

CPRE

Certified Professional for Requirements Engineering

Базовый уровень

Учебная программа

Условия использования

1. Физические лица и образовательные организации могут использовать эту программу в качестве основы для проведения семинаров при условии соблюдения авторского права и включения информации о нем в материалы семинара. Каждому, кто использует эту программу в рекламных целях, необходимо получить на это письменное согласие IREB e.V..
2. Каждое физическое лицо или группа лиц может использовать эту программу как основу для статей, книг или иных публикаций при условии соблюдения авторского права авторов и организации IREB и упоминании их как источников и владельцев документа в этих публикациях.

© IREB e.V.

Все права защищены. Никакая часть данной публикации не может быть воспроизведена, сохранена в информационно-поисковой системе или передана в любой форме и любыми средствами, электронными, механическими, фотокопированием, записью или иным способом без предварительного письменного разрешения либо авторов, либо организации IREB.

Благодарности

Эта программа была первоначально создана в 2007 году авторами: Карол Фрюхауф, Эммерих Фухс, Мартин Глинз, Райнер Грау, Колин Худ, Франк Худек, Питер Хрушка, Барбара Пич, Клаус Пол и Крис Рупп. При поддержке соавторов: Иан Алексардер, Джозеф Брудер, Самуел Фрикер, Гюнтер Халманс, Питер Йаесчке, Свен Краузе, Стеффан Лентз, Урте Паутз, Сьюзан Робертсон, Дирк Шюпферлинг, Йоханнес Стауб, Торстен Вайер и Джой Битти.

Версия 3.0 является крупной редакцией от авторов: Стен Бюне, Мартин Глинз, Ганс ван Люнхуд и Стефан Стаал. При поддержке соавторов: Карол Фрюхауф, Райнер Грау, Ким Лауэнрот, Крис Рупп и Камилла Салинези.

В рамках этой редакции предоставлены отзывы от: Ксавьер Франк, Карол Фрюхауф, Райнер Грау, Франк Худек и Торстен Вейер. Дополнительные отзывы предоставили: Вим Декутер и Ганс-Йорг Штеффе.

Рецензии были сделаны Кристофом Эбертом, Барбарой Пич и Крисом Руппом.

Утверждено к выпуску 22 июля 2020 года Советом IREB по рекомендации Ксавье Франка и Франка Худека.

Перевод на русский язык выполнен Анной Шведовой и Марией Гузенко

Мы благодарим всех за участие.

Авторское право © 2007–2022 на учебную программу принадлежит авторам, перечисленным выше. Имущественные права были переданы Международной организации проектирования требований. (IREB), Карлсруэ, Германия.

Преамбула

Летом 2017 года мы провели исследование на предмет актуальности существующей сертификации на Сертифицированного специалиста проектирования требований (CPRE) базового уровня (версия 2.2). Целью исследования было получение отзывов о практической маркетинговой актуальности сертификации от поставщиков услуг по обучению, а также от CPRE сертифицированных специалистов, практикующих в качестве проектировщиков

требований [MFeA2019]. Исследование показало, что существующая учебная программа CPRE базового уровня версии 2.2 в основном все еще удовлетворяет наиболее важным потребностям рынка и предоставляет кандидатам соответствующие знания о ПТ. Тем не менее, мы получили отзывы, что некоторые методы больше не используются на практике, в то время как другие отсутствуют в контексте итеративной и инкрементальной разработки. Эта обратная связь соответствовала собственным представлениям IREB об изменениях в области проектирования требований (ПТ). Поэтому мы решили провести крупную ревизию учебной программы CPRE базового уровня, удалив устаревшее содержимое и добавив новые элементы. Пересмотренная учебная программа отражает состояние современного подхода к ПТ, охватывая как водопадные, так и гибкие подходы к разработке и управлению требованиями.

Ожидается, что кандидаты, претендующие на сертификацию CPRE по этой программе, будут обладать определенными базовыми знаниями в области разработки систем с использованием водопадных и гибких подходов.

Цель документа

Данная учебная программа определяет базовый уровень сертификации "Сертифицированный специалист в области проектирования требований", установленный Международной организацией проектирования требований (IREB). Учебная программа предоставляет образовательным организациям основу для создания своих учебных материалов. Данная программа может быть использована студентами для подготовки к экзамену.

Содержание учебной программы

Базовый уровень учитывает потребности всех людей, вовлеченных в процесс проектирования требований. Сюда входят люди, выполняющие такие функции, как проектировщик требований, бизнес-аналитик, системный аналитик, владелец продукта или менеджер продукта, разработчик, менеджер проекта или ИТ менеджер, или эксперт области.

В данной учебной программе и соответствующем руководстве используется аббревиатура "ПТ" для проектирования требований.

Область применения

Базовый уровень CPRE передает основные принципы, которые в равной степени применимы к любому типу системы (например, к мобильным приложениям, информационным системам или киберфизическим системам). Кроме того, базовый уровень CPRE не предполагает какого-либо конкретного процесса разработки и не ориентирован на конкретную прикладную область. Учебные заведения могут предлагать обучение, ориентированное на конкретные типы систем, процессов или областей применения, при условии полного охвата учебных модулей данной учебной программы.

Уровень детализации

Уровень детализации этой программы позволяет проводить обучение и экзаменование, соответствующие международному уровню. Для достижения этой цели учебная программа содержит следующее:

- Общие задачи обучения
- Материалы с описанием образовательных задач

- Ссылки на дополнительную литературу (в случае необходимости)

Задачи обучения/Когнитивные уровни знаний

Всем модулям и образовательным блокам данной программы присваивается когнитивный уровень. Используются следующие уровни:

- **У1: Знать** (описывать, перечислять, характеризовать, узнавать, называть, запоминать, ...) — запоминать или извлекать ранее изученный материал.
- **У2: Понимать** (объяснять, интерпретировать, дополнять, резюмировать, обосновывать, классифицировать, сравнивать, ...) — понимать/конструировать значение из заданного материала или ситуации.
- **У3: Применять** (уточнять, писать, проектировать, разрабатывать, внедрять, ...) — применять знания и навыки в данных ситуациях.

Более высокие уровни включают в себя более низкие. Обратите внимание, что все термины в глоссарии, которые обозначены как основополагающие, должны быть изучены (У1), даже если они прямо не упомянуты в целях образования. Глоссарий доступен для скачивания на домашней странице IREB по адресу <https://www.ireb.org/downloads/#cpre-glossary>

Структура учебной программы

Учебная программа состоит из семи основных глав. Одна глава посвящена одному учебному модулю (М). Основные заголовки глав содержат когнитивный уровень их глав, который является наивысшим уровнем их подразделов. Рекомендуется время обучения, которое определяет тот минимум, который должен быть посвящен этой главе. Тренинговые компании могут свободно выделять больше времени, но должны сохранять пропорции между М. Важные термины, используемые в главе, перечислены в начале главы.

Пример

М 4 Практика по разработке требований (У3)

Продолжительность: 4 часа 30 минут

Термины: Источник требований, граница системы, контекст системы, выявление требований, обсуждение требований, валидация требований, заинтересованное лицо, модель Кано, конфликт

Этот пример показывает, что глава 4 содержит образовательные цели на уровне У3, и четыре с половиной часа предназначены для изучения материала в этой главе.

Каждая глава содержит подразделы. Их заголовки также содержат когнитивный уровень их содержания.

Номер задачи обучения (30) содержит номер подраздела главы документа, где находится ее подробное объяснение. Нумерация показывает, к какому подразделу они относятся.

Например, образовательная задача ОЗ 4.2.1 описана в подразделе 4.2.

Порядок тем в учебной программе

Порядок глав в этом учебном плане представляет собой логический порядок тем. Однако темы не обязательно должны изучаться именно в таком порядке. Обучающие учреждения имеют право преподавать материал в любом порядке (включая чередование тем из разных М), который они считают подходящим в контексте своего обучения и который соответствует их дидактическим концепциям.

Экзамен

Данная учебная программа является основой для экзамена на сертификат базового уровня.



Вопрос экзамена может охватывать материал нескольких глав программы. Все разделы (с М1 до М7) учебной программы могут быть включены в экзамен.

Формат экзаменационных вопросов – множественный выбор.

Экзамены могут быть проведены сразу после курса обучения, а также независимо от курсов (например, в экзаменационном центре). Список лицензированных IREB сертификационных органов можно найти на сайте <https://www.ireb.org>.

История версий

Version	Date	Comment
2.2.0	19 апреля 2017	Первоначальная версия (основано на английской версии 2.2)
2.2.1	31 августа 2017	Корректировка незначительных ошибок
3.0.0	1 апреля 2021	Второе масштабное изменение, отражающее состояние современного процесса ПТ, включающего в себя и водопадный и гибкий подход к выявлению и управлению требованиями.
3.0.1	1 января 2024	Новый корпоративный дизайн
3.0.2	26 февраля 2024	Корректировка макета для нового корпоративного дизайна

Оглавление

Оглавление	7
1 Введение и обзор проектирования требований (Y2)	9
1.1 Проектирование требований: Что (Y1)	9
1.2 Проектирование требований: Зачем (Y2)	10
1.3 Проектирование требований: Где (Y2)	10
1.4 Проектирование требований: Как (Y1)	10
1.5 Роль и задачи проектировщика требований (Y1)	11
1.6 Что нужно знать о проектировании требований (Y1)	11
2 Основные принципы проектирования требований (Y2)	12
2.1 Обзор принципов (Y1)	12
2.2 Пояснение принципов (Y2)	12
3 Результаты деятельности и практики документирования (Y3) 17	
3.1 Результаты деятельности в проектировании требований (Y2)	18
3.1.1 Характеристики результатов деятельности (Y1)	18
3.1.2 Категории и уровни абстракции (Y2)	19
3.1.3 Уровень детализации (Y2)	20
3.1.4 Аспекты, которые должны быть учтены в результатах деятельности (Y1)	20
3.1.5 Общие требования к документации (Y1)	21
3.1.6 Планирование результатов деятельности, которые будут использоваться (Y1)....	21
3.2 Результаты деятельности на основе естественного языка (Y2)	21
3.3 Результаты деятельности на основе шаблона (Y3)	22
3.4 Результаты деятельности на основе модели (Y3)	23
3.4.1 Роль моделей в проектировании требований (Y2)	23
3.4.2 Контекст моделирования (Y2)	24
3.4.3 Моделирование структуры и данных (Y3)	25
3.4.4 Моделирование функций и потоков (Y3)	25
3.4.5 Моделирование состояния и поведения (Y2)	26
3.5 Глоссарии (Y2)	26
Структура документации требований (Y2)	27

3.6	Прототипы в проектировании требований (Y1)	27
3.7	Критерии качества результатов деятельности и требований (Y1)	28
4	Практики разработки требований (Y3)	30
4.1	Источники требований (Y3)	30
4.2	Выявление требований (Y2)	32
4.3	Требования, переговоры и урегулирование конфликтов (Y2)	33
4.4	Валидация требований (L2)	34
5	Процесс и структура работы (Y3)	35
5.1	Влияющие факторы (Y2)	35
5.2	Аспекты процесса проектирования требований (Y2)	36
5.3	Выстраивание процесса разработки требований (Y3)	38
6	Практики управления требованиями (Y2)	40
6.1	Что такое управление требованиями? (Y1)	40
6.2	Управление жизненным циклом (Y2)	40
6.3	Контроль версий (Y2)	41
6.4	Конфигурации и базисы (Y1)	41
6.5	Атрибуты и представления (Y2)	41
6.6	Трассируемость (Y1)	42
6.7	Обработка требований (Y1)	42
6.8	Приоритизация (Y1)	43
7	Инструментальная поддержка (Y2)	44
7.1	Инструменты в проектировании требований (Y1)	44
7.2	Внедрение инструментов (Y2)	45
	Ссылки	46

1 Введение и обзор проектирования требований (У2)

Цель: Знать, что такое ПТ, и понимать назначение ПТ

Длительность: 1 час

Термины: Требование, спецификация требований, проектирование требований (ПТ), заинтересованные лица, система, проектировщик требований

Задачи обучения

30 1.1.1 Помнить базовую терминологию (У1)

30 1.2.1 Пояснить назначение ПТ (У2)

30 1.2.2 Перечислить признаки и причины недостаточного ПТ (У1)

30 1.3.1 Знать область применения ПТ (У1)

30 1.3.2 Понимать различные виды требований (У2)

30 1.4.1 Знать об основных задачах ПТ и о том, что процесс ПТ должен быть адаптирован для их выполнения (У1)

30 1.5.1 Характеризовать роль и задачи проектировщика требований (У1)

30 1.6.1 Помнить, что проектировщик требований должен изучать (У1)

1.1 Проектирование требований: Что (У1)

У людей и организаций есть желания и потребности для создания нового или развития существующего. Мы называем такие потребности *требованиями*.

