

Кыргызско-Российский Славянский университет им. Б.Н. Ельцина

Десятков Г.А., Лыченко Н.М., Манжикова С.Ц.

**Методические указания
по выполнению
выпускной квалификационной работы бакалавра
по направлению «Программная инженерия»**

Бишкек 2018

УДК 004:681.5
ББК 32.973-01
Д37

Рекомендовано к печати
Ученым советом Естественно-технического факультета КРСУ

Рецензенты:

канд. техн. наук, доцент Мусина И.Р.;
канд. техн. наук, доцент Хмелева И.В.

Десятков Г.А. и др.

Д37 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ВЫПУСКНОЙ
КВАЛИФИКАЦИОННОЙ РАБОТЫ БАКАЛАВРА ПО НАПРАВЛЕНИЮ
«ПРОГРАММНАЯ ИНЖЕНЕРИЯ» / Г.А.Десятков, Н.М. Лыченко, С.Ц. Манжикова.- Б.:
КРСУ, 2018. 54 с.

ISBN 978-9967-19-577-6

Методические указания содержат общие положения о выпускной квалификационной работе (ВКР) бакалавра по направлению «Программная инженерия», методические указания к её выполнению и требования к содержанию пояснительной записки к ВКР. Сформулированы требования к основным главам пояснительной записки: «Видение программного продукта», «Спецификация требований к программному продукту», «Проектирование и конструирование ПО», «Разработка тестов и тестирование ПО», «Руководство пользователя». Изложены технические требования к оформлению пояснительной записки к ВКР. В приложениях представлены примеры содержаний Спецификаций требований к программному продукту, примеры оформления библиографических записей, примеры титульного листа и заявления на тему ВКР.

Методические указания предназначены для студентов, обучающихся по направлению 09.03.04 (710400) «Программная инженерия» и могут быть использованы для подготовки специалистов университетами, где обучаются студенты по соответствующим направлениям.

Д 2404090000-18

УДК 004:681.5
ББК 32.973-01

ISBN 978-9967-19-577-6

© Г.А.Десятков, Н.М.Лыченко, С.Ц.Манжикова 2018

Содержание

1. Общие положения	4
1.1 Место ВКР в Основной образовательной программе бакалавриата	4
1.2 Цели выполнения и защиты ВКР	5
1.3 Тема ВКР	5
1.4 Научное руководство и консультирование	6
1.5 Преддипломная практика	7
1.6 Требования к ВКР	8
2. Требования к содержанию пояснительной записки к ВКР бакалавра.....	9
2.1 Содержание пояснительной записки к ВКР.....	9
2.2 Требования к Аннотации.....	9
2.3 Требования к Содержанию.....	10
2.4 Требования к Введению.....	10
2.5 Требования к Основной части	12
2.5.1 Содержание Главы 1 «Видение программного продукта»	12
2.5.2 Содержание Главы 2 «Спецификация требований к программному продукту».....	13
2.5.3. Требования к Спецификации требований к программному продукту.....	13
2.5.4 Требования к Главе 3 «Проектирование и конструирование ПО».....	21
2.5.5 Содержание Главы 4 «Разработка тестов и тестирование ПО.....	26
2.5.6 Требования к Главе 5 «Руководство пользователя»	26
2.5.7 Требования к Заключение.....	30
2.5.8 Требования к Списку использованных источников.....	30
2.5.9 Требования к Приложениям и их оформлению.....	30
3. Технические требования к оформлению пояснительной записки к ВКР.....	31
Список использованных источников.....	35
Приложение А. Образцы содержаний Спецификаций требований к программному продукту	36
Приложение Б. Примеры библиографических записей.....	44
Приложение В. Пример титульного листа ВКР.....	53
Приложение Г. Форма заявления на утверждение темы ВКР.....	54

1. Общие положения

1.1 Место выпускной квалификационной работы в основной образовательной программе бакалавриата

В соответствии с Образовательным стандартом высшего образования по направлению подготовки 09.03.04 «Программная инженерия» (уровень бакалавриата) (ФГОС3+) [1] и с Положением о Государственной итоговой аттестации выпускников КРСУ [2], защита выпускной квалификационной работы является обязательной составляющей государственной итоговой аттестации выпускников КРСУ.

Выпускная квалификационная работа (ВКР) представляет собой самостоятельную и логически завершенную работу на заданную тему, связанную с решением задач того вида или видов деятельности, к которым готовится студент бакалавриата (научно-исследовательская; аналитическая; проектная; технологическая; производственная; педагогическая; организационно-управленческая; сервисно-эксплуатационная) [1].

Разработанная в КРСУ в соответствии с требованиями ФГОС 3+ высшего образования по направлению 09.03.04 «Программная инженерия» Основная образовательная программа (ООП) ориентирована на научно-исследовательский вид профессиональной деятельности выпускника, т.е. является программой академического бакалавриата.

Для подготовки ВКР студенту назначается научный руководитель и, при необходимости, консультанты.

Выпускная квалификационная работа выполняется в период обучения в 7-8 семестрах [3].

В государственную итоговую аттестацию входят междисциплинарный экзамен и защита ВКР.

К государственной итоговой аттестации допускаются лица, успешно завершившие в полном объеме освоение ООП по направлению подготовки.

К защите ВКР допускаются студенты, успешно сдавшие междисциплинарный экзамен и представившие ВКР с отзывом руководителя в установленный срок. Защита ВКР проводится на заседании Государственной экзаменационной комиссии (ГЭК).

Сроки проведения Государственной итоговой аттестации и состав экзаменационных испытаний определяются приказом ректора КРСУ.

Результаты защиты определяются оценками по пятибалльной системе и являются основанием для принятия Государственной экзаменационной комиссией решения о присвоении степени бакалавра по направлению 09.03.04 «Программная инженерия» и выдаче диплома установленного образца.

ВКР выполняется и защищается в соответствии с Планом мероприятий по подготовке и защите ВКР бакалавра, подготавливаемым на кафедре ИВТ в начале 7 семестра.

1.2 Цели выполнения и защиты ВКР

При выполнении ВКР студенты должны показать свою способность и умение самостоятельно решать на современном уровне задачи из сферы профессиональной деятельности, аргументировать и защищать свою точку зрения.

Основными целями выполнения и защиты ВКР являются:

- систематизация и интеграция теоретических знаний и практических навыков по направлению «Программная инженерия»;
- выработка навыков самостоятельной аналитической работы, сбор и анализ информации из различных источников;
- применение полученных знаний при решении таких прикладных задач программной инженерии, как
 - разработка и анализ требований к программному обеспечению (ПО);
 - планирование процесса разработки программного обеспечения;
 - применение современных технологий разработки программных средств и контроль качества разрабатываемых программных средств;
 - разработка программной документации, в частности, Руководства пользователя.

1.3 Тема ВКР

Тематика ВКР должна быть направлена на решение профессиональных задач в области исследования, проектирования и разработки программных средств различного назначения.

Тематика ВКР разрабатывается преподавателями кафедры ИВТ и утверждается на заседаниях кафедры ИВТ. Студенту предоставляется право выбора научного руководителя ВКР и темы ВКР вплоть до предложения своей темы с необходимым обоснованием целесообразности её разработки.

После выбора руководителя и темы ВКР студент оформляет заявление на имя заведующего кафедрой, в котором указывается тема ВКР (см. Приложение Г). Заявление должно быть подписано студентом и руководителем ВКР.

