

## Классификация требований

Тип	Описание	Пример из требований к отчету загрузенность сотрудников по дереву задач
Бизнес– требования	Требования, содержащие высокоуровневые цели организации или заказчиков продукта. Содержат бизнес соображения, исходя из которых, целесообразно разрабатывать / дорабатывать данный продукт. Требования содержат информацию, описывающую финансовые, рыночные или другие отношения коммерческого характера, которые клиенты или разработчики собираются получить от использования продукта.	Сократить в 100 раз трудозатраты Менеджеров по качеству и Руководителей проекта по сбору метрик Ai, Wi проекта с 10 – 25 чел. час. до времени формирования отчета 0.1-0.25 чел. час.
Бизнес– правила	Требования, описывающие корпоративные политики, законодательные требования, требования стандартов и алгоритмы вычислений, обязательные для исполнения в данном продукте. Следует вносить по возможности без изменения формулировки текста исходного документа.	Общая трудоемкость, выделенная на решение задачи, должна разделяться между ролевыми функциями, как правило, следующим образом: <ul style="list-style-type: none"> <li>• 5 % - формулировка, регистрация задачи и разъяснение (при необходимости) его сути Исполнителю.</li> <li>• 80 % - исполнение задачи.</li> <li>• 10 % - текущий контроль и решение проблем на стадии исполнения задачи.</li> <li>• 5 % - контроль конечного результата.</li> </ul>
Функциона льные требования	Требования, определяющие ожидаемое поведение системы при определенных условиях и действия, которые система позволит выполнить пользователям. Функциональные требования описывают поведенческие (функциональные) характеристики продукта, какие входные данные должны быть преобразованы в выходные и какие для этого требуются операции.	Система должна выводить список всех сотрудников зарегистрировавших трудозатраты в рамках дерева задач.
Нефункцио нальные требования	Требования, описывающие насколько продукт соответствует определенному режиму работы или позволяет пользователям выполнять конкретные действия. Включают требования к производительности, надежности, безопасности, возможности обслуживания, а также любые дополнительные характеристики качества продукта, которые важны для клиентов и разработчиков.	Пример требования к безопасности: Система должна позволить вывести отчет пользователю только по тем задачам, в которых он является исполнителем.
Требования к внешнему интерфейсу	Требования, содержащие информацию, необходимую для правильной разработки интерфейсов с объектами, внешними по отношению к данному продукту. Подразумевают пользовательские, аппаратные, программные и коммуникационные интерфейсы.	В отчете должны быть отделены задачи, выводимые для каждого сотрудника.
Ограничени я	Требования, которые касаются дизайна и реализации, официально ограничивают возможности, доступные разработчику. Например, необходимость использования определённой архитектуры или сторонних библиотек.	Отчет должен быть реализован в рамках операционной системы отчетности.
Определени е данных	Требования, которые определяют формат данных, типы данных, допустимые значения и т.д.	Длительность представляется в формате целого количества дней.

## Рекомендации по написанию требований

### Рекомендаций по написанию требований.

- используйте полные предложения, с правильной грамматикой, правописанием и пунктуацией. Предложения и абзацы должны быть краткими и ясными;
- требования следует излагать последовательно, например, «Система будет» или «Пользователь будет», затем — активный глагол, а после — наблюдаемый результат. Укажите иницирующие условия или действия, вследствие которых система ведет себя определенным образом. Например, «Если запрошенный химикат найден на складе химикатов, система отобразит список всех хранимых на складе контейнеров с указанным химикатом». Вы можете использовать «должно» как синоним «будет», однако следует избегать «следовало бы», «может», «можно было бы» и аналогичных слов, из которых не ясно, необходимо ли действие;
- используйте действительный залог (например, «Система сделает то-то», а не «Произойдет то-то»);
- последовательно используйте термины и именно так, как они определены в словаре. Остерегайтесь синонимов и слов, близких по значению. Не следует в спецификации требований к ПО пытаться разнообразить лексику, чтобы заинтересовать читателя;
- нечеткие требования верхнего уровня следует детализировать, таким образом, чтоб они стали абсолютно ясны;
- при указании требования в форме «Пользователь будет...» идентифицируйте определенного исполнителя (например, «Покупатель будет...»);
- применяйте списки, рисунки, графики и таблицы, чтобы представить информацию визуально. Читателей утомляет большой объем сплошного текста;
- подчеркните наиболее значимые фрагменты информации. Здесь годятся: графики, последовательности, в которых первый элемент подчеркивается, повторы, пустое пространство и визуальный контраст, например затенения;
- требования, изложенные неясным языком, не поддаются проверке, поэтому избегайте двусмысленных и субъективных терминов. Заносите такие требования в таблицу, которую можно использовать при проверке..

