



# Практика формирования требований в ИТ проектах от A до Я. Часть З Функции системы и Границы проекта

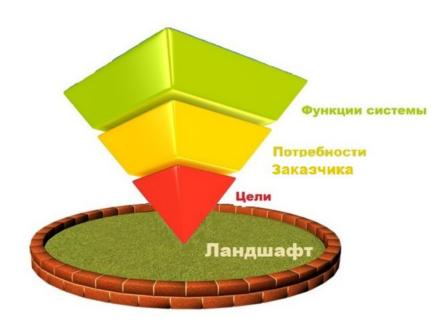
Промышленное программирование, Проектирование и рефакторинг, Программирование, Визуализация данных, Анализ и проектирование систем Tutorial

С частью 2 можно ознакомиться, перейдя по ссылке

# VI Определяем функции системы и границы проекта

Каждая модель ограничена в своих ответах, но нет ограничения на то, как и что моделирует модель, как нет ограничения на человеческую мысль

Дуглас Т. Росс



Когда основные потребности пользователей собраны и согласованы со всеми участниками, мы можем приступить к определению ключевых функций разрабатываемой системы, и уже на основании их примерно оценить стоимость и длительность проекта, направленного на создание конечного продукта. В результате этого процесса, как правило выясняется, что не хватает либо времен либо ресурсов, либо и того и другого для получения качественного результата в предусмотренные сроки. В этом случае, нам очен пригодится умение эффективно определять Границы проекта и управлять ими.

Цель данной группы работ: максимально полно определить набор функций, который должен выполнять целевой продукт, д удовлетворения выявленных потребностей заказчика. Отобрать те из них, которые, могут быть реализованы в рамках текупроекта.

Границы проекта (project scope) показывают, какая область конечного продукта будет реализована в текущем проекте. Другими словами, определяется черта между тем, что мы будем делать сейчас и тем, что отложим на потом или от чего вообще сможем отказаться. Для этого в арсенале команды должен быть инструмент, позволяющий не просто строить модели создаваемого продук помогающий наглядно очертить рамки автоматизируемых процессов, а также предоставлять возможность легко выносить процесс границу или включать их обратно. Это очень важно для осознания и более качественного планирования объемов работ. Подобный инструмент полезен не только для «борьбы» с непомерными желаниями заказчика, но и для маневров менеджмента со стороны разработчиков.

Для управления Границами проекта, как на начальных стадиях, так и в течение всего проекта, очень удобно использовать функциональное или процессное моделирование. Модели этого типа позволяют описывать события и последовательности исполня Бизнес процессов во времени.

Иногда, для определения границ, группа разработчиков пытается использовать не функции, а сущности предметной области. Хочу предостеречь Вас от такого подхода, так как он чреват следующими последствиями:

- в связи с недостаточной формализацией некоторые процессы могут быть реализованы не полностью, и это будет выявлено тол тогда, когда будут выпущены первые версии продукта;
- некоторые автоматизированные процессы не будут использованы в связи с тем, что связанные с ними процессы в проекте не реализованы;
- часть объектов системы не будет использоваться, потому что нет процессов, обрабатывающих их.

На рисунке 6.1 изображен процесс формализации требований к целевой системе, дополненный подпроцессом определения функц которые она должна выполнять.

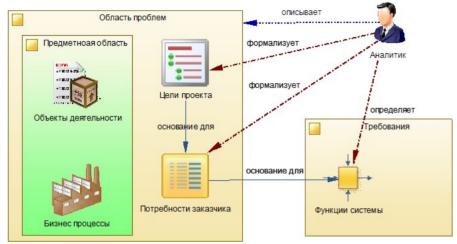


Рисунок 6.1 — модель процесса определения функций

#### 1. Используем нотацию IDEFO для определения функций системы и границ проекта

Наиболее удобной методикой функционального моделирования, с точки зрения определения границ проекта, на мой взгляд являет "старая добрая" методология проектирования SADT, использующая иерархическую декомпозицию сверху вниз. Применение диагр IDEF0 имеет следующие преимущества перед аналогами:

