

Требования

**для программного обеспечения:
рекомендации по сбору
и документированию**

Илья Корнипаев

УДК 004.588
ББК В32.50.41.25
К67

Корнипаев, Илья
К67 Требования для программного обеспечения: рекомендации по сбору и документированию — М.: Издательство «Книга по Требованию», 2016. — 118 с.

ISBN 978-5-517-60923-6.

Эта книга о том, как собирать, документировать и проверять требования. Она рассчитана на самый широкий круг читателей: начинающих аналитиков, проектировщиков, архитекторов, разработчиков, тестировщиков, руководителей проектов и других специалистов, задействованных в проектах по разработке программного обеспечения (ПО) на ранних стадиях.

Читатель, независимо от того, сотрудник ли он компании-разработчика, является ли он представителем заказчика или работает в ИТ-отделе компании, не связанной с разработкой ПО, может найти в книге полезные для себя советы и рекомендации.

Книга написана на основе более чем пятнадцатилетнего опыта автора, а также по материалам авторских курсов по разработке и управлению требованиями.

Оглавление

Предисловие	5
Главные вопросы	5
Зачем	5
Для кого	5
Что	6
Как читать эту книгу	6
Структура книги	6
Ограничение ответственности	7
Благодарности	7
Введение	8
Что такое требования?	10
Коммуникация при разработке систем	11
Область проблем и область решений	12
Количество требований и выбор решения	16
Проблемы, связанные с преждевременным переходом в область решений	18
Управление ожиданиями заказчика	19
Типы требований	20
Требования и качество	22
Требования и моделирование	24
Требования и изменения	26
Источник изменений — заказчик	26
Рынок как источник изменений	27
Разработчики как источник изменений	28
Другие характерные трудности	29
Высокая неопределенность	29
Разный культурный и образовательный уровень участников проекта	30
Противоречия	31
Использование естественного языка	31
Объем и глубина проработки требований	34
Обзор процесса разработки требований	35
Глава 1. Определение заинтересованных сторон и их представителей	37
Заинтересованные стороны в проекте	37
Определение представителей заинтересованных сторон для сбора требований	38
Диаграмма взаимодействия	40
Выбор представителя заинтересованной стороны	42
Заинтересованные стороны и их цели	44
Заинтересованные стороны и их проблемы	46
Заключение	49
Глава 2. Сбор требований	50
Обзор источников требований	50
Планирование сбора требований	50
Люди как источник для сбора требований	52
Подготовка к встрече	53

Интервью	58
С чего начинается встреча?	58
Как сесть?	60
Установить контакт	61
Введение	62
Вопросы и ответы	63
Окончание встречи	67
После встречи	67
Телефонные переговоры	68
Переписка	70
Групповые встречи	71
Семинар рабочей группы (Requirements Workshop)	72
Опросы	74
Другие способы получения требований	76
Наблюдение за работой людей	76
Временное выполнение аналитиком текущей работы будущих пользователей системы	78
Эксперты и консультанты	79
Системы	80
Документы	83
Обращения в службу поддержки	85
Прототипы	85
Опасные прототипы	88
Заключение	89
Глава 3. Работа с собранными требованиями	91
С чего начинается хороший документ	92
Структура требований	95
Текст требований	97
Атрибуты требований	102
Ключевые требования	104
Ключевые критерии производительности	105
Заключение	108
Глава 4. Проверка требований	109
Важность проверок	109
Виды проверок	110
Неформальные проверки коллегами (Peer Review)	111
Критерии проверки текста требований	113
Критерии законченного набора требований	114
Еще немного о проверках	114
Заключение	116
Литература	117

Предисловие

Моим учителям, научившим меня учиться.

Главные вопросы

В самом начале книги я постараюсь ответить на основные для себя вопросы: «зачем?», «для кого?» и «что именно?». Итак, начнем:

Зачем

Вопрос «Зачем?». Это поистине самый главный вопрос аналитика. Поэтому человек, который садится писать книгу, должен ответить себе на вопрос «зачем?», если он все еще претендует на то, чтобы считаться профессиональным аналитиком. Я взялся за эту работу в первую очередь для себя, потому что мне важно доказать себе самому, что я могу это сделать, что мне это нужно, мне это интересно. Сам, один, без принуждения, без денег, не рассчитывая ни на что. С другой стороны, хотелось бы, чтобы то, что ты делаешь, было бы полезно еще кому-то, поэтому я буду стараться сделать книгу максимально простой и компактной, ибо сложных и больших книг хватает. Возможно, к тому времени, как эта работа увидит свет, будут доступны книги и лучше, и проще, но я попытаюсь написать простую и полезную книгу. О том, насколько мне это удалось, судить вам.