Заявления студентов рассматриваются на заседании кафедры ИВТ. В результате обсуждения тема работы может быть скорректирована. Если тема ВКР рекомендована к утверждению, заявление студента подписывает заведующий кафедрой. На основании утвержденных заявлений студентов выпускается приказ о назначении тем ВКР и научных руководителей.

Изменение или уточнение темы ВКР и/или смена научного руководителя возможны не позднее, чем за один календарный месяц до представления ВКР секретарю ГЭК на основании личного заявления студента, согласованного с руководителем, на имя заведующего кафедрой. Изменение или уточнение темы ВКР утверждается заведующим кафедрой и оформляется

приказом ректора.

Можно выделить три основных категории (типа) тем ВКР:

1. Сравнительный анализ существующих решений – категория **аналитическая работа**.
2. Решение конкретной теоретической задачи - категория **научно-исследовательская работа**.
3. Решение конкретной прикладной задачи – категория **научно-практическая работа**.

Аналитическая работа может включать

1. Описание предметной области и анализируемых программных продуктов/решений/технологий.
2. Выделение задач их анализа/тестирования/сравнения.
3. Описание выбранной методики анализа и обоснование выбора определённого решения и/или технологии.
4. Описание инструментальных средств разработки/тестирования и обоснование сделанного выбора.
5. Подробное и качественное представление результатов анализа.
6. Обсуждение результатов анализа, выявление достоинств и недостатков, выработка рекомендаций для практического применения или для дальнейшей модификации.

Научно-исследовательская работа может включать

1. Обзор предшествующих достижений с оценкой научного вклада.
2. Подробное описание математической или иной модели и алгоритмов.
3. Теоретическая оценка сложности основных алгоритмов решения задачи.
4. Экспериментальная проверка адекватности модели (корректность, полнота, точность и т.п.).
5. Экспериментальная (эмпирическая) оценка вычислительной сложности предлагаемых решений и т.д.

Научно-практическая работа может включать

1. Обоснование значимости предлагаемой системы или технологии разработки ПО, обзор и сравнительный анализ существующих решений.
2. Подробное формализованное описание системы (общая архитектура, структура классов и их интерфейсы, объёмные характеристики и т.д.) или технологии.
3. Описание системы или технологии с точки зрения пользователя.
4. Обоснование оптимальности выбранных решений, в том числе на основе анализа их экспериментальных испытаний.
5. Сравнение предлагаемой системы или технологии с известными аналогами по функциональности, эффективности и удобству использования.
6. Демонстрацию системы и документацию к ней или данные по использованию/внедрению/апробации технологии.

1.4 Научное руководство и консультирование

Непосредственное руководство студентом при выполнении ВКР осуществляет научный руководитель.

Для руководства отдельными разделами ВКР, связанными со специфическими предметными областями, а также в тех случаях, когда тематика ВКР носит междисциплинарный характер, могут назначаться консультанты.

Руководителями ВКР могут назначаться сотрудники Университета, имеющие научно-педагогический стаж работы (профессорско-преподавательский состав, научные работники), представители сторонних организаций, привлеченные на договорных условиях.

Научный руководитель ВКР обязан:

- помочь студенту в выборе темы ВКР, разработке плана ее выполнения;
- оказать помощь в выборе методики проведения исследования;
- консультировать при подборе источников и фактического материала;
- осуществлять систематический контроль выполнения ВКР в соответствии с разработанным планом;
- информировать заведующего кафедрой ИВТ в случае несоблюдения студентом графика выполнения ВКР;
- давать квалифицированные рекомендации по содержанию ВКР;
- оценивать качество выполнения ВКР в соответствии с предъявляемыми к ней требованиями (отзыв научного руководителя).

План выполнения ВКР может включать следующие элементы:

- выбор и формулирование темы, разработку плана исследования и предварительного содержания ВКР;
- сбор и изучение исходного материала, поиск литературы и иных источников;
- анализ собранного материала,
- теоретическую разработку задачи;
- обсуждение предварительных результатов исследования с научным руководителем (консультантом);
- реализацию основных методов, моделирование, реализацию алгоритмов, проведение экспериментов и т.п.;
- обсуждение работы (на научно-исследовательском семинаре, на конференциях и т.п.) и анализ полученных результатов;
- оформление работы

и т.п.

Контроль за выполнением плана работы над ВКР осуществляет руководитель и/или консультант ВКР.

1.5 Преддипломная практика

Основой для выполнения ВКР является преддипломная практика. Преддипломная практика осуществляется в рассредоточенной и выездной формах [4]. Рассредоточенная форма преддипломной практики предполагает выполнение ВКР в ходе 8-го семестра, для чего в расписании занятий выделяется один день в неделю, освобожденный от аудиторных занятий. На этом этапе практики студент должен выполнить анализ существующих решений по заданной предметной области, описать бизнес-процессы и выполнить конструирование системы: разработать основные модели, алгоритмы и структуру проектного решения. Выездная форма преддипломной практики осуществляется в конце 8-го семестра, в организациях и учреждениях, в отделах и службах, ведущих деятельность, связанную с использованием современных информационных технологий. На этом этапе работы студент должен завершить разработку программной системы, выполнить ее тестирование, разработать программную документацию. По окончании практики студент должен подготовить Отчёт в виде презентации и выступить с докладом перед комиссией, состоящей из преподавателей отделения (кафедры) программной инженерии. В презентации необходимо отразить следующее:

- описание предметной области,
- основные понятия и определения,
- актуальность темы,
- цель и задачи ВКР,
- анализ текущего состояния по рассматриваемой теме,
- основные методы, модели, алгоритмы и т.д. для решения задач.

Студент должен продемонстрировать некоторые результаты исследования и проведенного эксперимента и прототип программного средства.

Студенты, не прошедшие преддипломную практику или получившие неудовлетворительную оценку на защите Отчета, к выполнению ВКР не допускаются.

1.6. Требования к ВКР

Бакалаврская выпускная работа представляет собой самостоятельное законченное исследование на заданную тему, написанное лично выпускником под руководством руководителя, свидетельствующее об умении выпускника работать с источниками, обобщать и анализировать фактический материал, используя теоретические знания и практические навыки, полученные при освоении образовательной программы. Бакалаврская работа может основываться на обобщении выполненных выпускником курсовых работ и содержать материалы, собранные выпускником в период производственной и преддипломной практик.

ВКР должна включать обоснование актуальности темы и ее связь с существующими аналогичными разработками. ВКР бакалавра должна иметь научную новизну и/или практическую значимость.

Содержание работы могут составлять:

- результаты теоретических исследований,
- разработка новых методологических подходов к решению научных или прикладных проблем,
- решение задач прикладного характера и т.д.

ВКР представляется в виде, который позволяет судить, насколько полно отражены и обоснованы содержащиеся в ней положения, выводы и рекомендации, их новизна и значимость. Совокупность полученных в такой работе результатов должна свидетельствовать о наличии у ее автора навыков аналитической, научно-исследовательской или научно-практической работы в избранной области профессиональной деятельности.

ВКР выполняется студентом по материалам, собранным им лично за время обучения, НИР, производственной и/или преддипломной практики.

Рекомендуемый объем пояснительной записке к ВКР - не более 60 страниц печатного текста без приложений. Пояснительная записка к ВКР оформляется в соответствии с системой стандартов по информации, библиотечному и издательскому делу [5].

2 Требования к содержанию пояснительной записки к ВКР бакалавра

2.1 Содержание пояснительной записки к ВКР

Пояснительная записка к ВКР бакалавра (ПЗ ВКР) **должна** содержать следующие структурные элементы:

- Титульный лист ¹⁾);
- Аннотации на русском, кыргызском, английском языках;
- Содержание;
- Введение;
- Основная часть;
- Заключение;
- Список использованных источников;
- Приложение 1. Глоссарий;
- Приложение 2. Листинг программного продукта;
- Приложения другие (если необходимо).

