**Тестирование документации**

**Документация**– это еще одна составляющая программного продукта любой уважающей себя организации, занимающейся разработкой программного обеспечения. Но не все организации уделяют достаточное количество времени разработке стоящей документации. Очень часто нам приходится иметь дело с толковым программным продуктом и невзрачным, непонятным и беспомощным документным сопровождением.

Попробуем собрать воедино критерии тестирования, образующие квинтэссенцию качественной документации. Думаю, будет справедливым, если мы опустим такое всем понятное правило, как грамматика, так как не в ней одной таится успешный релиз.

В целом, документация создается для возможности грамотно и без паники найти выход или решение из любой возникшей проблемной ситуации человеку из самым низким знанием принципов разработки программного обеспечения. От этого принципа необходимо отталкиваться, продумывая содержание и структуру наших мануалов.

И так, начнем:

• **Полнота и соответствие действительности.**Любая документация должна содержать описание именно той функциональности, которая присутствует в приложении. И данное описание должно касаться абсолютно всей функциональности, а не только наиболее значимой.

• **Навигация.**И не просто навигация, а удобная навигация. У пользователя никогда не должно возникать проблем с поиском необходимой ему информации. Древовидная структура файлов, закладки и прочее должны быть на видном месте сразу, как пользователь открывает документ. Алфавитный указатель, поиск – должно присутствовать все, что поможет найти решение или описание проблемы.

• Из пункта выше вытекает **структурированность документации**. Все документы должны находится в полном порядке, по разделам. Также, текст должен быть с четкой структурой, чтобы можно было в любой момент вспомнить, где остановился или понять, в каком абзаце содержится именно та информация, которая нам необходима.

• **Инструкции должны присутствовать везде**. Даже при выполнении абсолютно одинаковых манипуляций с программой необходимо пошаговое описание действий во всех случаях. Это может быть как прямое повторение инструкций, так и ссылка на уже существующие.

• **Термины и их значение.** В любой документации может использоваться масса терминов, аббревиатур и сокращений. Каждый из них должен иметь свое значение и расшифровку.

• **Доступность пользователю**. Документация должна быть максимально понятной для любой целевой аудитории.

• Если документация создана и для иностранных пользователей,то необходимо **привлечение специалистов**данного **лингвистического сектора**, вплоть до носителей языка.

**Основные принципы тестирования требований**

* Тестирование требований лучше проводить до старта разработки. Для этого нужно рассчитать необходимое время на проверку и заморозить тестируемую документацию до окончания проверки.
* Проводить тестирование требований могут как аналитики, так и тестировщики. Однако, для достижения лучшего результата описание и проверку требований следует поручать разным людям.
* Выявление дефектов по документации ничем не отличается от выявления дефектов по продукту: баги следует заносить в систему баг-трекинга как обычно.
* В том случае, когда проверка требований ведется параллельно с разработкой, крайне желательно предупредить команду разработки о найденных дефектах (чтобы они могли вовремя исправить ошибку).
* Уровень детализации требований (как и глубина тестирования) сильно зависит от уровня проекта. Нет смысла проверять время реакции на кнопку в проекте, который только запустился (если это, конечно, не относится к ключевому функционалу).

Существует еще много требований к составлению и тестированию документации. Сегодня мы рассмотрели основные положения. Но главное правило, которое поможет нам – это умение ставить себя на место пользователя, попавшего в определенную проблемную ситуацию.

**Свойства качественных требований**

В процессе тестирования требований проверяется их соответствие определённому набору свойств:

* **Атомарность, единичность**(atomicity). Требование является атомарным, если его нельзя разбить на отдельные требования без потери завершённости и оно описывает одну и только одну ситуацию.
* **Непротиворечивость, последовательность**(consistency). Требование не должно содержать внутренних противоречий и противоречий другим требованиям и документам.
* **Недвусмысленность**(unambiguousness, clearness). Требование должно быть описано без использования жаргона, неочевидных аббревиатур и расплывчатых формулировок, а также должно допускать только однозначное объективное понимание и быть атомарным в плане невозможности различной трактовки сочетания отдельных фраз.
* **Обязательность, нужность** (obligatoriness) **актуальность** (up-to-date). Если требование не является обязательным к реализации, то оно должно быть просто исключено из набора требований. Если требование нужное, но «не очень важное», для указания этого факта используется указание приоритета (см. «проранжированность по...»). Также должны быть исключены (или переработаны) требования, утратившие актуальность.
* **Прослеживаемость**(traceability). Прослеживаемость бывает вертикальной (vertical traceability) и горизонтальной (horizontal traceability). Вертикальная прослеживаемость позволяет соотносить между собой требования на различных уровнях требований, горизонтальная - соотносить требование с тест-планом, тест-кейсами, архитектурными решениями и т.д. Для обеспечения прослеживаемости часто используются специальные инструменты по управлению требованиями (requirements management tool) и/или матрицы прослеживаемости (traceability matrix).
* **Модифицируемость** (modifiability) - это свойство характеризует внесения изменений в отдельные требования и в набор требований. Можно говорить о наличии модифицируемости в том случае, если, при доработке требований, искомую информацию легко найти, а её изменение не приводит к нарушению иных описанных в этом перечне свойств.
* **Проранжированность по важности, стабильности, срочности** (ranked for importance, stability, priority). Важность характеризует зависимость успеха проекта от успеха реализации требования. Стабильность характеризует вероятность того, что в ближайшем будущем в требование не будет внесено никаких изменений. Срочность определяет распределение по времени усилий проектной команды по реализации того или иного требования.
* **Корректность** (correctness) ипроверяемость(verifiability). Фактически, эти свойства вытекают из соблюдения всех вышеперечисленных (или ,можно сказать, они не выполняются, если нарушено хотя бы одно из вышеперечисленных). В дополнение, можно отметить, что проверяемость подразумевает возможность создания объективного тест-кейса (тест-кейсов), однозначно показывающего, что требование реализовано верно и поведение приложения в точности соответствует требованию.

**Техники тестирования требований**

Тестирование документации и требований относится к разряду нефункционального тестирования (non-functional testing). Основные техники такого тестирования в контексте требований таковы:

* **Взаимный просмотр** (peer review) или «рецензирование» является одной из наиболее активно используемых техник тестирования требований и может быть представлен в одной из трёх следующих форм (по мере нарастания его сложности и цены):
* **Беглый просмотр** (walkthrough) может выражаться как в показе автором своей работы коллегам с целью создания общего понимания и получения обратной связи, так и в простом обмене результатами работы между двумя и более авторами с тем, чтобы коллега высказал свои вопросы и замечания. Это самый быстрый, дешёвый и часто используемый вид просмотра.

Для запоминания: аналог беглого просмотра — это ситуация, когда вы в школе с одноклассниками проверяли перед сдачей сочинения друг друга, чтобы найти описки и ошибки.

* **Технический просмотр** (technical review) выполняется группой специалистов. В идеальной ситуации - каждый специалист должен представлять свою область знаний. Тестируемый продукт не может считаться достаточно качественным, пока хотя бы у одного просматривающего остаются замечания.

Для запоминания: аналог технического просмотра — это ситуация, когда некий договор визирует юридический отдел, бухгалтерия и т.д.

* **Формальная инспекция** (inspection) представляет собой структурированный, систематизированный и документируемый подход к анализу документации. Для его выполнения привлекается большое количество специалистов. Само выполнение занимает достаточно много времени, потому этот вариант просмотра используется достаточно редко (как правило, при получении на сопровождение и доработку проекта, созданием которого ранее занималась другая компания).

Для запоминания: аналог формальной инспекции — это ситуация генеральной уборки квартиры (включая содержимое всех шкафов, холодильника, кладовки и т.д.).

* **Вопросы.**Следующей очевидной техникой тестирования и повышения качества требований является использование (повторное) техник выявления требований, а также (как отдельный вид деятельности) — постановка вопросов. Если хоть что-то в требованиях вызывает у вас непонимание или подозрение, то задавайте вопросы. Можно спросить представителей заказчика или же обратиться к справочной информации. По многим вопросам можно обратиться к более опытным коллегам при условии, что у них имеется соответствующая информация, ранее полученная от заказчика. Главное, чтобы Ваш вопрос был сформулирован таким образом, чтобы полученный ответ позволил улучшить требования.