

Министерство образования Республики Беларусь
Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий
Кафедра технологий программирования

Методические указания
к выполнению лабораторной работы №8

по дисциплине «**Надёжность программного обеспечения**»
для специальности 1-40 01 01 Программное обеспечение информационных
технологий

на тему «**Составление документа формулирования требований.**
Техническое задание»

Новополоцк, 2017 г.

Название: «Составление документа формулирования требований. Техническое задание».

Цель работы: ознакомиться с процессом формулирования требований. Научиться составлять техническое задания к программным продуктам.

Теоретическая часть

Процесс формулирования требований

Проекты по созданию программного обеспечения начинают свой жизненный цикл, когда у заказчика возникает необходимость заменить существующую систему или потребность в совершенно новой системе. Заказчиком может быть любой отдел вашей компании, отдельная компания или агентство, которое заключает с вами договор на выполнение работ. В качестве заказчика может также выступать рынок товаров массового производства с активным спросом на готовые программные продукты. Потребности заказчика могут быть представлены набором требований, где под *требованием (requirement)* понимается описание того, что способна выполнять система, либо описание некоторых условий функционирования, которые необходимо обеспечить с тем, чтобы система могла выполнять свои задачи. Требования определяют то, что должна выполнять система, но не то, как она должна это выполнять. Последнее означает, что в центре внимания требования находится задача заказчика и его деловая активность, но не то, как достигается ее решение.

Рисунок 1 служит иллюстрацией процесса формулирования требований к системе программного обеспечения. Мы исследуем этот процесс с точки зрения специалистов по тестированию, которые стремятся получить информацию, поддерживающую планирование тестирования на ранних стадиях разработки, стремятся обнаружить дефекты уже в самих требованиях, чтобы они не инициировали лавину ошибок на более поздних стадиях разработки.

Первым на рисунке 1 показан опрос заказчика с целью выявления требований, которые он предъявляет к программному продукту. Выявление требований выполняется в форме вопросов и ответов. С использованием слайдов, макетов и прототипов заказчику предлагаются различные варианты. Сбор пожеланий со стороны заказчика может осуществляться в виде серии интервью или путем использования технологии FAST (Facilitated Application Specification Techniques – технология упрощенной спецификации приложения), представляющей собой специальный тип собеседований с заказчиком, который облегчает выявление его требований.

