## Выработка требований к программному обеспечению и программному модулю

## Понятие требований к программному изделию

Проблемы, которые приходится решать в процессе создания программного изделия, обычно очень сложны. В частности, необходимо достаточно четко описать те возможности, которые должны быть реализованы, и те условия и ограничения, с учетом которых эти возможности должны предоставляться. Такое описание функциональных возможностей создаваемого программного изделия и накладываемых ограничений называется ***требованиями*** к этому изделию.

Требования к программному изделию определяются с учетом потребностей пользователей и функционального назначения программного изделия. Отдельные требования могут быть независимыми друг от друга или связаны между собой.

***Разработка требований*** – (requirements engineering) первый этап создания программного изделия. При этом предполагается, что у организации-заказчика уже существует четкое представление о том, зачем нужно создавать новое программное изделие (автоматизированную систему) и с какими другими изделиями (системами) оно должно взаимодействовать. Другими словами, у заказчика существует стратегия автоматизации предприятия в целом.

Разработка требований включает следующие виды работ:

1)***сбор***требований;

2) ***анализ*** требований;

3) ***документирование***требований.

Значение тщательно обоснованных и полных требований нельзя недооценивать. Такая недооценка, особенно со стороны заказчика и пользователей, приводит к затягиванию сроков разработки и выпуску недоброкачественной продукции. При отсутствии четких требований разработчик делает то, что он хочет или может сделать, а не то, что надо заказчику. При реализации сложных программных изделий до 70 % всех возникающих проблем непосредственно связаны с несовершенством требований и только 30 % являются результатом ошибок в процессе разработки. Нечеткость и неполнота требований, нерешенные вопросы и ошибки, допущенные на этапе их разработки, порождают на последующих этапах создания программного изделия трудные, часто неразрешимые проблемы и, в конечном счете, приводят к неуспеху всей работы в целом.

## Сбор требований

Сбор требований является основой всего процесса создания программного изделия. Первоначально сбор требований выполняют на этапе анализа автоматизируемого объекта, т.е. предприятия-заказчика и его отдельных подразделений. Затем требования могут уточняться на всех этапах жизненного цикла программного изделия.

Основными ***источниками требований*** являются:

- уже действующие на предприятии программные изделия, подлежащие совершенствованию, а если их нет, то коллектив, выполняющий функции, подлежащие автоматизации с учетом существующих документов, методик, стандартов и т.д.;

- представления будущих пользователей о новых возможностях, которые должны быть реализованы в новом программном изделии;

- известные программные изделия других предприятий.

Углубленное изучение нужд пользователей и личный опыт по созданию подобных программных изделий позволяет разработчику самостоятельно формулировать некоторые требования. При этом он должен всегда согласовывать эти формулировки с пользователями, так как ответственность за содержание требований несут именно они. Подавляющее же большинство требований пользователи должны сформулировать сами. Постоянное вовлечение будущих пользователей в процесс выработки решений на протяжении всего проектирования является одним из основополагающих принципов современных технологий разработки программных изделий.

Для сбора требований разработчик может использовать различные ***методы***:

* изучение действующей документации;
* интервьюирование пользователей (беседа);
* анкетирование пользователей;
* протоколирование деятельности пользователей;
* стажировку на рабочем месте;
* "мозговой штурм";
* создание и апробацию прототипа будущего программного изделия;
* использование собственных знаний и опыта.

На первом этапе при активном участии пользователей составляют описание существующей системы с выявлением её недостатков. Если программное изделие создается на "голом месте", то требования разрабатывают на основе анализа действующего ручного метода обработки и представлений пользователей о желаемой технологии функционирования автоматизируемого объекта.

Важным фактором, который необходимо учитывать при разработке программного изделия, является недостаточное понимание пользователями того, что они действительно хотели бы получить от разрабатываемой системы. Многие проектные решения в процессе разработки ПИ подвергаются многократным уточнениям и корректировкам по результатам обсуждений.

Кроме этого, следует иметь в виду то, что хороший специалист (в любой области) все базовые понятия своей профессии считает общеизвестными. Поэтому свои объяснения он начинает с того, чем следовало бы закончить. Прочее - по умолчанию, причем перечень установок, сделанных по умолчанию, остается неизвестным. Такой специалист охотно ответит на все вопросы, но надо еще знать, о чем его спрашивать.

