

Разработка видения продукта в методологии IBM Rational Unified Process

Аннотация

В статье рассматриваются вопросы формулировки основных требований к продукту, целей и границ проекта в концепции продукта на основе шаблона артефакта видения унифицированного процесса IBM Rational Unified Process.

Введение

Проект – это уникальная (в отличие от операций) деятельность, имеющая начало и конец во времени, направленная на достижение заранее определённого результата/цели, создание определённого, уникального продукта или услуги, при заданных ограничениях по ресурсам и срокам, а также требованиям к качеству и допустимому уровню риска [1]. «Временное» означает, что у любого проекта есть начало и непременно наступает завершение, когда достигаются поставленные цели, либо возникает понимание, что эти цели не могут быть достигнуты. «Уникальных» означает, что создаваемые продукты или услуги существенно отличаются от других аналогичных продуктов и услуг [2]. Уникальность продуктов или услуг проекта обуславливает необходимость последовательного уточнения их характеристик по мере выполнения проекта.

Под *управлением проектом* понимают применение к работам по проекту знаний, навыков, средств и технологий в целях удовлетворения потребностей всех заинтересованных сторон или для получения превосходящих их ожидания результатов [3]. Чтобы удовлетворить этим требованиям и ожиданиям, необходимо найти оптимальное сочетание между целями, сроками, затратами, качеством и другими характеристиками проекта. Управление проектами подчиняется четкой логике, которая связывает между собой различные области знаний и процессы управления проектами.

Одним из первых и фундаментальных этапов разработке любого программного обеспечения является разработка спецификации разрабатываемого продукта – «разработка требований». Разработка требований к программному обеспечению – процесс выявления, формулирования, анализа, документирования и верификации требований, подлежащих выполнению в продукте [4]. В его ходе системный аналитик формирует реестр требований, который ложится в документ или автоматизированную систему управления требованиями.

Иногда в него включается также процесс управления требованиями, являющийся частью более общего процесса управления изменениями в проекте.

Полнота и качество проведения разработки требований играют ключевую роль в успехе всего проекта. Это обусловлено тем, что ошибки, допущенные на этапе разработки требований, влияют на все содержание проекта и поэтому их исправления на более поздних этапах обходится весьма дорого [5]. В результате процесс управления требованиями занимает одно из главных мест во всех распространенных в настоящее время методологиях.

Текущее состояние управления проектами

В настоящее время существует несколько десятков стандартов управления проектами, из которых широкое применение на практике находят около десяти [6]. Предложено множество методик организации проектной деятельности, изучению дисциплины управления проектами уделяется большое внимание в ВУЗах, также существует большое число теоретических курсов обучающих организации разработки программного обеспечения.

Однако при всем богатстве выбора стандартов и методик по управлению проектами оказывается, что большинство проектов по разработке программного обеспечения не укладывается в отведенные сроки или оканчивается неудачей. Так по данным опубликованным известной консалтинговой компанией Standish Group в отчете о состоянии ИТ индустрии с красноречивым названием «Chaos Report» [7] только 26% проектов удается успешно завершить в установленные сроки и в пределах выделенных бюджетов.

В отчетах Standish Group [8] также отмечается, что:

- до 40% проектов в области внедрения информационных технологий оказываются неудачными или остаются незавершенными;
- расходы американских компаний на незавершенные проекты составляют 75 млрд. долл. в год;
- фактические расходы на реализацию 26% проектов составляют 189% от первоначальных оценок;
- более 60% проектов не обеспечивает планируемой рентабельности инвестиций.

Исследования причин низкой эффективности управления программными проектами ведутся уже более 30 лет, при этом многие компании проводившие исследования предлагали свои выходы из сложившейся ситуации, зачастую обещая универсальное решение проблемы. Однако если рассмотреть на динамику развития ситуации отраженную на Рис. 1, то легко заметить, что увеличение процентного отношения проектов заканчивающихся успешно и в

срок в основном идет за счет снижения относительного числа проектов зачисляющихся отменой разработки. В тоже время, относительный объем проектов превышающих запланированные сроки и бюджет за последние годы только увеличилось. Таким образом, можно говорить, что проблема создания и внедрения эффективной системы управления проектами по разработке программного обеспечения остается до сих пор не решенной для большей части проектов.

