



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Факультет компьютерных наук
Департамент программной инженерии

ОБЕСПЕЧЕНИЕ КАЧЕСТВА И ТЕСТИРОВАНИЕ

Семинар 1: Вводное занятие

Москва, 2020



ИНФОРМАЦИЯ О ПРЕПОДАВАТЕЛЕ

Татарников Андрей Дмитриевич

atatarnikov@hse.ru

andrewt0301@yandex.ru **(высылать ДЗ)**

Telegram канал: https://t.me/joinchat/AAAAAFTYn_XTYjzGzIP96w

Материалы: https://github.com/andrewt0301/QA_Testing_Course



ИНФОРМАЦИЯ О ПРЕПОДАВАТЕЛЕ

Опыт работы

- 1. 2005-2011.** SmartBear Software (<https://smartbear.com/>). QA Engineer (2005-2006), Tech Support (2006-2008), Software Developer (2008-2011).
- 2. 2011-2018.** ISP RAS (<https://www.ispras.ru>). *Researcher, Tech Lead.*
- 3. 2018-2019.** 1C (<https://1c.ru>). *Senior Software Developer.*
- 4. 2019-now.** Huawei (<https://www.huawei.com/>). *Senior Software Engineer.*



ИНФОРМАЦИЯ О КУРСЕ

Направленность занятий:

1. Получение практических навыков тестирования
2. Знакомство со средствами автоматизации процесса тестирования
3. Получения опыта разработки автоматизированных тестов

Основные средства:

1. Java, IntelliJ IDEA / Eclipse / Android Studio, Junit / TestNG, Gradle
2. Selenium, JMeter



СТРУКТУРА ЗАНЯТИЙ / БАЛЛЫ

1. Практические задания с оценкой
2. Сроки выполнения заданий (до следующего, иначе минус 2 балла еженедельно)
3. Работа на занятии
4. Посещаемость
5. Выставление оценки

$O_{\text{ауд.}} = ДЗ$

$O_{\text{накопл.}} = 0,5 * O_{\text{ауд.}} + 0,5 * O_{\text{контр. работа}}$

$O_{\text{результ.}} = 0,5 * O_{\text{накопл.}} + 0,5 * O_{\text{экз.}}$



ИНФОРМАЦИЯ О СТУДЕНТАХ

О группе:

1. Номер (подгруппа)
2. E-mail группы
3. Староста
4. По возможности — список студентов

Заголовки присланных ДЗ:

1. ФИО, Группа, Номер_ДЗ
 - пример — БПИ181 - Иванов - ДЗ №1



ОБЕСПЕЧЕНИЕ КАЧЕСТВА И ТЕСТИРОВАНИЕ

«Качество — это проблема всего коллектива»
Джеймс Уиттакер - «Как тестируют в Google»

Тестирование – выполнение программы с целью найти баги

Обеспечение качества – процессы, позволяющие предотвратить появление багов или обнаружить как можно раньше



ISO 9126

Функциональность	Определенный круг задач. Функциональность определяет, что именно делает данная программа.
Надежность	Способность ПО поддерживать определенный уровень работоспособности в заданных условиях. Надежность является вероятностной характеристикой работоспособности ПО.
Удобство использования	Удобство использования показывает, насколько ПО привлекательно, удобно в обучении работе с ним и при выполнении самой работы
Производительность	Способность ПО обеспечивать необходимую работоспособность по отношению к выделяемым для этого ресурсам.
Переносимость	Сохранение работоспособности ПО при изменении его окружения.
Удобство сопровождения	Трудоемкость анализа, исправления ошибок и внесения изменений в ПО.