Вещами, которые должны быть созданы или развиты, могут быть:

- *Продукты*, поставляемые клиентам
- *Услуги*, предоставляемые клиентам
- Любые другие *производные*, такие как устройства, процедуры или инструменты, которые помогают людям и организациям достичь определенной цели
- *Составляющие или компоненты* продуктов, услуг, или других производных

Все эти вещи можно рассматривать как *системы*. В этой учебной программе мы используем термин *система* для обозначения всех видов вещей, к которым предъявляются требования *со стороны заинтересованных лиц*. *Заинтересованные лица* – это лица или организации, которые оказывают влияние на требования к системе или которые будут взаимодействовать с этой системой.

Цель ПТ заключается в определении и управлении требованиями к системам таким образом, чтобы внедряемые и развертываемые системы удовлетворяли желаниям и потребностям заинтересованных лиц.

В ПТ мы различаем три вида требований [Glin2020]:

- *Функциональные требования* касаются результата или поведения системы, в процессе ее функционирования. Сюда включаются требования к данным или взаимодействию системы с окружающей средой.

- *Качественные требования* относятся к вопросам качества, которые не охватываются функциональными требованиями, таким как производительность, доступность, безопасность или надежность.
- *Ограничения* – это требования, которые ограничивают решение рамками того, что необходимо для выполнения заданных функциональных требований и требований к качеству.

1.2 Проектирование требований: Зачем (У2)

Надлежащее ПТ вносит дополнительную *ценность* в процесс разработки и развития системы:

- Снижение риска разработки неправильной системы
- Лучшее понимание проблемы
- Основа для оценки стоимости разработки и затрат
- Предпосылки для тестирования системы

Типичные признаки недостаточного ПТ это – пропущенные, неясны или некорректные требования. Это происходит в основном из-за:

- Спешки и желании сразу начать разработку системы
- Проблем коммуникации между заинтересованными лицами
- Предположений о том, что требования очевидны
- Недостаточных теоретических и практических знаний в области ПТ

1.3 Проектирование требований: Где (У2)

ПТ может быть применено к любой системе. Однако, основное прикладное значение ПТ имеет для систем, в которых программное обеспечение играет главную роль. Такие системы обычно состоят из программных компонентов, физических и организационных элементов.

Мы различаем:

- *Системные требования* — что должна делать система
- *Требования заинтересованных лиц* — чего хотят заинтересованные лица с их точки зрения
- *Требования пользователей* — чего хотят пользователи с их точки зрения
- *Требования области применения* — требования той области, где будет применяться система
- *Бизнес-требования* — бизнес-цели, задачи и потребности организации

1.4 Проектирование требований: Как (У1)

Основные задачи ПТ заключаются в выявлении (4.2), документировании (3), валидации (4.4) и управлении (6) требованиями. Инструментальная поддержка (X) может помочь в выполнении этих задач. Анализ требований и урегулирование конфликтов (4.3) требований происходит в процессе их (требований) выявления. Для надлежащего выполнения работ по ПТ должен быть разработан процесс ПТ с учетом различного рода аспектов (5).

1.5 Роль и задачи проектировщика требований (У1)

Проектировщик требований – это, как правило, не должность, а *роль*, которую выполняет тот, кто:

- Выявляет, документирует, валидирует и/или управляет требованиями в рамках своих обязанностей.
- Обладает глубокими знаниями в области ПТ.
- Может предложить потенциальное решение проблемы.

На практике в роли проектировщика требований могут выступать бизнес-аналитики, прикладные специалисты, владельцы продуктов, системные инженеры и даже разработчики.

1.6 Что нужно знать о проектировании требований (У1)

Эта учебная программа охватывает базовый набор навыков, который должен выучить проектировщик требований. Программа охватывает основополагающие принципы ПТ (2): как документировать различного вида требования (3), как разрабатывать требования с учетом различных практик (4), как выявить и применить подходящий процесс ПТ (5), как управлять существующими требованиями (6) и как применять инструментальную поддержку (8).

2 Основные принципы проектирования требований (У2)

Цель: Знать и понимать принципы ПТ

Продолжительность: 1 час 30 минут

Термины: Контекст, требование, проектирование требований (ПТ), заинтересованное лицо, общее понимание, валидация

Задачи обучения

30 2.1.1 Перечислить принципы ПТ (У1)

30 2.2.1 Помнить термины, связанные с принципами (У1)

30 2.2.2 Пояснить принципы и почему они важны (У2)

2.1 Обзор принципов (У1)

ПТ регулируется комплексом основополагающих принципов, которые применяются ко всем задачам, видам деятельности и практикам в области ПТ. Следующие девять принципов составляют основу практики, представленной в последующих главах этой программы.

1. Ориентация на ценности: Требования – это средство для достижения цели, а не сама цель
2. Заинтересованные лица: ПТ – это удовлетворение желаний и потребностей заинтересованных лиц
3. Общее понимание: Успешное развитие систем невозможно без общего базиса
4. Контекст: Изолированные системы не могут быть поняты
5. Проблема – Требование – Решение: Взаимосвязанные вершины треугольника
6. Валидация: Невалидированные требования бесполезны
7. Развитие: Изменение требований это – не инцидент, это – норма
8. Инновация: "То, что нужно" не значит "достаточно"
9. Систематическая и дисциплинированная работа: Мы не можем обойтись без этого в ПТ

2.2 Пояснение принципов (У2)

Принцип 1 – Ориентация на ценность: требования являются средством достижения цели, а не самой целью

Стоимость требования равна его выгоде за вычетом затрат на его выявление, документирование, валидацию и управление. Польза требования это – его вклад в:

- Создание систем, удовлетворяющих желаниям и потребностям заинтересованных лиц.
- Уменьшение риска ошибки и дорогостоящей доработки при разработке системы.

Принцип 2 – Заинтересованные лица: ПТ - это удовлетворении желаний и потребностей заинтересованных лиц

Поскольку в ПТ речь идет о понимании желаний и потребностей заинтересованных лиц, надлежащее взаимодействие с заинтересованными лицами является одной из основных задач ПТ. Каждое заинтересованное лицо имеет определенную роль в контексте создаваемой системы, например: пользователь, клиент, заказчик, оператор или администратор. Заинтересованное лицо может также иметь несколько ролей одновременно. Для описания ролей, в которых слишком много лиц, или когда лица неопределены, могут быть использованы вымышленные архетипичные описания, известные как *персонажи*. Недостаточно учитывать только требования конечных пользователей или клиентов. Иначе мы можем пропустить важные требования со стороны других заинтересованных лиц. Пользователи, предоставляющие отзывы о пользовании системой, также должны рассматриваться в качестве заинтересованных лиц.

Заинтересованные лица могут иметь разные потребности и точки зрения, что может привести к противоречивым требованиям. Выявление и разрешение таких конфликтов является задачей ПТ.

Вовлечение нужных людей в соответствующие роли заинтересованных лиц имеет решающее значение для успешного проведения ПТ. Практика определения, приоритизации и работы с заинтересованными лицами описана в 4.

Принцип 3 – Общее понимание: успешное развитие систем невозможно без общей основы

В процессе ПТ создается, прививается и обеспечивается общее понимание между вовлеченными сторонами: заинтересованными лицами, проектировщиками требований и разработчиками. Есть две формы общего понимания:

- *Явное общее понимание* (достигнутое благодаря документально оформленным и согласованным требованиям)
- *Неявное общее понимание* (на основе общих знаний о потребностях, видениях, контексте и т.д.)

Знания области, предыдущее успешное сотрудничество, общая культура и ценности, а также взаимное доверие являются факторами, способствующими общему пониманию, в то время как географическая удаленность, аутсорсинг или большие команды с высокой текучестью кадров являются препятствиями.

Проверенная практика достижения общего понимания включает в себя: создание глоссариев (3.5), создание прототипов (3.6) или использование существующей системы в качестве отправной точки. Практика оценки общего понимания включает в себя: примеры ожидаемых результатов, изучение прототипов или оценку затрат на реализацию того или иного требования. Наиболее важной практикой для уменьшения влияния недопонимания является использование процесса с короткими циклами обратной связи (5).

Принцип 4 – Контекст: нельзя полностью понять изолированные системы

Системы встроены в *контекст*. Без понимания этого контекста невозможно правильно определить систему. В ИТ контекст системы – это часть среды системы, которая имеет отношение к пониманию системы и ее требований. *Граница системы* – это граница между системой и окружающим ее контекстом. Изначально граница системы часто бывает неясна, и, кроме того, она может меняться со временем.

Выявление границы системы и определение внешних интерфейсов между системой и элементами контекста, с которыми она взаимодействует, являются важными задачами ИТ. В то же время требуется определить *объем* системы – диапазон вещей, которые будут сформированы и спроектированы при разработке системы. Необходимо также рассмотреть так называемую *границу контекста*, которая отделяет относящуюся к ИТ часть среды системы от остального мира.

При определении системы недостаточно учитывать только требования в границах системы. При ИТ также требуется рассмотреть:

- Изменения в контексте, которые могут повлиять на требования к системе.
- Требования реального мира, актуальные для системы (и как их сопоставить с требованиями к системе).
- Контекстные предположения, которые должны соблюдаться для того, чтобы система работала и отвечала реальным требованиям.

Принцип 5 – Проблема – Требование – Решение: взаимосвязанные вершины треугольника

Проблема возникает тогда, когда заинтересованные лица не удовлетворены сложившейся ситуацией. *Требования* отражают то, что необходимо заинтересованным лицам для того, чтобы избавиться от проблемы или минимизировать ее. Социально–техническая система, которая позволит удовлетворить эти требования, является *решением*.

Проблемы, требования и решения не обязательно возникают в таком порядке. Идеи решения могут создавать потребности пользователей, которые должны быть проработаны в виде требований и реализованы в конкретном решении. Как правило, это происходит при внедрении инноваций.

- Проблемы, требования и решения тесно переплетены: их нельзя рассматривать по отдельности.
- Тем не менее, проектировщики требований стремятся отделить проблемы, требования и решения друг от друга, насколько это возможно, при продумывании, коммуницировании и документировании. Такое разделение делает процесс работы над задачами ИТ проще.

