



Министерство науки и высшего образования
Российской Федерации
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный технический
университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Рубежный контроль №3

Реферат на тему:

«Методы и технологии управления инновационными проектами.»

Дисциплина: Экономика
Студент: Левушкин И. К.
Группа: ИУ7-72Б
Преподаватель: Герцик Ю.Г.

Москва, 2020 г.

Содержание

Введение	3
1 Водопадная модель	4
2 V-модель	5
3 Итеративная модель	7
4 Спиральная модель	9
5 Agile	12
5.1 Agile принципы	14
5.2 Agile: XP	14
5.3 Agile: Scrum	14
6 Заключение	15
Заключение	15
Список используемой литературы	16

Введение

У человечества за всю историю накопился внушительный список успешно реализованных сложных проектов. От строительства Пирамид в Гизе до отправки человека на Луну, самые смелые человеческие начинания требовали слаженной работы тысяч людей. А это подразумевает сложную систему управления проектами.

Управление проектами – это управление и организация всего, что нужно для достижения цели – вовремя и в рамках бюджета. Будь то разработка нового программного обеспечения, проведение маркетинговой компании или высадка человека на Марс – проектное управление позволяет добиться успеха.

Все проекты разные. Не существует идеальной системы управления проектами, подходящей для каждого из видов проектов. Также не существует системы, которая бы подходила каждому руководителю и была удобна для всех членов команды. Однако за время существования проектного управления было создано немало эффективных подходов, методик и стандартов, которые можно взять на вооружение.

1 Водопадная модель

Наиболее очевидным способом управления проектами выступает разделение процесса исполнения того или иного проекта на ряд последовательных этапов. Именно линейная структура является базой осуществления традиционного проектного управления.

Классический водопадный метод управления проектами предусматривает выделение следующих этапов:

1. этап инициации. На этом этапе определяются требованиями к проекту. Для этапа характерно проведение совещаний и «мозговых штурмов», результатом которых становится предполагаемый продукт проекта;
2. этап планирования. Данный этап характеризуется определением того, как будет осуществляться достижение цели, определенной ранее. Команда проекта осуществляет уточнение и детализацию целей и результатов проекта, определяет состав работ. Данная информация выступает основой формирования календарного плана и бюджета, оценки рисков и выявления заинтересованных лиц;
3. этап разработки. Реализация данного этапа осуществляется не у всех проектов, поскольку зачастую он является частью этапа планирования. Наиболее характерно наличие этого этапа для технологических проектов. Данный этап предназначен для определения конфигурации проекта и его продуктов, а также технических способов достижения необходимого результата;
4. этап реализации и тестирования. Этот этап включает в себя основную работу по проекту, то есть непосредственное создание продукта. В ИТ-проектах это может быть написание кода, в строительных – возведение сооружений и так далее. Суть этого этапа состоит в реализации разработанных планов проекта, осуществлении контроля определенных показателей. Также на

этом этапе проект подвергается тестированию на соответствие требованиям. Результатом тестирования является выявление недостатков и их исправление;

5. этап мониторинга и завершения проекта. Содержание данного этапа управления проектами может различаться в зависимости от сложности проекта. В определенных случаях этот этап может включать в себя просто передачу заказчику продукта проекта. В другом случае речь идет о длительном процессе взаимодействия с заказчиком, направленном на улучшение проекта и повышение степени удовлетворенности его результатом.

Классический метод управления проектами составляет основу для всех других методов управления проектами. Данный метод применяется по отношению к проектам, которые имеют строгие ограничения относительно последовательности их исполнения.

Основным преимуществом данного метода является то, что заказчик и исполнитель еще на первом этапе определяют, какой результат они хотят получить. Это является основой стабильности при осуществлении управления проектом, фундаментом упорядочения процесса его реализации.

Основным недостатком данного метода является отсутствие толерантности к изменениям. В настоящее время применение классического метода управления проектами в наибольшей мере относится к строительным и инженерным проектам.

2 V-модель

V-модель – это улучшенная версия классической каскадной модели. Здесь на каждом этапе происходит контроль текущего процесса, для того чтобы убедиться в возможности перехода на следующий уровень. В этой модели тестирование начинается еще со стадии написания требований, причем для каждого последующего

этапа предусмотрен свой уровень тестового покрытия.