- Декомпозиция при анализе автоматизируемых процессов производится сверху вниз от одного самого высокоуровневого к составляющим его подпроцессам. Последовательное разветвление процессов с уточнением их слой за слоем, позволяет с одностороны, не упустить важные деловые процессы из поля зрения команды, а с другой работать всегда с одним выбранным узлом содержащим небольшое количество элементов;
- Способ моделирования, при котором за основу берутся входящие потоки данных и управляющие на них воздействия, позволяе максимально точно и полно выявить все вложенные в процесс подпроцессы, обрабатывающие эти входящие данные и предостна выходе процесса результат исходящий поток;
- Обязательное для каждого результата процесса (исходящего потока) сопоставление другого процесса потребляющего его н входе (как входящий поток), позволяет не упустить функции в цепочке обработки и преобразования данных.

При помощи стандарта IDEF0 строится модель, описывающая основные функции, выполняемые системой и их взаимодействие в ви информационных потоков, в том числе с внешней средой. Таким образом на диаграммах IDEF0 легко читаются границы системы и окружение. Еще одно достоинство диаграмм этого типа, как было упомянуто выше, заключается в том, что вместо разработки одно большой, громоздкой модели, поэтапно создается несколько небольших, относительно простых моделей, вложенных одна в другунматрешки.

Таким образом, структура сложного процесса представляется в виде абстракции функций высокого уровня, которая раскладывает более детальные процессы, увеличивая степень точности, слой за слоем.

Этот вид моделирования позволит нам также определить процессы, которые были выпущены из поля зрения на предыдущих этапа

Если на этапе процессного моделирования будут обнаружены процессы, которые не описаны на стадии сбора информации, мы до. будем снова вернуться к этапу формирования Пользовательских историй и восполнить пробел. А теперь, чтобы все это нагроможд информации лучше улеглось в сознании, давайте рассмотрим конкретный пример.

## 2. Пример описания функции "Управление требованиями"

Хочу напомнить основные постулаты стандарта IDEFO. Графическую конструкцию стандарта составляют: понятие «Работа» (Activit обозначения действия, представленная в виде блока; четыре вида интерфейса: «Вход» (Input), «Выход» (Output), «Управление» (Control) и «Механизм» (Mechanism), представленный в виде дуг. Левая сторона блока предназначена для входов, верхняя — для управления, правая — для выходов, нижняя — для механизмов. Для более подробного изучения этой темы обратитесь к [2].

На первом шаге моделирования необходимо определить все потоки данных, поступающих в систему из вне (входные сигналы). В н случае это:

- Цели проекта;
- Участники и заинтересованные лица проекта;
- Потребности участников к функциональности целевого продукта проекта;
- Методики разработки требований;

Далее определяем все "продукты", генерируемые системой для внешнего использования (выходные сигналы):

- Артефакты проекта, в том числе Требования, Модели, Планы и т. д.;
- Целевой продукт проекта.

Описанные выше потоки данных и глобальная абстрактная функция системы, обрабатывающая эти потоки, отображены на Диаграм см. Рис. 6.2.

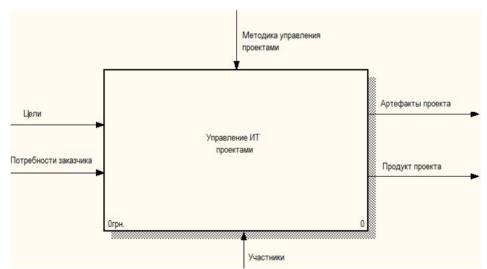


Рис. 6.2 – Функциональная модель системы Управления требованиями верхнего уровня

Проваливаясь во внутрь функции (блока «Работа») мы попадаем на следующий уровень абстракции функциональности системы. Вначале мы видим только потоки данных, выявленные на предыдущем этапе (уровне), см. рис. 6.3.