Для кого

Эта книга написана прежде всего для начинающих аналитиков, работающих в проектах по разработке программного обеспечения (ПО). Здесь и далее по тексту книги под аналитиками (IT-аналитиками) я подразумеваю людей, ответственных за сбор и разработку требований, независимо от того, какую еще работу они выполняют. Надеюсь, эта книга будет также полезна другим специалистам, занятым как разработкой ПО, так и разработкой любых других систем (содержащих или не содержащих ПО). К ним можно отнести инженеров, разработчиков, программистов, тестировщиков, специалистов QA, специалистов службы поддержки, руководителей проектов и других руководителей.

Я старался писать эту книгу так, чтобы она была понятна любому человеку, участвующему в разработке любой системы, как со

стороны заказчика, так и со стороны команды разработки, независимо от его образования и опыта. Однако стоит отметить, что опыт участия в проектах по разработке систем вообще и ПО в частности может существенно облегчить читателю понимание данного материала.

Что

Когда я провожу тренинги или читаю курс в академии, первый вопрос, который я задаю аудитории: «Кто из вас делал успешные проекты без формальных требований, без документа, карточек и т.п.?» Обычно поднимается несколько рук. Я поднимаю руку тоже. Да, я делал успешные проекты без формальных требований, потому что требования — это только **инструмент**, в умелых руках он помогает, а в неумелых — нет. Можно сделать успешный проект и без формализованных требований. С другой стороны, нет универсального механизма, универсального подхода, который бы детально, пошагово описывал 100%-но надежный путь получения успеха в проекте по разработке ПО. Не ждите, что в этой книге вы его найдете. В ней вы можете найти ряд важных, на мой взгляд, концепций, которые, возможно, помогут вам улучшить понимание того, что происходит на вашем проекте, и рекомендаций, которые, будучи примененными к месту, могут вам помочь добиться лучшего результата. Такова жизнь.

Как читать эту книгу

Эта книга писалась так, чтобы читатель, прочитав эту книгу от начала до конца, смог получить для себя максимальную пользу. Поэтому рекомендуется читать эту книгу именно таким образом.

Структура книги

Во введении обсуждаются вопросы общего характера, связанные со сбором и формализацией требований, такие как: отношение к требованиям как к инструменту достижения успеха проекта, определение проблемы и ее решения, связь требований и качества, изменения требований в ходе проекта и т.д. Эта глава необходима в книге для того, чтобы правильно расставить акценты и помочь вам определиться, использовать ли вам требования на проекте или нет. В заключение в этой главе кратко описывается процесс сбора

и разработки требований, который далее раскрывается в последующих главах. Следующие главы книги дают ответы на вопросы:

- Кто такие заинтересованные стороны и как их определить, проблемы, цели и задачи заинтересованных сторон;
- Как собирать требования, общаясь с людьми и используя другие источники;
- Как писать требования и как их структурировать;
- Как проверять требования.

Ограничение ответственности

Книга основана на реальных событиях. Все имена вымышлены. Все совпадения случайны.

Автор не несет ответственности за моральные страдания читателя при прочтении этой книги.

Автор не несет ответственности за неумелое применение этой книги читателем и нанесенный действиями читателя ущерб.

Все торговые марки и знаки принадлежат их уважаемым обладателям.

Благодарности

Выражаю огромную благодарность всей моей семье, людям, которые поддерживали меня и вдохновляли на эту работу. Особенно моей супруге Оле и брату Михаилу, принявшему также участие в редактировании книги.

Большое спасибо Вадиму Арнольдовичу Перекресту — за то, что поверил в меня и дал возможность преподавать, Елене Арсеньевой за мои тренинги, Александру Байкину — за Летние Аналитические Фестивали, отзывчивость и помощь.

Также выражаю свою благодарность людям, принявшим участие в моем становлении и росте в профессии аналитика: Павлу Яновскому, Павлу Грачеву, Олегу Ночке, Александру Заровскому, Алексею Емельянову, Алексею Мартынишину, Вагану Варданяну, Сергею Сорокопудову и Павлу Дранцеву.

Моя благодарность компаниям: VDI (и отдельно Анатолию Гавердовскому), «Ренессанс Капитал», «ВТБ Капитал», Eram Systems, «АТ Консалтинг», Luxoft и другим компаниям, а также их сотрудникам, которые терпели меня на протяжении многих лет.

Введение

Однажды на конференции Better Software я имел удовольствие слушать Карла Вигерса (Karl Wiegers) — автора книг и статей, посвященных требованиям. Человека, признаваемого всем IT-сообществом одним из столпов теории и практики управления требованиями. Его доклад (в моем вольном переводе) назывался «Космические истины о требованиях для ПО» (Cosmic Truths about Software Requirements). Его выступление произвело на меня тогда сильное впечатление. Я даже попытался его воспроизвести для своих коллег по слайдам Карла, как только вернулся в Москву. Начиная писать эту книгу, я подумал: а ведь это неплохая идея — начать с главного, чтобы потом на страницах книги шаг за шагом раскрывать, объяснять и иллюстрировать на примерах несколько простых, но важных истин:

№ 1. Первая простая истина, которую я давно осознал, состоит в том, что многое уже придумано, и если так — значит, об этом уже где-то написано. Это нужно найти и прочитать, чтобы применить для решения задач, стоящих передо мной. (Не правда ли, звучит банально? ☺) Иными словами, не нужно изобретать велосипед, для того чтобы построить велосипед. Я постараюсь изложить свой собственный опыт на этих страницах, а также дать вам полезные, на мой взгляд, ссылки, чтобы вам было что почитать и над чем подумать. В то же время я не стал делать список литературы огромным, потому что знаю: если список большой, никто им пользоваться не станет (я, например, точно). Примеры из моего личного опыта и опыта моих коллег выделены отступом и курсивом.