Принцип 6 – Валидация: невалидированные требования бесполезны

В конечном счете, мы должны подтвердить, что разработанная система удовлетворяет желаниям и потребности заинтересованных лиц. Для того чтобы с самого начала контролировать риск появления неудовлетворенных заинтересованных сторон, валидация требований должна начинаться уже в ходе ПТ. Мы должны проверить:

- Было ли достигнуто согласие в отношении требований между заинтересованными лицами,
- Желания и потребности заинтересованных лиц должным образом удовлетворяются,
- Контекстные предположения (см. Принцип 4 выше) являются обоснованными.

Практика валидации требований описана в 4.4.

Принцип 7 – Развитие: изменение требований - не инцидент, а норма

Системы и их требования подвержены *развитию*. Это значит, что они постоянно меняются. Например, просьбы об изменении того или иного требования или набора требований к системе могут быть вызваны:

- Меняются бизнес-процессы
- Конкуренты запускают новые продукты или услуги
- Клиенты меняют свои приоритеты или мнения
- Меняются технологии
- Появляются изменения в законах или нормативных актах
- Есть обратная связь от пользователей системы с запросами добавить или изменить функциональность

Кроме того, требования могут меняться при получении обратной связи от заинтересованных лиц при утверждении требований, в связи с нахождением неточностей в ранее утвержденных требованиях или в связи с изменившимися потребностями.

Как следствие, перед проектировщиками требований стоят две на первый взгляд противоречивые цели:

- Позволять требованиям меняться
- Сохранять требования стабильными

Подробности о том, как этого добиться, описываются в 6.7.

Принцип 8 – Инновация: "то, что нужно" не значит "достаточно"

Предоставление заинтересованным лицам именно того, что они хотят, устраняет возможность создать системы, которые удовлетворяют потребности заинтересованных сторон лучше, чем они ожидают. Целью правильного ПТ является не только удовлетворение

потребностей заинтересованных лиц, но осчастливить их, взволновать, дать гарантию безопасности. Вот в чем, в конечном счете, заключается инновация.

ПТ формирует инновационные системы:

- Начиная с реализации новых функциональностей и обеспечения простоты использования.
- Заканчивая реализацией новых прорывных идей.

В 4.2 обсуждается несколько методов стимулирования инноваций в области ПТ.

Принцип 9 – Систематическая и дисциплинированная работа: мы не можем обойтись без этого в ПТ

Мы должны использовать корректные процессы и практики для систематического выявления, документирования, валидации и управления требованиями, независимо от того, какой процесс разработки используется в действительности. Даже если система разрабатывается по принципу ad-hoc, систематический и дисциплинированный подход к ПТ улучшает качество получаемой в итоге системы.

В ПТ нет единого процесса или практики, которые бы хорошо работали в любой конкретной ситуации или, по крайней мере, в большинстве ситуаций. Систематическая и дисциплинированная работа означает, что проектировщик требований:

- Адаптирует свои процессы и практики под конкретную проблему, контекст и окружающую среду.
- Не использует всегда один и тот же процесс и набор практик.
- Не использует без раздумья последние успешно примененные процессы и практики по ПТ.

Для каждого ПТ должны быть выбраны процессы, методы и рабочие решения, которые наилучшим образом подходят для конкретной ситуации. Подробности подробно описаны в разделах 3, 4, 5 и 6.

3 Результаты деятельности и практики документирования (У3)

Цель: Понять фундаментальную роль результатов деятельности в ПТ и создавать результаты деятельности

Длительность: 7 часов

Термины: Результат деятельности, результаты деятельности на основе естественного языка, результаты деятельности на основе шаблонов, результаты деятельности на основе моделей, глоссарий, критерии качества, спецификация требований

Задачи обучения

- 30 3.1.1 Знать характеристики результатов деятельности ПТ и знать часто используемые типы результатов деятельности (У1)
- 30 3.1.2 Знать, для чего можно использовать каждый результат деятельности, и знать срок службы результатов деятельности (У1)
- 30 3.1.3 Пояснить различные категории и уровни абстракции для требований, включая то, как выбрать соответствующие уровни абстракции и уровни детализации (У2)
- 30 3.1.4 Знать аспекты, которые должны учитываться в результатах деятельности, и взаимосвязи между этими аспектами (У1)
- 30 3.1.5 Перечислить общие требования к документации (У1)
- 30 3.1.6 Описать, почему стоит планировать, какие результаты деятельности будут использоваться (У1)
- 30 3.2.1 Знать результаты деятельности на базе естественного языка, их преимущества и недостатки (У1)
- 30 3.2.2 Пояснить правила написания хороших требований на основе естественного языка (У2)
- 30 3.3.1 Знать категории результатов деятельности на основе шаблонов, их преимущества и недостатки (У1)
- 30 3.3.2 Описать индивидуальное требование и историю пользователя, используя шаблон фразы (У3)
- 30 3.3.3 Описать вариант использования с помощью шаблона формы (У3)
- 30 3.4.1 Понимать роли, назначение и использование моделей в ПТ (У2)
- 30 3.4.2 Понимать преимущества и ограничения моделирования в ПТ (У2)
- 30 3.4.3 Знать термины: модель, язык моделирования, модель деятельности, диаграмма деятельности, модель класса, диаграмма классов, контекстная модель, контекстная диаграмма, модель области определения, модель целей, модель взаимодействия, модель бизнес-процесса, диаграмма последовательности, диаграмма состояний, диаграмма автоматов, вариант использования, диаграмма вариантов использования (У1)
- 30 3.4.4 Понимать, как выбрать подходящий тип модели для описания требований в данной ситуации (У2)
- 30 3.4.5 Понимать и интерпретировать простые UML модели следующих типов: контекстные модели, варианты использования и диаграммы вариантов использования, модели области определения, модели классов, модели деятельности, модели бизнес-процессов и диаграмм состояний (У2)

- 30 3.4.6 Определить простую модель данных системы или объектов области определена, используя диаграмму классов UML (У3)
- 30 3.4.7 Определить простую системную функцию или бизнес–процесс по диаграмме активности UML (У3)
- ЕО 3.5.1 Пояснить назначение глоссариев и как их создавать (У2)
- 30 3.6.1 Знать часто используемые документы по спецификациям требований (У1)
- 30 3.6.2 Пояснить, какие структуры документа служат какому назначению, а также критерии структурирования документа (У2)
- 30 3.7.1 Знать различные виды прототипов и для чего они используются (У1)
- 30 3.8.1 Знать критерии качества для единичных требований (У1)
- 30 3.8.2 Знать критерии качества результатов деятельности (У1)

3.1 Результаты деятельности в проектировании требований (У2)

Результат деятельности – это записанный промежуточный или конечный результат, полученный в процессе работы. В ПТ существует целый ряд результатов деятельности, начиная, например, с временных графических эскизов появившихся из набора историй пользователей, и заканчивая официально выпущенными документами, содержащими спецификации договорных требований, объемом в сотни страниц.

3.1.1 Характеристики результатов деятельности (У1)

В ПТ используются различные виды результатов деятельности. Они характеризуются назначением, репрезентативностью, размерами и продолжительностью жизни. На практике для достижения поставленных задач встречаются следующие результаты деятельности. Обратите внимание, что результат деятельности может содержать другие результаты деятельности.

- Результат деятельности по одному требованию включают в себя индивидуальные требования и истории пользователей.
- Результаты деятельности по набору требований включают варианты использования, графические модели того или иного типа (3.4), описания задач, описания внешних интерфейсов, эпикей.
- Результаты деятельности по документам требований включают в свою структуру системные требования, бизнес–требования, требований заинтересованных лиц и пользователей, продуктовый и спринтовые беклоги, а также карты историй пользователей.
- В другие результаты деятельности входят глоссарии, текстовые примечания, графические эскизы и прототипы.

Результаты деятельности могут быть *представлены* в различных формах:

- На основе естественного языка (X)
- На основе шаблона (3.3)
- На основе модели (3.4)
- Другое представление, такие как чертежи или прототипы (3.6)

Большинство результатов деятельности *хранится* в электронном виде в виде файлов, в базах данных или в инструментах ПТ. Неофициальные, временные результаты деятельности могут также храниться на других носителях, например на бумаге или в карточках на Канбан доске.

По сроку службы результаты деятельности разделяются на три категории:

- *Временные результаты деятельности:* созданы в процессе коммуникации и для создания общего понимания.
- *Рабочие результаты деятельности:* разрабатываются в несколько итераций с течением времени; требуются некоторые метаданные (6.5), может потребоваться управление изменениями.
- *Долгосрочные результаты деятельности:* входят в *базис требований* или в *релиз*; необходим полный набор метаданных; необходимо управление изменениями.

Временный результат деятельности может быть преобразован в рабочий результат деятельности (путем его сохранения и добавления метаданных). Аналогичным образом, временный или рабочий результат деятельности может стать долгосрочным, если он будет включен в базис или релиз.

3.1.2 Категории и уровни абстракции (У2)

Как описано в 1.3, мы проводим различие между системными требованиями, требованиями заинтересованных лиц, требованиями пользователей, требованиями области применения и бизнес-требованиями. Некоторые результаты деятельности, такие как индивидуальные требования, эскизы или процессные модели, встречаются во всех этих категориях. Другие результаты деятельности непосредственно относятся к определенными категориями.

Когда бизнес-требования и требования заинтересованных лиц представлены в виде долгосрочных результатов деятельности, таких как спецификации бизнес-требований, спецификации требований заинтересованных лиц или видение проекта, они предшествуют спецификации системных требований. Например, при договорных отношениях, когда клиент заказывает у поставщика разработку системы, клиент часто создает и передает поставщику спецификацию требований заинтересованных лиц. Далее поставщик использует это в качестве основы для составления спецификации системных требований. В других проектах изначальными могут быть бизнес-требования, требования заинтересованных лиц и системные требования.

Требования обычно существуют на различных уровнях абстракции — от, например, высокоуровневых требований к новому бизнес-процессу до хорошо детализированных требований, например, реакции конкретного программного компонента на неординарное событие.

Требования часто организованы в три слоя абстракции:

- Бизнес
- Системный
- Компонентный

Выбор правильного уровня абстракции зависит от того, что должно быть определено. Однако важно не смешивать требования, которые находятся на разных уровнях абстракции. В небольших результатах деятельности (например, история пользователя) и средних (например, вариант использования) требования должны быть на более или менее одном и том же уровне абстракции. В

больших результатах деятельности, таких как спецификация системных требований, требования разных уровней абстракции должны быть разделены на уровне структуры документа (0).

3.1.3 Уровень детализации (У2)

Уровень детализации, до которого должны быть описаны требования, зависит, в частности, от нескольких факторов:

- Контекст проблемы и разработки
- Степень общего понимания проблемы
- Степень свободы, делегированная дизайнерам и программистам
- Наличие оперативной обратной связи с заинтересованными лицами в ходе разработки и реализации
- Стоимость в сравнении с необходимостью детализированной спецификации
- Предъявляемые стандарты и регулятивные ограничения

Чем выше уровень детализации требований, тем меньше риск того, что в итоге получится что-то неожиданное или неучтенное. Однако, с увеличением уровня детализации увеличивается и стоимость спецификации.

3.1.4 Аспекты, которые должны быть учтены в результатах деятельности (У1)

При установлении требований к результатам деятельности необходимо учитывать различные аспекты.