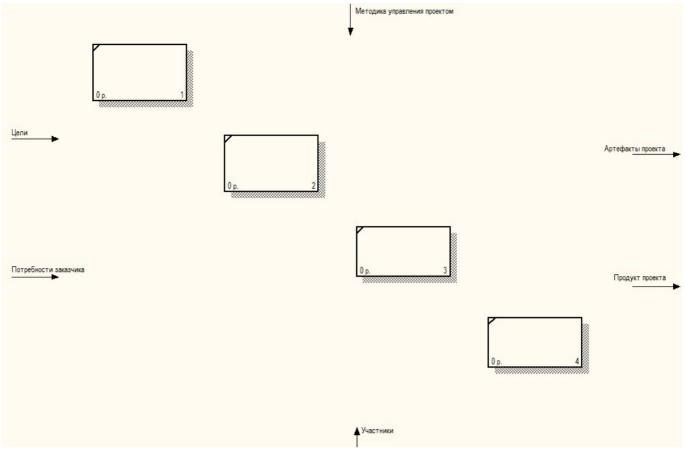


Рис.6.3 – Начало работы по детализации функции Управление ИТ проектами

Для каждого такого потока необходимо определить процесс, обрабатывающий (преобразующий) или использующий его, добавляя диаграмме новые блоки «Работ» (функции), связанные с ним.

Таким образом, при каждом последующем шаге (проваливаясь внутрь процесса) необходимо: каждому потоку данных, выявленном предшествующем уровне, сопоставить функцию (процесс) для его обработки. В результате такого моделирования, мы получим перечень функций, детализирующих вышестоящий абстрактный процесс см. Рис. 6.3.

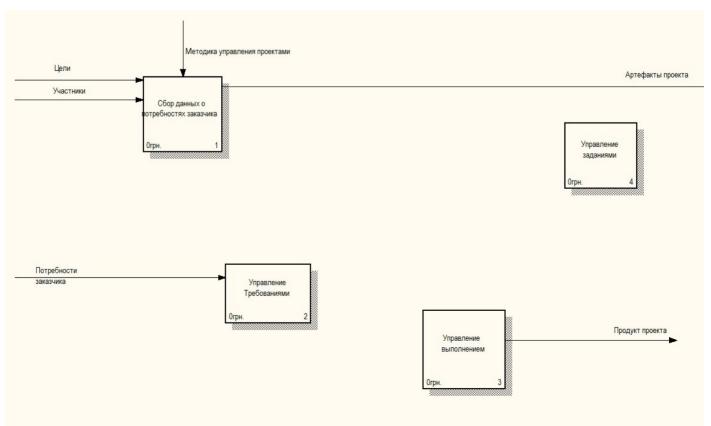


Рис. 6.4 — Определение подпроцессов для функции Управление ИТ проектами

В нашем проекте, согласно изложенным в главе IV целям и выявленным на первом этапе потокам данных, необходимо автоматизировать следующую группу процессов:

- Сбор данных о потребностях заказчика, предъявляемых к функциональности продукта;
- Учет системных требований к целевому продукту;
- Управление пулом работ, направленных на реализацию системных требований и, как результат на создание целевого продуг
- Учет выполнения работ по реализации системных требований и, как результат продвижение проекта к цели;

Один процесс «Управление заданиями» у нас не получает данные из вне и не предоставляет ничего во внешнее окружение. Получ — он вспомогательный процесс, обеспечивающий функционирование других. Он пока стоит обособлено см. рис. 6.4.

Поскольку большинство процессов логически связаны между собой, необходимо установить все потоки данных, обеспечивающие связи. Таким образом от функции к функции на диаграмме появляются дуги. Позже, каждая такая дуга (связь), входящая в блок «Работы», должна будет, при моделировании следующего вложенного уровня, "получить" свой процесс обработки.

Пример модели этих функций с уже установленными взаимосвязями отображены на Диаграмме см. Рис. 6.5.

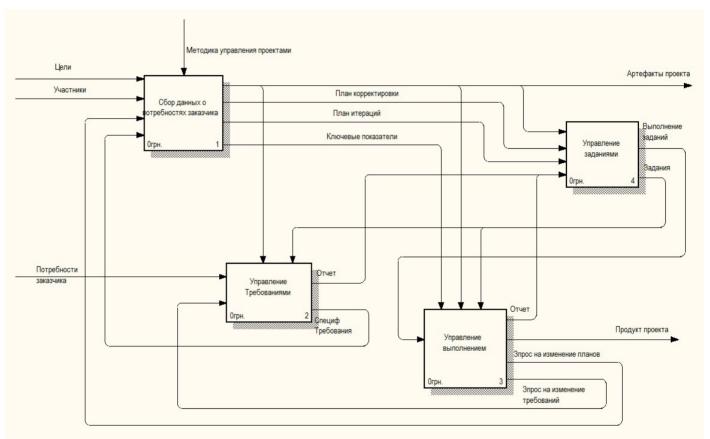


Рис.6.5 – Функциональная модель системы Управления требованиями

Если такие диаграммы используют в документе необходимо сопровождать их подробным описанием. Дальше приведен пример описания:

На рисунке видно, что функциональную архитектуру нашего проекта представлена в виде четырех доменов:

- 1. Сбор потребностей заказчика. Группа процессов по учету и анализу целей разработки продукта, основных сценариев его использования и действующих лиц пользователей продукта.
- 2. Управление спецификациями требований. Процессы формирования функциональных требований на основании выявленных потребностей заказчика и будущих пользователей целевого продукта.
- 3. Управление формированием заданий. Процессы формирования заданий, направленные на удовлетворение выявленных потребностей заказчика, путем реализации функциональных требований. Фиксация результатов, при выполнении заданий исполнителями, путем изменения состояния требований, по которым оно сформировано.
- 4. Управление выполнением. Процессы, выполняющие мониторинг и фиксацию приращения функционала, позволяющего реализовывать потребности пользователей в целевом продукте.

В первый функциональный блок "А1", в качестве входящих параметров, направим данные о целях и пожеланиях заказчика, заинтересованных лицах и т.п. Данные поступают из внешнего окружения системы в виде документов –заявок на разработку или в устной форме. Цель процесса — преобразовать их в информацию, пригодную для передачи в блок "А2" — «Управление спецификациями требований». Результат деятельности первого блока, в виде Пользовательских историй и отчетов, также пригоде для внешнего использования и доступен внешним системам и пользователям в виде печатных форм или экспорта данных в файлы. Поэтому стрелка из блока разделяется и выходит еще и за границу диаграммы, в вышестоящую функцию.

Из диаграммы видно, что блок "А1" имеет обратную связь от блока "А2", в виде управляющей информации, доводящей о степени реализации требований, сформированных по Пользовательским историям (дуга «Отчет о реализации требований»). Эта связь позволяет отследить ход и полноту реализации Пользовательской истории в конечном продукте по цепочке, через спецификации требований.

Второй блок, как показано на схеме, получает на вход обработанные пожелания заказчика в виде Пользовательских историй, потребностей пользователей и т.п. уже в формализованном виде. На основании этих данных в нем формируются функциональные требования к разрабатываемому продукту и формализуются в виде спецификаций требований. Дальше эти спецификации передак четвертый блок "А4", отвечающий за выставление задания исполнителям на их реализацию. Из диаграммы видно, что задания могу выставляться и на выполнение работ с требованиями (дуга «Задания», входящая в качестве управления в блок "А2"). Обратите внимание на то, что во второй функциональный блок возвращаются данные об исполнении заданий по спецификациям, что позволя этом домене определить приращение функциональности, полученное в ходе разработки.

В третий блок направим из блока "А2", в качестве управляющих инструкций, ключевые показатели спецификаций. На основании их можно определить степень достижения заданной функциональности, используя отчет о выполнении задний, выставленных по этим спецификациям исполнителям. Отчет поступает в виде входящих параметров из блока "А4".

Несмотря на все мои старания, этот раздел получился тяжелым и утомительным. Но для понимания концепции важно было разобраться, как это все работает. Далее, будем описывать вложенные функции уже не так подробно.

## 3. Пример описания функции "Сбор потребностей заказчика"

Продолжаем декомпозицию выявленных функций, раскладывая каждый домен на более мелкие, детальные функции. Для этого используем наши Пользовательские истории (речь о них шла в предыдущей части публикации). При их описании будем определять насколько полно мы «покрываем» ими потребности заказчика.

Заглянем в блок A1 на рисунке 6.4, представляющий домен «Сбор потребностей заказчика». Его детализация показана на рисунке Все потоки данных, которые входили в блок A1 на рисунке 6.4, соответственно попали и в детализирующую его диаграмму см. рис.