Следует помнить, что не все позиции, указанные в пояснениях к элементам, возможно или целесообразно расписывать в пояснительной записке. Конкретное наполнение информацией структурных элементов ПЗ ВКР определяется спецификой разрабатываемой программной системы, при этом последовательность изложения сохраняется.

¹⁾ см. Приложение В.

2.2 Требования к Аннотации

Аннотация - краткая (не более одной страницы) информация о работе. Аннотация должна содержать:

- текст аннотации;
- сведения об объеме работы в страницах, количестве глав, иллюстраций, таблиц, приложений, использованных источников;
- перечень ключевых слов.

Текст аннотации должен отражать:

- объект исследования или разработки;
- цель работы;
- методы исследования или разработки;
- научная новизна и/или практическая значимость работы;
- основные результаты работы;
- апробация работы (внедрение, публикации, выступления на конференциях); область применения,
- перспективы продолжения исследования или разработки.

Некоторые из этих сведений могут отсутствовать.

Перечень ключевых слов должен включать от 5 до 10 слов или словосочетаний из текста работы, которые в наибольшей мере характеризуют ее содержание и обеспечивают возможность информационного поиска. Ключевые слова приводятся в именительном падеже и печатаются строчными буквами в строку через запятую.

2.3 Требования к Содержанию

Содержание (до 2-х листов) перечисляет вышеуказанные структурные элементы ВКР, начиная с Введения. Главы, параграфы и подпараграфы основной части ПЗ к ВКР нумеруются (см. раздел Основная часть). Введение, Заключение, Список использованных источников не нумеруются. Особое внимание следует обратить на соответствие номеров страниц в содержании и по тексту ПЗ. Рекомендуется с этой целью использовать режим автоматического составления содержания.

2.4 Требования к Введению

Во введении (объем до 10 страниц) должна быть представлена следующая информация:

- 1) актуальность темы;
- 2) краткая характеристика объекта и предмета исследования;
- 3) оценка современного состояния проблемы (степень проработанности с указанием источников) и сравнительный анализ систем-аналогов;
- 4) цели и задачи исследования;

5) краткое описание структуры ПЗ.

Объектом исследования могут быть:

- процесс или явление, порождающие проблемную ситуацию и взятые исследователем для изучения;
- материальная или идеальная система;
- совокупность (класс) сходных явлений и ситуаций, а не просто отдельное явление (конкретная ситуация).

Предметом исследования могут быть:

- структура системы, закономерности взаимодействия элементов внутри системы и вне ее, закономерности развития, различные свойства, качества и т. д.;
- элементы и/или свойства объекта, подлежащие исследованию, наиболее полно отражающие выявленную проблему;
- аспект проблемы, при исследовании которого формируется представление о целостном объекте с выделением главных, наиболее существенных признаков.

Объект и предмет исследования как научные категории соотносятся как общее и частное, первичным является объект исследования (более широкое понятие), вторичным — предмет исследования, в котором выделяется определенное свойство объекта исследования.

Предмет исследования чаще всего перекликается с определением темы ВКР.

Цель работы. При формулировании цели работы рекомендуется использовать следующие выражения: «определить ...», «исследовать ...», «выявить ...», «разработать ...» «программный продукт (систему), методику, модель, инструменты, методы, механизмы, способы, критерии, требования, основы ... ».

Задачи исследования — действия, которые необходимо выполнить для достижения поставленной цели исследования. Наиболее распространенными могут быть следующие формулировки задач исследования: «провести анализ ...», «выявить ...», «определить ...», «сформулировать ...», «исследовать ...», «разработать ...», «провести апробацию ...», «внедрить ...».

В качестве конкретного примера вышеназванных понятий, которые необходимо раскрыть во введении, можно привести следующие формулировки:

объект исследования — процессы продвижения «тиражных» программных продуктов малых компаний-разработчиков на корпоративном рынке субъектов малого и среднего бизнеса с использованием инструментов интернет-маркетинга;

предмет исследования — методы, модели и механизмы поддержки принятия решений на разных стадиях разработки программы продвижения «тиражных» программных продуктов на корпоративном рынке с использованием инструментов интернет-маркетинга;

цель исследования — разработка комплекса моделей и инструментальных средств поддержки принятия решений при продвижении «тиражных» программных продуктов на корпоративном

рынке субъектов малого и среднего бизнеса с использованием инструментов интернет-маркетинга;

задачи исследования, которые необходимо решить для достижения цели исследования, выполненного в рамках бакалаврской работы:

- 1) определить специфику маркетинговой деятельности продуктовых IT-компаний, специализирующихся на разработке и тиражировании собственных ПП;
- 2) провести анализ методов и инструментов продвижения тиражных продуктов на корпоративном рынке, а также пакетов прикладных программ поддержки принятия решений при организации этих процессов;
- 3) разработать функциональную модель организации процесса продвижения программного продукта на рынок с учетом специфики использования Интернет как основного канала коммуникаций;
- 4) разработать и исследовать модели и алгоритмы поддержки принятия решений на разных стадиях процесса продвижения программного продукта на корпоративном рынке малых и средних компаний с использованием инструментов интернет-маркетинга;
- 5) разработать программный комплекс поддержки принятия решений при организации продвижения ПП на корпоративный рынок субъектов малого и среднего бизнеса с использованием инструментов интернет-маркетинга;
- 6) провести практическую апробацию полученных результатов и внедрение программного комплекса.

2.5 Требования к Основной части

Основная часть должна содержать следующие обязательные главы: Видение программного продукта, Спецификация требований к программному продукту, Проектирование и конструирование ПО, Разработка тестов и тестирование ПО, Руководство пользователя. При работе над этими главами рекомендуется использовать источники [6-10]. Далее представлены требования к главам, параграфам и подпараграфам, при этом нумерация соответствует нумерации в Содержании пояснительной записки к ВКР.

2.5.1 Содержание Главы 1 «Видение программного продукта»

- 1.1. Введение
 - 1.1.1. Цель
 - 1.1.2. Контекст
 - 1.1.3. Определения, акронимы и сокращения
 - 1.1.4. Ссылки
 - 1.1.5. Краткое содержание
- 1.2. Позиционирование программного продукта
 - 1.2.1. Деловые преимущества

- 1.2.2. Определение проблемы
- 1.2.3. Определение позиции программного продукта
- 1.3. Описание пользователей
 - 1.3.1. Сведения о пользователях
 - 1.3.2. Пользовательская среда
 - 1.3.3. Профили пользователей
 - 1.3.4. Ключевые потребности пользователей
- 1.4. Краткий обзор программного продукта
 - 1.4.1. Контекст использования системы
 - 1.4.2. Сводка возможностей
 - 1.4.3. Предположения и зависимости
- 1.5. Возможности продукта
- 1.6. Ограничения
- 1.7. Показатели качества
 - 1.7.1. Применимость
 - 1.7.2. Надежность
- 1.8. Старшинство и приоритеты
- 1.9. Другие требования к программному продукту
 - 1.9.1. Применяемые стандарты
 - 1.9.2. Системные требования
 - 1.9.3. Эксплуатационные требования
- 1.10. Требования к документации
 - 1.10.1. Руководство пользователя
 - 1.10.2. Интерактивная справка
 - 1.10.3. Руководства по установке и конфигурированию, файл ReadMe
- 1.11. Маркировка и пакетирование