1. Требования классифицируются по видам требований (1.1):
 - а) Функциональные требования
 - б) Требования к качеству
 - с) Ограничения
2. Функциональные требования сосредоточены на различных аспектах системы:
 - а) Структура и данные
 - б) Функция и поток
 - с) Состояние и поведение
3. В итоге, требования можно понять только в контексте (Принцип 4 в 2):
 - а) Контекст системы
 - б) Граница системы и внешние интерфейсы

Существует множество взаимосвязей и зависимостей между упомянутыми аспектами. Например, запрос, отправленный пользователем (контекст) может инициировать переход в состояние (состояние и поведение), который инициирует действие, за которым следует другое действие (функция и поток), требующее данных (структура и данные) для предоставления результата пользователю (контекст) в течение заданного интервала времени (качество).

Некоторые результаты деятельности фокусируются на определенном аспекте и не затрагивают другие. Особенно это касается моделей требований (3.4). Другие результаты деятельности, такие как спецификация системных требований, охватывают все эти аспекты.

Когда различные аспекты документируются в отдельных результатах деятельности или в отдельных главах одного и того же документа, эти результаты деятельности или главы должны быть согласованы друг с другом.

3.1.5 Общие требования к документации (Y1)

Независимо от используемых техник, при создании результатов деятельности применяются следующие рекомендации:

- Выберите тип результата деятельности, который наиболее соответствует *назначению*.
- *Избегайте дублирования* – делайте ссылки на содержимое, вместо того чтобы повторять одно и то же содержимое еще раз.
- *Убедитесь в отсутствии несоответствий* между результатами деятельности, особенно когда они охватывают различные аспекты.
- *Используйте термины* как определено в глоссарии.
- *Структурируйте* результаты деятельности должным образом.

3.1.6 Планирование результатов деятельности, которые будут использоваться (Y1)

Условия каждого проекта и каждой предметной области отличаются друг от друга, поэтому изначально должен быть определен набор итоговых результатов деятельности. Поэтому необходимо согласовать следующие вопросы:

- В каких результатах деятельности и для какой цели должны быть зафиксированы требования?
- Какие уровни абстракции необходимо учитывать?
- До какого уровня детализации должны быть задокументированы требования по каждому уровню абстракции?
- Каким образом требования должны быть представлены в этих результатах деятельности?

Результаты деятельности, которые будут использоваться, должны быть определены на ранней стадии проекта. У этого есть несколько преимуществ:

- Помогает планировать затраты и ресурсы.
- Гарантирует, что будут использованы соответствующие нотации.
- Гарантирует, что все итоги зафиксированы в соответствующих результатах деятельности.
- Гарантирует, что информация не перемешана, и нет необходимости в "итоговом ревью" документов.
- Помогает избежать избыточности, что приводит к меньшему объему работ и удобной эксплуатации.

3.2 Результаты деятельности на основе естественного языка (Y2)

С самого начала систематического ПТ для практического описания требований в основном использовался естественный язык.

Результаты деятельности на основе естественного языка обладают рядом преимуществ:

- Произвольный естественный язык является чрезвычайно выразительным и гибким.
- Почти любые требования любого аспекта могут быть выражены естественным языком.
- Естественный язык используется в повседневной жизни и преподается в школе, поэтому не требуется специальной подготовки для чтения и понимания текстов на естественном языке.

Эти преимущества ведут к тому, что тексты, написанные на естественном языке, часто могут интерпретироваться неоднозначно, что является проблемой при описании требований. Кроме того, процесс выявления неточностей, пропусков и несоответствий в таких текстах является сложным и дорогостоящим.

Для написания хороших требований, основанных на естественном языке, рекомендуется:

- Писать короткие и хорошо структурированные предложения.
- Определить и придерживаться единой терминологии (3.5).
- Избегать расплывчатых или двусмысленных терминов и фраз.
- Знать подводные камни технической литературы, перечисленные ниже.

При написании технической документации на естественном языке следует избегать известных подводных камней и быть осторожным [GoRu2003].

Требуется избегать:

- Неполные описания
- Неопределенные существительные
- Неполные условия
- Неполные сравнения

Использовать с осторожностью:

- Пассивный залог
- Универсальные квантификаторы (такие как "все" или "никогда")
- Номинализация (т.е. существительные, производные от глагола, например, "аутентификация")

3.3 Результаты деятельности на основе шаблона (УЗ)

Результаты деятельности на основе шаблона используются для устранения некоторых недостатков результатов деятельности на естественном языке за счет использования заранее подготовленных шаблонов для описания требований.

- *Шаблоны фразы* обеспечивают predetermined синтаксическую структуру для фразы, которая описывает требование, например, индивидуальное требование или историю пользователя.
- *Шаблоны формы* предоставляют набор predetermined полей в заполняемой форме, например, для написания варианта использования или требований к качеству.
- *Шаблоны документа* обеспечивают predetermined структуру для документа с требованиями.

Различные шаблоны были описаны в литературе. [ISO29148] и [Rupp2014] содержат шаблоны фраз для индивидуальных требований. [MWHN2009] [Cohn2004] содержит часто используемые шаблоны фраз для историй пользователя и [Cock2001] описывает шаблоны

форм для вариантов использования. [Laue2002] содержит шаблон для описания задач. [ISO29148] и [RoRo2012] содержат шаблоны документов для всей спецификации. Кроме того, клиент может потребовать использовать в проекте свои шаблоны, уникальные для клиента.

Преимущества шаблонных результатов деятельности:

- Обеспечить четкую, многократно используемую структуру
- Помогает выявить наиболее актуальную информацию
- Делает представление требований и спецификаций требований единообразным
- Улучшает общее качество требований и спецификаций требований

Недостатки и подводные камни шаблонных результатов деятельности:

- Люди часто сосредотачиваются на форме заполнения шаблона, а не на его содержании.
- Аспекты, изначально не включенные в шаблон, скорее всего, будут упущены.

3.4 Результаты деятельности на основе модели (Y3)

Требования, представленные на естественном языке, имеют ограничения [Davi1993], в частности, в отношении получения общего представления о наборе требований и понимания взаимосвязи между требованиями. Требования спроектированные на базе модели устраняют эти ограничения.

3.4.1 Роль моделей в проектировании требований (Y2)

Модель – это абстрактное представление существующей части реальности или области реальности, которая должна быть создана. Реальность представляет собой любой возможный набор элементов, явлений или концепций, в том числе и другие модели. Сточки зрения модели, смоделированная область реальности называется оригиналом.

В ПТ модели помогают понять взаимосвязь и взаимодействие между требованиями и дают представление о наборе требований. Это достигается в первую очередь путем сосредоточения внимания на конкретных аспектах — например, на поведении — и абстрагировании от всех других аспектов. Использование графических нотаций в моделировании позволяет получить общее представление информации. Тем не менее, модели могут быть также представлены и неграфическим образом, например, с помощью таблиц.

Требования на основе модели имеют преимущества по сравнению с требованиями, представленными на естественном языке:

- Взаимосвязь и взаимодействие между требованиями легче понять с помощью представления их графически, чем при описании их на естественном языке.
- Сосредоточение внимания на одном аспекте снижает когнитивную нагрузку при изучении смоделированных требований.
- Языки моделирования требований имеют определенный синтаксис, что снижает риск двусмысленности и упущений.

Модели также имеют ограничения:

- Сложно поддерживать модели, которые фокусируются на различных аспектах, согласующихся друг с другом.
- Для понимания причинно-следственных связей необходимо интегрировать информацию, полученную от разных моделей.
- Модели ориентированы в первую очередь на функциональные требования; большинство требований к качеству и ограничений не могут быть легко выражены с помощью моделей.
- Ограниченный синтаксис языка графического моделирования подразумевает, что не каждый релевантный элемент информации может быть выражен в модели.

Поэтому часто комбинируются представление требований на основе модели и на основе естественного языка.

В ПТ модели можно использовать для:

- *Определения* (прежде всего функциональных) требований частично или даже полностью, как средство замены текстового представления требований.
- *Декомпозиции* сложной реальности на четко определенные и дополняющие друг друга аспекты; каждый аспект представлен определенной моделью.
- *Перефразирования* текстовых требований с целью улучшения их понимания, в частности, для представления взаимосвязей между требованиями.
- *Проверки* текстовых требований с целью выявления пропусков, неясностей и несоответствий.

Для построения моделей используются *языки моделирования*. Некоторые языки моделирования, например, UML [OMG2017] или BPMN [OMG2013], были стандартизированы. Когда требования указываются в нестандартном языке моделирования, требуется легенда, объясняющая синтаксис и семантику используемого языка моделирования.

Существует много типов моделей, которые могут быть использованы для ПТ. Проектировщик требований должен понимать, какой тип модели лучше всего подходит для описания требований конкретной ситуации.

3.4.2 Контекст моделирования (Y2)

Модели, которые фокусируются на аспекте контекста, описывают структурное встраивание системы в ее окружение и взаимодействие между системой и участниками в контексте системы.

Контекстные модели определяют систему и участников, взаимодействующих с системой, в контексте системы. Контекстная модель также описывает интерфейсы между системой и ее контекстом (например, с точки зрения того, какая информация является предметом обмена).

Контекстные диаграммы используются в качестве языка графического моделирования для выражения контекстных моделей. Для контекстных диаграмм нет стандартизированной нотации. Для составления контекстной модели могут быть использованы контекстные диаграммы из структурированного анализа [DeMa1978] или блочно-линейные диаграммы [Глинз 2019]. В SysML диаграммы определений блоков могут быть адаптированы для выражения контекстных диаграмм, для отображения системы и участников используются

стереотипные блоки. Обратите внимание, что диаграммы определений блоков не входят в базовый уровень CPRE.

[OMG2017] В языке моделирования UML *диаграммы вариантов использования* являются средством моделирования системы и ее контекста с точки зрения вариантов использования системы и участников в контексте системы, которые взаимодействуют с системой через эти варианты использования.

Варианты использования моделируют динамическое взаимодействие между участником в контексте системы и системой с точки зрения участника. Варианты использования в основном пишутся на базе шаблонов форм на естественном языке (3.3) или с помощью диаграмм активности UML (3.4.4).

3.4.3 Моделирование структуры и данных (УЗ)

Модели, которые фокусируются на аспектах структуры и данных, описывают требования к статическим структурным свойствам системы или области.

Статические модели области определения описывают (бизнес) объекты и их взаимоотношения в интересующей области. Они могут быть выражены с помощью UML диаграммы [OMG2017] классов.

Модели классов в первую очередь определяют классы системы, их атрибуты и связи. Классы представляют собой осязаемые и неосязаемые объекты в реальном мире, о которых система должна знать для выполнения своих задач. Диаграммы классов UML обычно используются в качестве языка моделирования для моделей классов.

В проектировании систем *диаграммы определений блоков* языка моделирования SysML описывают концептуальные сущности системы и взаимосвязи между ними.

Моделирование используется во многих областях. Например в строительстве используются *информационные модели зданий* (BIM) для моделирования элементов, необходимых для планирования, строительства и управления зданиями и другими конструкциями.

Обратите внимание, что диаграммы определений блоков и информационные модели зданий не входят в программу базового уровня CPRE.

3.4.4 Моделирование функций и потоков (УЗ)

Модели, ориентированные на функциональные и потоковые аспекты, описывают требования к последовательности действий, необходимые для получения требуемых результатов от заданных входов или действий, необходимых для выполнения (бизнес) процессов, в том числе к потоку управления и данных между действиями и к тому, кто за какое действие отвечает.

Модели деятельности используются для описания функций системы. В языке моделирования UML [OMG2017] для описания моделей поведения используются *диаграммы деятельности*. В них есть элементы для моделирования действий и потока управления между действиями. Диаграммы деятельности могут также показать, кто за какое действие отвечает. Усовершенствованные элементы моделирования (не входящие в программу базового уровня CPRE) предоставляют средства для моделирования потока данных.