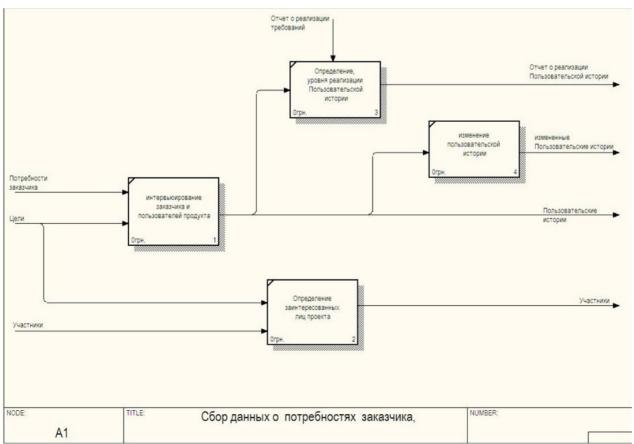


Рис. 6.6 – Схема домена Сбор потребностей заказчика

Функционально домен мы разделили на четыре процесса:

- 1. Интервьюирование заказчиков и пользователей целевого продукта. Процесс фиксации Пользовательской истории, в том числе цели. Выполняется проверка наличия подобных историй, зафиксированных ранее. Определяются противоречащие друг другу сценарии. Этот блок должен "покрывать" Пользовательские истории US01, US02
- 2. Изменение Пользовательской истории. Процесс управления изменениями в описании потребностей заказчика продукта, включинициацию процесса изменения связанных с ней системных требований. Этот блок должен "покрывать" Пользовательские ист US02.
- 3. Фиксация заинтересованных лиц проекта. Выявляются все лица, так или иначе связанные с проектом. Этот процесс должен "покрывать" Пользовательские истории US04
- 4. Определение, уровня реализации Пользовательской истории. Осуществляется мониторинг системных требований, связанных с Пользовательской историей, определяются их текущие состояния и уровень реализации в рамках конечного продукта. Этот бло должен "покрывать" Пользовательские истории US11. Блок, в качеств управляющего интерфейса, получает «Отчет о реализаци требования», поступающий из блока "A2" «Управление спецификациями требований», на основании которого рассчитывает уровень выполнения.

## 4. Пример описания функции "Управления спецификациями требований"

Следующее уточнение произведем с доменом «Управление спецификациями требований проекта» (A2). На рисунке 6.7 изображен диаграмма этой модели.

Функционально домен делим на четыре процесса:

- 1. Определение бизнес-процессов. Выполняется разработка функциональной архитектуры целевого продукта, в том числе, определяется важность и приоритетность автоматизируемых процессов управление границами проекта. Этот блок должен "покрывать" Пользовательские истории US05, US06. В качестве управляющего интерфейса блок получает «Задания», поступак из блока "А4" «Управление заданиями» и инициирующие выполнение работ.
- 2. Разработка спецификаций требований. Разрабатываются нотации логической и физической структуры продукта, формируется контент проекта с детальным описанием требований к целевому продукту, на основе которого будет осуществляться взаимодействие всех участников. Этот блок должен "покрывать" Пользовательские истории US07. В качестве управляющего воздействия блок получает описание стандартов, которым должны соответствовать спецификации.
- 3. Управление изменениями требований. Фиксируются заявки на изменение ранее утвержденных характеристик целевого продук том числе инициируется процесс формирования заданий на реализацию этих изменений. Этот блок должен "покрывать" Пользовательские истории US10.
- 4. Управление состоянием требований. Осуществляется мониторинг процесса продвижения проекта, в части реализации требова Эта функция должна "покрывать" Пользовательские истории US11. В качестве управляющего интерфейса блок получает отчеты «Выполнение Заданий» поступающие из блока "А4" «Управление заданиями», для определения текущего состояния и уровреализации проекта.