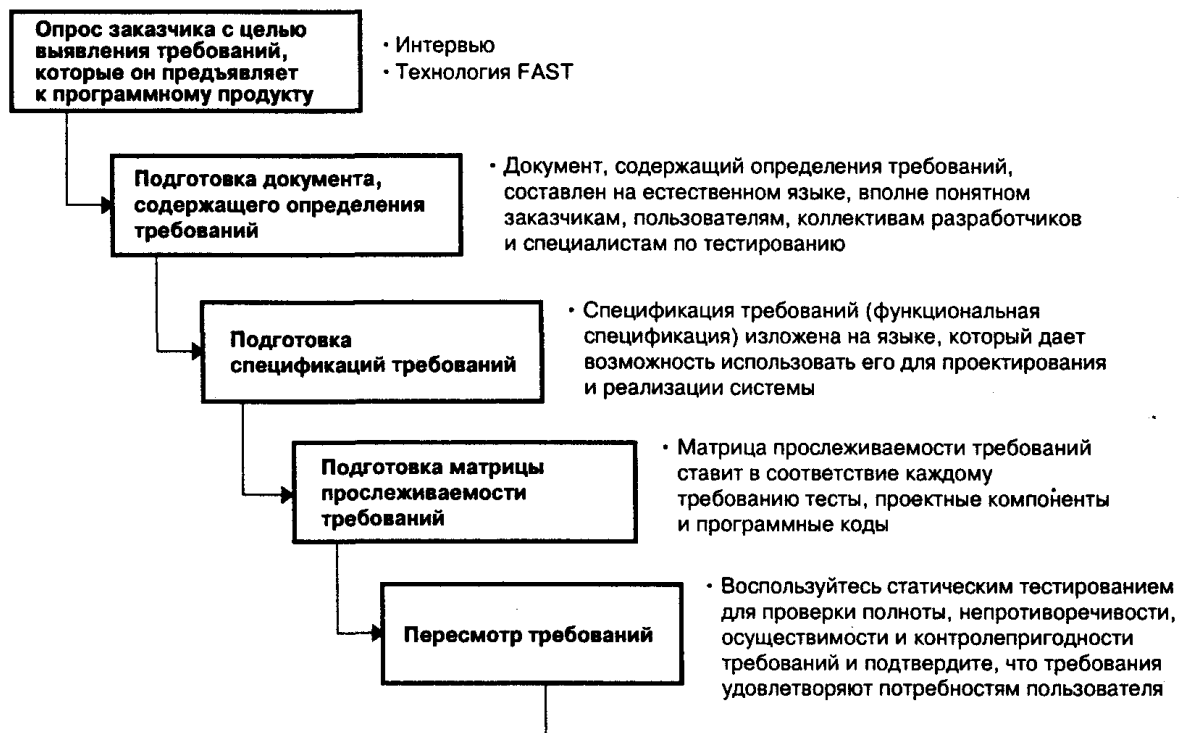


Рисунок 1 – Процесс формулирования требований

По мере получения от пользователя, требования должны фиксироваться в виде документа, известного как *документ определения требований* (*requirements definition document*). Этот документ оформляется в виде списка требований, сформулированных в результате собеседований с заказчиком. Он представляет собой соглашение между заказчиком и организацией, выполняющей разработки, о том, что должно создаваться. Определение требований представляет собой письменный документ на естественном языке, вполне понятный как заказчику, так и коллективам, которые занимаются разработкой и сопровождением системы.

Достоинство документа определения требований состоит в том, что его легко понять, но в то же время ему не хватает точности и технических деталей, необходимых для проектирования и разработки программного продукта. В силу этого обстоятельства должен быть подготовлен другой документ, известный как *спецификация требований* (*requirement specification*) или *функциональная спецификация* (*functional specification*).

Как только документ определения требований и спецификации требований будут готовы, можно приступить к построению *матрицы прослеживаемости требований* (*requirements traceability matrix*). Назначение этой матрицы состоит в том, чтобы поставить в соответствие каждому требованию тесты, компоненты проекта и программный код. В идеальном случае в этой работе принимают участие коллективы разработчиков и специалистов по тестированию, в результате чего со временем появятся связанные с каждым требованием проектные компоненты, тесты программных модулей, тесты комплексных и приемочных испытаний. При правильном использовании матрица прослеживаемости требований

представляет собой инструментальное средство, которое помогает каждому специалисту, принимающему участие в процессе разработки, выполнять работу, непосредственно связанную с потребностями заказчика, и не тратить понапрасну время на решение ненужных задач. С точки зрения группы тестирования этот инструмент может с успехом использоваться для планирования тестирования и проектирования тестов, обеспечивающих хорошее тестовое покрытие.

Выявление требований

Обмен информацией между заказчиком и разработчиком в процессе определения требований к программному продукту настолько важен для успеха всего проекта, что для достижения его эффективности затрачиваются значительные усилия и средства. Одним из факторов, способных снизить эффективность такого обмена, является недостаточный объем знаний заказчиком методов разработки программных продуктов. Довольно часто заказчик не настолько разбирается в процессе разработки программного обеспечения, чтобы быть способным выразить свои требования в форме, понятной специалисту по разработке системы.

Процесс определения системы программного обеспечения можно сравнить с проектированием дома по индивидуальному проекту: может случиться так, что вы не настолько хорошо разбираетесь в строительстве, чтобы точно сказать, что вам нужно. Обмен мнениями, по всей видимости, должен включать осмотр уже построенных домов, изучение планировки помещений с целью выбора понравившейся, а также изучение некоторого множества других индивидуальных проектов и чертежей. Что касается систем программного обеспечения, то часто полезно продемонстрировать в работе существующее программное обеспечение, ознакомиться с соответствующими документами или прототипами программного продукта и обсудить рабочую среду, в которой будет использоваться программный продукт. Результаты такой работы с заказчиком вызывают непосредственный интерес и у группы, проводящей тестирование. Специалисты этой группы желают знать, как заказчик намерен использовать программное обеспечение, дабы спроектировать реалистичные тесты для системных и приемочных испытаний.

Один из классов методов, которые могут быть использованы для выявления требований, получил название *технологии FAST* (Facilitated Application Specification Techniques – упрощенной спецификации приложения). Наиболее распространенным подходом в рамках реализации технологии FAST является разработанный компанией IBM *метод JAD* (Joint Application Development – совместная разработка приложения). Другой метод, *JAR* (Joint Application Requirement – совместная разработка требований к приложению), был предложен Гэри Коббом (Gary Cobb).