ISO 25010

Надежность (reliability)	<ul style="list-style-type: none">o Зрелость (maturity)o Способность к восстановлению (recoverability)o Устойчивость к ошибкам (fault tolerance)o Работоспособность (availability, иногда также переводится как доступность)
Защищенность (security)	<ul style="list-style-type: none">o Конфиденциальность (confidentiality)o Целостность (integrity)o Строгое выполнение обязательств (неотвергаемость, non-repudiation)o Авторизуемость (операций, accountability)o Аутентичность (authenticity)
Удобство сопровождения (maintainability)	<ul style="list-style-type: none">o Удобство проверки (testability)o Анализируемость (analyzability)o Модифицируемость (modifiability, удобство внесения изменений + стабильность)o Модульность (modularity)o Повторная используемость (reusability)
Переносимость	<ul style="list-style-type: none">o Адаптируемость (adaptability)o Удобство замены (replaceability)o Удобство установки (installability)

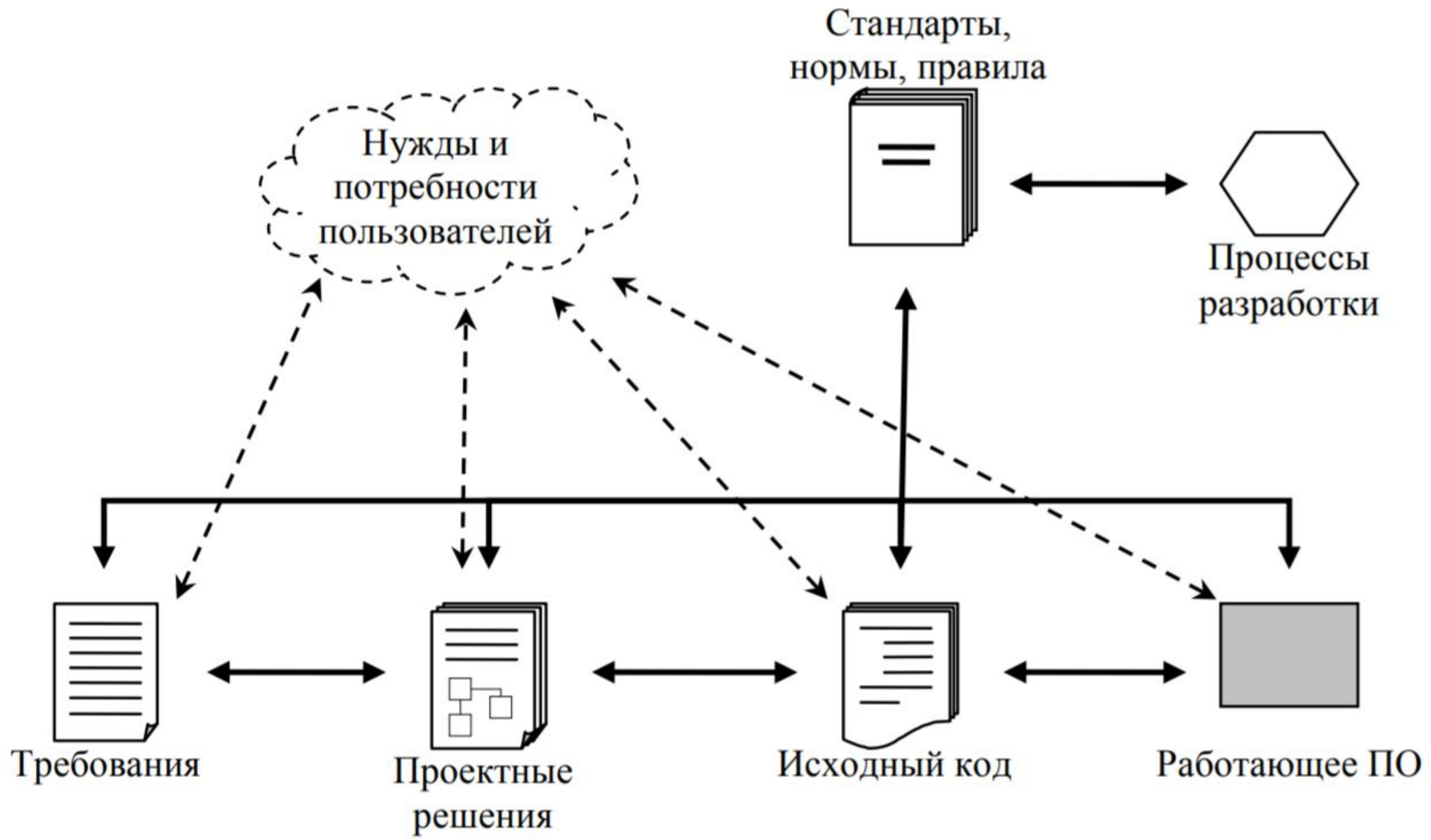
ТЕСТИРОВАНИЕ

- Тестированием называется проверка соответствия поведения проверяемой системы требованиям, выполняемая по результатам реальной работы этой системы в некотором конечном наборе специально созданных ситуаций. При этом проверяемая система обычно называется тестируемой системой (system under test или SUT по-английски).
- SWEBOOK



МЕТОДЫ КОНТРОЛЯ КАЧЕСТВА ПО

- Верификация
 - Проверяется за счет сопоставления его с другим артефактом, на основе которого первый должен был быть разработан или которому он должен соответствовать
- Валидация
 - Обозначает проверку некоторого артефакта разработки на соответствие конечным целям, для достижения которых это ПО предназначено, т. е., нуждам и потребностям его пользователей и заказчиков.