Еще более сложной проблемой является терминологический барьер. Специалист дает объяснения на привычном ему профессиональном жаргоне. Программист, не понимая объяснений, задает свои вопросы на языке, непонятном специалисту. Взаимопонимание требует огромных усилий и достигается не всегда. Наиболее курьезные результаты получаются в тех случаях, когда один и тот же термин означает в двух разных областях совершенно различные понятия.

Независимо от метода, применяемого при сборе первичных сведений, проектировщик обязан получить ответы на следующие вопросы:

- что требуется от системы?

- зачем это нужно?

- кому это необходимо?

- насколько это важно?

- чем можно измерить степень соблюдения требования?

С целью облегчения вовлечения пользователей в процесс разработки, необходимо интенсивно использовать представление проектных решений в графической форме, обеспечивающей достаточную наглядность и не требующей для понимания их сущности специальной подготовки в области информационных технологий.

Большое значение для обеспечения единства понимания между проектировщиками и будущими пользователями, уточнения существующих и выявления дополнительных требований имеет разработка и оценка демонстрационного прототипа. Прототип - это не завершенная, но уже работающая версия программного изделия. Создается на ранних этапах жизненного цикла и используется для исследования и выяснения концепции функционирования и требований к программному изделию, а также для экспериментальной проверки требований, структур данных и алгоритмов. После апробации прототипа техническое описание системы и структуры данных уточняют.

# Анализ требований

После завершения сбора требований осуществляют их ***анализ***, в процессе которого, как минимум, необходимо выявить:

1) одинаковые требования, сформулированные по-разному различными пользователями, а также требования, учтенные в других требованиях;

2) взаимосвязанные требования;

3) противоречивые и взаимоисключающие требования;

4) трудно реализуемые и нереализуемые требования.

Наиболее квалифицированные пользователи могут привлекаться в качестве экспертов для разрешения противоречий между требованиями, выдвинутыми различными группами пользователей.

В результате анализа требований с учетом имеющихся ресурсов (времени на разработку и денежных средств), а так же с учетом взаимоисключающих требований формируют несколько вариантов реализации программного изделия.

## Документирование требований. Уровни детализации требований

Требования формулируются в виде отдельных предложений. Однако сам термин ***требования*** может трактоваться по-разному, так как на разных этапах создания программного изделия требуется разная детализация требований. В некоторых случаях под требованиями понимаются лишь самые обобщенные утверждения о функциональных возможностях и ограничениях программного изделия. Другая крайняя ситуация — детальное описание каждой автоматизируемой функции будущего программного изделия.

Если компания хочет выиграть контракт на разработку большого программного изделия, она вынуждена, пока решение не принято, представлять требования в самом обобщенном виде, чтобы, с одной стороны, удовлетворить требования заказчика, а с другой — иметь возможность для маневра при конкуренции с другими компаниями-разработчиками. После того как контракт выигран, компания должна представить заказчику более подробное описание программного изделия с указанием всех выполняемых ею функций.

Некоторые проблемы, возникающие в процессе разработки требований, порождены отсутствием четкого понимания различия между этими разными уровнями требований.

Чтобы различить требования разных уровней детализации, используют термины***пользовательские требования*** (user requirements) для обозначения обобщенных требований и ***системные требования*** (system requirements) для детализированного описания выполняемых программным изделием функций. Кроме требований этих двух уровней применяется еще более детализированное описание программного изделия, предназначенное для программистов - ***проектная системная спецификация*** (software design specification). Три перечисленных вида требований можно определить следующим образом.

***Пользовательские требования***, как правило, описывают программное изделие в обобщенном виде. Они пишутся в первую очередь для заказчика, поэтому должны описывать функции и ограничения так, чтобы они были понятны даже пользователю, не имеющему специальных технических знаний в области информационных технологий. Эти требования должны определять только внешнее поведение программного изделия, избегая по возможности определения его структурных характеристик. Пользовательские требования должны быть написаны естественным языком с использованием простых таблиц, а также наглядных и понятных схем и диаграмм.

Вместе с тем при описании требований на естественном языке могут возникнуть различные проблемы, связанные с неточностью и "размытостью" требований.

1. *Отсутствие четкости изложения.* Иногда нелегко изложить какую-либо мысль естественным языком четко и недвусмысленно, не сделав при этом текст многословным и трудночитаемым. Естественно, разработчики интерпретируют требования, допускающие двоякое толкование, так, чтобы систему было проще реализовать. Но это толкование может не совпадать с ожиданиями заказчика. Такая ситуация приводит к разработке новых требований и внесению изменений в систему. Это, в свою очередь, ведет к задержке сдачи готовой системы и ее удорожанию.