№ 2. Вторая простая истина заключается в том, что не существует универсального средства или шаблона, который бы помог решить любую задачу, сделать любой проект. На практике это выражается в том, что существует великое множество различных сред разработки и систем поддержки процесса разработки, включая системы управления требованиями. Также существует как минимум несколько популярных методологий, определяющих тот или иной процесс разработки, и все они применяются. Если взять, к примеру, требования, то вы можете найти несколько различных методологий и стандартов, определяющих процесс разработки требований, шаблоны документов и т.п. У каждого средства, инструмента, методологии,

среды и т.п. находится немало сторонников и противников, поэтому на страницах этой книги я не буду вам навязывать никаких подходов, методов, средств и инструментов. Все, что я буду излагать, — это рекомендации, основанные на собственном опыте и опыте моих коллег, которым я доверяю. Использовать эти рекомендации или нет — это ваше дело. Более того, если вы после прочтения этой книги решите вообще исключить требования как формальный документ из своего процесса разработки, это тоже ваше личное дело. Ведь не секрет, что очень многим командам удавалось делать успешные проекты, не имея на руках формального документа с требованиями. Например, весь мир гибкой разработки ПО (Agile software development) живет с карточками (User Stories), содержание которых весьма и весьма далеко от формальных требований, как их понимают многие авторы и описывают международные стандарты.

№ 3. Еще одна простая истина состоит в том, что старые проверенные средства работают. Лучше использовать их, чем пытаться внедрить что-то новое и модное и при этом загубить проект и свои отношения руководством с клиентом. Иными словами, быть умеренным консерватором не так уж и плохо.

№ 4. Следующая простая истина заключается в том, что в основе любого успешного проекта лежит хорошее взаимопонимание заказчиков и исполнителей. И не секрет, что работа аналитика во многом состоит в том, чтобы понимать и правильно интерпретировать пожелания заказчика, а затем передать это разработчику на понятном для него языке.

№ 5. А если вы согласились с тем, что в основе любого успешного проекта лежит взаимопонимание, то следующая неожиданная истина состоит в том, что работа аналитика, собирающего требования, имеет мало общего с инженерной работой. Сбор требований — это работа, похожая во многом на работу доктора: проанализировать состояние больного — поставить диагноз — предложить метод лечения. Перефразируя известную поговорку про врачей, можно сказать: «Плох то аналитик, от беседы с которым клиенту не становится легче». При этом знания в технической области, несомненно, важны. В первую очередь — чтобы общаться с разработчиками,

а также для понимания текущей ситуации на стороне заказчика и анализа причин возникновения проблем.

№ 6. А теперь спросите себя: хотите ли вы доверить свое здоровье доктору, который ничего не знает? Ответ очевиден — нет. Тем не менее в моей практике я неоднократно сталкивался с тем, что на позицию аналитика выдвигаются люди, которые явно обладают недостаточными знаниями как в предметной области проекта, так и в области применяемых технологий разработки ПО. Это, на мой взгляд, серьезно усложняет задачу команды и увеличивает проектные риски (которые нередко просто перекадываются на плечи клиента и разработчиков). А истина заключается в том, что аналитик должен знать и предметную область проекта, и применяемые технологии разработки.

№ 7. Отсюда мы можем плавно перейти к заключительной истине. Для того чтобы добиться успеха, необходимо прилагать усилия. Работа аналитика — это всегда творческий труд. Достаточно сложно представить себе ситуацию, когда работа аналитика превратилась в рутину, не требующую от него включения в работу головы, поэтому на каждом проекте, на каждой задаче аналитик должен прилагать усилия для того, чтобы помочь своей команде добиться хорошего результата. На мой взгляд, простая фиксация того, что требует заказчик, в формальном документе просто недопустима. Если вы не готовы постоянно прилагать усилия, работа аналитика не для вас.

Итак, эта книга — набор рекомендаций, но не догм, о том, в каком направлении, на мой взгляд, аналитик может приложить свои усилия, чтобы помочь своей команде сделать успешный проект.

Что такое требования?

В этой книге требования рассматриваются в самом широком контексте. Я намеренно не привожу определение этого понятия, чтобы сразу же не загонять читателя в определенные узкие рамки, характерные для той или иной методологии или стандарта. Под требованиями можно понимать как цели, которых желает достичь ваш клиент, так и формальную постановку задачи разработчикам. Это могут быть как слова, произнесенные вслух, так и документ,