### Примеры требований

#### **Бизнес-требования**

##### **Бизнес-цели**

- Уменьшить потери продуктов в кафетерии на 50% в течение 6 месяцев после первого выпуска системы.  
Масштабы: стоимость продуктов, выбрасываемых каждую неделю персоналом кафетерия.  
Способ измерения: исследование инвентарных журналов кафетерия,  
Показатели в прошлом (2002г., первоначальное исследование): 30%  
Планируемые показатели: менее 15%.  
Обязательные показатели: менее 20%.
- Уменьшить оперативные расходы кафетерия на 15% в течение 12 месяцев после первого выпуска системы.
- Увеличить среднее эффективное рабочее время каждого сотрудника на 20 минут в день в течение трех месяцев после первого выпуска системы.

##### **Критерии успеха**

- 75% сотрудников, в настоящее время пользующихся кафетерием, должны начать использовать Cafeteria Ordering System в течение 6 месяцев после первого выпуска системы.
- Достичь увеличения среднего рейтинга по ежеквартальному опросу об удовлетворенности работой кафетерия на 0,5 балла в течение 3 месяцев после первого выпуска системы и на 1,0 балла в течение 12 месяцев после первого выпуска системы.

#### **Бизнес-правила**

- Если заказ доставлен, то клиент должен оплатить его посредством удержания из зарплаты.
- Стоимость заказа подсчитывается как сумма цен единиц каждого блюда, умноженных на количество заказанных единиц этого блюда, плюс соответствующий налог с продаж, плюс плата за доставку, если заказ доставляется в пункт, расположенный вне зоны бесплатной доставки.
- Только работники кафетерия, назначенные менеджером кафетерия менеджерами меню, могут создавать, изменять или удалять меню кафетерия.
- Передача данных по сети, включающая финансовую или подпадающую под учет личную информацию, должна проходить со 128-битным шифрованием.
- Только штатные сотрудники могут регистрироваться для совершения каких-либо покупок в компании посредством удержания из зарплаты.
- Сотрудник может зарегистрироваться для оплаты питания в кафетерии посредством удержания из зарплаты, если не более 40% его начисленной зарплаты удерживается в настоящее время по другим причинам.

#### **Функциональные требования**

- Система должна позволять клиенту указывать количество единиц каждого блюда, которое он хочет заказать.
- Меню на текущую дату должно показывать только те блюда, которые хотя бы в одном экземпляре есть в инвентарном списке кафетерия.
- Система должна отображать меню для указанной даты.

## Рекомендации по написанию требований

- Если заказ должен быть доставлен и все еще есть резервы времени доставки на дату заказа, клиент должен указать место доставки.
- Если дата доставки заказа — текущий день, а крайний срок приема заказов уже прошел, то система должна известить клиента, что уже слишком поздно размещать заказ на сегодня. Клиент должен либо изменить дату, либо отменить заказ.
- Система должна позволять клиенту заказывать дополнительные наборы блюд на ту же или другие даты. Включение нескольких наборов блюд в один заказ регулируют Бизнес-правило-3 и Бизнес-правило-4.
- См. Бизнес-правило-11.
- Система должна выводить названия заказанных блюд, сумму к оплате, метод оплаты и инструкции по доставке.