2.5.2 Содержание Главы2 «Спецификация требований к программному продукту»

- 2.1. Введение
 - 2.1.1. Назначение документа
 - 2.1.2. Цели создания системы, решаемые задачи и область применения
 - 2.1.3. Определения, акронимы и сокращения
 - 2.1.4. Ссылки
 - 2.1.5. Краткое содержание
- 2.2. Общее описание
 - 2.2.1. Позиционирование программного продукта
 - 2.2.2. Функциональность программного продукта

- 2.2.3 Пользовательские характеристики
- 2.2.4 Ограничения
- 2.2.5 Предположения и зависимости
- 2.2.6 Распределение требований
- 2.3. Специфические требования
 - 2.3.1. Внешние требования к интерфейсам
 - 2.3.2. Функциональные требования
 - Спецификация варианта использования ВИ 1
 - Спецификация варианта использования ВИ 2
 - 2.3.3. Требования к производительности
 - 2.3.4. Логические требования к базе данных
 - 2.3.5 Ограничения дизайна
 - 2.3.6. Ограничения проектирования
 - 2.3.7. Соответствие стандартам
 - 2.3.8. Атрибуты программного продукта

2.5.3. Требования к Спецификации требований к программному продукту

Спецификация требований отвечает на вопрос: ЧТО И НАСКОЛЬКО ХОРОШО НАДО СДЕЛАТЬ?

Документ Спецификация требований (SRS- – Software Requirements Specification) соответствует стандарту IEEE Standard 830-1998, «IEEE Recommended Practice for Software Requirements Specifications»

2.1. Введение

2.1.1. Назначение

Необходимо указать, что представляет собой данный документ, в связи с чем он разработан и для кого предназначен; необходимо задать целевую аудиторию Спецификации требований.

Пример:

Назначение документа «Спецификация требований к» - определение требований к системе и согласование их с заказчиком. Документ включает общее видение продукта, формализованные требования и описание объектов систем. Данный документ является достаточным для разработки программного продукта.

2.1.2. Цели создания системы, решаемые задачи и область применения

В этот подраздел входят сведения, которые описывают существующую технологию, цели и назначение создания системы, а также эффекты от ее внедрения. По своей сути данный подраздел является бизнес-требованиями (Business requirements) к разрабатываемой системе.

2.1.2.1. Краткое описание деятельности заказчика и существующей технологии

2.1.2.2. Цели создания системы. Эффекты от ее внедрения

2.1.2.3. Назначение системы и область применения

Краткое описание деятельности заказчика и существующей технологии (Описание объекта автоматизации). Деятельность заказчика и существующая технология обычно описываются "естественным" языком. Однако, в случае, когда этого требует специфика проекта (например, была проведена предварительная работа по описанию процессов предметной области заказчика и построена модель AS IS) можно, и даже нужно дополнять текстовое описание графическими моделями. Это поможет команде разработчиков более грамотно ориентироваться в существующем бизнесе заказчика и глубже понять его потребности. Вообще, традиционно считается, что связующим мостом между заказчиком и разработчиками при разработке автоматизированной системы является аналитик, и именно он и только он должен вникать в тонкости деятельности заказчика. И зачастую разработчики не удосуживаются разобраться в предметной области, которую они автоматизируют. Однако практика показывает, что на тех проектах, где разработчики не ограничиваются только непосредственно требованиями, а пытаются разобраться в том, как, собственно, устроен сам бизнес, вероятность успеха более высокая.

В некоторых случаях данный подраздел стоит вынести отдельным разделом и вставить туда оценку текущего состояния автоматизации, если на ее средства необходимо будет опираться. В этом разделе может также приводиться описание оргструктуры и процессов предметной области заказчика. Описание в этом случае должно сопровождаться соответствующими диаграммами – моделями AS IS.

Цели создания системы. Эффекты от ее внедрения. Цели создания системы показывают, каким именно образом внедрение системы повлияет положительно на бизнес заказчика. Любая работа начинается с того, что формулируется цель выполнения этой работы. Далее, согласно цели, разрабатывается последовательность шагов (задач), которая должна привести к достижению этой поставленной цели.

При определении формулировок целей желательно, чтобы цель отвечала критериям SMART. Т.е., была конкретной (S - specific), измеримой (M - measurable), достижимой (A - attainable), релевантной (R – relevant, актуальный) и имела сроки исполнения (T - time-bound). Иначе говоря, для того, чтобы определить, достиг ли проект своей цели, необходимо формулировать конкретные и измеримые цели, имеющие непосредственное отношение к решаемой проблеме и достижимые в определенные отрезки времени. При этом желательно указать, как именно можно будет измерить результаты деятельности. Поэтому, для того, чтобы

сделать определение положительного влияния внедрения системы более наглядным, в документе приводят эффекты от внедрения системы. Эффекты оперируют числовыми показателями, как правило, полученными на основе имеющихся статистических данных. Например, "уменьшить расходы на обслуживание бизнес-процесса на 10% после внедрения системы по сравнению с текущими показателями (данные берутся из квартальных отчетов)", "увеличить количество обрабатываемых документов с 20 до 50 в день в течение первых 6 месяцев эксплуатации системы" и т.д. Приведенные расчеты должны быть выполнены еще на этапе подготовки к проекту, и являются основанием для принятия решения об автоматизации. Наличие подобных расчетов показывает серьезность подхода заказчика к разработке АС и позволяет избежать многих рисков, свойственных проектам подобного рода. Может так случиться, что разработка системы обойдется дороже, чем ожидаемый эффект от ее внедрения.

Не все изменения после внедрения системы можно описать "числовым" языком. Но все же к этому надо стремиться.

Назначение системы и область применения. Здесь необходимо:

- определить название программного продукта;
- объяснить, что программный продукт будет делать;
- описать типы/категории пользователей и преимущества использования ими программного продукта.

В подавляющем большинстве случаев создание информационной системы преследует автоматизацию бизнес-процессов Заказчика. Однако, как правило, Заказчик не требует сквозной автоматизации своих процессов (что, по правде сказать, возможно только при высочайшей культуре процессного управления), а желает просто получить автоматизацию каких-то отдельных функций (сбор и накопление информации, формирование отчетов, управление потоками данных и т.п.) Назначение системы должно отвечать на вопрос о том, какие задачи с точки зрения Заказчика создаваемый продукт должен решать. При этом должны быть определены как автоматизируемые бизнес-процессы, так и задачи, выполнение которых должна обеспечивать проектируемая система для автоматизации перечисленных процессов. Таким образом, назначение системы довольно точно определяет границы проекта.

Представленная в данном разделе функциональность АС является верхнеуровневой. Например:

Система предназначена для автоматизации процессов планирования и исполнения бюджета, мониторинга и анализа финансовых показателей.

Система обеспечивает автоматизацию сбора, обработки, хранения и формирования данных, а также обеспечивает информационную поддержку следующих процессов:

-....

-....

Объектами автоматизации являются:.....

Подробный перечень автоматизируемых функций системы, обеспечивающих реализацию ее назначения, приводится в разделе "Функциональность программного продукта".

2.1.3. Определения, акронимы и сокращения – указать ссылку на Приложение 1.

2.1.4. Ссылки – указать ссылки на используемые источники из Списка использованных источников.

2.1.5. Краткое содержание – описать содержимое остальной части Спецификации требований; пояснять организацию Спецификации требований.