Обычно во время сеанса технологии FAST рекомендуется следовать тому или иному набору условий из числа приведенных ниже:

- и заказчики, и разработчики принимают участие в совещании, посвященном вопросам определения требований;

– установлены правила подготовки и участия в совещании, которым следуют обе стороны. Должна быть установлена атмосфера, содействующая успеху совещания;

– совещание проводится под управлением посредника. Таким посредником может быть заказчик, разработчик или кто-то посторонний, приемлемый для обеих сторон;

– для фиксации требований используются такие средства, как лекционные плакаты с рейкой, настенные индикаторные панели или простая ленточная бумага;

– цель совещания – определить задачи или потребности заказчика, предложить решение, обсудить различные подходы и определить набор требований.

Настоятельно рекомендуется участие в этом совещании специалиста по отладке или даже несколько упростить регламент совещания по технологии FAST с таким расчетом, чтобы в этом совещании могли использоваться элементы статического тестирования. Технология JAR особо благоприятствует встроенному статическому тестированию, которое в терминологии JAR называется «вспомогательная поддержка».

Одним из видов информации, которую можно извлечь в результате проведения мероприятий по выявлению требований, является соглашение между заказчиком и разработчиком относительно приоритетов требований. Например, каждое требование может быть отнесено в одну из следующих категорий:

– наиболее важные требования;

– требования, выполнение которых крайне желательно;

– требования, выполнение которых желательно, но не обязательно.

Благодаря такому распределению приоритетов упрощается выполнение плана разработок. Например, можно составить план, который ставит выполнение наиболее важных требований в рамках первой версии программного продукта, реализация желательных требований во второй версии и всех остальных – в третьей версии. Аналогично, группа тестирования может начать разработку тестов для требований с наибольшим приоритетом в первую очередь. Такой тип нарастающих поставок или поставок версиями, часто используется в тех случаях, когда решающее значение имеет соблюдение временного графика работ.

Нужно рассмотреть специальный случай, когда от специалистов по тестированию требуется провести испытание программных продуктов, к которым не предъявлено никаких требований, либо случай, когда не все требования выявлены. К сожалению, упомянутые ситуации встречаются довольно часто; например, данные группы Standish Group, в которых первой из причин неудачи проекта или нарушение сроков его разработки была названа неполнота требований.

В ситуации, когда спецификация требований отсутствует, у вас, возможно, не остается другого выбора, как только написать собственный документ определения требований. В зависимости от того, сколько времени

выделено на завершение тестовых работ, полученное определение требований может оказаться не таким исчерпывающим, как этого бы хотелось, но очень важно, чтобы были определены все ключевые требования. Невозможно тестировать программное обеспечение, если нет более-менее точных определений необходимых функциональных средств. Если вы столкнулись с затруднениями при определении требований, возможно, потребуется провести JAR-сеанс с генеральным разработчиком и, желательно, с представителями группы маркетинга и заказчиком. По меньшей мере, можно будет взять интервью у представителей разработчика и группы маркетинга и сформулировать по возможности полные определения требований, насколько это позволяет отпущенное вам время.

Определение требований

Документ определения требований содержит все требования, предъявляемые к системе и сформулированные на естественном языке, благодаря чему они становятся понятными и заказчику, и тем, кто принимает участие в разработке проекта. В перспективе тестирования мы заинтересованы в получении из этого документа информации, достаточной для того, чтобы иметь возможность приступить к планированию и разработке тестов. Для наших целей определения требований должны обладать следующими свойствами:

- каждое требование должно быть снабжено уникальным идентификатором, чтобы можно было однозначно ссылаться на него при планировании тестового покрытия, при проектировании тестовых случаев и в отчетах по результатам тестирования;

- требования должны быть представлены с точки зрения пользователя системы. Системные и приемочные испытания должны быть спроектированы на основе определений требований, следовательно, эти определения должны быть сформулированы в перспективе системного уровня. Этот принцип препятствует появлению требований, затрагивающих внутренние свойства системы и требующих детальных знаний программного кода для успешного тестирования. Такие требования должны возникать на поздних стадиях разработки и охватываться модульным тестированием и проверкой взаимодействия и функционирования компонентов системы;

- должны быть включены как *функциональные (functional)*, так и *нефункциональные (nonfunctional)* требования. Функциональные требования суть требования, описывающие услуги и функции, которые должна выполнять разрабатываемая система. Нefункциональные требования описывают ограничения, накладываемые на работу системы, например, количество одновременно работающих пользователей, и стандарты, которым должна соответствовать система;

- документ определения требований должен находиться под управлением конфигурациями. Это, по меньшей мере, означает, что данный документ подпадает под управление версиями и что все версии документа должны быть помещены в безопасное хранилище, подобное, например, каталогу, содержимое которого обычно дублируется. Если требования

подвергаются изменениям, мы должны иметь возможность проследить, чтобы соответствующие изменения были внесены в тестовые случаи системных и приемочных испытаний.