Модели процессов используются для описания бизнес-процессов или технических процессов. Они могут быть выражены с помощью UML диаграммы деятельности или специфическими языками моделирования процессов, такими как BPMN [OMG2013]. В рамках программы базового уровня CPRE используются только UML диаграммы активности для моделирования процессов.

Модели историй области применения описывают визуальные истории о том, как участники взаимодействуют с устройствами, артефактами и другими предметами в области применения, обычно используя специфические для домена символы. Они являются средством понимания области применения, в которой будет работать система. Однако они не попадают в рамки программы базового уровня CPRE.

3.4.5 Моделирование состояния и поведения (У2)

Модели, ориентированные на состояние и поведение, описывают требования к поведению системы или компонента области с точки зрения статусно-зависимых реакций на события или динамики взаимодействия компонентов.

Диаграмма состояний (диаграмма автомата) моделирует события, которые запускают переход из одного состояния в другое, и действия, которые необходимо выполнить при переходе из одного состояния в другое. *Диаграмма состояний* [Hare1988] – это диаграмма автомата с состояниями, декомпозированными иерархически и/или ортогонально. Диаграмма автомата, в том числе и диаграмма состояний, могут быть построены с помощью UML [OMG2017] *диаграммы состояний автоматов* (также известной как *диаграмма состояний*).

Модели взаимодействия показывают динамическое взаимодействие между объектами или участниками. UML *диаграммы последовательностей* являются популярным средством для описания взаимодействия между объектами. Моделирование взаимодействия не входит в рамки базового уровня CPRE.

3.5 Глоссарии (У2)

В начале процесса ПТ, где участвуют несколько человек, существует риск отсутствия общего понимания терминов – разные люди интерпретируют один и тот же термин по-разному. Для минимизирования этой проблемы общепринятые описания терминов заносятся в глоссарий.

Глоссарий представляет собой централизованный свод определений для: контекстно-зависимых терминов, повседневных терминов с особым значением в данном контексте, сокращений и акронимов. Должны быть описаны синонимы (различные термины, обозначающие одно и то же). Гомонимы (использование одного и того же термина для разных вещей) следует избегать или указывать это.

Для глоссариев действуют следующие правила:

- Управляйте глоссарием централизованно.
- Глоссарий актуализируется на протяжении всего процесса разработки системы.
- Определите лицо или небольшую группу, которая отвечает за глоссарий.
- Используйте единый стиль и структуру глоссария.
- Привлекайте заинтересованных лиц и добивайтесь согласования терминологии.
- Сделайте глоссарий доступным для всех участников.
- Сделайте использование глоссария обязательным.

- Проверьте что во всех результатах деятельности глоссарий используется корректно.

Структура документации требований (Y2)

Документы спецификаций требований (3.1.1) включают в себя несколько результатов деятельности ПТ. Поэтому важно организовать такие документы с помощью четко определенной структуры, чтобы создать последовательный и удобный в обслуживании набор требований. Кроме самих требований, документ с требованиями может содержать дополнительную информацию и пояснения – например, глоссарий, условия приемки, информацию о проекте или описание технической реализации.

Структурах документации может отличаться от классически принятой.

Часто используемые документы:

- Спецификация требований заинтересованных лиц
- Спецификация требований пользователей (входит в спецификации требований заинтересованных лиц, охватывает только требования пользователей)
- Спецификация системных требований
- Спецификация бизнес-требований
- Видение

Часто используются альтернативные структуры документов:

- Беклог продукта
- Беклог спринта
- Карта историй

Выбор структуры документа и организация выбранной структуры зависят от:

- Выбранный процесс разработки (5)
- Тип разработки и область применения
- Договорные соглашения (клиент может прописать использование определенной структуры документа)
- Размер документа

Шаблоны документа могут помочь структурировать спецификацию требований. Шаблоны приведены в литературе [Vole2020], [RoRo2012], а также в стандартах [ISO29148]. Шаблоны также могут быть заимствованы из предыдущих аналогичных проектов или могут быть предложены заказчиком. Организация также может принять решение о создании шаблона в качестве внутреннего стандарта.

3.6 Прототипы в проектировании требований (Y1)

В ПТ *прототипы* являются средством наглядного уточнения и валидации требований на примере. В частности, прототипы могут быть использованы в том случае, если заинтересованные лица не хотят создавать и согласовывать результаты деятельности на естественном языке, по шаблонам или моделям.

Исследовательские прототипы [LiSZ1994] используются для создания общего понимания, уточнения требований или подтверждения требований на различных уровнях точности. Они выбрасываются после использования.

- *Вайрфреймы* – это прототипы с низкой точностью, созданные из простых материалов или инструментов эскизного моделирования, которые служат, в первую очередь, для обсуждения и проверки дизайнерских идей и концепций пользовательского интерфейса.
- *Макеты* – прототипы средней точности. При описании цифровых систем используются реальные экраны, клики и переходы, но без реальной функциональности. Они служат, прежде всего, для описания и проверки пользовательских интерфейсов.
- *Нативные прототипы* – это прототипы высокой точности, которые реализуют критические части системы до такой степени, что заинтересованные лица могут использовать прототип для проверки, будет ли спроектированная часть системы, работать и вести себя так, как ожидается.

В зависимости от степени точности, исследовательские прототипы могут быть дорогостоящими, поэтому всегда должен быть компромисс между стоимостью и получаемой выгодой.

Эволюционные прототипы[LiSZ1994] – это пилотные системы, представляющие собой основу разрабатываемой системы. Окончательная система развивается за счет постепенного расширения и совершенствования пилотной версии системы в течение нескольких итераций.

3.7 Критерии качества результатов деятельности и требований (У1)

Требование должно отвечать конкретным критериям качества, чтобы считаться хорошим требованием. В современном ПТ с ценностно-ориентированными подходами (Принцип 1 в 2) степень соответствия критерию качества должна соответствовать ценности, создаваемой этим требованием. Это означает, что требования не обязательно должны полностью соответствовать всем критериям качества — чем выше ценность требования, тем больше оно должно соответствовать критериям качества, чтобы снизить риск провала.

Адекватность и понятность являются важнейшими критериями качества единичного требования. Без них требование является бесполезным или даже вредным, независимо от соответствия всем остальным критериям.

Критерии качества для *единичных требований*:

- Адекватность (описывает истинные и согласованные потребности заинтересованных лиц)
- Необходимость
- Однозначность
- Полнота (автономность)
- Понятность
- Проверяемость

Как описано выше в 3.1.1, требования обычно фиксируются в различных результатах деятельности, которые покрывают единичные или множественные требования.

Приведенные выше критерии качества должны использоваться для создания хороших единичных требований в результате деятельности. Для результатов деятельности, которые включают более одного требования, дополнительно следует учитывать следующие критерии качества.

Критерии качества результатов деятельности, покрывающие *множественные требования*:

- Непротиворечивость
- Безызбыточность
- Полнота (ни одного известного и уместного требования не пропущено)
- Модифицируемость
- Трассируемость
- Согласованность

4 Практики разработки требований (УЗ)

Цель: Понять использование практик для определения источников требований, выявления требований, определения и разрешения конфликтов, а также для валидации требований

Продолжительность: 4 часа 30 минут

Термины: Источник требований, граница системы, контекст системы, выявление требований, обсуждение требований, валидация требований, заинтересованное лицо, модель Кано, конфликт

Задачи обучения

- 30 4.1.1 Определить границы системы с целью сосредоточения внимания на значимых требованиях (УЗ)
- 30 4.1.2 Помнить значимые источники для создаваемой системы (У1)
- 30 4.1.3 Идентифицировать заинтересованные лица и составить их список (УЗ)
- 30 4.1.4 Понимать преимущества управления заинтересованными лицами (У2)
- 30 4.2.1 Понимать, как модель Кано может помочь в выявлении правильных требований (У2)
- 30 4.2.2 Понимать разницу между методами сбора, методами проектирования и методами генерирования идей (У2)
- 30 4.2.3 Понимать, как выбрать правильную технику выявления для данной ситуации (У2)
- 30 4.3.1 Помнить различные типы конфликтов (У1)
- 30 4.3.2 Понимать, какие действия необходимы для разрешения конфликтов (У2)
- 30 4.3.3 Понимать, как применять подходящие методы разрешения конфликтов (У2)
- 30 4.4.1 Понимать, почему документы с требованиями должны быть валидированы (У2)
- 30 4.4.2 Помнить о четырех важных аспектах валидирования требований (У1)
- 30 4.4.3 Знать, как применять соответствующие методы валидации требований (У2)

4.1 Источники требований (УЗ)

Качество и полнота требований в значительной степени зависят от используемых источников требований. Потеря значимого источника приведет к неполному пониманию требований или к неполному требованию. Выявление источников требований – это итеративный и рекурсивный процесс, требующий постоянного пересмотра.

Общее понимание (Принцип 3 в 2) контекста разрабатываемой системы является необходимым условием для того, чтобы иметь возможность определить значимые источники требований. Область между границей системы и границей контекста называется (системным) контекстом (Принцип 4 в 2). Контекст (системы) необходим для общего понимания разрабатываемых требований и для определения первоначальных источников требований.

Источники требований классифицируются по трем типам:

- Заинтересованные лица
- Документы
- Системы

Заинтересованные лица системы (см. определение [Glin2020]; см. также Принцип 2 в разделе 2) являются основным источником требований. Заинтересованные лица системы включают [BiSp2003]:

- Пользователи
- Спонсоры
- Разработчики
- Представители власти
- Клиенты

Выявление заинтересованных лиц должно происходить систематически, список создается в начале разработки, и должен актуализироваться на протяжении всего процесса разработки.

[Coop 2004]Для всех систем с пользовательским интерфейсом *конечные пользователи* системы образуют группу заинтересованных лиц, которая представляет особый интерес для проектировщика требований. В случае большого числа пользователей их следует объединить в группы, чтобы облегчить процесс выявления. Практическим способом описания больших групп пользователей является использование персонажей .

Потенциальными источниками для выявления значимых заинтересованных лиц являются:

- Чеклисты типичных групп и ролей заинтересованных лиц
- Организационные структуры
- Документация по бизнес-процессам
- Маркетинговые отчеты
- Первоначальные заинтересованные лица для определения *дополнительных* заинтересованных лиц

Заинтересованные лица должны быть зафиксированы и актуализироваться в списке заинтересованных лиц, содержащем (по крайней мере) следующую информацию:

- Имя
- Функция (роль)
- Дополнительные личные и контактные данные
- Временная и территориальная доступность в течение проекта
- Значимость
- Область и глубина экспертизы
- Цели и заинтересованность в рамках проекта

Проблемы с заинтересованными лицами могут возникнуть, если права и обязанности заинтересованного лица не ясны или если его потребности не удовлетворены в достаточной степени. Управление [Bour2009] взаимоотношениями заинтересованных лиц является эффективным способом решения проблем с заинтересованными лицами.

В большинстве системных контекстов доступно еще больше источников. Они также должны быть рассмотрены для успешности новой системы, так как большинство заинтересованных лиц не говорят об очевидном: это их "подсознательные" требования (4.2).

Дополнительные источники требований включают в себя:

- Существующие и устаревшие системы
- Документы о процессах
- Нормативно-правовые документы

- Регламенты компании
- (Маркетинговая) информация о потенциальных будущих пользователях

Другой источник требований можно найти, анализируя похожие ситуации в совершенно разных областях.