2. *Смешение требований.* В пользовательских требованиях отсутствует четкое разделение на функциональные и нефункциональные требования, на системные цели и проектную информацию.

3. *Объединение требований.* Несколько различных требований к программному изделию могут описываться как единое пользовательское требование.

***Системные требования*** — это более детализированное описание пользовательских требований. Системные требования описывают систему максимально подробно, включая ее входные и выходные данные, исключения из правил обработки данных и т.д. Такое описание также называют ***функциональной спецификацией***. Она служит основой для заключения контракта между покупателем программного изделия и его разработчиком и предназначена для руководящего технического состава компании-разработчика и для менеджеров проекта.

Системные требования определяют, что должна делать система, не показывая при этом, как и пользовательские требования, механизма реализации. Но, с другой стороны, для полного описания системы требуется детализированная информация о ней, которая по возможности должна включать всю информацию о системной архитектуре. На то существует ряд причин.

1. Первоначальная архитектура системы помогает структурировать спецификацию требований. Системные требования должны описывать подсистемы, из которых состоит разрабатываемая система.

2. В большинстве случаев разрабатываемая система должна взаимодействовать с уже существующими системами. Это накладывает определенные ограничения на архитектуру новой системы.

3. В качестве внешнего системного требования может выступать условие использования для разрабатываемой системы специальной архитектуры.

В документе, содержащем требования к программному изделию, желательно отделять пользовательские требования от более детализированных системных требований. Иначе неподготовленный читатель пользовательских требований может "потонуть" в технических подробностях, понимание которых требует определенных профессиональных знаний.

***Проектная системная спецификация*** — это подробное описание структуры программного изделия, которое будет основой для его проектирования и последующей реализации. Эта спецификация дополняет и детализирует функциональную спецификацию (системные требования). Проектная системная спецификация является документом, который ориентирован на разработчиков программного обеспечения.

Чтобы избежать проблем неоднозначности понимания требований, разработаны специальные методы их формализованного описания, которые структурируют спецификацию и уменьшают размытость определений. Кроме этого, разработаны другие подходы, например специальные языки описания требований, которые используются относительно редко.

# Виды требований к программному изделию

Требования к программному изделию можно разделить на группы:

1) Требования к изделию (системе) в целом;

2) Требования к отдельным видам обеспечения;

3) Требования к отдельным подсистемам или модулям;

4) Требования к отдельным функциям;

5) Организационные требования;

6) Внешние требования, характеризующие предметную область (окружающую среду).

Кроме этого требования делят на функциональные и нефункциональные.

***Функциональные требования*** определяют поведение программного изделия: это перечень действий и функций, которые должны быть реализованы.

***Нефункциональные требования*** описывают характеристики программного изделия и его окружения, а не его поведение. Здесь может быть приведен перечень ограничений, накладываемых на выполняемые действия и функции. Например, временные ограничения, ограничения на процесс разработки системы, стандарты и т.д.

В действительности четкой границы между этими типами требований не существует. Например, требования, касающиеся безопасности системы, можно отнести к нефункциональным. Однако при более детальном рассмотрении такое требование можно отнести к функциональным, поскольку оно порождает необходимость включения в систему средства авторизации пользователя. Поэтому, рассматривая далее эти виды требований, мы должны всегда помнить, что данная классификация в значительной степени искусственна.

## Функциональные требования

Функциональные требования зависят от типа разрабатываемой системы и от потребностей пользователей. Эти требования описывают поведение системы, действия и функции, которые она выполняет. Должно быть определено, как программа должна реагировать на те или иные входные данные, как она должна себя вести в определенных ситуациях. В некоторых случаях указывается, что программа не должна делать.

В функциональных требованиях также определяют:

- работа каких пользователей должна автоматизироваться;

- какая информация должна храниться;

- по каким формулам и алгоритмам и в какой последовательности она должна обрабатываться;

- какие документы должны формироваться.

Функциональные требования к системе в целом определяются целями ее создания.

Спецификация (полный набор) функциональных требований должна быть комплексной и непротиворечивой. Комплексность подразумевает описание (определение) всех необходимых функций (системных сервисов).

Непротиворечивость означает отсутствие несовместимых и взаимоисключающих требований. На практике для больших и сложных систем крайне трудно разработать комплексную и непротиворечивую спецификацию функциональных требований. Причина кроется частично в сложности самой разрабатываемой системы, а частично — в несогласованных опорных точках зрения на то, что должна делать система. Эта несогласованность может не проявиться на этапе первоначального формулирования требований — для ее выявления необходим более глубокий анализ спецификации. Когда несогласованность системных функций проявится на каком-либо этапе жизненного цикла программного изделия, в системную спецификацию придется внести соответствующие изменения.