2.2. Общее описание

Этот раздел SRS должен описывать общие факторы, оказывающие влияние на продукт и требования к нему. В этом разделе не приводятся специфические требования. В нем подготавливается основа для требований, которые детально определяются в подразделе 3 «Специфические требования», и приводится информация, облегчающая их понимание.

Этот раздел обычно состоит из шести подразделов:

- Позиционирование продукта;
- Функции продукта;
- Характеристики пользователей;
- Ограничения;
- Предположения и зависимости;
- Распределение требований.

2.2.1 Позиционирование программного продукта

Этот подраздел позиционирует продукт среди других связанных продуктов. Если продукт является независимым и полностью самодостаточным, это следует отразить здесь. Если SRS определяет продукт, который является компонентом большей системы, как это часто бывает, то данный подраздел должен связать требования к большей системе с функциональностью программного обеспечения и определить интерфейсы между системой и программным обеспечением.

Могут быть полезными блок-схемы, показывающие основные компоненты большей системы, связи между ними и внешние интерфейсы.

Этот подраздел должен также описывать, как программное обеспечение работает под действием различных ограничений. Например, эти ограничения могут включать:

- Системные интерфейсы;
- Пользовательские интерфейсы;

- Аппаратные интерфейсы;
- Программные интерфейсы;
- Коммуникационные интерфейсы;
- Память;
- Операции;
- Требования к адаптируемости на месте.

2.2.2 Функциональность продукта

В этом подразделе SRS следует представить сводку основных функций, выполняемых системой. Иногда сводка функций, необходимых для данной части, берется прямо из соответствующего раздела спецификации верхнего уровня (если она есть), которая размещает некоторые функции в программном продукте. Уместна Use Case диаграмма верхнего уровня.

2.2.3. Характеристики пользователей

Этот подраздел SRS должен описывать общие характеристики предполагаемых пользователей, включая уровень образования, опыт и техническую грамотность. В нем не следует устанавливать специфические требования, но следует привести причины, по которым некоторые специфические требования заданы далее в Разделе 3 SRS.

2.2.4. Ограничения

В этом подразделе должны быть приведены описания всего того, что может ограничить действия разработчика. Они включают:

- правовые вопросы;
- аппаратные ограничения (например, требования к длительности сигналов);
- интерфейсы с другими приложениями;
- параллельные операции;
- функции аудита;
- функции управления;
- языковые ограничения высшего порядка;
- протоколы синхронизации сигналов;
- требования к надежности;
- критичность приложения;
- соображения безопасности и секретности.

2.2.5. Предположения и зависимости

В этом подразделе следует перечислить все факторы, которые влияют на требования, устанавливаемые SRS. Эти факторы не являются проектными ограничениями, а скорее относятся к их изменениям, которые могут повлиять на требования SRS. Например, может быть сделано предположение, что некоторая операционная система будет доступна для оборудования, на которое ориентирован программный продукт. Если в действительности операционная система

недоступна, потребуется соответствующее изменение SRS.

2.2.6. Распределение требований

Этот подраздел должен представлять требования, которые могут быть отложены до будущих версий системы.

2.3. Специфические требования

В этом разделе специфицируются требования в такой степени детальности, чтобы удовлетворить требования разработчиков и тестировщиков.

2.3.1. Требования к внешним интерфейсам – детализировать все входы и выходы системы (дополнить, не дублируя, информацию, представленную в 2.2.4)

- 2.3.1.1. Пользовательские интерфейсы
- 2.3.1.2. Аппаратные интерфейсы
- 2.3.1.3. Программные интерфейсы
- 2.3.1.4. Коммуникационные интерфейсы

2.3.2. Функциональные требования

Этот раздел содержит детализированные спецификации всех вариантов использования программной системы. В случае необходимости *допускается использование диаграмм деятельности или состояний* для пояснения сценария того или иного варианта использования.

Ниже представлен шаблон спецификации варианта использования.

Наименования и краткое описание. В этом разделе указывается: наименование варианта использования, актеры варианта использования, краткое (в один абзац) описание варианта использования.

Поток событий

.1. Основной поток событий

Перечисляются шаги основного сценария, начиная от триггера и вплоть до достижения гарантии успеха.

.2. Альтернативные потоки событий

Каждый из альтернативных сценариев описывается в отдельном параграфе, в том же стиле, что и основной поток событий. Альтернативные сценарии описывают поведение системы при любых отклонениях от основного сценария, а также поведение в исключительных ситуациях.

Специальные требования

Здесь перечисляются нефункциональные требования, имеющие непосредственное отношение именно к этому варианту использования.

Предусловия

События, описываемые предусловиями или постусловиями, должны быть состояниями, которые пользователь может наблюдать. Предусловие описывает состояние, в котором система должна находиться до начала исполнения прецедента.

Постусловия

- что гарантируется актерам-участникам. Например – в случае неудавшейся транзакции все данные, имевшиеся в системе до её начала, сохраняются неизменными.

Корректно сформулированное постусловие должно быть истинным при любом возможном сценарии прецедента, а не описанном в основном потоке.

Точки расширения

Данный параграф определяет положение точек, расширяющих поток событий.

2.3.3. Требования к производительности

Этот подраздел должен задавать как статические, так и динамические численные требования, предъявляемые в целом к программному обеспечению или к взаимодействию человека с программой. Статические численные требования могут включать следующие:

- Число поддерживаемых терминалов;
- Число одновременно поддерживаемых пользователей;
- Объем и тип обрабатываемой информации.

Динамические численные требования могут включать, например, число транзакций и задач, или объем данных, обрабатываемых в некоторый период в условиях как нормальной, так и пиковой нагрузки.

Все эти требования следует формулировать в терминах измеримых величин.

Например:

95% транзакций должны обрабатываться менее чем за 1 секунду

вместо:

Оператор не должен ждать, пока завершится транзакция.

2.3.4. Логические требования к базе данных

Здесь следует задать логические требования к информации, которая должна размещаться в базе данных. Они могут включать следующие:

- Типы информации, используемой различными функциями;
- Сущности и отношения между ними;
- Частоту использования;
- Ограничения целостности;
- Возможность доступа;
- Требования к хранению данных.

2.3.5. Ограничения проектирования

Ограничения проектирования - формулировки условий, модифицирующих требования и сужающих выбор возможных решений по их реализации.

Например, выбор платформы реализации и/или развертывания (протоколы, серверы приложений, баз данных, ...), которые, в свою очередь, могут относиться, например, к внешним интерфейсам.

Ограничение соответствия стандартам должно задавать ограничения, вытекающие из существующих стандартов и правил. Они могут включать следующее:

1. форматы отчетов;
2. именованние данных;
3. бухгалтерские процедуры;
4. протоколирование работы.

Другой пример ограничения: требования к программному обеспечению по трассировке вычислительных действий. Подобные трассировки необходимы для некоторых приложений для выполнения требований правовых или финансовых стандартов. Требование протоколирования работы может, например, устанавливать, что при изменениях в платежной ведомости прежнее и новое значения должны записываться в файл трассировки.

2.3.6. Атрибуты программной системы

Некоторые атрибуты программного обеспечения могут служить требованиями. Важно, чтобы требуемые атрибуты были заданы таким образом, чтобы можно было объективно проверить выполнение данного требования.

2.3.6.1. Надежность

Следует перечислить факторы, необходимые для установления требуемого уровня надежности программного обеспечения.

2.3.6.2. Доступность

Следует перечислить факторы, призванные гарантировать определенный уровень доступности системы, такие как контрольные точки, восстановление и перезапуск.