4.2 Выявление требований (У2)

В течение процесса выявления задача проектировщика требований это – понимать желания и потребности заинтересованных лиц, при этом быть уверенным, что требования из всех значимых источников были собраны, путем применения соответствующих методов для их выявления. Важнейшим моментом в выявлении информации является превращение неявных требований, пожеланий и ожиданий в четкие требования.

При выявлении требований крайне важно знать характер и важность каждого требования. Они могут меняться от проекта к проекту, а также с течением времени. Модель [KaeA1984] Кано классифицирует требования по трем соответствующим категориям:

- Привлекательные (синонимы: восхищающие факторы, неосознанные требования)
- Желаемые (синонимы: основные факторы, осознанные требования)
- Ожидаемые (синонимы: базовые факторы, подсознательные требования)

Существует множество различных методов определения этих категорий требований. Мы различаем:

- Методы сбора
- Методы проектирования и генерации идей

Методы сбора – это устоявшиеся приемы выявления [ВаСС2015] требований, которые помогают выявить желаемые и ожидаемые требования путем изучения различных источников.

Можно выделить четыре основные категории:

- Методы анкетирования
- Методы сотрудничества
- Методы наблюдения
- Методы на основе артефактов

[Kuma2013][Osbo1979] *Методы проектирования и генерации идей* сосредоточены на стимулировании творчества. Эти методы (вкратце: *креативные методы*) используются для создания новых или инновационных требований, которые часто являются привлекательными. Они направлены на создание идей для поиска решений определенного вопроса, проблемы или цели. Популярными примерами таких методов являются мозговой штурм или методы аналогии.

За пределами сообщества разработчиков программного обеспечения появился более широкий термин "*методы проектирования*". Методы проектирования включают в себя творческие методы генерации идеи и предоставляют дополнительные исследовательские или комбинированные методы для разработки и получения дальнейшего понимания данной идеи. Популярные методы для этой цели включают создание прототипов (например, макетов), сценариев и сторибордов.

Более широкое понятие, связанное с проектированием и генерированием идей, – это *дизайнерское мышление*. Существуют различные подходы, такие как *d.school* [Sdsc2012] и *Designing for Growth* [LiOg2011], предлагающие большой набор методов, которые могут быть использованы для выявления требований.

Методы выявления должны позволять определять всевозможные требования – как функциональные, так и качественные, а также ограничения. На практике требованиям к качеству и ограничениям часто уделяется меньше внимания.

Для выявления *требований к качеству* в качестве чеклиста следует использовать такую модель качества, как стандарт [ISO25010] ISO 25010. Эта модель также может быть полезна при определении количественных требований.

Ограничения могут быть выявлены путем рассмотрения возможных ограничений возможного пространства решения – например, технических, правовых, организационных, культурных или экологических.

Выбор правильной техники распознавания является ключевой компетенцией, которая зависит от множества различных факторов, например:

- Тип системы
- Модель жизненного цикла разработки программного обеспечения
- Вовлеченные лица
- Организационная структура

Наилучшие результаты обычно достигаются с помощью комбинации различных техник выявления требований. [CaDJ2014] представляет систематический подход к выбору методов.

4.3 Требования, переговоры и урегулирование конфликтов (Y2)

Методы выявления требований сами по себе не обеспечивают согласованность, полноту, соответствие и т.д. результирующего набора требований в целом. (3.7). Однако, для окончательного утверждения все заинтересованные лица должны понимать и согласовать все требования, которые к ним относятся. Если некоторые заинтересованные лица не согласны, то эта ситуация признается как конфликт, который должен быть урегулирован соответствующим образом. Подходящие методы разрешения конфликтов должны выбираться на основе типа конфликта и контекста ситуации. Это требует глубокого понимания природы конфликта требований и взглядов заинтересованных лиц.

Задачи по выявлению и разрешению конфликтов:

- Идентификация конфликта
- Анализ конфликта
- Решение конфликта
- Документирование разрешения конфликта (принятые решения)

Важно различать типы [Moog2014] конфликтов. Следующие типы конфликтов часто требуют внимания проектировщика требований:

- Предметный конфликт
- Конфликт данных
- Конфликт интересов
- Конфликт ценностей

- Конфлик отношений
- Структурный конфликт

Для успешного разрешения конфликтов могут применяться общепринятые методы:

- Соглашение
- Компромисс
- Голосование
- Определение вариантов
- Отмена

Кроме того, существует несколько вспомогательных методов, например:

- Рассмотрение всех фактов
- Плюс–минус–интересно
- Матрица решений

Подходящие методы разрешения конфликтов должны выбираться на основе типа конфликта и контекста ситуации.

4.4 Валидация требований (L2)

Валидация требований является важным шагом на пути к созданию успешной системы (Принцип 6 в 2). Обеспечение качества требований на первом этапе позволит сократить объем затраченных усилий в дальнейшем. Под валидацией требований понимается проверка качества набора имеющихся требований и индивидуальных требований (подробнее см. 3.7).

Важными аспектами, которые необходимо учитывать при валидации требований, являются:

- Привлечение правильных заинтересованных лиц
- Разделение процессв диагностики и исправления ошибок
- Валидация с различных точек зрения
- Повторная валидация

Существует несколько методов валидации (например [GiGr1993], [OleA2018]). Эти методы валидации часто классифицируются как:

- *Методы обзора*, в том числе:
 - Пошаговый разбор
 - Инспекции
- *Исследовательские методы*, например:
 - Прототипирование
 - Альфа- и бета-тестирование
 - А/В тестирование [KoTh2017]
 - Создание минимального жизнеспособного продукта (MVP)
- *Разработка образцов*

Эти методы различаются по степени формальности и затраченным усилиям. Выбор метода зависит от таких факторов, как модель жизненного цикла разработки программного обеспечения, зрелость процесса разработки, сложность и критичность системы, любые законодательные или нормативные требования и/или необходимость аудиторской проверки.

5 Процесс и структура работы (У3)

Цель: Пояснить концепции процессов ПТ и применять соответствующие конфигурации процессов

Продолжительность: 1 час 15 минут

Термины: Процесс, процесс ПТ

Задачи обучения

- 30 5.1.1 Знать важные факторы, влияющие на процесс ПТ (У1)
- 30 5.1.2 Понимать, как и на что эти факторы влияют (У2)
- 30 5.2.1 Понимать аспекты, которые следует учитывать при выстраивании процесса ПТ (У2)
- 30 5.3.1 Знать типичные конфигурации процесса ПТ (У1)
- 30 5.3.2 Понимать этапы выстраивания процесса ПТ (У2)
- 30 5.3.3 Выбирать и применять соответствующие конфигурации процесса ПТ для простых условий разработки системы (У3)

Процесс требует формирования и структурирования работы в области ПТ, которая будет выполняться в данном контексте. Поскольку не существует универсального процесса ПТ (1.4), необходимо выстроить свой процесс ПТ, который соответствовал бы данному контексту разработки и системы.

Процесс ПТ формирует информационный поток и модель коммуникации между различными участниками (например, клиентами, пользователями, проектировщиками требований, разработчиками, тестировщиками), а также определяет результаты деятельности, которые будут использоваться или производиться. Таким образом, процесс ПТ обеспечивает рамки для выявления, документирования, валидации и управления требованиями.

5.1 Влияющие факторы (У2)

На конфигурацию процесса ПТ влияют многие факторы. Основные факторы:

- Общая совместимость процесса: процесс RE должен соответствовать общему процессу разработки системы.
- Контекст разработки
- Возможности и наличие заинтересованных лиц
- Общее понимание
- Сложность и критичность разрабатываемой системы
- Ограничения
- Имеющиеся сроки и бюджет
- Волатильность требований
- Опыт проектировщиков требований

Анализ влияющих факторов дает информацию о том, как выстроить процесс ПТ. Влияющие факторы также ограничивают пространство возможных конфигураций процесса. Например, когда заинтересованные лица доступны только в начале проекта, нельзя выбирать процесс, который опирается на постоянную обратную связь от заинтересованных лиц.

5.2 Аспекты процесса проектирования требований (У2)

Есть три решающих аспекта, которые необходимо учитывать при выстраивании процесса [Глинз 2019] ПТ.

Временной аспект: линейный против итеративного

В линейном процессе требования выявляются заранее, во время одной из фаз процесса. В итеративном процессе требования выявляются постепенно, начиная с общих целей и некоторых начальных требований, а затем добавляя или изменяя требования при каждой итерации.

Критерии выбора *линейного* процесса ПТ:

- Процесс разработки системы является плановым и, в основном, линейным.
- Заинтересованные лица знают свои требования и могут сформулировать их заранее.
- В качестве контрактной основы для аутсорсинга проектирования и внедрения системы, когда требуется полная спецификация требований.
- Регулирующие органы требуют полную, официально выпущенную спецификацию требований на ранней стадии разработки.

Критерии выбора *итеративного* процесса ПТ:

- Процесс разработки системы является итеративным и гибким.
- Многие требования не известны заранее, но будут возникать и изменяться в процессе разработки системы.
- Заинтересованные лица доступны таким образом, когда можно организовать только короткие циклы обратной связи для снижения риска разработки неправильной системы.
- Продолжительность разработки позволяет провести более двух итераций.
- Необходима возможность легко изменять требования.

Аспект цели: предписывающий против исследовательского

В предписывающем процессе ПТ спецификация требований представляет собой договор: все требования являются обязательными и должны быть выполнены. В процессе исследовательского ПТ изначально известны только цели, а сами требования должны быть изучены.

Критерии выбора *предписывающего* процесса ПТ:

- Заказчику необходим зафиксированный договор на разработку системы.
- Функциональность и объем имеют приоритет над затратами и сроками.
- Разработка указанной системы может быть предложена на конкурсной основе или на условиях аутсорсинга.

Критерии выбора *исследовательского* процесса ПТ:

- Заинтересованные лица изначально имеют лишь смутное представление о своих требованиях.
- Заинтересованные лица активно вовлечены в процесс и обеспечивают постоянную обратную связь.
- Сроки и стоимость имеют приоритет над функциональностью и объемом.

- Изначально неясно, какие требования будут фактически реализовываться и в каком порядке.

Целевой аспект: ориентированность на клиента против ориентированности на рынок

В процессе ПТ, ориентированном на клиента, система заказывается клиентом и разрабатывается поставщиком. В рыночно-ориентированном процессе ПТ система разрабатывается как продукт или услуга для рынка, ориентированная на определенные сегменты пользователей.

Критерии выбора *клиенто-ориентированного* процесса ПТ:

- Системой будет пользоваться в основном та организация, которая заказала систему и оплатила ее разработку.
- Важные заинтересованные лица в основном связаны с организацией заказчика.
- Могут быть определены отдельные личности в роли заинтересованных лиц.
- Заказчик хочет получить спецификацию требований, которая может служить в качестве договора.

Критерии выбора *рыночно-ориентированного* процесса ПТ:

- Организация-заказчик намерена продавать систему в каком-то сегменте рынка как продукт или услугу.
- Потенциальные пользователи не могут быть идентифицированы.
- Проектировщики требований должны разработать требования таким образом, чтобы они соответствовали предполагаемым потребностям целевых пользователей.
- Владельцы продуктов, специалисты по маркетингу, дизайнеры и системные архитекторы являются основными заинтересованными лицами.

