

# Составление программной документация

Хорошая программная документация – будь это документ, содержащий спецификацию требований для программистов или тестеров, технический документ для внутренних пользователей, руководство для использования программного обеспечения или файл подсказок для пользователей – помогает человеку, работающему с программным обеспечением, понять его характерные черты и функции.

Следуйте советам - как писать программную документацию для технических и конечных пользователей.

# Совет 1

*Определите, какая информация должна быть упомянута*

Документы о требованиях к программному обеспечению служат справочным руководством для дизайнеров интерфейса пользователя, программистов, которые пишут код и тестеров, которые проверяют, работает ли программное обеспечение как следует.

Точная информация зависит от самой программы, однако может включать следующее:

файлы, созданные командой разработчиков, базы данных, вызываемые во время программной операции, и служебные программы третьей стороны.

Функции и подпрограммы. Здесь указывается, что делает каждая функция и подпрограмма, включая входные и выходные значения. Программные переменные и постоянные и как они используются в приложении.

Общая структура программы. Для приложений на основе диска вам, вероятно, понадобится описание отдельных блоков и библиотек программы, в то время, как для веб-приложений понадобятся описание страниц, которые использую файлы.

# Совет 2

*Решите, как много документации должно быть в программном коде и как много должно быть отделено.*

Чем больше технической документации создано в программном коде, тем проще будет обновлять этот код, как и документацию, касающуюся различных версий оригинального приложения.

Как минимум документация в программном коде должна объяснять функции, подпрограммы, программные постоянные и переменные. Если программный код довольно длинный, его можно оформить в виде справочного файла, в котором можно делать поиск по ключевым словам или указателям. Это будет большим плюсом для приложений, где логика программы разделена на много страниц и включает номера вспомогательных файлов, как и в определенных веб-приложениях.

Некоторые языки программирования, например Java или NET Framework (Visual Basic.NET, C #), имеют свои собственные стандарты для кода документации. В таких случаях следуйте стандартным указаниям - сколько документации следует включать в программный код.

# Совет 3

## *Выберите подходящий инструмент.*

В какой-то мере это определяется языком, на котором код написан, будь это C++, C#, Visual Basic, Java, или PHP - для каждого существуют свой собственный инструмент. В других случаях используемый инструмент определяется типом требуемой документации. Текстовый редактор «Microsoft Word» – подходящий инструмент для создания отдельных текстовых файлов документации, которая будет простой и краткой. Для длинных текстовых файлов многие разработчики технической документации предпочитают выбирать программу «Adobe FrameMaker».

Файлы подсказки для документации программного кода могут писаться с помощью любого инструмента, например «RoboHelp», «Help and Manual», «Doc-To-Help», «MadCap Flare», или «HelpLogix».

# Написание программной документации для конечных пользователей

Определите **пользователей** приложения для вашей документации.

Хотя функциональные причины для программной документации – помочь пользователям понять, как использовать приложение, есть и другие причины, например, помочь в продвижении товара на рынке, улучшение образа компании и самое главное, уменьшение затрат на техническую поддержку. В определенных случаях документация требуется для соблюдения определенных правил и юридических требований. Ни в коем случае программная документация не должна заменять плохой дизайн интерфейса. Если экран приложения требует много объяснительной документации, то лучше сменить дизайн на что-то более интуитивное.

# Понимайте аудиторию, для которой вы пишете документацию

В большинстве случаев пользователи программного обеспечения мало знают о компьютерах помимо задач приложения. Есть несколько способов определить, как согласовать их потребности с документацией.

1. Посмотрите на профессии, которыми владеют ваши потенциальные пользователи. Системный администратор, скорее всего, будет экспертом в использовании приложений программного обеспечения, в то время как оператор ввода данных, скорее всего, владеет приложением, которое он или она в настоящий момент использует для ввода данных.
2. Посмотрите и на самих пользователей. Хотя их должности в целом определяют то, чем люди занимаются, но бывают значительные различия в том, как определенные должности используются в данной организации. Проведя собеседование с потенциальными пользователями, вы можете сложить свое мнение - соответствует ли название должности выполняемым обязанностям.



Посмотрите существующую документацию. Документация для предыдущих версий программного обеспечения дает примерное понятие, что пользователю нужно знать об использовании программы. Однако помните, что конечные пользователи не заинтересованы в том, как работает программа, им важно знать, что они могут с ней делать. Определите задания, которые необходимы для этой работы и какие задачи должны быть выполнены до того, как эти задания могут быть выполнены.

# Определите соответствующий формат(ы) документации

Программная документация может быть структурирована в одном из двух форматов – справочное руководство и инструкция по пользованию. Иногда лучше выбрать смешанный вариант из этих двух форматов. Справочное руководство призвано пояснять инструментарий программного оборудования (кнопки, таблицы, поле и панель диалога) и как этот инструментарий работает. Многие файлы подсказки написаны в этом формате, а контекстные подсказки помогают показать нужную тему после того как пользователь щелкает на кнопку «помощь» на нужном экране.

Инструкция по пользованию поясняет, как использовать программное обеспечение для выполнения определенной задачи. Инструкция по пользованию часто имеет вид печатного руководства или формат PDF, хотя некоторые файлы подсказки включают темы о том, как выполнять определенную задачу. (Эти темы справки обычно не являются контекстными, хотя могут быть гиперссылками) Инструкция по пользованию часто имеет форму справочника с описание задачи и пошаговой инструкцией.

# Решите, какой должны быть формат (форматы) документации

Программная документация для конечных пользователей может быть одного или нескольких форматов: печатное руководство, документы в формате PDF, файлы подсказки или онлайн-справка. Каждый из этих форматов создан, чтобы показать пользователю, как использовать каждую программную функцию – будь это краткий обзор или руководство. Как в случае с файлами подсказки и онлайн-справкой, документация может иметь демонстрационное видео или текст с картинками. Файлы подсказки и онлайн-справка должны иметь указатели, поиск по ключевым словам, что позволит пользователю быстро найти требуемую информацию. Хотя инструменты для файлов подсказок могут автоматически создавать указатели, лучше это делать вручную, используя термины, которые пользователи, скорее всего, станут искать.

# выберите подходящий инструмент для создания документации

Печатные руководства или формат PDF могут писаться в текстовых редакторах, таких как «Word» или «FrameMaker», в зависимости от длины и сложности руководства.

Файлы подсказок можно писать с помощью таких средств разработки как «RoboHelp», «Help and Manual», «Doc-To-Help», «Flare», «HelpLogix», or «HelpServer».

# Советы

*Текст должен быть простым для чтения, картинки должны располагаться как можно ближе к тексту, к которому они относятся.*

Разбейте документацию на разделы и логические темы. Каждый раздел или тема должна касаться определенного вопроса, будь это одна программа или задача. Смежные вопросы должны быть обозначены «Смотреть также» с гиперссылкой, если требуется.

Все инструменты для создания документации, которые были перечислены выше, могут дополняться программой по созданию скриншотов, например «Snagit», если в документации требуется определенное количество скриншотов. Как и с другой документацией скриншоты должны пояснять, как работает программное обеспечение, а не вводить пользователя в заблуждение.

Также очень важен тон написания документации, особенно если она пишется для конечных пользователей. Используйте второй лицо «вы», вместо третьего лица «пользователи».