### Нефункциональные требования

#### Требования к производительности

- Система должна обслуживать 400 пользователей в период пиковой активности с 8:00 до 10:00 по местному времени, со средней продолжительностью сеанса 8 минут.
- Все Интернет-страницы, генерируемые системой, должны полностью загружаться не более чем за 10 секунд по модемному соединению со скоростью 40кб/сек.
- Загрузка ответов на запросы на экран должна занимать не более 7 секунд после того, как пользователь отослал запрос.
- Система должна выводить пользователю сообщение о подтверждении не более чем через 4 секунды после того, как пользователь отправляет информацию системе.

#### Требования к безопасности

- Все сетевые транзакции, включающие финансовую или поддающуюся учету личную информацию, должны быть зашифрованы согласно Бизнес-правилу-33.
- Пользователи обязательно регистрируются для входа в Cafeteria Ordering System для выполнения всех операций, кроме просмотра меню.
- Клиенты должны регистрироваться для входа в систему согласно политике ограниченного доступа к компьютерным системам по Бизнес-правилу-35.
- Система должна позволять клиентам просматривать только заказы, размещенные ими лично, но не другими клиентами.

#### Атрибуты качества ПО

##### *Доступность*

- Cafeteria Ordering System должна быть доступна пользователям корпоративной сети интранет и клиентам удаленного доступа по коммутируемой линии 99,9% времени между 5:00 и полночью по местному времени и 95% времени между полночью и 5:00 по местному времени.

##### *Надежность*

- Если соединение между пользователем и системой разрывается до того, как заказ подтвержден или отменен, Cafeteria Ordering System должна позволять пользователю восстановить незавершенный заказ.

## Рекомендации по написанию требований

### Требования к внешнему интерфейсу

#### Интерфейсы пользователя

- Экраны вывода Cafeteria Ordering System должны соответствовать «Process Impact Internet Application User Interface Standard, Version 2.0».
- Система должна обеспечивать ссылку на справку на каждой HTML странице, объясняющую, как пользоваться этой страницей.
- Интернет-страницы должны предоставлять полную возможность навигации и выбор блюд только при помощи клавиатуры, в дополнение к использованию мыши и клавиатуры.

#### Программные интерфейсы

##### *Инвентарная система кафетерия*

- Cafeteria Ordering System должна передавать количество единиц заказанных блюд инвентарной системе кафетерия через программный интерфейс.
- Cafeteria Ordering System должна опрашивать инвентарную систему кафетерия для определения наличия запрашиваемого блюда.
- Когда инвентарная система кафетерия сообщает Cafeteria Ordering System, что определенного блюда нет в наличии, Cafeteria Ordering System должна убирать это блюдо из меню на текущую дату.

##### *Система расчета зарплат*

Cafeteria Ordering System должна сообщаться с системой расчета зарплат через программный интерфейс, выполняя следующие операции:

- Позволять клиенту регистрироваться для оплаты через удержания из зарплат;
- Позволять клиенту отменять регистрацию для оплаты посредством удержания из зарплат;
- Проверять, зарегистрирован ли клиент для оплаты посредством удержания из зарплат;
- Передавать запрос на оплату приобретенного набора блюд;
- Возвращать полностью или частично предыдущую оплату, если клиент отменил заказ, или не был удовлетворен им, или заказ не был доставлен согласно подтвержденным инструкциям по доставке.

#### Интерфейсы передачи информации

- Cafeteria Ordering System должна посылать клиенту e-mail с подтверждением принятия заказа, ценой и инструкциями по доставке.
- Cafeteria Ordering System должна посылать клиенту e-mail с сообщением о любых проблемах, возникших с заказом или его доставкой после принятия заказа.

### Ограничения

#### Ограничения дизайна и реализации

- Документация системы по конструкции, коду и сопровождению должна соответствовать Process Impact Intranet Development Standard, версия 1.3
- Система должна использовать текущую версию корпоративного стандарта процессора базы данных Oracle.
- Весь код HTML должен соответствовать стандарту HTML 4.0.