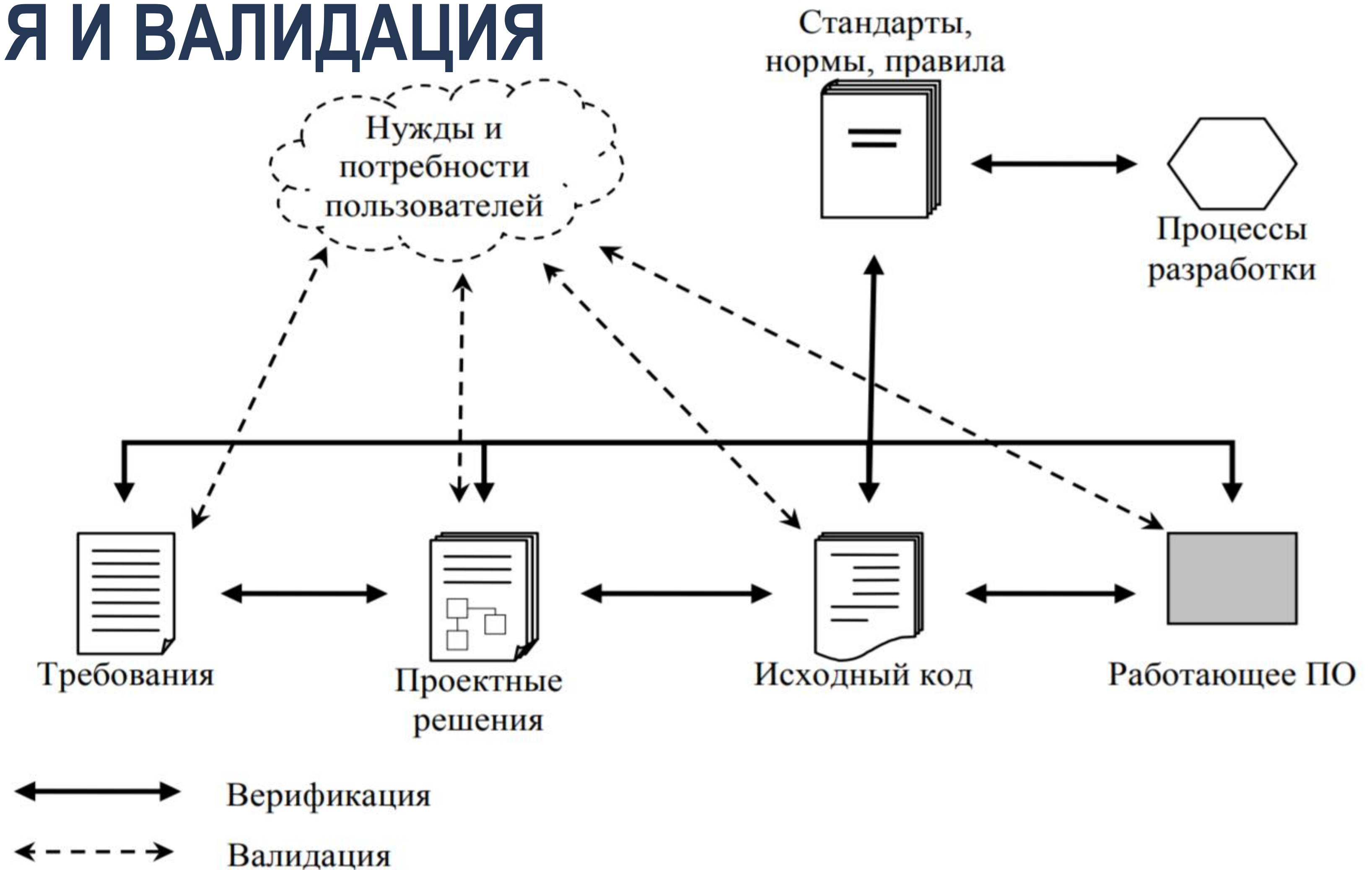


↔ Верификация
 ↔ Валидация

ВИДЫ ОШИБОК

defect	Иногда, хотя и достаточно редко, так называют произвольный дефект программной системы, будь то сбой, полностью разрушающий данные системы, неточно прорисованная буква на кнопке графического интерфейса пользователя или нестандартное форматирование исходного кода.
failure	Часто ошибкой или сбоем называют наблюдаемое нарушение требований, проявляющееся при некотором сценарии работы рассматриваемой системы.
fault	Ошибка в коде программы, вызывающая сбой, и состоящая в неправильном использовании какой-то конструкции языка программирования, употреблении лишней конструкции или в пропуске необходимой.
error	Ошибка аналитика, архитектора или программиста, заключающаяся в неправильном понимании определенного требования или ограничения, в том, что какое-то требование забыто, или, наоборот, используется лишнее требование.

ВЕРИФИКАЦИЯ И ВАЛИДАЦИЯ





ТЕСТИРОВАНИЕ В ПРОЦЕССЕ РАЗРАБОТКИ

1. Анализ требований
2. Разработка тестового плана
3. Разработка тестов разного уровня
4. Тестирование
5. Анализ результатов тестирования



НЕКОТОРЫЕ РОЛИ В ОБЕСПЕЧЕНИИ КАЧЕСТВА И ТЕСТИРОВАНИИ

- *Разработчик* (Software Engineer, SWE)
- *Разработчик в тестировании* (Software Engineer in Test, SET)
- *Инженер по тестированию* (Test Engineer, TE)
- *Инженер по требованиям*
- *Site Reliability Engineering (SRE)*
- *DevOps*



МЕТОДЫ ОБЕСПЕЧЕНИЯ КАЧЕСТВА И ТЕСТИРОВАНИЯ

1. Анализ требований
2. Инспекции кода
3. Статический анализ
4. Функциональное тестирование
 - 4.1. Ручное
 - 4.2. Автоматизированное
 - 4.2.1. Модульное
 - 4.2.2. Интеграционное
 - 4.2.3. Системное
5. Тестирование производительности / нагрузочное тестирование