Для каждого уровня тестирования разрабатывается отдельный тест-план, то есть во время тестирования текущего уровня, происходит разработка стратегии тестирования следующего. при создании тест-планов,определяются ожидаемые результаты тестирования и указываются критерии входа и выхода для каждого этапа.

В V-модели каждому этапу проектирования и разработки системы соответствует отдельный уровень тестирования. Здесь процесс разработки представлен нисходящей последовательностью в левой части условной буквы V, а стадии тестирования – на ее правом ребре. Соответствие этапов разработки и тестирования показано горизонтальными линиями.

Основные преимущества V-модели разработки

- строгая этапизация;
- планирование тестирования и верификация системы производятся на ранних этапах;
- улучшенный, по сравнению с каскадной моделью, тайм-менеджмент;
- промежуточное тестирование.

Основные недостатки V-модели разработки

- недостаточная гибкость модели;
- собственно создание программы происходит на этапе написания кода, то есть уже в середине процесса разработки;
- недостаточный анализ рисков;
- нет работы с параллельными событиями и возможности динамического внесения изменений.

V-модель используется:

- В проектах, в которых существуют временные и финансовые ограничения;
- Для задач, которые предполагают более широкое, по сравнению с каскадной моделью, тестовое покрытие.

3 Итеративная модель

Итеративная разработка ПО — это процесс создания программного обеспечения, который осуществляется небольшими этапами, в ходе которых ведется анализ полученных промежуточных результатов, выдвигаются новые требования и корректируются предыдущие этапы работы.

Жизненный цикл проекта при итерационной разработке разбит на последовательность итераций, каждая из которых, по сути, является проектом в миниатюре, то есть включает в себя все процессы разработки ПО (сбор и анализ требований, составление спецификаций, непосредственную реализацию, тестирование и запуск), но в рамках одной итерации разрабатывается не весь проект, а только его версия или отдельная часть.

Как правило, цель каждой итерации — это получение версии ПО, включающей в себя как новые или преработанные возможности, реализованные в ходе текущей итерации, так и функциональность всех предыдущих итераций. Результат же финальной итерации содержит всю требуемую функциональность продукта.

Бюджет и сроки, необходимые для реализации финальной версии обычно изначально не устанавливаются, так как не определяется общий объём работ и требования формируются по ходу реализации.

Основные преимущества итеративной модели разработки

- Снижение рисков — раннее обнаружение конфликтов между требованиями, моделями и реализацией проекта; большая фо-

кусировка на основных задачах; динамическое формирование требований и управление ими.

- Организация эффективной обратной связи проектной команды с потребителем, создание продукта, реально отвечающего его потребностям.
- Быстрый выпуск минимально ценного продукта (MVP) и возможность вывести продукт на рынок и начать эксплуатацию гораздо раньше.

Основные недостатки итеративной модели разработки

- Проблемы с архитектурой и накладные расходы — при работе с хаотичными требованиями и без проработанного глобального плана архитектура приложения может пострадать, а на её приведение к адекватному виду могут потребоваться дополнительные ресурсы. По сути, за возможность менять требования в ходе создания продукта, приходится так или иначе расплачиваться.
- Нет фиксированного бюджета и сроков, а также нужна сильная вовлеченность Заказчика в процесс — для некоторых Заказчиков это неприемлемые условия сотрудничества с разработчиком, им лучше подойдет водопадная модель.

Итеративная модель используется:

- В больших проектах;
- В проектах с неопределенными требованиями;
- В проектах, которые носят инновационный характер и основаны на бизнес-гипотезах, требующих проверки.

4 Спиральная модель

Спиральную модель можно описать как повторяющуюся последовательность циклов разработки с непрерывным контролем рисков.

Ниже приведена схема механизма разработки ПО, использующая спиральную модель.