2.3.6.3. Безопасность

Следует перечислить факторы, защищающие программное обеспечение от случайного или злонамеренного доступа, использования, модификации, разрушения или разглашения.

Специфические требования в этой области включают необходимость:

- Использования криптографии;
- Хранение логов или истории;
- Назначать некоторые функции различным модулям;
- Ограничивать коммуникации между некоторыми областями программы;
- Проверять целостность данных для критических переменных.

2.3.6.4. Поддерживаемость

Следует перечислить атрибуты программного обеспечения, относящиеся к легкости поддержки самого программного обеспечения. Это могут быть некоторые требования, относящиеся к модульности, интерфейсам, сложности и т.д. Не следует помещать здесь требования лишь потому, что принято считать их хорошим тоном разработки.

2.3.6.5. Переносимость

Следует перечислить атрибуты программного обеспечения, касающиеся легкости переноса программного обеспечения на другие компьютеры и/или операционные системы. Они могут включать следующее:

- доля компонентов с машинно-зависимым кодом;
- доля машинно-зависимого кода;
- использование испытанного переносимого языка;
- использование особенного компилятора или подмножества языка;
- использование особенной операционной системы.

2.5.4. Требования к Главе 3 «Проектирование и конструирование ПО»

Этот раздел ПЗ ВКР отвечает на вопрос: КАК ЭТО ДЕЛАЛОСЬ?, т.е. описывает процесс разработки ПО.

3.1. Разработка диаграммы классов

Следует помнить, что разработка диаграммы классов фактически ведётся на протяжении всего периода проектирования ПС, изменяется и уточняется по мере приближения к завершению проекта. Сама диаграмма классов является исключительно важным компонентом UML модели ПС, т.к. практически полностью определяет статическую организацию разрабатываемой системы, обосновывает её архитектуру и обеспечивает качество кода.

Классы для разрабатываемой системы можно «находить» двумя основными путями.

1. Внимательно разрабатывая, а затем читая спецификацию вариантов использования ПС. Следует вспомнить, что спецификация вариантов использования представляет собой достаточно подробное описание возможных сценариев работы ПС и/или с ПС, в которых принимают участие различные объекты и/или субъекты, например, Покупатель авиабилета, специальная форма для ввода запроса параметров авиарейса, окно с сообщением конкретного содержания и т.п. Очевидно, что эти объекты и субъекты должны быть представлены в программе в качестве экземпляров соответствующих классов. На диаграмме классов нужно отобразить классы сущностей, классы границ и классы управлений.

Те действия, которые могут осуществлять субъекты и объекты, также как правило, описываются в спецификации вариантов использования, будут не чем иным как методами, или операциями объектов программы. Например, Покупатель авиабилета может быть зарегистрирован в системе, может быть проверен, редактирован, удалён; Форма или Окно могут быть открыты, закрыты, могут быть изменены их размеры. Соответствующими методами могут стать, например, зарегистрировать (), редактировать(), удалить(), изменить (размеры) и т.п. Величины, перечисляемые в скобках, являются параметрами методов, и они могут менять свои значения от одного сценария к другому.

2. Разработанная спецификация вариантов использования позволяет создать столько диаграмм последовательности, сколько сценариев работы ПС и/или с ПС предусмотрено для её практического использования. В этом случае объекты и субъекты, участвующие в конкретном сценарии, располагаются в заголовке диаграммы. Между ними пересылаются сообщения

различного характера. Характер сообщений «маркируется» стереотипами. Посылка и приём сообщения активизируют тот или иной объект, создавая их фокусы активности.

Помня о том, что объект – это экземпляр определённого класса, читая диаграммы последовательности, следует создать соответствующие классы и показать их на диаграмме классов. Второй путь следует из унифицированного процесса разработки ПС.

Особое значение при любом подходе к построению диаграммы классов имеет установление отношений между классами. На диаграмме классов не может быть отношений только одного типа! Язык UML располагает шестью типами отношений, совокупность которых может вполне детально представить характер взаимосвязей и взаимодействия классов в процессе функционирования ПС.

При разработке диаграммы классов следует помнить, что понятие объекта «исторически» является первичным по отношению к понятию класса. Объекты – это конкретные сущности, которые существуют во времени и пространстве. В понятие программного объекта вкладывалась простая идея, что программные объекты должны моделировать физические объекты реального мира и поэтому они должны вобрать в себя черты и характеристики физических объектов. Однако, создавая понятие в интересах какой-либо задачи, мы отвлекаемся (абстрагируемся) от несущественных характеристик конкретных объектов, определяя только существенные характеристики. Например, в абстракции «часы» целесообразно выделить характеристику «показывать время», отвлекаясь от таких характеристик конкретных часов как форма, цвет, материал, цена, изготовитель. Можно сказать, что поиск, выделение, определение объектов в качестве экземпляров программных классов сводится к формированию абстракций и представляет собой процесс абстрагирования. Каждая абстракция фиксирует основные характеристики объекта, которые отличают его от других видов объектов и обеспечивают ясные понятийные границы. Выбор и правильное описание набора абстракций для заданной предметной области (т.е. области деятельности заказчика ПО) является главной задачей объектно-ориентированной разработки.

Объект обладает **индивидуальностью, состоянием и поведением**. Структура и поведение подобных объектов определены в их общем классе. Говорят, что состояние объекта представляет суммарный результат его поведения [10]. Таким образом, анализируя предметную область, можно определить объекты, свойственные именно этой конкретной области деятельности и сформировать основные классы сущностей. Имена этих классов должны соответствовать реальным объектам. Совокупность имён этих объектов составляет терминологический словарь предметной области – так называемый тезаурус. На практике применяется метод проверки корректности и полноты набора найденных объектов (классов), которые нужно создать в программе: сравниваются перечень программных объектов (классов сущностей) и терминологический словарь предметной области. Они должны совпадать.

Далее анализируется поведение объектов (с помощью UML диаграмм последовательностей) и выявляются действия, которые совершают объекты или действия, которые совершаются над ними – так происходит уточнение программных абстракций. При этом могут появиться специальные программные классы, например, метаклассы, шаблоны, классы-функции и т.д. И так - итерация за итерацией - уточняется диаграмма классов.

3.2. Разработка диаграммы (диаграмм) последовательностей

Один из способов разработки диаграмм последовательностей описан выше.

Другой состоит в том, чтобы «выносить» на диаграмму последовательности того или иного сценария объекты тех классов, которые уже разработаны и представлены на диаграмме классов. При этом возможно обнаружить, что уже имеющихся классов не хватает для реализации сценария или сценариев, тогда должно быть выполнено изменение, дополнение диаграммы классов, редактирование их атрибутов и/или операций (методов). Таким образом, разработка диаграммы классов происходит итеративно.

3.3. Разработка диаграммы компонентов

Статические и динамические модели описывают логическую организацию системы, отражают логический мир программного приложения. Модели реализации обеспечивают представление системы в физическом мире, рассматривая вопросы упаковки логических элементов в компоненты и размещения компонентов в аппаратных узлах.

Компонентная диаграмма — первая из двух разновидностей диаграмм реализации, моделирующих физические аспекты объектно-ориентированных систем. Компонентная диаграмма показывает организацию набора программных компонентов и зависимости между ними.

По своей сути компонент является физическим фрагментом реализации системы, который включает в себе программный код (исходный, двоичный, исполняемый), сценарные описания или наборы команд операционной системы (имеются в виду командные файлы).

Язык UML дает следующее определение компоненту.

Компонент — физическая и заменяемая часть системы, которая соответствует набору интерфейсов и обеспечивает реализацию этого набора интерфейсов.