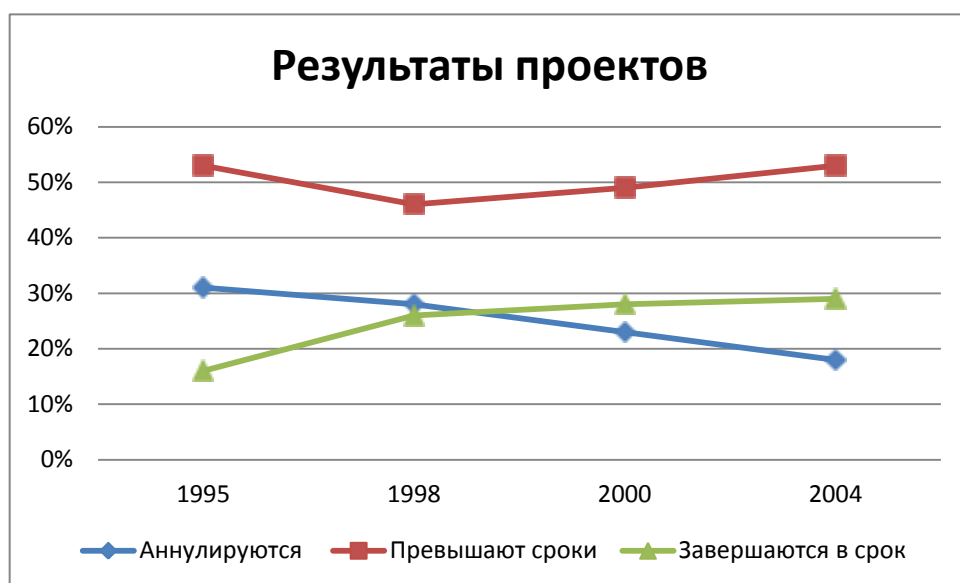


Рис. 1 Изменение результатов работы над проектами

Та же компания приводит в своих отчетах основные факторы достижения успеха в проектах с точки зрения участников проектов, представленные в Таб. 1, из которого видно, что одними наиболее критичными для успеха проекта является реальная поддержка руководства и налаженное взаимодействие с пользователями и руководитель проекта. Вторыми по важности идут постановка цели и границ проекта, обеспечивая в сумме 22% влияния на успех проекта. И только в конце списка упоминается влияние формального планирования, стабильности требований, выбранных методик и стандартов. В целом из данной таблицы можно сделать вывод, что организационные вопросы влияют на успех проекта гораздо больше, чем методологические или технологические.

Критерий успеха	Влияние
Поддержка со стороны высшего руководства	18,00%
Участие пользователей в проекте	16,00%
Опытный менеджер проекта	14,00%
Четко поставленные цели	12,00%
Минимизация границ проекта	10,00%

Использование технологических стандартов	8,00%
Стабильные требования	6,00%
Использование формальных методов	6,00%
Достоверные оценки	5,00%
Другое	5,00%

Таб. 1 Постановка проблемы

Таким образом, одной из главных причин провала проектов является нечеткая постановка целей и неясности в выделении потребителей разрабатываемого программного обеспечения, его ключевых характеристик и преимуществ, могут привести к тому, что разработанный продукт не оправдает ожидания пользователей, так как не будет обладать нужной функциональностью. В результате разработанное программное обеспечение, пусть и сделанное вовремя и в соответствии с поставленными требованиями оказывается невостребованным из-за отсутствия потребителей как таковых – и число подобных случаев довольно велико.

К увеличению сроков выполнения проектов часто приводит отсутствие фиксации границ проекта. Границы проекта задают, что проект должен делать, а что нет. Отсутствие четких ограничений на проекте, приводит к проблеме накопления небольших дополнений, когда, следуя новым пожеланиям заказчика, функциональность разрабатываемого продукта постоянно разрастается. С другой стороны разработчик может значительно уменьшить объем работы в результате функциональность разработанного продукта окажется недостаточной, и она не будет использоваться на практике.