## Рекомендации по написанию требований

- Все сценарии должны быть написаны на Perl.

### Определение данных

#### Заказ

##### Общее описание

Объект предназначен для описания заказов пользователей на получение химикатов со склада.

##### Атрибуты

- **Request ID** – 6-значный, сгенерированный системой, последовательный номер, состоящий из целого числа, начинающегося с 1, уникальным образом идентифицирующий каждый заказ.
- **Единицы количества** – [«граммы»| «килограммы»| «миллиграммы»]. Текстовая строка, состоящая из 9 символов, указывающая единицы, связанные с количеством запрошенного химиката.
- **Дата размещения запроса** – дата, когда клиент разместил заказ; формат ММ/ДД/ГГГГ
- **Описание запроса** – текстовое описание запроса; максимум 1024 знака
- **Цена заказа** – стоимость заказа в долларах и центах
- **Дата доставки** – дата, когда заказ должен быть доставлен; формат ММ/ДД/ГГГГ; по умолчанию – текущая дата.

## Рекомендации по написанию требований

### Определение уровня детализации требований.

- Избегайте длинных повествовательных абзацев, которые содержат несколько требований. Наличие в требовании таких слов, как «и», «или» и «также», предполагает, что несколько требований могли быть объединены. Это не означает, что вы не можете использовать союз «и», но если вы делаете это, проверяйте, соединяет ли он две части одного требования или два отдельных требования.
- Никогда не используйте «и/или» в требованиях; это оставляет читателю свободу маневра. Такие выражения, как «пока не» и «кроме» также указывают на наличие нескольких требований: «Кредитная карточка покупателя будет считаться действительной для платежей до тех пор, пока не истечет ее срок действия». Разделите это положение на два — для двух условий: когда срок действия кредитной карты истек и когда она действительна.
- Попробуйте отдельно описать требования, которые можно протестировать. Если вам удастся придумать несколько вариантов тестирования, значит, необходимый уровень детализации достигнут. Если ваши тесты многочисленны и разнообразны, вероятно, несколько требований соединены вместе, их следует разделить на более простые.
- Составляйте требования настолько подробно, чтобы при удовлетворении требования задача клиента была бы выполнена, однако избегайте ненужных ограничений разработки.
- Требования должны быть сформулированы достаточно подробно, чтобы риск непонимания был минимальным, для этого необходимо учесть знания и опыт разработчиков. Если разработчики предлагают несколько способов удовлетворения требования и все они приемлемы, значит, особенности продукта и детализация изложения выбраны верно.
- Точно сформулированные требования повышают вероятность того, что ожидания клиентов будут реализованы; менее строгие формулировки дают разработчикам больше свободы для интерпретаций. Если разработчику по составленной спецификации не удастся абсолютно ясно представить себе ожидания клиентов, следует включить дополнительную информацию, чтобы снизить риск последующих исправлений.

### Показатели качества требований

**Корректность.** Совокупность требований считается корректной только в том случае, если она содержит только те требования, которые должны быть реализованы в программном продукте. Кроме того, корректность может быть установлена на основании соответствия Спецификации требований фактическим потребностям заказчика и пользователей.

**Однозначность.** Требование является однозначным, если оно допускает единственное толкование. Совокупность требований должна быть однозначной как для своих разработчиков, так и для тех, кто будет пользоваться ей в дальнейшем.

**Полнота.** Совокупность требований является полной, если она включает:

- все требования, относящиеся к функциональности, производительности, ограничениям, атрибутам, внешним интерфейсам;
- определение всех реакций продукта и алгоритмы преобразований на все реализуемые классы входных данных во всех реализуемых ситуациях (включая ошибочные входные данные);
- определения всех нестандартных терминов, все необходимые внутренние и внешние ссылки и определение используемых единиц измерения.