Оглавление одного из возможных вариантов документа, содержащего системные требования, представлено на рисунке 2. Это оглавление соответствует стандарту IEEE Standard 830: *The IEEE Guide to Software Requirements Specifications* – Руководящие принципы IEEE по составлению спецификации требований к программному обеспечению.

Документ определения требований (Requirements Definition Document)	
Оглавление (Table of content)	
1. Введение (Introduction)	4
1.1. Назначение (Purpose).....	4
1.2. Область применения (Scope).....	4
1.3. Обзор (Overview).....	4
2. Общее описание (General Description)	4
2.1. Перспектива продукта (Product perspective).....	4
2.2. Функция продукта (Product function).....	4
2.3. Характеристика пользователя (User Characteristics).....	4
2.4. Ограничения общего характера (General Constraints).....	4
2.5. Допущения и зависимости (Assumption and Dependencies).....	5
3. Особые требования (Specific Requirements)	5
3.1. Функциональные требования (Functional Requirements).....	5
3.2. Требования внешнего интерфейса (External Interface Requirements).....	5
3.3. Требования к производительности (Performance Requirements).....	5
3.4. Проектные ограничения (Design constraints).....	5
3.5. Атрибуты (Attributes).....	6
3.6. Другие требования (Other Requirements).....	6
Ссылки (References)	7
Приложение 1. Сокращения (Appendix 1 — Acronyms)	7
Приложение 2. Терминология (Appendix 1 — Definition of Terms)	7

Рисунок 2 – Пример оглавления документа определения требований к программному продукту

Документ определения требований, показанный на рисунке 2, включает общее описание, которое показывает, какими достоинствами обладает данный продукт перед другими продуктами этого типа, и описание его функциональных возможностей. В нем дано краткое описание основных свойств продукта и указано, какой квалификацией должен обладать пользователь, чтобы работать с ним. В то же время документ не содержит подробного описания функциональных средств. Раздел специальных требований содержит определения функциональных требований, требований к производительности, к интерфейсам и ряд других требований. Рассматриваемый документ должен содержать определения сокращенных обозначений (акронимов) и технической терминологии, используемой для описания продукта.

Как только документ определения требований будет написан, он должен быть подвергнут статическому тестированию с целью проверки

требований на полноту, непротиворечивость, осуществимость, контролепригодность, однозначность и релевантность.

Типы требований. Выше мы определили требования как описание того, что способна сделать система, либо как условия эксплуатации, которые необходимо обеспечить для того, чтобы система была способна выполнять возложенные на нее задачи. Это подразумевает использование широкого диапазона различных сведений, поэтому полезно разбить требования на категории. Пример набора таких категорий представлен следующим списком:

– **Функциональные средства.** Этот набор требований определяет, какие функции должен выполнять данный программный продукт на системном или пользовательском уровне. Для ясности может быть указано, чего *не должен* делать данный продукт.

– **Интерфейсы.** Эта категория требований описывает входы, получаемые из внешних систем, и выходы, направляемые во внешние системы. Накладываются ли на эти интерфейсы какие-то ограничения, связанные с форматами данных и носителями информации?

– **Данные.** Эти требования описывают входные и выходные данные системы. Какой при этом используется формат? Какие данные нужно сохранять? Какой объем данных поступает в систему и из системы, и с какой скоростью передачи? С какой точностью должны выполняться вычисления?

– **Производительность.** Требования этой категории описывают проблемы масштабирования и синхронизации, например, сколько пользователей одновременно должна обслуживать система, сколько транзакций должна выполнять система в единицу времени, как долго пользователь должен ждать ответа на свой запрос. Следует соблюдать особую осторожность при выражении этих требований в числовых значениях, поскольку в дальнейшем их придется подтверждать.

– **Пользователи и человеческий фактор.** Эта категория требований предъявляется к тем, кто будет работать с системой в смысле их квалификации и опыта. Они также определяют необходимый уровень удобства и простоты использования программного продукта, например, сколько отдельных действий требуется для выполнения рутинной операции в системе? Насколько отчетливо выражена обратная связь после каждого действия?