Видение проекта в методике RUP

Одной из наиболее распространенных является унифицированный процесс управления проектами фирмы IBM Rational Unified Process (RUP). Он был разработан Филиппом Крачтенем (Philippe Kruchten), Иваром Якобсоном (Ivar Jacobson) и другими сотрудниками компании «Rational Software» в качестве дополнения к языку моделирования UML. RUP опирается на проверенные практикой методы анализа, проектирования и разработки программного обеспечения, методы управления проектами. Процесс обеспечивает прозрачность и управляемость процесса и позволяет создавать продукты в соответствии с требованиями заказчика на момент ее сдачи, а также в соответствии с возможностями инструментальных средств поддержки разработки.

В тоже время, модель RUP описывает абстрактный общий процесс, на основе которого организация или проектная команда должна создать конкретный специализированный

процесс, ориентированный на ее потребности. Именно эта черта RUP вызывает основную критику — поскольку он может быть чем угодно, его нельзя считать ничем определенным. В результате такого общего построения RUP можно использовать и как основу для самого что ни на есть традиционного водопадного стиля разработки, так и в качестве гибкого процесса [9].

В основе RUP лежат следующие принципы [10]:

- Ранняя идентификация и непрерывное (до окончания проекта) устранение основных рисков.
- Концентрация на выполнении требований заказчиков к исполняемой программе (анализ и построение модели прецедентов (вариантов использования)).
- Ожидание изменений в требованиях, проектных решениях и реализации в процессе разработки.
- Компонентная архитектура, реализуемая и тестируемая на ранних стадиях проекта.
- Постоянное обеспечение качества на всех этапах разработки проекта (продукта).
- Работа над проектом в сплочённой команде, ключевая роль в которой принадлежит архитекторам.

Для определения целей и границ проекта в подходе RUP используется документ «Видение проекта» (Vision) — это краткое описание сути будущего продукта [11]. В нем кратко описывается, что это за продукт, каковы цели и задачи его создания, кто его пользователи и каковы основные возможности будущей системы. Видение позволяет с помощью нескольких абзацев ознакомить с сутью проекта любое заинтересованное лицо, собрать основные требования и дать общее представление о целях, которые поставлены перед продуктом.

Видение проекта разрабатывается на основе анализа потребностей бизнеса. Главная функция документа – подтверждение и согласование единого видения целей, задач и результатов всеми участниками проекта. Видение определяет *что и зачем* делается в проекте. Видение проекта это ключевой документ, который используется для принятия решений в ходе всего проекта, а также на фазе приемки — для подтверждения результата[9].

В том или ином виде видение присутствует практически во всех методиках, иногда называясь «Концепция проекта» [12]. Также в различных методиках различается детальность проработки и степень формализации видения проекта, но при этом ответы на основные вопросы присутствуют в каждом варианте документа.

Шаблон видения проекта

Видение проекта в концепции RUP строится, как документ, последовательно детализирующий потребителей, их потребности, цели и границы проекта, таким образом, чтобы читатель, не знакомый с предметной областью, читая документ, постепенно погружался в нее. Поэтому каждый последующий пункт детализирует информацию предыдущих пунктов. Рассмотрим, более подробно содержание документа Видения проекта в методике RUP[13].

Введение

Во введении описывается цель создания документа и принципы его использования. Задача этого пункта – дать читателю документа возможность понять, что описано в этом документе и для чего предназначена описываемая система:

1. Цель документа – формальная цель разработки документа. Обыкновенно, целью документа объявляется сбор, анализ и определение высокоуровневых потребностей заинтересованных лиц и конечных пользователей системы, а также характеристик системы необходимых для удовлетворения этих потребностей. Детали того, как система отвечает данным потребностям, записываются в модели прецедентов и, возможно, других документах спецификации.
2. Область действия документа – краткое описание границ этого документа. Указывается продукт или приложение, которое будет создаваться или изменяться. Формулируется цель приложения, версии и новые предоставляемые функции в развитие существующей системы.
3. Определения, аббревиатуры и сокращения – перечень всех необходимых для понимания сокращений либо ссылка на отдельный документ глоссария.
4. Ссылки – в этом разделе приводится полный список источников, на которые существуют ссылки в Видении или к которым можно обратиться за справкой. Информация может быть представлена ссылкой на приложение к видению или ссылкой другой документ.
5. Обзор документа – здесь описывается, остальное содержимое Видения и способ структурирования информации в нем. Целью данного раздела является дать читателю возможность быстро сориентироваться в структуре документа.

