulting.net

Что лучше: сдача проекта «под ключ», точное следование стандартам, соблюдение всех этапов планирования или постоянное совершенствование? Какой из методов удобен, а какой бесконечно устарел? Стоит ли ради модной нынче новой методологии отказаться от старого? Или лучше всего держать баланс?

ИТ-разработки тоже бывают разные, как и условия работы, и, в зависимости от типа проекта, подходы могут меняться. Какая методология разработки лучше: Agile или Waterfall?

Прежде чем ответить на этот вопрос, давайте сначала поближе познакомимся с этими методологиями.

Agile – гибкость, постоянное совершенствование, творческий подход

Гибкая методология разработки (англ. Agile software development) – серия подходов к разработке программного обеспечения, ориентированных на использование итеративной разработки, динамическое формирование требований и обеспечение их реализации в результате постоянного взаимодействия внутри самоорганизующихся рабочих групп, состоящих из специалистов различного профиля.

Agile применяется как эффективная практика организации труда небольших групп (которые делают однородную творческую работу) в объединении с их управлением комбинированным (либеральным и демократическим) методом.

Большинство гибких методологий нацелены на минимизацию рисков путём сведения разработки к серии коротких циклов, называемых итерациями, которые обычно делятся две-три недели. Каждая итерация сама по себе выглядит как программный проект в миниатюре и включает все задачи, необходимые для выдачи мини-прироста по функциональ-

ности: планирование, анализ требований, проектирование, программирование, тестирование и документирование. Хотя отдельная итерация, как правило, недостаточна для выпуска новой версии продукта, подразумевается, что гибкий программный проект готов к выпуску в конце каждой итерации. По окончании каждой итерации команда выполняет переоценку приоритетов разработки.

Agile-методы делают упор на непосредственное общение лицом к лицу. Большинство Agile-команд расположены в одном офисе, иногда называемом по-английски bullpen. Как минимум, она включает и «заказчиков» (product owner – заказчик или его полномочный представитель, определяющий требования к продукту; эту роль может выполнять менеджер проекта, бизнес-аналитик или клиент). Офис может также включать тестировщиков, дизайнеров интерфейса, технических писателей и менеджеров.

Основной метрикой Agile-методов является рабочий продукт. Отдавая предпочтение непосредственному общению, Agile-методы уменьшают объём письменной документации по сравнению с другими методами. Это привело к критике этих методов как недисциплинированных.

В феврале 2001 в штате Юта США был выпущен «Манифест гибкой методологии разработки программного обеспечения». Он являлся альтернативой управляемым документацией, «тяжеловесным» практикам разработки программного обеспечения, таким как «метод водопада» (Waterfall), явившимся золотым стандартом разработки в то время. Данный манифест был одобрен и подписан представителями методологий: экстремального программирования, Crystal Clear,



DSDM, Feature driven development, Scrum, Adaptive software development, Pragmatic Programming. Гибкая методология разработки использовалась многими компаниями и до принятия манифеста, однако вхождение Agile-разработки в массы произошло именно после этого события.

Как на практике выглядит работа по Agile методологии

Первым моим проектом с гибкой методологией разработки Agile была разработка с нуля мобильного приложения для приобретения билетов на поезд. В данном проекте мы старались творчески подходить при разработке данного приложения, так как не было четкого технического задания, а только требование разработать клиентский интерфейс для покупки билетов на базе программного обеспечения компании АСУ «Экспресс» (автоматизированная система управления резервированием мест и билетно-кассовыми операциями, предназначенная для бронирования мест в поездах дальнего следования). Лицензию на право использования системы приобрели практически все страны СНГ и Прибалтики. Система «Экспресс-3» является полноправным членом международного сообщества систем резервирования мест, активно взаимодействует с другими европейскими электронными системами резервирования в рамках вычислительных сетей ОСЖД/МСЖД) для продажи билетов.

Мы разрабатывали, тестировали, анализировали, делились мнениями с членами команды, дорабатывали, обновляли программу, и так продолжалось, пока данный продукт не был приведен в финальную версию. После клиент также предложил свои идеи, которые мы учли и доработали продукт. После того, как мы завершили разработку продукта, решили запустить в pilotную эксплуатацию, и во время работы программы клиенты начали делиться своими мнениями и замечаниями, предлагать улучшения. На основе мнений и замечаний клиентов мы дорабатывали и обновляли свой продукт.

Этот процесс постоянен, он продолжается внутри самоорганизующихся рабочих групп, состоящих из специалистов различного профиля.

Так на практике выглядит работа по Agile методологии.