– **Физическая среда.** Эта категория требований определяет, где должно функционировать оборудование. Установлено ли оборудование более чем в одном месте? Имеют ли место ограничения по температуре, влажности или другие ограничения, связанные с окружающей средой?

– **Безопасность.** Эта категория требований описывает, как осуществляется доступ к системе и как осуществляется управление ее данными. Данная категория является подходящим местом для описания, как должны дублироваться системные данные. Как часто следует выполнять дублирование? На каких носителях сохраняются дублированные данные?

– **Документация.** Эта категория требований связана с документацией и тем, должна ли она быть представлена в печатном виде или выдается в интерактивном режиме. Кроме того, здесь же определяется категория лиц, для которых эта документация предназначена.

– **Устранение неисправностей.** Эта категория требований описывает, как система реагирует на возникновение неисправностей. Будет ли система обнаруживать неисправность и выдавать аварийные сигналы? Какой должна быть средняя длительность промежутка времени между двумя неисправностями? Каким должно быть максимальное время простоя?

– **Сопровождение.** Эти требования определяют, как производится устранение проблем, обнаруженных в системе, как усовершенствованные версии системы поставляются заказчику и как пользователи должны переходить на усовершенствованную версию системы.

– **План поставки версий.** Если требованиям назначены приоритеты, возможно, необходимо будет определить временной график поставки версий, который показывает, реализации каких требований уделяется внимание в конкретной версии.

Выполняя статическое тестирование на документах с формулировками требований, возможно, возникнет желание воспользоваться приведенным выше списком с целью обеспечения полноты тестирования. Если вы ведете предысторию статического тестирования сразу нескольких проектов, этот список можно расширить таким образом, чтобы он соответствовал номенклатуре программных продуктов, создаваемых вашей компанией. Если обстоятельства заставляют вас проводить тестирование в условиях неполного набора требований, на основе предложенного списка, скорее всего, придется определить, какие тесты нужны помимо тех, что образуют покрытие задокументированных требований.

Составление технического задания программных продуктов

Структура технического задания при написании программных продуктов:

- 1 Введение.
 - 1.1 Основания для разработки.
 - 1.2 Назначение.
 - 1.3 Область применения.
- 2 Требования к программному продукту.
 - 2.1 Требования к функциональным характеристикам.
 - 2.2 Требования к графическому интерфейсу.
 - 2.3 Требования к надежности.
 - 2.3.1 Требования к обеспечению надежного функционирования программы.
 - 2.3.2 Время восстановления после отказа.
 - 2.3.3 Отказы из-за некорректных действий пользователей системы.
- 3 Условия эксплуатации.

- 3.1 Климатические условия эксплуатации.
- 3.2 Требования к квалификации и численности персонала.
- 3.3 Требования к составу и параметрам технических средств.
- 3.4 Требования к информационной и программной совместимости.
 - 3.4.1 Требования к информационным структурам и методам решения.
 - 3.4.2 Требования к исходным кодам и языкам программирования.
 - 3.4.3 Требования к программным средствам, используемым программой.
 - 3.4.4 Требования к защите информации и программ.
- 3.5 Специальные требования.
- 4 Требования к программной документации.
 - 4.1 Предварительный состав программной документации.
- 5 Технико-экономические показатели.
 - 5.1 Экономические преимущества разработки.
- 6 Стадии и этапы разработки.
 - 6.1 Стадии разработки.
 - 6.2 Этапы разработки.
 - 6.3 Содержание работ по этапам.
- 7 Порядок контроля и приемки.
 - 7.1 Виды испытаний.
 - 7.2 Общие требования к приемке работы.

Содержание разделов технического задания

В разделе «**Введение**» указывают наименование программы, а также следующие подразделы:

- основания для разработки (должны быть указаны: документ (документы), на основании которых ведется разработка; организация, утвердившая этот документ, и дата его утверждения; наименование и (или) условное обозначение темы разработки).
- назначение (описывается функциональное и эксплуатационное назначение программного продукта).
- область применения (краткая характеристика области применения программы и объекта, в котором её используют).