Позиционирование

В данном разделе содержится описание профиля потенциальных пользователей продукта и основных проблем, которые позволяет решить разработка этого продукта. Этот раздел не используется для формулировки конкретных требований – в нем только содержатся

обоснования, почему возникла необходимость разработки данного продукта и почему были выдвинуты те или иные требования.

1. Необходимость в продукте – в данном пункте кратко описывается: почему данный продукт будет востребован потребителем и как возникла возможность для разработки продукта.
2. Постановка проблемы – это один из самых важных пунктов, в нем описываются проблемы, вставшие перед пользователями, которые может решить разрабатываемый продукт. Явная и четкая формулировка проблем пользователя решаемых продуктом, помогает гарантировать наличие потребителя у создаваемого продукта, обеспечить достаточное число потребителей, избежать реализации не нужного функционала, реализовать весь необходимый функционал. При этом для каждой проблемы указывается:
 - a. Формулировка проблемы.
 - b. Кого она затрагивает.
 - c. Как она появилась.
 - d. Как поможет ее успешное решение.
3. Формулировка позиции продукта – здесь описывается целевая аудитория, разрабатываемый продукт, его конкуренты и ключевые отличия от них. В данном пункте проводится обобщенный анализ конкурентных преимуществ нового решения и дается общее определение на высоком уровне абстракции особого положения, которое продукт должен занять в линейке решений для данной предметной области. Для каждой целевой группы указывается следующая информация:
 - a. Группа потребителей
 - b. Описание задач, которые эта группа будет решает.
 - c. Категория нового продукта.
 - d. Ключевая выгода – почему новый продукт будут покупать.
 - e. Перечисление основных конкурентов.
 - f. Список новых возможностей и характеристик, которые новый продукт предлагает этой группе потребителей.

Описание заинтересованных сторон и пользователей

Одной из наиболее часто встречающихся и критичных проблем в разработке программного обеспечения является неявное или неясное обозначение круга потребителей продукта и их интересов. Зачастую это приводит к тому, что даже у качественно сделанного продукта оказывается недостаточная аудитория, либо продукт покрывает, только часть нужд

пользователей или не обладает нужными характеристиками. В любом из этих случаев, новый продукт не сможет получить достаточного распространения на рынке и возможно не окупит затраты на разработку и запуск продукта. Поэтому в данном разделе важно внимательно составить список заинтересованных лиц, перечислив их всех, и сформулировать их потребности. При этом заинтересованные лица должны корректно представлять интересы конечных пользователей.

1. Состояние рынка – в данном пункте описываются ключевые показатели рынка сбыта, важные для разработки продукта. При этом необходимо оценить размер рынка и его потенциальный рост, указав число возможных пользователей, либо возможного дохода, например на основе оценки текущих расходов пользователей на решение задач, которые решает новый продукт. Кроме того, следует соотнести новый продукт с целями самой организации разработчика (особенно это важно при разработке для внутренних заказчиков). Для этого надо описать:
 - a. Как продукт связан с использованием новых технологий.
 - b. Как повлияет выход нового продукта на репутацию организации.
 - c. Насколько продукт соответствует стратегическим целям компании.
2. Описание заинтересованных лиц – здесь перечисляются все заинтересованные лица, которые будут использовать систему или с которыми она будет связана. Заинтересованным в проекте является лицо, удовлетворение интересов которого повышает ценность продукта. При этом под понятием «заинтересованного лица» понимается не конкретные люди, работающие с системой, а функциональные роли, которые они играют в этот момент. Для каждого заинтересованного лица, необходимо указать роль в проекте, потребности и их важность, критерий успеха удовлетворения каждой потребности, текущие и предлагаемые способы удовлетворения потребностей. Можно выделить три категории заинтересованных лиц:
 - a. Первая категория заинтересованных лиц – это пользователи системы. Их потребности легко учесть, поскольку они будут непосредственно привлекаться к определению и использованию системы.
 - b. Вторую категорию составляют не прямые пользователи, а также те, на кого воздействуют только последствия разработки. Этих заинтересованных лиц можно найти в соответствующей в "окрестностях" среды конкретного приложения. Например, это могут быть инвесторы проекта, владельцы организации или субподрядчики по разработке системы.
 - c. Кроме людей, также необходимо учесть интересы сопряженных технических систем и внешних организаций, например государственных структур. Такой

список составляется для обеспечения учета ключевых интересов заинтересованных лиц при планировании функциональности продукта.