ТРЕБОВАНИЯ

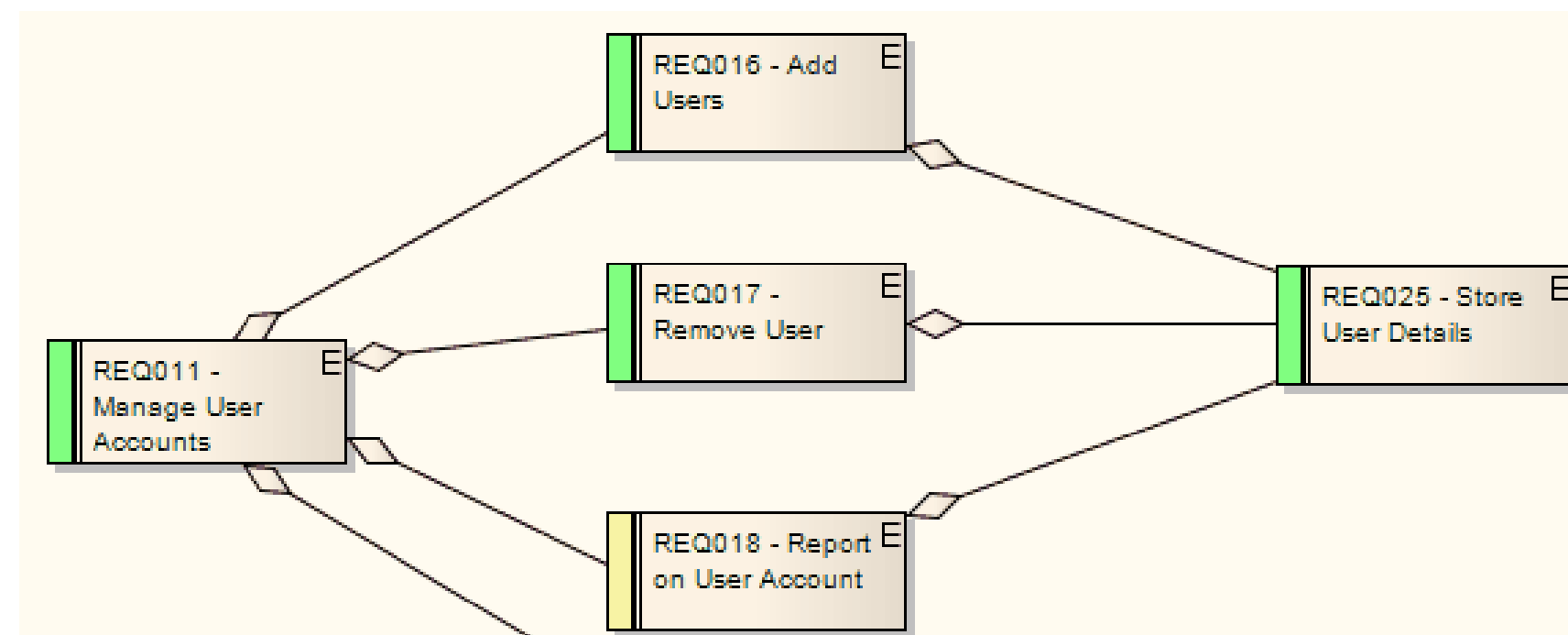
- Требования к программному обеспечению — совокупность утверждений относительно атрибутов, свойств или качеств программной системы, подлежащей реализации.

ПРЕДСТАВЛЕНИЯ ТРЕБОВАНИЙ

```
module "
  attr target_object: 'function'
```

```
section 'Actions'
action RET
do
  Return $CODE.
end
  attr return: $CODE
```

```
action RET_NOTHING
do
  Function **must** retur
end
```