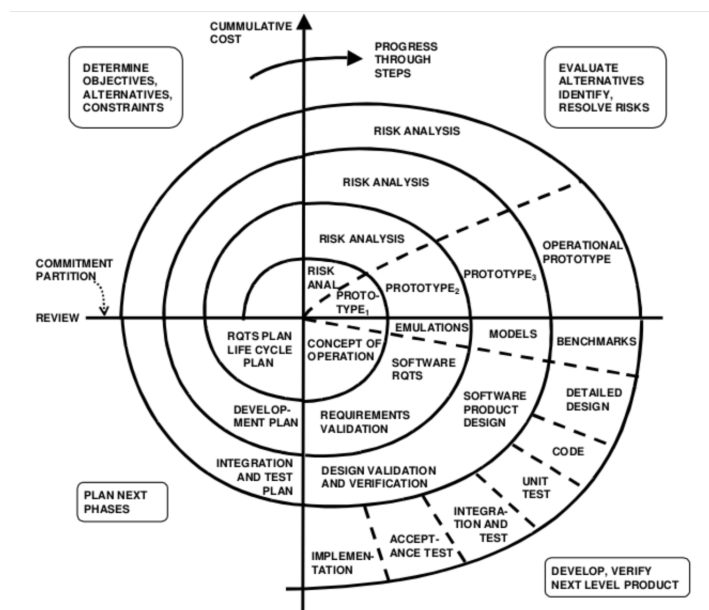


Рисунок 1: Спиральная модель.

Исходя из схемы, видно, что спиральная модель состоит из четырех главных повторяющихся стадий. В ходе процесса разработки проект несколько раз проходит через все эти фазы. Каждая такая итерация называется *спиралью*.

Четыре главные фазы это:

1. **Определение целей, альтернатив, ограничений, или фаза планирования.** С этой стадии начинается работа над проектом. Команда разработчиков формулирует цели проекта, основные требования (такие как, например, Business Requirement Specifications, или BRS, System Requirement Specifications, или SRS), возможный дизайн и т.д. На последующих спиральях требования формируются согласно отзывам, полученным от

заказчика. Именно поэтому постоянная коммуникация между заказчиком и командой крайне важна.

2. **Анализ, определение и разрешение рисков** является одной из самых значимых стадий разработки. В данном контексте, риски — это возможные события и состояния проекта, препятствующие достижению командой разработчиков поставленных целей. Существует довольно обширный диапазон возможных рисков, от тривиальных и легко преодолимых, до крайне серьезных. Главной задачей для команды разработчиков является выявление всех возможных рисков и присвоение им определенного уровня приоритета на основе их значимости. Следующим шагом является разработка возможных стратегий преодоления этих рисков. В итоге этих действий возможны изменения в последующих стадиях разработки. В качестве результата работы на этом этапе создается прототип.
3. **Фаза разработки.** На этом этапе происходит разработка и последующее тестирование продукта. Во время первой итерации, когда общие требования еще не так четко сформулированы, разрабатывается так называемый концепция будущего продукта (Proof Of Concept), которая необходима для получения отзыва заказчика. На последующих витках спирали рабочие версии продукта, или билды (builds), отправляются заказчику. Это позволяет получить более детальный отзыв и четче сформулировать требования.
4. **Планирование следующей фазы.** На этом этапе вся полученная информация используется для планирования дальнейших этапов разработки.

Основные преимущества спиральной модели разработки

- Мониторинг рисков является одной из главных особенностей, делающих данную модель особенно привлекательной в том случае, если вам предстоит управление большим, сложным

и дорогостоящим проектом. Более того, проект будет более прозрачным, поскольку спиральная модель изначально была спроектирована таким образом, чтобы каждая итерация тщательно анализировалась;

- Заказчик может увидеть работающую версию продукта уже на ранних стадиях жизненного цикла ПО;
- Изменения могут быть внесены на поздних стадиях разработки;
- Проект может быть разделен на несколько частей и те из них, которые, согласно анализу, окажутся более рискованными, могут быть реализованы на ранних стадиях. Такой подход может снизить трудности, связанные с управлением проектом;
- Строгий контроль над документацией, как результат постоянного анализа рисков.