Интерфейс — очень важная часть понятия «компонент». Интерфейс — список операций, которые определяют услуги класса или компонента. Образно говоря, интерфейс — это разъем, который торчит из ящичка компонента. С помощью интерфейсных разъемов компоненты стыкуются друг с другом, объединяясь в систему.

Очень важна взаимосвязь между компонентом и интерфейсом. Возможны два способа отображения взаимосвязи между компонентом и его интерфейсами. В первом, свернутом способе, интерфейс изображается в форме пиктограммы подобно тому, как это делается на

диаграмме вариантов использования. Во втором способе используется развернутая форма изображения интерфейса, в которой могут показываться его операции. Компонент, который реализует интерфейс, подключается к нему отношением реализации. Компонент, который получает доступ к услугам другого компонента через интерфейс, подключается к интерфейсу отношением зависимости.

По способу связи компонента с интерфейсом различают:

- экспортируемый интерфейс — тот, который компонент реализует и предлагает как услугу клиентам;
- импортируемый интерфейс — тот, который компонент использует как услугу другого компонента.

У одного компонента может быть несколько экспортируемых и несколько импортируемых интерфейсов.

Тот факт, что между двумя компонентами всегда находится интерфейс, устраняет их прямую зависимость. Компонент, использующий интерфейс, будет функционировать правильно вне зависимости от того, какой компонент реализует этот интерфейс. Это очень важно и обеспечивает гибкую замену компонентов в интересах развития системы.

Повторное использование — магистральный путь развития программного инструментария. Создание нового ПО из существующих, работоспособных программных компонентов приводит к более надежному и дешевому коду. При этом сроки разработки существенно сокращаются.

Основная цель программных компонентов — допускать сборку системы из двоичных заменяемых частей. Они должны обеспечить начальное создание системы из компонентов, а затем и ее развитие — добавление новых компонентов и замену некоторых старых компонентов без перестройки системы в целом. Ключ к воплощению такой возможности — интерфейсы. После того как интерфейс определен, к выполняемой системе можно подключить любой компонент, который удовлетворяет ему или обеспечивает этот интерфейс. Для расширения системы производятся компоненты, которые обеспечивают дополнительные услуги через новые интерфейсы.

Реализация системы может включать большое количество разнообразных компонентов:

- исполняемых элементов;
- динамических библиотек;
- файлов данных;
- справочных документов;
- файлов инициализации;
- файлов регистрации;
- сценариев;
- файлов установки.

В языке UML для компонентов определены следующие стереотипы:

- Библиотека (library) - определяет первую разновидность компонента, который представляется в форме динамической или статической библиотеки.

- Таблица (table) - также определяет первую разновидность компонента, который представляется в форме таблицы базы данных.
- Файл (file) - определяет вторую разновидность компонента, который представляется в виде файлов с исходными текстами программ.
- Документ (document) - определяет вторую разновидность компонента, который представляется в форме документа.
- Исполнимый (executable) - определяет третий вид компонента, который может исполняться в узле (Узел – это элемент диаграммы размещения. Узлами могут быть, например, сервер, рабочая станция, сканер и др. аппаратные устройства, входящие в состав компьютерной системы).

3.3.1. Компоновка основного (основных) программного модуля

Разработка диаграммы компонентов предполагает использование информации не только о логическом представлении модели системы, но и об особенностях ее физической реализации. В первую очередь, необходимо решить, из каких физических частей или файлов будет состоять программная система. На этом этапе следует обратить внимание на такую реализацию системы, которая обеспечивала бы возможность повторного использования кода за счет рациональной декомпозиции *компонентов*, а также создание объектов только при их необходимости.

Общая производительность программной системы существенно зависит от рационального использования вычислительных ресурсов. Для этой цели необходимо большую часть описаний классов, их операций и методов вынести в динамические библиотеки, оставив в исполняемых *компонентах* только самые необходимые для инициализации программы фрагменты программного кода.

После общей структуризации физического представления системы необходимо дополнить модель интерфейсами и схемами базы данных. При разработке *интерфейсов* следует обращать внимание на согласование различных частей программной системы. Включение в модель схемы базы данных предполагает спецификацию отдельных таблиц и установление информационных связей между ними.

Завершающий этап построения диаграммы компонентов связан с установлением и нанесением на диаграмму взаимосвязей между *компонентами*, а также отношений реализации. Эти отношения должны иллюстрировать все важнейшие аспекты физической реализации системы, начиная с особенностей компиляции исходных текстов программ и заканчивая исполнением отдельных частей программы на этапе ее выполнения. Для этой цели можно использовать различные графические стереотипы *компонентов*.

При разработке диаграммы *компонентов* следует придерживаться общих принципов создания моделей на языке UML. В частности, в первую очередь необходимо использовать уже имеющиеся в языке UML и общепринятые графические и текстовые стереотипы. В большинстве типовых проектов этого набора достаточно для представления *компонентов* и зависимостей между ними.

Если же проект содержит физические элементы, описание которых отсутствует в языке UML, то следует воспользоваться механизмом расширения. В частности, можно применить дополнительные стереотипы для отдельных нетиповых *компонентов* или помеченные значения для уточнения отдельных характеристик *компонентов*.

Наконец, следует обратить внимание на то, что диаграмма компонентов, как правило, разрабатывается совместно с диаграммой развертывания, на которой представляется информация о физическом размещении *компонентов* программной системы по ее отдельным узлам.

3.3.2. Модуль _____

3.3.3. Модуль _____

3.3.4. Модуль взаимодействия с пользователем

3.3.5. Модули, реализующие дополнительные функции

3.3.6. Библиотечные программы (и т. д.)

Рекомендации, сформулированные выше, касаются принципов и приёмов выделения любых компонентов программных систем.

Язык UML имеет возможности показать, какие именно классы и/или файлы входят в состав того или иного компонента.

2.5.5 Содержание Главы 4 «Разработка тестов и тестирование ПО»

Этот раздел отвечает на вопрос: КАК ЭТО РАБОТАЕТ?

4.1. Разработка плана тестирования включает в себя: выбор среды тестирования, разработку сценариев тестирования, определение исходных данных для тестов.

4.2. Модульное тестирование подразумевает тестирование методов классов, функций и процедур.

4.3. Интеграционное тестирование или тестирование сборки включает в себя: выбор технологии тестирования, тестирование пользовательских и программных интерфейсов, тестирование БД (желательно).

4.4. Системное тестирование может включать в себя: тестирование производительности, нагрузочное тестирование (для систем, ориентированных на неопределенное количество пользователей, например, сайтов), тестирование других нефункциональных требований.

2.5.6. Требования к Главе 5 «Руководство пользователя»

Руководство пользователя (*User guide*) - документ, предназначенный для пользователя, отвечает на вопрос: КАК ИСПОЛЬЗОВАТЬ программный продукт, чтобы практически воспользоваться реализованными функциональными возможностями, заявленными ранее в Спецификации требований к функционалу?

Основная цель Руководства пользователя заключается в обеспечении пользователя необходимой информацией для самостоятельной работы с программой или автоматизированной системой.

Руководство пользователя должно отвечать на следующие вопросы: что это за программа, что она может, что необходимо для обеспечения ее корректного функционирования и что делать в случае отказа системы.

В пояснительную записку к ВКР включаются следующие основные разделы документа Руководство пользователя:

- назначение системы;
- условия применения системы;
- подготовка системы к работе;
- описание операций;
- аварийные ситуации.