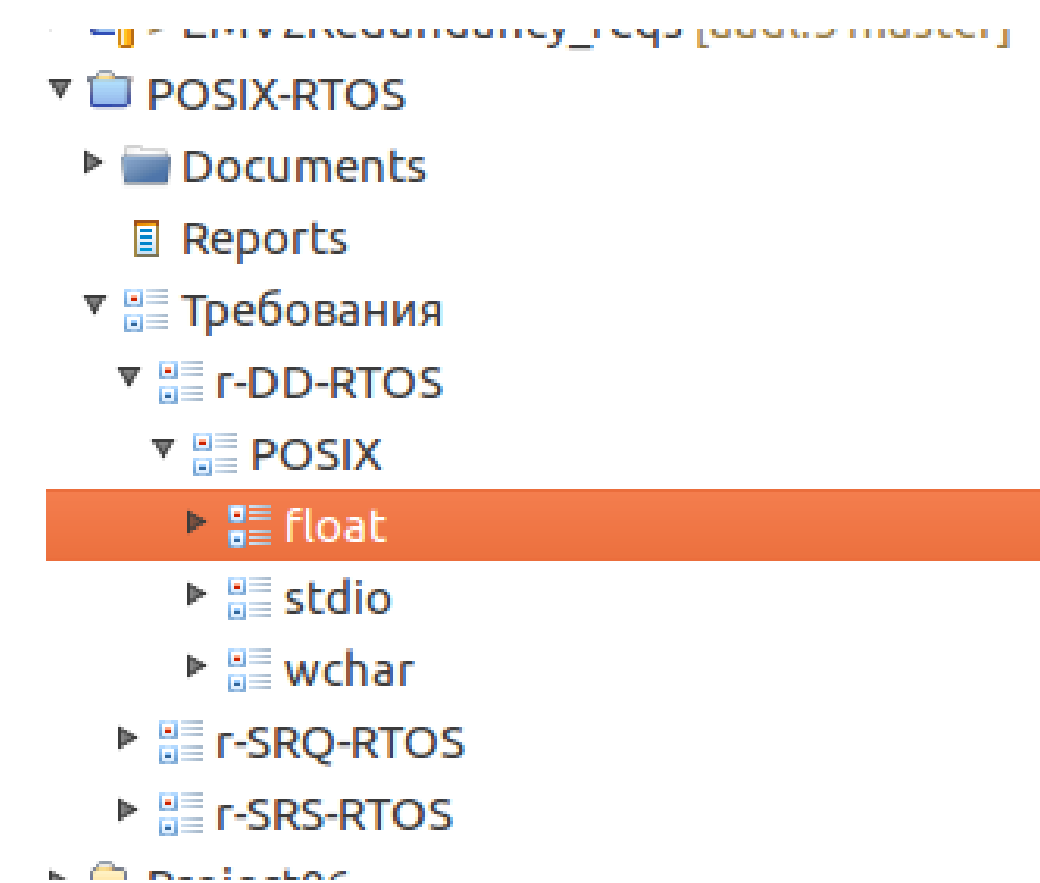
Created by GUY MILEVA 3 days ago.

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:	-	% Done:	
Category:	Administration		
Target version:	-		
Resolution:			

Description

Currently, you can perform administrative operations to projects on admin#projects page, except for close page to close/reopen the project.

I think close/reopen button should also be on admin#projects page because close/reopen and archive/unarchive



The Open Group Base Specifications Issue 7
IEEE Std 1003.1-2008, 2016 Edition
[Copyright](#) © 2001-2016 The IEEE and The Open Group

POSIX.1-2008 is simultaneously IEEE Std 1003.1™-2008 and The Open Group Issue 7. This 2016 Edition includes IEEE Std 1003.1-2008/Cor 1-2013 incorporated into IEEE Std 1003.1-2008 (the base document). The Tech discovered since the approval of IEEE Std 1003.1-2008

Introduction

This introduction is not part of IEEE Std 1003.1, 2016 Edition, Standard Operating System Interface (POSIX).

This standard was developed, and is maintained, by a joint working group Applications Standards Committee, members of The Open Group, and Committee 1. This joint working group is known as the Austin Group.

The Austin Group arose out of discussions amongst the parties which started in and formation of the group in September 1998. The purpose of the Austin Group open systems interfaces that are the POSIX® 1003.1 (and former 1003.2) star Single UNIX Specification

ИСТОЧНИКИ ТРЕБОВАНИЙ

- Техническое задание (а по-английски — requirements specification, спецификация требований).
- Стандарты, регламентирующие характеристики, функции и состав систем, работающих в определенной предметной области.
- Внутренние ограничения задач, не соблюдая которые, невозможно решить их правильно.
- Анализ уже существующих систем для решения схожих задач.
- Требования сформулированные на основе явно высказываемых пожеланий пользователей системы
- Невысказанные явно потребности и нужды пользователей создаваемой системы.