**Согласованность.** Требования являются согласованными, если отсутствуют противоречия между любыми двумя подмножествами требований.

**Наличие ранжирования требований.** Для каждого требования к продукту следует определить следующие характеристики:

- важность,
- сложность,
- стабильность,
- приоритет.

Ранжирование требований важно для оптимальной организации процесса проектирования и разработки программного обеспечения.

Ранжирование требований по важности обеспечивает более полное согласование ожиданий Заказчика и усилий исполнителя.

Рекомендуется следующий минимальный набор степеней важности:

- «обязательное» – требование должно быть выполнено и будет проходить проверку при сдаче версии продукта;
- «рекомендуемое» – требование целесообразно, согласовано с Заказчиком, оно улучшает характеристики продукта, однако его отсутствие либо неполное удовлетворение не является основанием для отказа от приемки;
- «опциональное» – требование желательно с точки зрения разработчика, целесообразность его со стороны Заказчика в текущий момент не подтверждена.

Ранжирование требований по сложности служит базой для оценки объема работы по проекту и его планирования.

Рекомендуется следующий минимальный набор значений показателя сложности:

- «сложное» - для реализации требования необходимо разработать отдельный модуль, пакет или подсистему;
- «средняя» - требование может быть реализовано одним или небольшим числом связанных программных компонентов (классов, объектов, др.);



## Проверка требований

- «простая» - требование может быть реализовано одним или несколькими связанными методами одного программного компонента.

Для показателя стабильности требований вводится следующая шкала:

- «высокая» - требование не будет пересмотрено в ходе проекта;
- «средняя» - в ходе проекта возможно изменение данного требования;
- «низкая» - вероятнее всего, данное требование будет уточнено в ходе работы по проекту.

Ранжирование требований по приоритету определяет последовательность реализации требований при разработке продукта. К требованиям высшего приоритета относятся наиболее важные, стабильные и архитектурно-значимые требования, а также требования, связанные с повышенными технологическими рисками.

**Реализуемость.** Требование является реализуемым, если его реализация и/или эксплуатация приемлемы с точки зрения ресурсозатрат и надежности.

**Тестируемость.** Требование является проверяемым, если существует процесс (автоматический или ручной), приемлемый с точки зрения ресурсозатрат, который способен однозначно проверить соответствие полученного результата данному требованию.

**Способность к модификации.** Совокупность требований считается модифицируемой только в том случае, если ее структура и стиль позволяют легко внести любые изменения в требования, не нарушая при этом самой структуры и стиля. Данное качество достигается путем независимого определения каждого требования, не смешанным с определением других требований, а также отсутствием избыточности (это означает, что одно и то же требование не повторяется).

**Возможность трассировки.** Требования будут прослеживаемы, если источник для данного требования ясен из документа и если имеется возможность связать формулировки требований с вытекающими из них формулировками других документов, базирующихся на требованиях. Если до составления требований к продукту был утвержден документ «Концепция продукта», то должна быть установлена связь требований к продукту с требованиями, определенными в концепции. Необходимо поддерживать прослеживаемость вперед и назад: прослеживаемость вперед обеспечена, если требования имеют уникальный идентификатор в проекте, назад – если имеются прямые ссылки в поясняющих требованиях комментариях. Также необходимой является прослеживаемость от требований к их реализации. Требованию должен сопоставляться продукт, в котором оно будет реализовано, релиз продукта, и проект.