Waterfall – соблюдение стандартов, четкое планирование, точные сроки

Каскадная модель (англ. waterfall model, иногда переводят как модель «Водопад») – модель процесса разработки программного обеспечения, в которой процесс разработки выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки.

В качестве источника названия часто указывают статью, опубликованную У. Ройсом (W. Royce) в 1970 году, при том, что сам Ройс использовал итеративную модель разработки.

В 1970 году в своей статье Ройс описал в виде концепции то, что сейчас принято называть «каскадная модель», и обсуждал недостатки этой модели. Там же он показал, как эта модель может быть доработана до итеративной модели.

В оригинальной каскадной модели Ройса, следующие фазы шли в таком порядке:

1. Определение требований
2. Проектирование

3. Конструирование (также «реализация» либо «кодирование»)
4. Воплощение
5. Тестирование и отладка (также «верификация»)
6. Инсталляция
7. Поддержка

Какова практика работы по методу Waterfall

Мой первый ИТ-проект в рамках международной компании – проекте по внедрению АБС Globus (сейчас Т24) компании Temenos в Национальном банке Республики Узбекистан – начался в 2002 году.

Для реализации ИТ-проекта по внедрению АБС Globus компания выбрала метод Waterfall, так как у компании имеются четко соблюдаемые стандарты внедрения. Проект был разделен на несколько этапов, и у каждого этапа был свой срок.

Все внедрялось в рамках плана поэтапно.

1 этап. Анализ требований клиента и составления документа.

2 этап. GAP-анализ.

3 этап. Построение продукта.

4 этап. Предварительное тестирование.

5 этап. UAT-тестирование и исправление ошибок.

6 этап. Миграция.

7 этап. Запуск в производственную эксплуатацию и поддержка.

При внедрении и доработке готового софта для определенного клиента данный метод удобен, так как на этапе анализа четко определяются все требования клиента, включая GAP-анализ (требования клиента, которые не покрываются стандартным функционалом продукта), и составляется план внедрения программного обеспечения.

Следуя каскадной модели, разработчик переходит от одной стадии к другой строго последовательно. Сначала полностью завершается этап «определение требований», в результате чего получается список требований к ПО. После того как требования полностью определены, происходит переход к проектированию, в ходе которого создаются документы, подробно описывающие для программистов способ и план реализации указанных требований. После того как проектирование полностью выполнено, программистами выполняется реализация полученного проекта. На следующей стадии процесса происходит интеграция отдельных компонентов, разрабатываемых различными командами программистов. После того как реализация и интеграция завершены, производится тестирование и отладка продукта; на этой стадии устраняются все недочёты, появившиеся на предыдущих стадиях разработки. После этого программный продукт внедряется и обеспечивается его поддержка – внесение новой функциональности и устранение ошибок.

Тем самым, каскадная модель подразумевает, что переход от одной фазы разработки к другой происходит только после полного и успешного завершения предыдущей фазы, и что переходов назад либо вперёд или перекрытия фаз – не происходит.

Во всех остальных проектах по внедрению АБС Т24 в странах СНГ (Казахстан, Азербайджан, Грузия, Украина) применялся этот же метод, так как он был удобен и для разработчика, и для клиента. Согласно этой методологии мы старались добиваться сдачи проекта в срок.

Во всех постсоветских странах у нас привычно все базировалось на основах бухгалтерского учета, и все банки выбирают Account Base платформу, и им сложно привыкать к зарубежным системам, которые строятся на основе сделок/контрактов. Поэтому у них имеется четкая методология основ бухгалтерского учета, где все прописано, и вы обязаны соблюдать данные правила. В данных проектах метод Agile скорее всего был бы неудобен, так как все привело бы к бесконечному изменению изначальных требований и постоянному обновлению, которое привело к срыву сроков. Поэтому метод Waterfall представляется наиболее оптимальным при таких условиях работы.

Что выбирать – Agile или Waterfall?

Выбор методологии разработки компания делает сама, исходя из поставленных перед собой задач и с учетом выгодного положения для себя. В своей практике мы выбираем для каждого нового проекта удобный для себя метод. Выбор удобного метода разработки помогает успешно завершить проект. У каждого метода есть свои преимущества и недостатки, которые мы постараемся подробно рассмотреть.

При сравнении Waterfall и Agile необходимо осознание того, что Waterfall – это подход, хорошо описанный и детализированный. А вот Agile – это, скорее, набор практик и принципов, которые, так или иначе, поддерживают различные методологии гибкой разработки проектов.