РАБОТА С ТРЕБОВАНИЯМИ

1. Спецификация требований
 2. Цель проекта
 3. Архитектура
 4. Функциональные и нефункциональные требования
 5. Неполнота / противоречивость
- Выделение требований
 - Систематизация и описание требований
 - Валидация и верификация требований.



ХАРАК ТЕРИС ТИКИ

Адекватность	Соответствие реальным потребностям пользователей ПО.
Однозначность	Отсутствие двусмысленностей и возможностей разного толкования.
Полнота	Отражение в требованиях всех существенных потребностей и всех ситуаций, в которых система должна будет функционировать
Непротиворечивость	Согласованность между разными элементами требований.
Систематичность представления	Должны быть описаны в рамках некоторой системы с четким указанием места каждого требования среди остальных, с определением связей и зависимостей между ними и приоритетности для различных заинтересованных лиц.
Прослеживаемость	Четко определенные связи с модулями разрабатываемой системы, частями проектной документации и тестами и т.д.
Проверяемость	Возможность для каждого требования однозначно установить при помощи некоторых действий, выполнено это требование или нет.
Модифицируемость	Возможность внесения изменений в набор требований с максимально быстрым отслеживанием последствий такой модификации



ИНСТРУМЕНТЫ

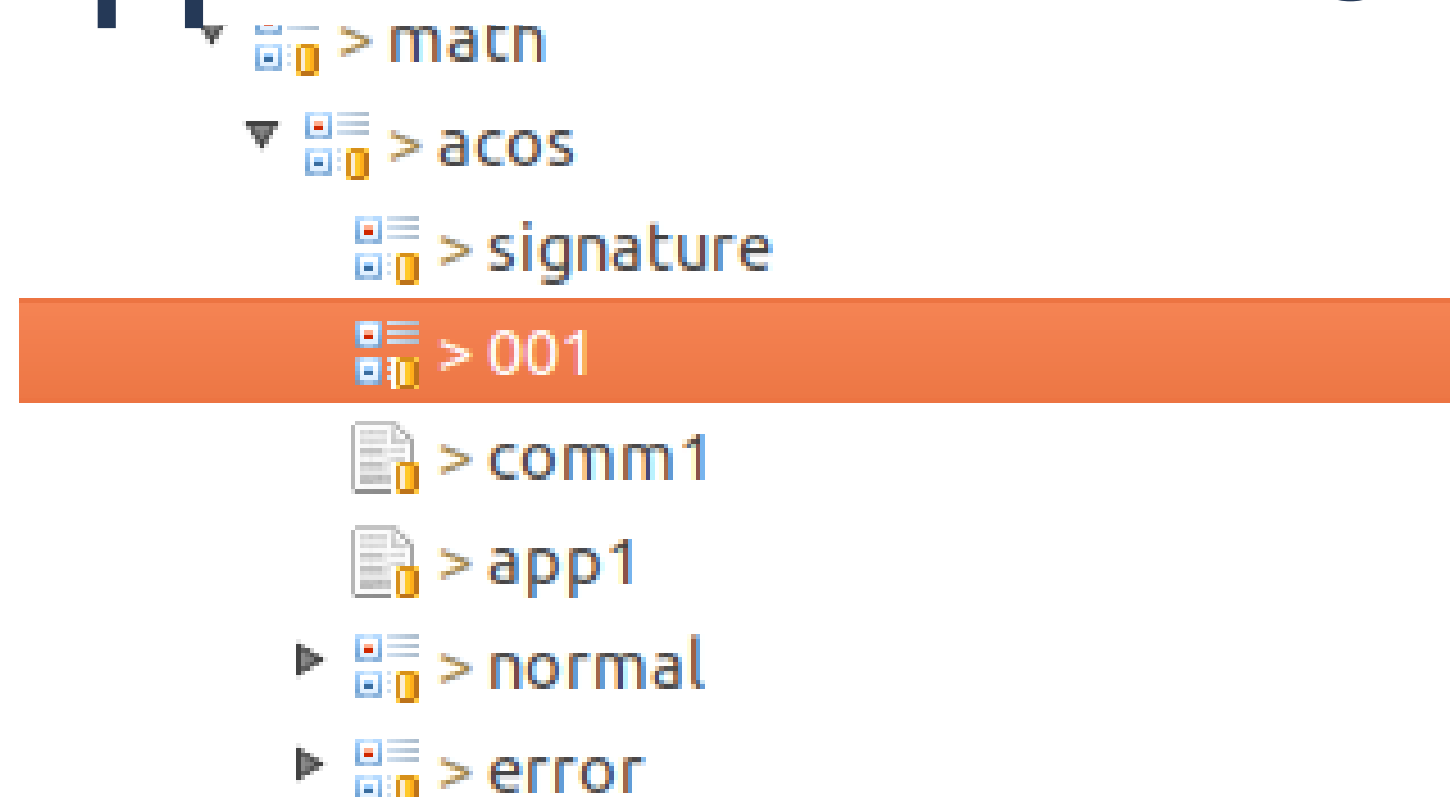
- Неспециализированные — офисные и т. п. (Google doc)
- Инструменты для поддержки разработки проектов (Redmine), системы Issue-трекеры
- Инструменты поддержки разработки полного цикла - Jama, Polarion
- Специализированные продукты
– IBM Doors, Doors NG, ProR, **Requality**