3. Описание пользователей – должны быть перечислены все идентифицированные на данный момент пользователи. Отдельно перечисляются все пользователи, являющиеся заинтересованными лицами. Для каждого пользователя, не являющегося заинтересованным лицом, указывается: заинтересованное лицо, представляющее его интересы, потребности и их важность, критерий успеха удовлетворения каждой потребности, текущие и предлагаемые способы удовлетворения потребностей.
4. Окружение пользователя – в данном пункте содержится описание рабочего окружения конечных пользователей, характеристики организации, в которой будет использоваться продукт, условия работы продукта. С одной стороны эти условия, в конечном счете, будут влиять на нефункциональные требования, которые будут сформулированы в техническом задании на разработку системы. С другой – они также детализируют целевую аудиторию продукта, что позволяет точнее оценить рынок сбыта и перспективы продукта. Для этого необходимо описать рабочую среду каждого типа пользователей, например:
 - a. Число людей вовлеченных в решение задачи.
 - b. Длительность выполнения задачи.
 - c. Длительность отдельных действий.
 - d. Ограничения, накладываемые окружением, например: работа в движении, обработка потока данных в реальном времени, критичность сбоев в работе, финансовая безопасность и т.д.
 - e. Платформы и операционные системы, используемые пользователями.
 - f. Сторонние приложения, с которыми продукт должен интегрироваться.
5. Альтернативы и конкуренты – здесь перечисляются все конкуренты разрабатываемого продукта. Часто можно услышать заявления, что у продукта нет конкурентов, однако практически у всех продуктов есть конкуренты, и кроме покупки другого продукта это может быть использование самодельного решения или отказ от решения проблемы. Например, для программы создания резервной копии, кроме продукта-конкурента, альтернативой может быть использование ручного копирования или отказ от создания копий в принципе. При этом для дальнейшей детализации потребностей заинтересованных лиц бывает полезно составить таблицу, в которой будет перечислены по приоритетам ключевые потребности заинтересованных лиц, и для каждой потребности указано:
 - a. Заинтересованных лиц, которые она затрагивает

- b. Приоритет потребности с точки зрения интересов бизнеса
- c. Текущие решение потребности, плюсы и минусы таких решений.
- d. Предлагаемое решение потребности, плюсы и минусы этого решения.

Обзор продукта

Данный раздел содержит высокоуровневый обзор возможностей продукта, интерфейсы с другими приложениями и конфигурацию систем.

1. Формула продукта – предлагается общее определение на высоком уровне абстракции особого положения, которое продукт должен занять в линейке решений для данной предметной области.
2. Перспективы продукта – указываются планы развития продукта. Прогнозируется время жизни продукта.
3. Набор основных возможностей – перечисляются основные выгоды заинтересованных лиц, а также характеристики системы, которые обеспечивают эти выгоды. Функции должны описываться на естественном языке пользователя. Эти функции составляют основу для определения продукта, а также управления проектом в целом. Впоследствии эти возможности будут раскрываться в техническом задании в функциональные требования к системе.
4. Предположения и зависимости – перечисляются все факторы, влияющие на характеристики системы, указанные в данном документе. Перечисляются все предположения, которые в случае изменения, приведут к изменению данного документа. А также перечисляются связанные технические системы для выявления необходимости сопряжения с ними.
5. Стоимость и цена продукта – указывается предполагаемая стоимость разработки и ценовая категория, по которой продукт будет поставляться конечному потребителю.
6. Лицензирование и установка – кратко описываются характеристики процедуры установки продукта и требования по его лицензированию.