Один из повторяющихся пунктов критики: при Agile-подходе часто пренебрегают созданием плана («дорожной карты») развития продукта, равно как и управлением требованиями, в процессе которого и формируется такая «карта». Гибкий подход к управлению требованиями не подразумевает далеко идущих планов (по сути, в данной методологии управления требованиями просто не существует), а подразумевает возможность заказчика вдруг и неожиданно в конце каждой итерации выставлять новые требования, часто противоречащие архитектуре уже созданного и поставляемого продукта. Такое иногда приводит к катастрофическим «авариям» с массовым ре-факторингом и переделкам практически на каждой очередной итерации.

Также считается, что работа в Agile мотивирует разработчиков решать все поступившие задачи простейшим и быстрейшим возможным способом, при этом зачастую не обращая внимания на правильность кода с точки зрения требований нежеллежащей платформы. Подход – «работа-



ет, и ладно», при этом не учитывается, что может перестать работать при малейшем изменении или же дать тяжёлые к воспроизведению дефекты после реального развёртывания у клиента. Это приводит к снижению качества продукта и накоплению дефектов.

А методику Waterfall довольно часто критикуют за недостаточную гибкость и объявление самоцелью формальное управление проектом в ущерб срокам, стоимости и качеству. Тем не менее, при управлении большими проектами формализация часто являлась очень большой ценностью, так как могла кардинально снизить многие риски проекта и сделать его более прозрачным. Поэтому даже в PMBoK¹ 3-й версии формально была закреплена только методика «каскадной модели» и не были предложены альтернативные варианты, известные как итеративное ведение проектов.

Сильные стороны Waterfall:

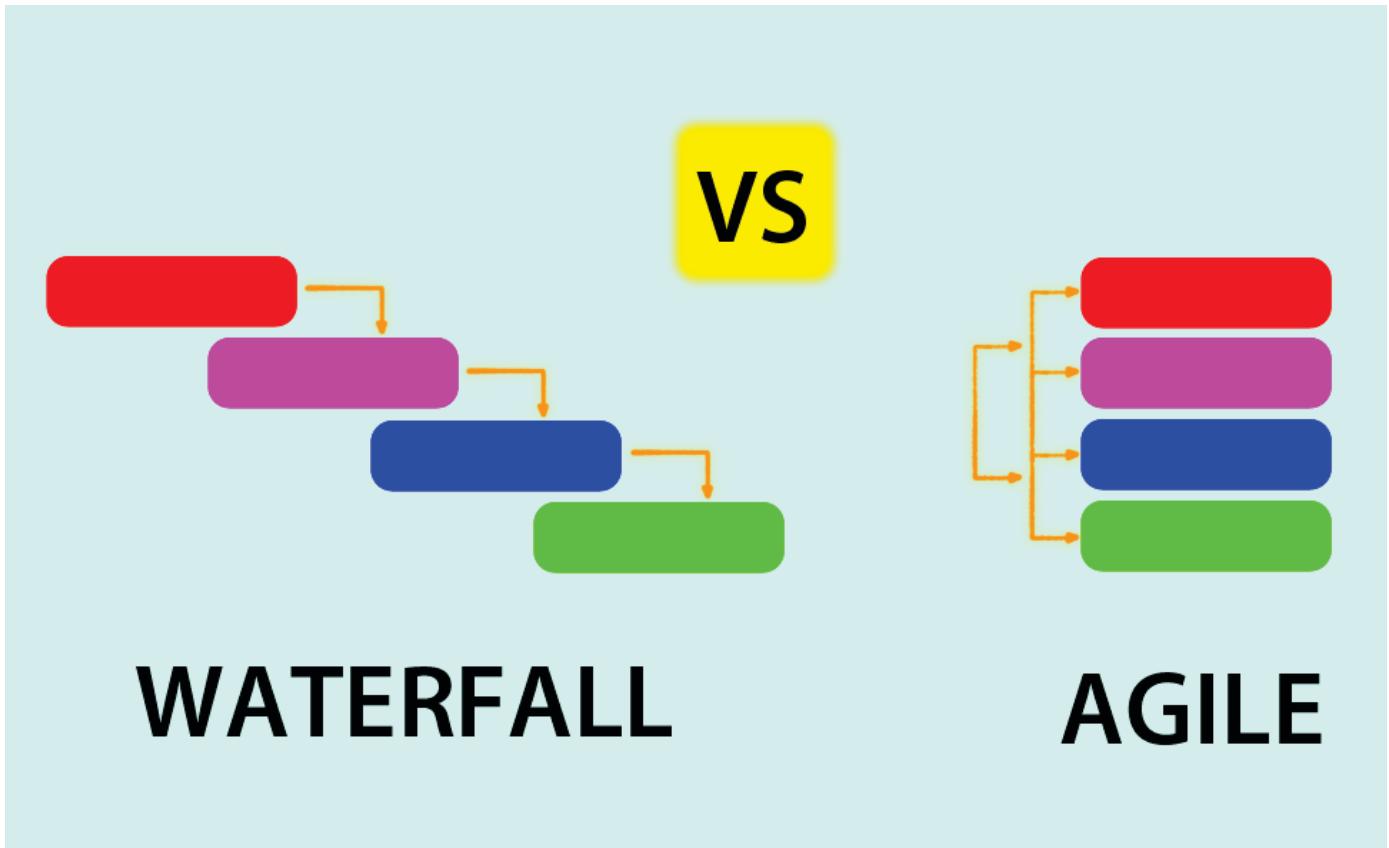
- Легок для понимания и использования;
- Детально структурирован, что облегчает его применение к малоопытным командам;
- Задает стабильные требования к проекту/продукту с самого старта;
- Проекты легко контролируются, отслеживаются ресурсы, риски, время;
- Качество имеет первоочередной приоритет по сравнению со стоимостью и временем.