# Нефункциональные требования

Нефункциональные требования не связаны непосредственно с функциями, выполняемыми системой. Они связаны с такими интеграционными свойствами системы, как ***надежность, производительность****(время отклика),* ***размер системы***и др. (см. рис. 8.1).

Другими словами, нефункциональные требования связаны с критериями, определяющими

1) качество выполнения отдельных функций и программного изделия в целом;

2) условия функционирования системы;

3) ограничения на систему, например на пропускную способность устройств ввода-вывода, на форматы представления данных и т.д.

Как и для функциональных требований, различают нефункциональные требования к системе в целом и требования к отдельным ее компонентам и видам обеспечения. Более того, каждое функциональное требование может иметь несколько своих нефункциональных.

Нефункциональные требования, относящиеся к системе в целом, как правило, более значимы и критичны, чем отдельные функциональные требования. Ошибка, допущенная в функциональном требовании, может снизить качество системы, в то время как ошибка в нефункциональных требованиях может сделать систему неработоспособной.

***Требования к изделию (системе)*** описывают эксплуатационные свойства программного изделия. Сюда относятся требования к производительности системы, объему необходимой памяти, надежности, переносимости системы на разные компьютерные платформы, удобству эксплуатации и т.д.

***Организационные требования*** отображают взаимоотношения между заказчиком и разработчиком. Эти требования относятся не к самому программному изделию, а к технологическому процессу его создания, стандартам разработки, к языку программирования, к методам проектирования и т.д. Например, может быть задан перечень стандартов качества, накладываемых на процесс разработки, или может быть указано, что проектирование должно выполняться только определенными инструментальными средствами (CASE-средствами), и приведено описание процесса проектирования, которому необходимо следовать.

Организационные требования включают также выходные требования, которые определяют сроки изготовления программного изделия, перечень необходимой документации и т.д.

***Внешние требования*** учитывают факторы, внешние по отношению к разрабатываемой системе и процессу ее разработки. Они включают требования, определяющие взаимодействие данной системы с другими системами, юридические требования, следование которым гарантирует, что система будет разрабатываться и функционировать в рамках существующего законодательства, а также этические требования. Последние должны гарантировать, что система будет приемлемой для пользователей или заказчика.

Нефункциональные требования основываются на бюджетных ограничениях, учитывают организационные возможности компании-разработчика и возможность взаимодействия разрабатываемой системы с другими информационными системами, а также такие внешние факторы, как правила техники безопасности, законодательство о защите интеллектуальной собственности и т.п.

Основная проблема нефункциональных требований состоит в том, что их выполнение трудно проверить. Часто они пишутся для того, чтобы отобразить общие цели заказчика системы, такие, как простота эксплуатации, возможность восстановления после сбоев или быстрый ответ на запросы пользователя. Реализация подобных требований может оказаться сложной для системных разработчиков, поскольку они нечетко сформулированы и открывают простор для различных толкований.

# Функции разработчиков при определении требований

К созданию программных изделий причастны следующие категории лиц: пользователи, проектировщики и прочие специалисты по информационным технологиям.

Термин "пользователи" относится к широкой группе людей, заинтересованных в новой системе. Пользователи являются источником большинства требований и другой дополнительной информации, касающейся области применения ПИ. Они отвечают за правильность формулирования и полноту выявления требований. В тех случаях, когда возникают противоречия между требованиями, эти конфликты должны разрешать пользователи, обладающие более глубокими знаниями предметной области, а также ответственные за финансирование проекта.

В процессе определения требований к ПИ основная задача проектировщиков - четко сформулировать требования пользователей и ожидаемые результаты. Проектировщики помогают пользователям лучше осознать и сформулировать свои требования к новой системе. Они должны разумно подходить к выбору пользователей для получения тех или иных сведений. Проектировщики отвечают за документирование требований и за соответствие им остальных проектных документов.

В определении требований к ПИ могут принимать участие также специалисты по планированию ресурсов вычислительных средств, оценке качества выполнения функций ПИ и анализу риска. Их профессиональные знания необходимы для того, чтобы убедиться в реалистичности предъявляемых требований, оценить затраты на их соблюдение и влияние на сложившуюся информационную инфраструктуру.

<https://habr.com/ru/articles/501468/>

<https://habr.com/ru/articles/432844/>