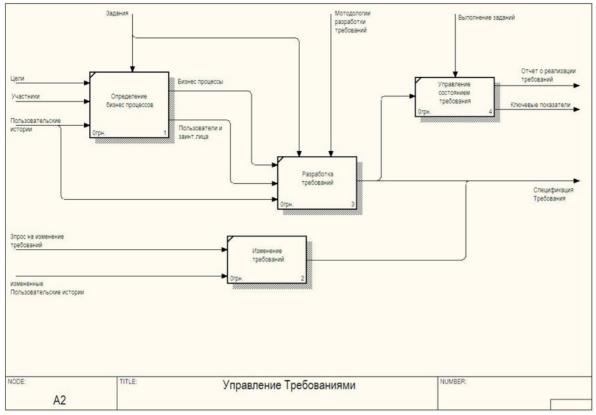


Рис. 6.7 – Схема домена Управления спецификациями требований проекта

#### 5. Пример описания функции "Управление заданиями"

На рисунке 6.8 изображена диаграмма, представляющая домен Управления Заданиями проекта (А4). Функционально домен мы разделили на четыре процесса:

- 1. Формирований заданий на основании спецификаций требования. Формируются задания на реализацию требования в том числ проектирование, разработку, тестирование, развертывание, пилотное внедрение. Этот блок должен "покрывать" Пользователь истории US08.
- 2. Формирование заданий на итерацию. Осуществляется управление выставлением заданий по плану итерации. Эти требования в из стандартного процесса управления проектом с использованием итерационного подхода.
- 3. Формирование заданий при наступлении риска. Осуществляется управление выставлением заданий по плану устранения последствий риска.
- 4. Учет выполнения заданий. Формируется отчет о выполнении задания. Происходит изменение статуса объектов проекта по результатам выполнения заданий. Этот процесс должен "покрывать" Пользовательские истории US09.

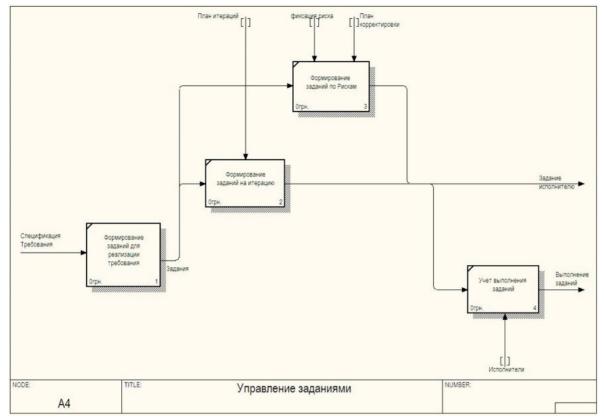


Рис. 6.8 – Схема домена Управления Заданиями проекта

При моделировании этого домена, у нас появились процессы, которым мы не можем сопоставить ни одну из Пользовательских ист Поэтому восполним пробел. Для этого, мы должны вынести проблему на совместное обсуждение команды с заказчиками и сформировать новые Пользовательские истории.

Для процесса 5.2 опишем следующую Пользовательскую историю:

US12. На основании плана итераций проекта отобрать пул задач, которые необходимо реализовать на данном этапе. Цель: Получить список задач для реализации в текущей итерации проекта.

Процесс 5.3 затрагивает несколько Пользовательских историй:

US13. Подготовить требования, которые необходимо будет реализовать, при наступлении прогнозируемого риска. Цель: Выработать альтернативные решения для реализации потребности заказчика, при возникновении проблем.

В случае наступления риска, выполняется пользовательская история US8.

## 6. Пример описания функции "Управление выполнением"

На рисунке 6.9 изображена диаграмма, представляющая домен Управления выполнения проекта. Реализация этого домена немног выходит за рамки нашего проекта, но тесно с ним связана. Поэтому рассмотрим и его.