Подсказки и нюансы

- Представленные выше критерии – это *эвристика*, а не строгие правила. Например, передачу разработки системы на внешний подряд предпочтительно делать с использованием предписывающего процесса ПТ, а не исследовательского, поскольку договор между заказчиком и поставщиком, как правило, основывается на общей спецификации требований. Однако можно также договориться о заключении контракта на аутсорсинг на основе исследовательского процесса в области ПТ.
- Линейные процессы ПТ работают только при наличии в них сложного процесса управления изменениями требований.
- Линейные процессы ПТ подразумевают длинные циклы обратной связи: для снижения риска разработки неправильной системы требования должны быть качественно отвалидированы.
- При выборе процесса ПТ часто выбираются вместе *линейный* и *предписывающий*.
- Исследовательские процессы ПТ, как правило, также являются итеративными процессами (и наоборот).
- В рыночно-ориентированном процессе обратная связь с пользователями является единственным средством проверки того, действительно ли продукт удовлетворяет потребности целевого сегмента пользователей.
- Рыночно-ориентированный аспект плохо сочетается с линейным и предписывающим аспектами.

5.3 Выстраивание процесса разработки требований (УЗ)

В контексте разработки конкретной системы выстраивать процесс ПТ должны лица, отвечающие за ПТ. 5.15.2[Глинз 2019] На основе анализа воздействующих факторов может быть использована подходящая комбинация аспектов, описанных в , . Ниже описаны три типичные комбинации.

Совместный процесс ПТ: итеративный и исследовательский и клиентоориентированный

Основной пример применения:	поставщик и заказчик тесно сотрудничают; заинтересованные стороны активно вовлечены как в процессы ПТ, так и в процессы разработки
Типичные результаты деятельности:	Беклог продукта с пользовательскими сценариями и/или описаниями задач, прототипами
Типичный информационный поток:	Непрерывное взаимодействие между заинтересованными лицами, владельцами продуктов, проектировщиками требований и разработчиками; может включать в себя и обратную связь с пользователями

Договорной процесс ПТ: как правило, линейный (иногда итерационный), предписывающий и клиентоориентированный

Основной пример применения:	Спецификация требований представляет собой договорную основу для разработки системы лицами, не вовлеченными в разработку спецификации и не имеющими тесного взаимодействия с заинтересованными лицами после этапа проектирования требований.
Типичные результаты деятельности:	Классическая спецификация системных требований, состоящая из моделей и требований, описанных естественным языком
Типичный информационный поток:	В первую очередь от заинтересованных сторон к проектировщикам требований

Продукто-ориентированный процесс ПТ: итеративный и исследовательский и рыночно-ориентированный

Основной пример применения:	Организация определяет и разрабатывает программное обеспечение с целью продажи или распространения его в качестве продукта или услуги
Типичные результаты деятельности:	Беклог продукта, прототипы
Типичный информационный поток:	Взаимодействие между владельцем продукта, маркетингом, проектировщикам требований, дизайнерами, разработчиками и (возможно) быстрая обратная связь с потребителями/пользователями

Обратите внимание на то, что могут быть такие контексты системы и процессов разработки, когда ни одна из вышеупомянутых комбинация не подходит. Например, нормативные ограничения могут требовать использование процесса, который соответствует конкретным стандартам, таким как ISO/IEC/IEEE 29148 [ISO29148].

При выстраивании процесса ПТ мы рекомендуем использовать пятиступенчатую процедуру:

1. Анализ влияющих факторов (5.1)
2. Оценка критериев аспектов (5.2)
3. Выстраивание процесса (5.3)
4. Определение результатов деятельности (3)
5. Выбор соответствующих практик

6 Практики управления требованиями (У2)

Цель: Понять необходимость и преимущества управления требованиями

Длительность: 2 часов

Термины: Управление требованиями, управление изменениями, трассируемость, атрибуты требований, жизненный цикл требований, приоритизация

Задачи обучения

- 30 6.1.1 Знать, что такое управление требованиями и зачем оно (У1)
- 30 6.2.1 Пояснить, почему результаты деятельности по проектированию требований нуждаются в модели жизненного/статусного цикла (У2)
- 30 6.3.1 Настроить концепцию версионирования требований для конкретного проекта (У2)
- 30 6.4.1 Знать способы применения конфигураций требований и базиса требований (У1)
- 30 6.5.1 Знать назначение атрибутов требований (У1)
- 30 6.5.2 Определить подходящий набор атрибутов для требований для конкретного проекта (У2)
- 30 6.5.3 Пояснить назначение представлений требований и перечислить представления требований (У2)
- 30 6.6.1 Перечислить причины для трассируемости требований (У1)
- 30 6.6.2 Пояснить различия между неявной и явной трассируемостью (У1)
- 30 6.6.3 Знать, как может быть задокументирована явная трассируемость (У1)
- 30 6.7.1 Знать, как управлять изменениями при линейном (на основе плана) и гибком подходах (У1)
- 30 6.8.1 Знать назначение приоритизации и знать основные критерии оценки (У1)
- 30 6.8.2 Перечислить шаги по приоритизации требований (У1)
- 30 6.8.3 Перечислить различные категории методов приоритизации (У1)

6.1 Что такое управление требованиями? (У1)

Управление требованиями – это процесс управления существующими требованиями, зафиксированными в различных результатах деятельности. В частности, это включает в себя [Glin2020] хранение, изменение и трассируемость требований. Управление требованиями может происходить по-разному и на разных уровнях в зависимости от выбранного процесса разработки и контекста — см., пример, [Leff2011], [Rupp2014], [WiBe2013]. Вне зависимости от обстоятельств, задача управления требованиями заключается в поддержании требований таким образом, чтобы все роли в проекте могли работать эффективно и результативно.

6.2 Управление жизненным циклом (У2)

Под управлением жизненным циклом понимается процесс отслеживания всех результатов деятельности с точки зрения их статуса в течение их жизненного цикла. Каждое задокументированное требование и каждый результат деятельности имеет свой жизненный цикл: оно создается, затем оценивается и уточняется, рассматривается, перерабатывается, объединяется, согласовывается и так далее. Для того чтобы можно было определить, в каком

состоянии находится результат деятельности, необходима модель жизненного цикла, определяющая каждое возможное состояние жизненного цикла и переход из одного состояния в другое. Текущее состояние результата деятельности всегда должно быть однозначным, включая (как правило) историю его переходов.

6.3 Контроль версий (Y2)

Контроль версий требований относится к процессу отслеживания всех результатов деятельности в процессе их развития. Любое изменение результата деятельности ведет за собой создание новой версии. Версионирование позволяет проследить историю результата деятельности от его создания и восстановить результат деятельности в любой более ранней версии. Для этой цели требуется как минимум иметь:

- Номер версии для однозначной идентификации версии результата деятельности.
- ▶ История изменений, чтобы определить, что было изменено.
- Хранилище результатов деятельности.

Версионирование должно применяться для всех результатов деятельности [WiVe2013]. Номер версии обычно состоит как минимум из двух частей: версии и инкремента.

6.4 Конфигурации и базисы (Y1)

Конфигурация требований – это последовательный набор результатов деятельности, которые содержат требования. Каждая конфигурация имеет конкретное назначение и содержит не более одной версии каждого результата деятельности [Glin2020]. Назначением конфигураций является, например, обзор набора результатов деятельности или оценка затрат на разработку.

Базис – это стабильная, с контролируруемыми изменениями *конфигурация* результатов деятельности, используемая для планирования релизов или других этапов реализации проекта [Glin2020].

Конфигурации имеют следующие свойства:

1. Логическая связь
2. Согласованность
3. Уникальность
4. Неизменность
5. Базис, к которому можно сделать откат

6.5 Атрибуты и представления (Y2)

Атрибуты необходимы для документирования важных метаданных результата деятельности и обычно используются для ответа на ряд важных вопросов в течение жизненного цикла проекта или продукта.

Цель атрибуции требований заключается в том, чтобы дать возможность членам команды и другим заинтересованным лицам получить информацию о необходимых им требованиях в любой момент времени в ходе проекта.

Определение соответствующего набора атрибутов зависит от информационных потребностей различных заинтересованных лиц в проекте. Существующие стандарты, например [ISO29148], дают обзор наиболее значимых атрибутов.

Представление – это ограниченный набор из общего набора требований, который содержит только то, что представляет интерес в настоящий момент. С технической точки зрения, представление – это комбинация настроек фильтрации и сортировки, которую можно сделать доступной для других участников, сохранив выбранную комбинацию.

Есть три типа представления:

- *Выборочное представление*
- *Проективное представление*
- *Агрегированное представление*

В большинстве случаев представление требований – это сочетание выборочных, проективных и агрегированных представлений.

6.6 Трассируемость (Y1)

Трассируемость [GoFi1994] – это возможность проследить требование *до его источника* (т.е. до заинтересованных лиц, документов, правовых актов и т.д.) и *далее до результатов деятельности* (например, до тестовых сценариев), а также и *до других требований*, от которых зависит это требование.

Трассируемость является необходимым условием для управления требованиями и часто напрямую требуется стандартами, законами и нормативными актами. Ведение трассируемости, по сути, означает поддержание взаимозависимости между различными результатами деятельности на уровне требований, а также их отношений с предшествующими и последующими результатами.

Трассируемость может быть задокументирована *неявно* путем структурирования и стандартизации результатов деятельности или *явно* путем сопоставления результатов деятельности друг с другом с помощью разного вида уникальных идентификаторов [HuJD2011]. Общими формами представления являются гиперссылки, ссылки, матрицы, таблицы или графики.

6.7 Обработка требований (Y1)

Требования не статичны. Изменения в требованиях происходят по разным причинам и должны быть обработаны должным образом (Принцип 7 в 2), например, путем создания формального *запроса на изменения*, или путем добавления нового элемента в *беклог продукта*.

Принятие решения, планирование и контроль за введением изменения зависят от подхода к разработке и от того, в какой момент происходит изменение.

При *линейном подходе* решение об изменении часто принимается Группой по контролю за изменениями (в проектной деятельности) или Группой по советам к изменениям (в операционной деятельности). При более *итеративном* подходе владелец продукта включает изменение в беклог продукта и соответствующим образом расставляет приоритеты.

6.8 Приоритизация (У1)

Не все требования одинаково важны [Davi2005. Оценка и приоритизация используются для определения наиболее значимых требований для следующего релиза или доработки продукта.

Определение затрат на реализацию требований является основой для их приоритизации, часто определяемой с использованием множественных критериев оценки, таких как ценность для бизнеса, срочность, усилия, зависимость и другие.

Приоритет требования описывает важность одного требования по сравнению с другими требованиями в соответствии с определенными критериями [Glin2020]. Сама по себе *расстановка приоритетов* осуществляется на основе одного или нескольких критериев; это в основном зависит от выбранного метода расстановки приоритетов.