Сильные стороны Agile:

- Итеративная разработка;
- Использование временных рамок (time boxes);
- Конечный пользователь вовлечен в процесс с самого начала;



¹ Свод знаний по управлению проектами (англ. Project Management Body of Knowledge, PMBoK) представляет собой сумму профессиональных знаний по управлению проектами. Project Management Institute – всемирная некоммерческая профессиональная организация по управлению проектами – использует этот документ в качестве основного справочного материала для своих программ по профессиональному развитию. Является Американским национальным стандартом, в котором описываются суть процессов управления проектами в терминах интеграции между процессами и взаимодействий между ними, а также цели, которым они служат. Эти процессы разделены на пять групп, называемых «группы процессов управления проектом».



- Быстрое получение первой/пробной версии продукта для тестирования;
- Легко воспринимаются корректировки и изменения в процессе разработки.

Слабые стороны Waterfall:

- Все требования должны быть определены и детально описаны до начала разработки;
- Дорого и медленно;
- Чувствителен к изменениям;
- Мало возможностей для конечного пользователя повлиять на цели проекта и требования к продукту;
- Зачастую проблемы выявляются на этапе тестирования;
- Много документации, много технической документации, которая не понятна конечному пользователю или заказчику.

Слабые стороны Agile:

- Может привести к низкому качеству продукта;
- Риск никогда не достигнуть закрытия/завершения проекта;
- Могут возникнуть проблемы с расширяемостью продукта.

Когда использовать Waterfall:

- Требования к продукту предельно ясны и стабильны;
- Известны используемые технологии и инструменты;
- Проект большой, дорогой и сложный;

Примеры: внедрение новой версии известного продукта; внедрение ERP систем.

Когда использовать Agile:

- Конечный пользователь вовлечен в проект со старта;
- Четко определены бизнес-цели проекта/продукта;
- Небольшой или средний проект, относительно короткий по времени;
- Состав команды стабильный;

- Команда с высоким уровнем профессионализма;
- Технические требования приемлемые, коррелируются с технологиями, которые собираются быть использованными для разработки;
- Система может быть модульной.

Гибридный вариант

Начиная с PMBoK 4-й версии удалось достичь компромисса между методологиями, приверженными формальному и поступательному управлению проектом, с методологиями, делающими ставку на гибкие итеративные методы. Таким образом, начиная с 2009 года, формально Институтом управления проектами (PMI) предлагается гибридный вариант методологии управления проектами как стандарт, сочетающий в себе как плюсы от методики «Водопада», так и достижения итеративных методологий.

Какой вывод можно сделать? Два подхода со своими плюсами и минусами, каждый из которых прекрасно подходит для применения в проектах с совершенно разными входными данными и требованиями.

Не исключена ситуация, когда обаими методами можно реализовать поставку одного и того же продукта, но на старте каждого из проектов входные данные по таким ключевым параметрам, как стоимость, время, квалификация команды, требования к качеству, конечный пользователь и т.д., существенно влияют на выбор методологии.

Задача руководителя проекта – выбрать наиболее подходящий способ для достижения целей проекта. Waterfall, RUP, Scrum, RAD, XP, FDD, TDD и другие методологии вам помогут этого добиться, если вы понимаете разницу между ними, их принципы, слабые и сильные стороны. В некоторых случаях, это не выбор между методологиями, а правильная комбинация подходов для каждого из этапов проекта. Чтобы не было недопонимания с клиентом, необходимо выбрать верную методологию.