Раздел «**Требования к программному продукту**» должен содержать следующие подразделы:

- требования к функциональным характеристикам (указываются требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т.п.);
- требования к графическому интерфейсу (указываются требования к оформлению и удобству использования программного продукта и т.п.);
- требования к надежности (указываются требования к обеспечению надежного функционирования (обеспечения устойчивого

функционирования, контроль входной и выходной информации, время восстановления после отказа, возможные отказы из-за некорректных действий пользователя и т.п.)).

Раздел **«Условия эксплуатации»** может делиться на подразделы:

- климатические условия эксплуатации (указываются температуру окружающего воздуха, относительную влажность и т.п. для выбранных типов носителей данных), при которых должны обеспечиваться заданные характеристики);
- требования к квалификации и численности персонала (описывают вид обслуживания программного продукта, а также необходимое количество и квалификацию персонала);
- требования к составу и параметрам технических средств (указывают необходимый состав технических средств с указанием их основных технических характеристик);
- требования к информационной и программной совместимости (указываются требования к информационным структурам на входе и выходе и методам решения, исходным кодам, языкам программирования программным средствам, используемым программой. При необходимости должна обеспечиваться защита информации и программ);
- требования к маркировке и упаковке (в общем случае указывают требования к маркировке программного изделия, варианты и способы упаковки);
- требования к транспортированию и хранению (указываются для программного изделия условия транспортирования, места хранения, условия хранения, условия складирования, сроки хранения в различных условиях);
- специальные требования (описываются особые условия, не входящие в вышеописанные подразделы).

В разделе **«Требования к программной документации»** должен быть указан предварительный состав программной документации и, при необходимости, специальные требования к ней.

В разделе **«Технико-экономические показатели»** должны быть указаны:

- ориентировочная экономическая эффективность;
- предполагаемая годовая потребность;
- экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

В разделе **«Стадии и этапы разработки»** устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), а также, как правило, определяют сроки разработки и исполнителей.

В разделе **«Порядок контроля и приемки»** должны быть указаны виды испытаний и общие требования к приемке работы.

В приложениях к техническому заданию, при необходимости, приводят:

- перечень научно-исследовательских и других работ, обосновывающих разработку;
- схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке;
- другие источники разработки.

Практическая часть

Содержание задания

Составить техническое задание для программного продукта. В качестве варианта взять курсовой проект по дисциплине «Языки программирования», согласовать с преподавателем. Обязательными для написания технического задания являются все пункты, описанные в теоретической части. Заказчиком считаем преподавателя, компанией-разработчиком и исполнителем – студента, выполнявшего курсовой проект. Пример документа формулирования требований приведен в книге «Быстрое тестирование» страницы 294 – 312.

Порядок выполнения работы

- 1 Ознакомиться с теоретической частью.
- 2 Согласовать вариант с преподавателем.
- 3 Составить техническое задание (**обязательными являются ВСЕ пункты, указанные ниже в содержании отчета**).
- 4 Показать отчёт преподавателю.
- 5 После защиты, выполненного задания, скинуть отчёт преподавателю (в названии указать **фамилию и номер лабораторной**, например, Ivanov8).

Содержание отчета

- 1 Титульный лист (**на титульном листе вместо варианта указать название тестируемого программного продукта**).
 - 2 Введение (включает пункты: «Основания для разработки», «Назначение», «Область применения»).
 - 3 Требования к программному продукту (включает пункты: «Требования к функциональным характеристикам», «Требования к графическому интерфейсу», «Требования к надежности»).
 - 4 Условия эксплуатации (включает пункты: «Климатические условия эксплуатации», «Требования к квалификации и численности персонала», «Требования к составу и параметрам технических средств», «Требования к информационной и программной совместимости», «Специальные требования»).
 - 5 Требования к программной документации (включает пункт: «Предварительный состав программной документации»).
 - 6 Техничко-экономические показатели (включает пункт: «Экономические преимущества разработки»).
 - 7 Стадии и этапы разработки (включает пункты: «Стадии разработки», «Этапы разработки»).
 - 8 Порядок контроля и приёмки (включает пункты: «Виды испытаний», «Общие требования к приемке работы»).
 - 9 Вывод о проделанной работе.
- Защита работ проводится индивидуально.

Контрольные вопросы

- 1 Дайте определение понятию «требование»?
- 2 Из каких этапов состоит процесс формулирования требований?
- 3 Перечислите достоинства и недостатки документа определения требований.
- 4 Какого назначение матрицы прослеживания требований?
- 5 Назовите свойства для определения требований.
- 6 Перечислите категории требований.
- 7 Перечислите разделы, необходимые при написании технического задания.