Характеристики продукта

Характеристики продукта это высокоуровневые возможности, необходимые для удовлетворения потребностей пользователей продукта. Для каждой характеристики должен быть указан ее приоритет, статус (например, предложена, утверждена), сложность, риск. Функции, реализуемые продуктом, должны описываться на естественном языке пользователя. Эти функции составляют основу для определения продукта, а также управления проектом в целом. Впоследствии эти возможности будут раскрываться в техническом задании в функциональные требования к системе.

Ограничения для продукта

Описываются ограничения для продукта, дополняющие список проектных ограничений.

1. Требования к качеству – данный пункт определяет требования к производительности, надежности, удобству использования и другим характеристикам качества программного продукта.
2. Применяемые стандарты – здесь перечисляются стандарты, которым должен соответствовать продукт: коммуникационные стандарты, стандарты совместимости платформ, а также стандарты качества и безопасности.
3. Системные требования – определяются все системные требования, необходимые для поддержки приложения, операционные системы и сетевые платформы, конфигурации, память. Например, что среднее время отклика интерфейса программы при работе до 10 одновременных пользователей не должно превышать 10 сек.
4. Требования к производительности – в данном пункте перечисляются основные вопросы связанные с производительностью разрабатываемого продукта. К вопросам производительности относятся фактор нагрузки, создаваемой пользователями, надежность, время ответа системы. Например, что система должна работать без перезагрузки сервера не менее одного месяца и должна обрабатывать не менее 1Гб новых данных в месяц.
5. Требования к документации – здесь описывается, какую документацию необходимо разработать для поддержки успешного внедрения и эксплуатации продукта. Например, описывается цель и содержание руководства, рассматривается его желаемый объем, потребность в глоссарии. Указываются ограничения, связанные с форматированием и печатью.

Выводы

Исследование показало, что для эффективной фиксации целей, и управления границами проекта необходимо разрабатывать концепцию системы. Примером разработки такой концепции, может служить создание документа видения проекта в унифицированном процессе RUP. Данный документ позволяет задания цели проекта, зафиксировать его границы, четко определить пользователей системы, удовлетворить их ключевые потребности, позиционировать продукт на рынке, осуществлять долгосрочное планирование и контролировать объем реализуемой функциональности.

Список литературы

- [1] Краев В. Н. Ползунова Н. Н., *Исследование систем управления*. М.: Академический проект, 2004.
- [2] Либерзон В И, *Основы управления проектами*. М.: Нефтяник, 1997.
- [3] PMI, *Руководство к своду знаний по управлению проектами (Руководство РМВОК)*. М.: Институт Управления Проектами, 2004.
- [4] Карл И. Вигерс, *Разработка требований к программному обеспечению*. М.: Русская Редакция, 2004.
- [5] Ричард Ньютон, *Управление проектами от А до Я*. М.: Альпина Паблишерз, 2009.
- [6] Ньюэлл Майкл В., *Управление проектами для профессионалов. Руководство по подготовке к сдаче сертификационного экзамена*. М.: КУДИЦ-ПРЕСС, 2008.
- [7] David Rubinstein, "Standish Group Report: There's Less Development Chaos Today," *SD Times*, 1 марта 2007.
- [8] Standish Group International. (2010, март) Main page. [Online]. www.standishgroup.com
- [9] Архипенков С. (2010, Март) Лекции по управлению программными проектами. [Online]. http://www.arkhipenkov.ru/resources/sw_project_management.pdf
- [10] Поллис Гари, Огастин Лиз, Лоу Крис, and Мадхар Джас, *Разработка программных проектов на основе Rational Unified Process (RUP)*. М.: Бином-Пресс, 2005.
- [11] Филипп Крачтен Пер Кролл, *Rational Unified Process - это легко. Руководство по RUP для практиков*. М.: КУДИЦ-Образ, 2004.
- [12] Майкл С. В. Тернер, *Основы Microsoft Solution Framework*. Сп-Б.: Русская Редакция, Питер, 2008.
- [13] IBM. (2010, март) Шаблоны RUP для MS Word. [Online]. http://www.it4business.ru/docs/hub/rup_wd_tmpl.zip