Этапы расстановки приоритетов:

- Определение целей и ограничений приоритизации
- Определение желаемых критериев оценки
- Определение соответствующих заинтересованных лиц
- Определение требований, которые должны быть приоритизированы
- Выбор метода приоритизации
- Выполнение приоритизации

Методы приоритизации могут быть классифицированы:

- Ad-hoc методы приоритизации
- Аналитические методы приоритизации

7 Инструментальная поддержка (У2)

Цель: Дать общий обзор роли инструментов ПТ и аспектов их применения

Продолжительность: 30 минут

Термины: Инструмент, инструмент ПТ

Задачи обучения

30 7.1.1 Знать различные типы инструментов ПТ (У1)

30 7.2.1 Пояснить, на что следует обратить внимание при внедрении инструментов ПТ (У2)

7.1 Инструменты в проектировании требований (У1)

В процессе ПТ могут быть применены инструменты, которые поддерживают выполнение поставленных задач и активностей. Поскольку процессы ПТ достаточно индивидуальны (), существующие инструменты ПТ зачастую сосредоточены только на определенных аспектах ПТ и редко поддерживают все виды деятельности. Перед выбором инструмента проектировщики требований должны решить, какие задачи и виды деятельности в процессе ПТ должны поддерживаться и каким образом. Есть инструменты, которые поддерживают:

- Управление требованиями:
 - Определение и хранение атрибутов требований
 - Приоритизация требований
 - Управление версиями и конфигурациями
 - Отслеживание и трассировка требований
 - Управление изменениями в требованиях
- Управление процессом ПТ:
 - Метрики и отчетность по процессу ПТ
 - Метрики и отчетность по качеству продукта
 - Управление рабочим процессом ПТ
- Документирование требований:
 - Совместное использование требований
 - Создание общего понимания требований
- Моделирование требований
- Сотрудничество в области ПТ
- Тестирование/симуляция требований

Инструменты часто сочетают в себе вышеупомянутые возможности. Для обеспечения надлежащего охвата всех задач, связанных с ПТ, можно использовать сразу несколько инструментов.

Иногда для документирования или управления требованиями используются иные виды инструментов, например, офисное программное обеспечение или трекеры задач. Эти инструменты имеют ограничения и должны использоваться только тогда, когда процесс ПТ находится под контролем, а требования сошлосованы и достаточно стабильны.

7.2 Внедрение инструментов (У2)

Выбор инструмента для ПТ ничем не отличается от выбора инструмента для других целей. Прежде, чем выбирать инструмент, надо определиться с целью, контекстом и требованиями [Fugg1993].

Надлежащий инструмент можно подобрать только после того, как будут внедрены надлежащие процедуры и методы ПТ. Внедрение инструмента требует наличия четких зон ответственности и процедур по ПТ. Следующие аспекты должны рассматриваться в процессе внедрения инструментов:

- Учитите все расходы в течение жизненного цикла, выходящие за рамки затрат на лицензию
- Оцените необходимые ресурсы
- Используйте пилотные проекты, чтобы избежать рисков
- Оцените инструмент согласно определенным критериям
- Проинструктируйте сотрудников о том, как пользоваться инструментом

Ссылки

- [AnPC1994] Annie I. Antón, W. Michael McCracken, Colin Potts: Goal Decomposition and Scenario Analysis in Business Process Reengineering. CAiSE (Conference on Advanced Information Systems Engineering), 1994, 94–104.
- [BaCC2015] К. Бакстер, Си. Карредж, Кей. Кейн: Понимание ваших пользователей: Практическое руководство по методам исследования пользователей, 2-е издание. Морган Кауфман, Берлингтон, 2015.
- [BiSp2003] К. Биттнер, Ай. Спенс: Моделирование вариантов использования. Пирсон Эдьюкейшн, Бостон, 2003.
- [Bour2009] Л. Борн: Управление взаимоотношениями с заинтересованными лицами: Модель зрелости для организационной реализации. Гоуэр Пабблишинг Лтд, Бирлингтон, 2009.
- [CaDJ2014] D. Карризо, О. Дайсти, Н. Джуристо: Систематизация выбора техники сбора требований. Информационные и программные технологии 2014, 56(6): 644–669.
- [Cock2001] А. Кокберн: Написание эффективных вариантов использования. Эддисон-Уэсли, Бостон 2001.
- [Cohn2004] М. Кон: Прикладные сценарии пользователей. Разработка программного обеспечения методом Agile. Эддисон-Уэсли, Бостон, 2004.
- [Coop 2004] А. Купер: Психбольница в руках пациентов. Почему высокие технологии сводят нас с ума и как восстановить душевное равновесие. Кью, Индианаполис, 2004.
- [Davi2005] А. М. Дэвис: Просто управление требованиями где разработка программного обеспечения идет в ногу с маркетингом. Издательство Дорсет Хаус, Нью-Йорк, 2005.
- [Davi1993] А. М. Дэвис: Требования к программному обеспечению. Объекты, функции и состояния, 2-е издание, Прентис Холл, Нью-Джерси, 1993.
- [DeMa1978] Т. ДеМарко: Структурный анализ и спецификация системы. Юрдон Пресс, Нью-Йорк, 1978.
- [Fugg1993] А. Фадджетта: Классификация CASE методов. IEEE Компьютер 1993, 26 (12): 25–38.
- [GiGr1993] Т. Гилб, Ди. Грэм: Проверка программного обеспечения. Эддисон Уэсли, Бостон, 1993 год.
- [Глинз 2019] М. Глинз: Проектирование требований I. Корс ноутс, Университет Цюриха, 2019 год. <https://www.ifi.uzh.ch/en/reqg/courses/archives/hs19/re-i.html#resources>. Последне посещение: июль 2020 года.
- [Glin2020] М. Глинз: Глоссарий терминов проектирования требований. Версия 2.0. <https://www.ireb.org/en/downloads/#cpre-glossary>. Последне посещение: июль 2020 года.

- [GoFi1994] О. Готэл, Э. Финкельштейн: Анализ проблемы отслеживания требований. 1-я Международная конференция по разработке требований, Колорадо Спрингс, 1994. 94–101.
- [GoRu2003] Р. Гетц, Си. Рупп: Психотерапия для системных требований. 2-я Международная конференция по когнитивной информатике в рамках IEEE (ICC'03), Лондон, 2003 г. 75–80.
- [GRL2020] Goal oriented Requirement Language. University of Toronto, Canada <https://www.cs.toronto.edu/km/GRL>. Последнее посещение – май 2020 года.
- [Hare1988] D. Харел: О визуальных формализмах. Коммуникации ACM 1988, 31 (5): 514–530.
- [HoSch2020] S. Хофер, Эйч. Швентнер: Сторитейлинг предметной области — метод совместного моделирования. Доступно в Линпаб, <http://leanpub.com/domainstorytelling>. Последне посещение: июль 2020 года.
- [HuJD2011] Е. Халл, Кей. Джексон и Джей. Дик: Проектирование требований. Спрингер, 3-е издание, 2011.
- [ISO29148] ISO/IEC/IEEE 29148: Инжиниринг систем и программного обеспечения – Процессы жизненного цикла – Инжиниринг требований, Международная организация по стандартизации, 2018.
- [ISO19650] ISO 19650: Организация и перевод в цифровой формат информации о зданиях и строительных работах, включая информационное моделирование зданий (BIM) – Управление информацией с использованием информационного моделирования зданий – Часть 1 и 2, Международная организация по стандартизации, 2018 г.
- [ISO25010] ISO/IEC/IEEE25010:2011: Требования к качеству систем и программного обеспечения и его оценка (SQuaRE). Модели качества систем и программного обеспечения. Международная организация по стандартизации, Женева, 2011 год.
- [Jack1995] М. А. Джексон: Требования и спецификации программного обеспечения: практический лексикон, принципы и предрассудки. Эддисон-Уэсли, Нью-Йорк, 1995.
- [Jack1995b] М. Джексон: Мир и машина. 17-я Международная конференция по программной инженерии 1995 г. (ICSE 1995). 287–292.
- [KaeA1984] N. Кано и др.: Привлекательное качество и обязательное качество. Журнал японского общества контроля качества 1984, 14(2): 39–48. (на японском)
- [KoTh2017] Р. Кохави, Эс. Томк: Удивительная сила онлайн-экспериментов – получение максимальной отдачи от A/B и других контролируемых тестов. Гарвардский бизнес-обзор, сентябрь–октябрь 2017 года: 74–82.
- [Kuma2013] V. Кумар: 101 метод проектирования – Структурированный подход к внедрению инноваций в вашу организацию. Джон Уайли и сыновья, Хобокен, 2013 год.
- [Laue2002] S. Лаузен: Требования к программному обеспечению. Стили и методы. Эддисон-Уэсли, Харлоу, 2002.

- [Leff2011] D. Леффингуэлл: Agile требования к программному обеспечению. Практики бережливых требований для команд, программ и предприятий. Эддисон-Уэсли, Бостон, 2011.
- [LiOg2011] J. Лидтка, Ти. Огилви: Проектирование для роста: набор инструментов проектирования для менеджеров. Пресса Колумбийского университета, 2011.
- [LiSZ1994] Н. Лихтер, М. Шнайдер-Хуфшмидт, Эйч. Зуллиховен: Прототипирование в промышленных программных проектах – устранение разрыва между теорией и практикой. Операции IEEE по разработке программного обеспечения 1994, 20 (11): 825–832.
- [MFeA2019] D. Мендес Фернандес, Икс. Фрэнк, Н. Сейфф, М. Фельдерер, М. Глинз, М. Калиновский, А. Волгельсанг, Эс. Вагнер, Эс. Бюне, К. Лауэнрот: Проповедуем ли мы то, что практикуем? Исследование практической значимости программы проектирования требований – дело IREB. CIBSE 2019: 476–487.
- [Moor2014] С. W. Мур: Процесс медиации – практические стратегии разрешения конфликтов, 4-е издание. Джон Уайли и сыновья, Хобокен, 2014 год.
- [MWHN2009] А. Мавин, Пи. Уилкинсон, Э. Харвуд и М. Новак: Простой подход к синтаксису требований (EARS). 17-я Международная конференция по проектированию требований IEEE (RE'09), Атланта, Джорджия, 2009. 317–322.
- [OleA2018] К. Олсен и др.: Сертифицированный тестер, программа базового уровня – версия 2018. Международная квалификационная комиссия по тестированию программного обеспечения, 2018.
- [OMG2013] Группа управления объектами: модель и нотация бизнес-процесса (BPMN), версия 2.0.2. OMG document formal/2013–12–09 <http://www.omg.org/spec/BPMN>. Последне посещение: июль 2020 года.
- [OMG2017] Группа управления объектами: Унифицированный язык моделирования OMG (OMG UML), версия 2.5.1. Документ OMG формальный/2017–12–05. <https://www.omg.org/spec/UML/About-UML/>. Последне посещение: июль 2020 года.
- [OMG2019] Группа управления объектами: Язык моделирования систем OMG (OMG SysML™), версия 1.6. Документ OMG формальный/2019–11–01. <https://www.omg.org/spec/SysML/>. Последне посещение: Январь 2022 года.
- [Osbo1979] А. Ф. Осборн: Прикладное воображение, 3-е пересмотренное издание. Сыновья Чарльза Скрибнера, Нью-Йорк, 1979.
- [RoRo2012] S. Робертсон и Джей. Робертсон: Освоение процесса требований, 3-е издание. Эддисон-Уэсли, Бостон, 2012.
- [Rupp2014] С. Рупп: Проектирование и управление требованиями, 6. Auflage. Хансер, Мюнхен, 2014 год. (на немецком).
- [Sdsc2012] Стэнфордская школа: Введение в проектное мышление. Институт дизайна Хассо Платтнера, Стэнфорд, 2012. <https://dschool-old.stanford.edu/groups/designresources/wiki/36873>. Последне посещение: июль 2020 года.

- [vLam2009] Аксель ван Ламсвеерде: Инженерия требований: От целей системы к моделям UML и спецификациям программного обеспечения. Чичестер: John Wiley & Sons, 2009.
- [Vole2020] Волер: Ресурсы требований. <https://www.volere.org>. Последне посещение: июль 2020 года.
- [WiVe2013] К. Вигерс и Джей. Битти: Требования к программному обеспечению, 3-е издание. Майкрософт Пресс, Редмонд, 2013.