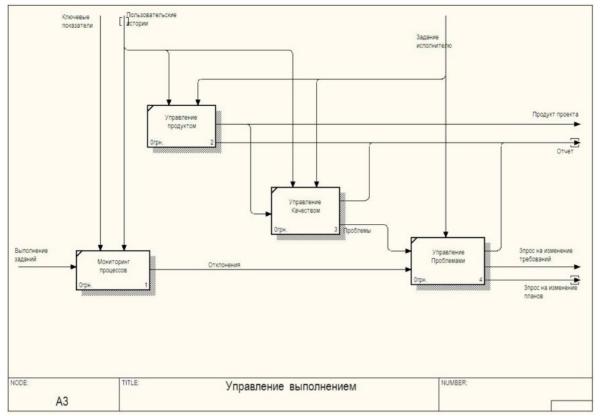


Рис. 6.9 — Схема домен Управления выполнения проекта

Функционально домен мы разделили на четыре процесса:

- 1. Управление продуктом проекта. Фиксируется выпуск продукта. Формируются отчеты о выпуске
- 2. Управление качеством. Производится анализ исправления замечаний предыдущей проверки. Определяется соответствие Выпу требованиям. Оформляется отчет о контроле качества. Выявляются новые проблемы.
- 3. Управление проблемами. Реагирование на отклонения в ходе выполнения процессов. Выполняется приоритезация ошибок, формируются запросы на изменение.
- 4. Мониторинг и анализ процессов. Выполняется сравнение плановых значений ключевых показателей с реальными. Определяют значения отклонения. Отслеживаются индикаторы триггеры («признаки рисков», «симптомы риска»), указывающие на то, ч события риска произошли или произойдут в ближайшее время. Этот процесс должен "покрывать" Пользовательские истории L

#### 7. Подведем итоги процесса определения функций системы и границ проекта

Таким образом, в несколько проходов, слой за слоем уточняется и детализируется функциональная модель разрабатываемого прои определяются границы его контуров. В результате этой деятельности мы получаем подробную процессную модель, которую необходимо воплотить в жизнь. Как видно из диаграмм, помимо перечня автоматизируемых функций, также обозначены все информационные потоки, связывающие их.

Теперь если мы хотим вынести какую-либо функцию за рамки проекта или его этапа, мы можем проанализировать зависимости и избежать «провисания» остальных функций в линейке технологического процесса.

В итоге, опираясь на разработанные нами диаграммы, мы можем на первом этапе вынести за рамки проекта функции группы «АЗ Управления выполнением», а также функции «А4.2 Формирование заданий на итерацию» и «А4.3 Формирование заданий при наступлении риска». Из диаграммы видно, что в результате система лишится потока данных — «задания исполнителям», обусловленных работами по нивелированию рисков.

Теперь, всякий раз, когда заказчик предлагает Вам включить в продукт новую функциональность, Вы должны сначала зафиксирова изменения на диаграмме Процессов (функций), определить степень изменений и их влияний для системы в целом.

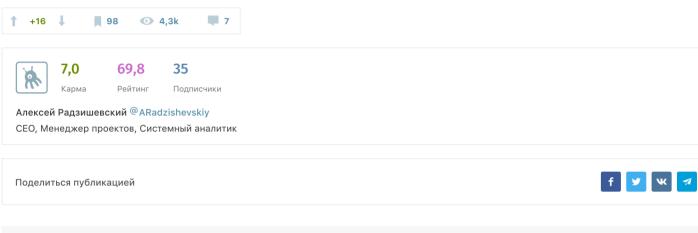
В следующей части мы будем детализировать процессы, включенные в рамки системы ссылка