Основные недостатки спиральной модели разработки

- Мониторинг рисков требует дополнительных ресурсов, а значит, эта модель может оказаться весьма затратной. Каждая итерация требует отдельной экспертизы, что делает управление проектом сложнее. Именно поэтому спиральная модель плохо подходит для небольших проектов;
- Большое количество промежуточных стадий разработки. Как следствие — большой объем документации;
- На самых ранних стадиях дата завершения работы над проектом может быть неизвестна, что также усложняет контроль над процессом разработки

Спиральная модель используется в проектах, где:

- предстоит работа со средним или высоким уровнем возможных рисков;

- заказчик не может предоставить достаточно четкий список требований к конечному продукту или эти требования достаточно сложные;
- ожидаются значительные изменения в процессе разработки.

5 Agile

Agile — итеративная модель разработки, в которой программное обеспечение создают инкрементально с самого начала проекта, в отличие от каскадных моделей, где код доставляется в конце рабочего цикла.

Основа гибкой методологии — разбиение проектов на маленькие рабочие кусочки, называемые пользовательскими историями. Согласно приоритетности задачи решают в рамках коротких двухнедельных циклов (итераций).

12 принципов, которые составляют Agile Methodology, можно поделить на 4 главные идеи:

- Приоритет людей и общения над инструментами и процессами;
- Приоритет работающего продукта над полной документацией;
- Приоритет сотрудничества с заказчиков над утверждением контракта;
- Приоритет готовности меняться над следованием первоначально созданному плану.

Эти идеи раскрываются в 12 принципах Agile Manifesto:

1. работающий конкурентоспособный продукт, удовлетворяющий заказчика — лучший показатель прогресса и измеритель эффективности;

2. оперативная и бесперебойная поставка продукта, удовлетворяющего заказчика;
3. адаптивность продукта к новым требованиям, которые могут повысить его ценность и конкурентоспособность (возможность внесения изменений на любом этапе разработки);
4. простота и прозрачность технических решений, документации, процессов и инструментов, чтобы не создавать лишней работы; частая поставка функционирующего продукта (раз в месяц/-неделю или ещё чаще);
5. постоянный темп работы всех участников проекта на протяжении всего его срока;
6. минимизация организационных и информационных барьеров, лучший путь передачи информации — это личный разговор лицом к лицу;
7. тесное и ежедневное общение исполнителей с заказчиком в течение всего проекта;
8. мотивация участников проекта и обеспечение их всеми необходимыми условиями работы, поддержкой и доверием;
9. самоорганизация и самоконтроль команды проекта;
10. непрерывное улучшение профессиональных компетенций команды проекта;
11. систематический анализ и постоянный поиск возможностей оптимизации командной и индивидуальной работы.

Основные преимущества agile-модели разработки

- быстрота, адаптивность и фокус на главном;
- Отсутствие бюрократии и периодичность поставок работающего продукта с постепенным наращиванием его функциональных возможностей.

Основные недостатки agile-модели разработки

- снижение важности регламентирующей и технической документации может привести к ее нерелевантности или даже к фактическому отсутствию;
- краткосрочное планирование не всегда учитывает необходимость масштабирования продукта, что влечет ошибки в архитектуре;
- появление новых требований после нескольких итераций приводит к кардинальным изменениям архитектуры и переделкам уже созданных решений;
- накопление дефектов и снижение качества продуктов вследствие решения проблем самым простым и быстрым, но не всегда самым правильным способом.

Agile-модель используется не только в управлении ИТ-проектами, а используется как эффективная практика организации труда небольших групп и творческих команд вместе с либеральными и демократическими методами менеджмента.

6 Заключение

В данной работе был проведен анализ наиболее популярных методов и технологий, используемых для управления инновационными проектами.

Были даны основные положения каждого из методов, разобраны их преимущества и недостатки, а также место их применения.

В итоге, поставленная цель, заключающаяся в исследовании методов и технологий управления инновационными проектами, достигнута.

Список литературы

- [1] Основные методологии разработки [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/edison/blog/269789/>, свободный – (11.12.2020)
- [2] Проектный менеджмент в IT - как это? [Электронный ресурс]. – Режим доступа: <https://worksection.com/blog/it-project-management.html>, свободный – (11.12.2020)
- [3] Топ-7 методов управления проектами: Agile, Scrum, Kanban, PRINCE2 и другие [Электронный ресурс]. – Режим доступа: <https://www.pmservices.ru/project-management-news/top-7-metodov-upravleniya-proektami-agile-scrum-kanban-prince2-i-drugie/>, свободный – (11.12.2020)