РАЗРАБОТКА КАТАЛОГА ТРЕБОВАНИЙ

- С нуля — описывая отдельные требования и формируя их в определенную структуру
- Из имеющихся документов — спецификации на ПО, технического задания, документов по регулятивным ограничениям

ВЫДЕЛЕНИЕ ТРЕБОВАНИЙ



These functions shall compute the principal value of the arc cosine of their argument x . The value of x should be in the range $[-1, 1]$.

An application wishing to check for error situations should set *errno* to zero and call *feclearexcept*(FE_ALL_EXCEPT) before calling these functions. On return, if *errno* is non-zero or *fetestexcept*(FE_INVALID | FE_DIVBYZERO | FE_OVERFLOW | FE_UNDERFLOW) is non-zero, an error has occurred.

Представляемое число при этом вычисляется по следующим правилам

- если $E \neq 0$ и $E \neq 2^k - 1$ (порядок не состоит из одних нулей или одних единиц)

$$x = (-1)^S \cdot 2^{(E-B)} \cdot (1 + M/2^{n-k-1})$$

это нормализованные числа

- если $E = 0$

$$x = (-1)^S \cdot 2^{(-B+1)} \cdot (M/2^{n-k-1})$$

это денормализованные числа

- если $E = 2^k - 1$

при $M = 0$, $x = (-1)^S \cdot \infty$ (используются для представления бесконечных или

ВИДЫ ДЕЯТЕЛЬНОСТИ И ДОКУМЕНТЫ

Вид деятельности	Первичные документы	Вторичные документы
Анализ требований	Модели предметной области, концепция системы, составленные заказчиками и пользователями требования	Описание требований к ПО
Проектирование	Требования к ПО	Описание архитектуры, проектная документация
Кодирование	Проектная документация	Код, проектная документация на отдельные компоненты
Тестирование	Требования к ПО, проектная документация, код	Тестовые планы и наборы тестовых вариантов

ЛИТЕРАТУРА

1. Роман Савин – «Тестирование Дот Ком, или Пособие по жестокому обращению с багами в интернет-стартапах»
2. Сэм Канер – «Тестирование программного обеспечения»
3. Джеймс Уиттакер – «Как тестируют в Google»
4. Стив Макконнелл – «Совершенный код»
5. Борис Бейзер – «Тестирование черного ящика. Технологии функционального тестирования программного обеспечения и систем»
6. Иан Соммервилл – «Инженерия программного обеспечения»

СПАСИБО! ВОПРОСЫ?



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