## Проверка требований

Неоднозначные термины	Способы их улучшения
Приемлемый, адекватный	Определите, что понимается под приемлемостью и как система это может оценить
Практически выполнимо	Не заставляйте разработчиков определять, что под этим понимается. Поставьте пометку «Требуется уточнения» и определите дату, к которой эту проблему следует разрешить
По меньшей мере, как минимум, не более чем, не должно превышать	Укажите минимальное и максимальное допустимые значения
Между	Определите, указаны ли конечные точки
Зависит от	Определите природу зависимости. Обеспечивает ли другая система ввод данных в вашу систему, надо ли установить другое ПО до запуска вашей системы и зависит ли ваша система от другой при выполнении определенных расчетов или служб?
Эффективный	Определите, насколько эффективно система использует ресурсы, насколько быстро она выполняет определенные операции и насколько она проста в обращении
Быстрый, моментальный	Укажите минимальную приемлемую скорость, при которой система выполняет определенное действие
Гибкий	Опишите способы изменения системы в ответ на изменения условий или бизнес-потребностей
Улучшенный, лучший, более быстрый, превосходный	Определите количественно, насколько лучше или быстрее стали показатели в определенной функциональной области
Включает; включает в себя, но не ограничен этим; и т.д.; и т.п.; такой как	Список элементов должен включать все возможности. В противном случае его нельзя использовать при разработке или тестировании
Максимизируйте, минимизируйте, оптимизируйте	Укажите минимальное и максимальное допустимые значения определенного параметра
Обычно, в идеальном варианте	Также опишите поведение системы при нештатных или неидеальных условиях
Необязательно	Укажите, кто делает выбор: системы, пользователь или разработчик
Разумный, при необходимости, при соответствующих условиях	Объясните, как это выполнить
Устойчивый к сбоям	Определите, как система должна обрабатывать исключения и реагировать на неожиданные условия работы
Цельный, прозрачный, элегантный	Выразите ожидания пользователя, применяя характеристики продукта, которые можно наблюдать
Несколько	Укажите сколько или задайте минимальную и максимальную границы диапазона

## Проверка требований

Неоднозначные термины	Способы их улучшения
Не следует	Старайтесь формулировать требования в позитивной форме, описывая, что именно система будет делать
Реальный	Поясните этот термин
Достаточный	Укажите, какая степень чего-либо свидетельствует о достаточности
Поддерживает, позволяет	Дайте точное определение, какие действия система будет выполнять, поддерживая конкретную возможность
Дружественный, простой, легкий	Опишите системные характеристики, которые будут отвечать потребностям пользователей и его ожиданиям, касающимся легкости и простоты применения продукта
Надежная	<p>Надежностью называется вероятность работы ПО без сбоев в течение определенного периода времени. Иногда одной из характеристик надежности считают устойчивость к сбоям. Для измерения надежности ПО используют такие показатели, как процент успешно завершенных операций и средний период времени работы системы до сбоя. Определите количественные требования к надежности, основываясь на том, насколько серьезными окажутся последствия сбоя и оправдана ли цена повышения надежности.</p> <p><i>Пример. Не более пяти из тысячи начатых экспериментов могут быть потеряны из-за сбоев ПО.</i></p>
Интуитивно понятная, удобная для пользователя	<p>При обсуждении этих терминов нужно стараться выделить параметры, поддающиеся измерению. Узнайте, должна ли новая система соответствовать каким-либо стандартам или соглашениям, касающимся пользовательского интерфейса, и должен ли последний быть совместим с другими часто используемыми системами. Вы можете сформулировать такое требование следующим образом:</p> <p><i>Пример. Всем функциям меню File должны соответствовать быстрые клавиши, нажимаемые одновременно с Ctrl. Командам меню, которые также присутствуют в меню File пакета Microsoft Word XP, должны соответствовать те же быстрые клавиши, что и в Word.</i></p> <p>Кроме того, удобство и простота использования определяется и тем, насколько легко новые или непостоянные пользователи научатся работать с продуктом. Простота обучения также поддается исчислению и измерению:</p> <p><i>Пример. Химик, который прежде никогда не использовал Chemical Tracking System, должен не более чем 30 минут разобраться, как правильно запросить химикат.</i></p>

## Замечания по показателям качества требований

### I. Корректность

#### 1. Требования к проекту, процессам, технологиям:

- *Создаваемая в ходе проекта документация должна основываться на шаблонах пользовательской документации. Оформление документации должно выполняться в соответствии с регламентами создания документации в компании. В каждом документе перед его утверждением должна быть идентифицирована версия программного продукта, к которой относится документ.*
- *Вся пользовательская документация должна проходить тестирование.*
- *Разработка системы ведется под один сервер приложений Oracle 9i AS. Tr*