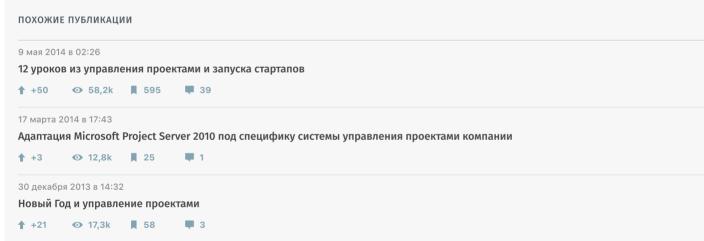
https://habrahabr.ru/post/336950/.com

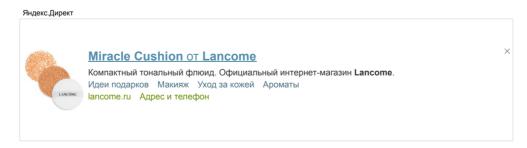
Список литературы

- 1. Якобсон А., Буч Г., Рамбо Дж. «Унифицированный процесс разработки ПО» (2004)
- 2. Дэвид А. Марк и Клемент МакГоуэн «Методология структурного анализа и проектирования SADT»
- 3. Коберн «Современные методы описания функциональных требований» (2002)
- 4. Леффингуэлл Дин, Уидрих Дон «Принципы работы с требованиями к ПО» (2002)
- 5. Карл И. Вигерс «Разработка требований к программному обеспечению» (2002)
- 6. Элизабет Халл, Кен Джексон, Джереми Дик «Разработка и управление требованиями практическое руководство пользователей» (2005)
- 7. Скотт Амблер «Гибкие технологии: экстремальное программирование и унифицированный процесс разработки» (2005)
- 8. Гринфилд Джек, Кейн Шорт «Фабрики разработки программ» (2007)
- 9. Алистэр Коуберн «Каждому проекту своя методология»
- 10. Вольфсон Борис- «Гибкие методологии разработки»
- 11. Лешек А. «Анализ требований и проектирование систем»
- 12. Фримен Эрик, Фримен Элизабет «Паттерны проектирования» (2011)
- 13. Эванс Эрик «Предметно-ориентированное Проектирование» (2011)
- 14. ГОСТ 34.602-89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание создание автоматизированной системы»

Метки: формирование требований, проектирование, требования, управление проектами, менеджмент продукта



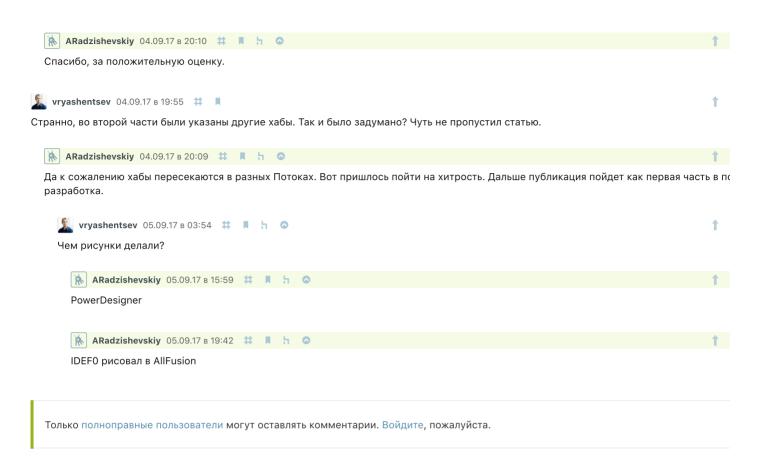


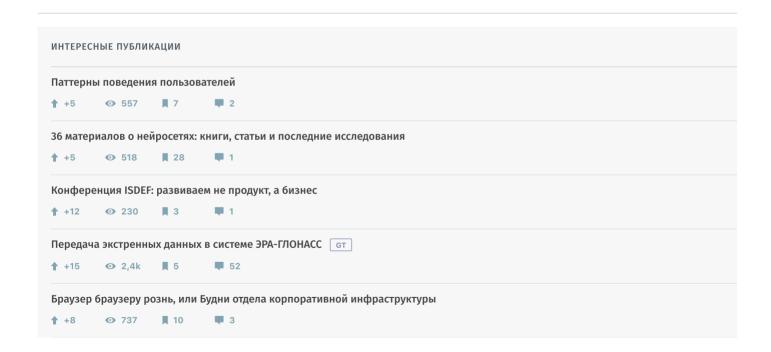


# Комментарии 7

**Solution** WizardryIB 04.09.17 в 14:29 # ■

- - 1. Спасибо за интересный подход и глубокую декомпозицию плана управления проектом.
  - 2. на рис. 6.5 напутано с потоками данных (ключевые показатели, планы, ...), но не суть.
  - 3. Некоторые процессы лучше вынести в предпроект.





Аккаунт	Разделы	Информация	Услуги	Приложения
Войти	Публикации	О сайте	Реклама	<b>З</b> агрузите в <b>ДОСТУПНО В</b>
Регистрация	Хабы	Правила	Тарифы	App Store Google
	Компании	Помощь	Контент	
	Пользователи	Соглашение	Семинары	

07.09.2017