#### 2. Формулировки требований, содержащие технические решения, тесты:

- *«система должна отображать следующие данные, получаемые посредством вызова соответствующих методов Generic API: 1. accumulatorId; 2. accumulatorName; 3. accumulatorClass 4. accumulatorType»*
- *На данный момент функционал сервиса проверяется следующими автотестами:*
  - *uap\_errConn. Проверяется случай, когда ни на один номер дозвониться не получается, попутно проверяется проигрывание приветствия и общая логика.*

### II. Однозначность.

#### 1. Не однозначные формулировки требований:

- *«Система должна обеспечивать переносимость на различные аппаратные и операционные системы»*
- *«Система должна поддерживать возможность отправки детализации разговоров по факсу, e-mail. При этом в системе ничего не должно пересоздаваться.»*
- *«Система периодически либо по запросу администратора обновляет локальную базу с настройками телефонов, получая изменения из центральной базы.»*
- *«Массив должен позволять классифицировать трафик по протоколам передачи данных. Можно считать достаточно надёжным определение протокола по используемому IP-порту.»*
- *«поддержка и корректное отображение непальского и григорианского календаря совместно, возможность выбора пользователем одного из календарей основным, корректная работа пользователей с календарем при вводе даты в соответствующие поля»*

#### 2. Не детализированные формулировки требований:

- *«Подсистема отправки периодически и по запросу от подсистемы приема заказов проверяет наличие в пуле заказов со статусом "принят" и "ошибка при отправке". При наличии таких заказов система достает из пула наиболее старый заказ и формирует SMS сообщения для указанной в заказе модели телефона и с настройками, которые указаны в заказе. При этом, если источник заказа сообщения - SMS, то*

## Проверка требований

*формируется SMS с инструкцией по применению данных настроек, либо с информацией о том как и где можно получить такую инструкцию. Вариант поведения задается в настройках. В случае второго варианта в настройках задается текст с информацией. После этого система отправляет сначала SMS с информацией, а потом SMS с настройками через SMSC на номер, указанный в заказе. В случае возникновения ошибки при отправке у заказа выставляется статус "ошибка при отправке" и инкрементируется счетчик попыток. При достижении числа попыток отправки определенного значения, задающегося в настройках, статус заказа выставляется в "исчерпано число попыток отправки".»*

- *«Массив должен содержать показатели объема переданного трафика: входящий, исходящий, в сумме входящий и исходящий, разность между входящим и исходящим. Трафик измеряется в мегабайтах и количеством переданных пакетов. Также массив должен содержать связь между этими единицами измерения - средний размер пакета.»*

### III. Полнота.

1. *Функциональные требования не выявлены.*
2. *Нефункциональные требования не определены.*

### IV.Согласованность.

1. Цели не трассируются к представленным требованиям. Какие из представленных требований повышают производительность, надежность, эргономичность.

#### 2.2. Цели

*Разработка системы XXXX преследует цели:*

- *Повышение производительности при многопользовательском информационном обслуживании абонентов оператором связи*

#### 3.5. Нефункциональные требования

*Нефункциональные требования не определены.*

### V. Способность к модификации.

1. Требование *«Максимальное количество номеров для переадресации - 5»* должно быть в ограничениях, а не в требованиях к web-интерфейсу.

### VI.Возможность трассировки.

1. При согласовании спецификации требований при проверке требований на тестируемость были созданы тестовые случаи. В процессе согласования были выявлены недостатки требований. Требования изменили, не сохранив нумерацию. Созданные тестовые случаи ссылаются на другие требования. Требуется затраты на анализ изменившихся требований и «перепривязка» к изменившимся требованиям.
2. В спецификации содержатся требования без идентификаторов.