5.1. Назначение системы

Данный раздел документа Руководство пользователя должен содержать информацию о назначении системы, ее целях и задачах.

Пример:

«Корпоративный интранет портал предназначен для повышения корпоративной культуры организации эффективного взаимодействия сотрудников.

Основной целью Порта является создание единого информационного пространства предприятия и оптимизация работы сотрудников путем облегчения коммуникаций между ними и оптимизации ряда бизнес-процессов.»

5.2. Условия применения системы

Данный раздел документа Руководство пользователя должен включать все те факторы, которые необходимы для корректной работы системы. Здесь можно выделить несколько подразделов:

- Требования к аппаратному обеспечению – сюда можно включить требования к конфигурации компьютера пользователя, программное обеспечение необходимое для работы Системы, а также наличие дополнительного оборудования (принтер, сканер и т.п.), если таковое необходимо;
- Квалификация пользователя – данный подраздел должен содержать требования к навыкам и знаниям пользователя (*пример: «Пользователи должны обладать навыками работы с операционной системой Windows XP»*).

5.3. Подготовка системы к работе

Данный раздел документа Руководство пользователя должен содержать пошаговую инструкцию для запуска приложения. К этапу подготовки системы к работе можно отнести установку дополнительных приложений (при необходимости), идентификацию, аутентификацию и т.п.

5.4. Описание операций

Это основной раздел документа Руководство пользователя, который содержит пошаговую инструкцию для выполнения той или иной функции системы (сценария варианта использования системы, описанного в Спецификациях вариантов использования (см. п. 2.3.2)). В этом разделе Руководства необходимо выделить подразделы, описывающие функции, и действия (операции), которые необходимо совершить для их выполнения.

Для каждой операции обработки данных указывается:

- 1) наименование;
- 2) условия, при соблюдении которых возможно выполнение операции;
- 3) подготовительные действия;
- 4) основные действия в требуемой последовательности;
- 5) заключительные действия;
- 6) ресурсы, расходуемые на операцию.

Пример:

«5.4.1 Согласование проекта. Данный процесс предназначен для организации работы сотрудников, участвующих в разработке и согласовании проекта. Автор проекта создает запись в Системе и прикрепляет пакет необходимой документации, далее проект передается на согласование руководящими лицами. Руководители после ознакомления с проектом могут подтвердить его или отправить на доработку Автору.

5.4.1.1 Создание проекта. Для того, чтобы создать Проект, необходимо на панели «...» нажать на кнопку «...» и в появившейся форме заполнить следующие поля:

- Наименование проекта;
- Описание проекта;

Следующие поля заполняются автоматически:

- Дата создания проекта – текущая дата;
- Автор – ИФО и должность автора проекта.»

5.5. Аварийные ситуации

Данный раздел документа Руководство пользователя должен содержать пошаговые инструкции действий пользователя в случае отказа работы Системы:

- 1) Действия в случае несоблюдения условий выполнения технологического процесса, в том числе при длительных отказах технических средств.
- 2) Действия по восстановлению программ и/или данных при отказе магнитных носителей или обнаружении ошибок в данных.

- 3) Действия в случаях обнаружении несанкционированного вмешательства в данные.
- 4) Действия в других аварийных ситуациях

Если к пользователю не были предъявлены особые требования по администрированию операционной системы и т.п., то можно ограничиться фразой «При отказе или сбое в работе Системы необходимо обратиться к Системному администратору».

Руководство пользователя помимо текстовых описаний содержит изображения: [снимки экрана](#) – скрин-шоты, а при описании аппаратных средств — простые и понятные рисунки или фотографии. Используется стиль и язык, доступный предполагаемой аудитории, использование [жаргона](#) сокращается до минимума либо всё подробно объясняется.

Особенности изложения Руководства пользователя (*полезно учитывать и при написании других разделов пояснительной записки*)

Помните, что [стиль изложения](#) в устной речи или в деловом письме отличается от стиля технической документации, и в частности, Руководства пользователя.

Стиль Руководства должен быть нейтрально-формальным – использование стилистически окрашенных слов отвлекает пользователя от сути.

Пишите кратко и логично. Не давайте лишних деталей, не дублируйте информацию. Последовательность упоминания информации в Руководстве пользователя должна совпадать с последовательностью действий пользователя:

Хорошо: In File menu, select **Save item**.

Хуже: Select **Save** item from File menu.

Используйте повелительное наклонение, не употребляйте вежливые обороты (please, could и т.д.)

Хорошо: Click **Logout** to log out current user account from the system.

Хуже: It is needed to click **Logout** to log out current user account from the system.

Хуже: If user wants to log out current user account from the system(s) he needs to click **Logout**.

Структурируйте информацию. Часто можно встретить совет, что надо стараться избегать списков, однако, структурированная по шагам информация всегда лучше воспринимается.

Хорошо:

To create project:

1. Click the **Create** button on toolbar.
2. On the **Create Project** overlay fill in all mandatory fields.
3. Click the **Save button** to save the project.

Хуже: To create project click the **Create** button on toolbar, on the **Create Project** overlay fill in all mandatory fields, click the **Save** button to save the project.

Не используйте будущее или прошлое время. У ПО нет прошлого или будущего: всё случается в настоящем как прямой результат конкретного действия пользователя. Как только событие случается, ПО реагирует.

Хорошо: When user clicks the **Start** button, the program **starts** the process.

Хуже: When user clicks the **Start** button, the program **will start** the process.

Проверяйте значение слов. Надо стараться максимально избегать ошибок, связанных с недостаточным знанием языка.

Например, глагол «press» означает нажатие клавиши на клавиатуре, а «click» – нажатие кнопки или значка в окне программы при помощи мыши, а «hit» вообще является жаргонным словом.

Разумеется, орфографические ошибки недопустимы.

Не используйте синонимы для одного и того же термина. Определитесь с терминологией и придерживайтесь ее в рамках всего документа.

Например, не допускайте, чтобы в одной части документа выпадающий список назывался dropdown, а в другой точно такой же элемент – combobox или dropdown list. Это путает пользователя.

Разумно используйте сокращения и исключите жаргон.

Считается, что сокращения использовать не стоит, но если длинный термин употребляется несколько раз, то при первом упоминании в тексте надо писать полное название и рядом — аббревиатуру в скобках, а далее по тексту можно использовать только аббревиатуру. Аббревиатура должна содержаться и расшифровываться в Глоссарии (Приложение 1). Не используйте жаргонные слова, метафоры и термины, заимствованные из языка, отличного от языка Руководства.

2.5.7 Требования к Заключению

Заключение (не более 3 страниц) должно содержать:

- основные результаты выполненной работы;
- рекомендации по конкретному применению результатов работы;
- рекомендации по направлениям дальнейших исследований.

Основные результаты работы должны быть сформулированы в следующем виде:

«Выполнен анализ ... (обоснование актуальности)»; «Предложена функциональная (объектно-ориентированная) модель ...»; «Усовершенствована модель (методика)...»; «Результаты работы внедрены на предприятии...»; «Проведен анализ отечественных и зарубежных источников (аналогов)...»; «Предложена классификация существующих аналогов, алгоритмов, моделей ...»; «Разработан ПП ...»; «Разработана математическая модель (методика) на базе ... с учетом оптимизации параметров ...».

Фактически в Заключении повторяются в утвердительном тоне цель и задачи ВКР, заявленные во Введении.

2.5.8 Требования к Списку использованных источников

Список использованных источников (не менее одной страницы) оформляется в соответствии с требованиями (см. Приложение Б). На все источники, представленные в Списке, необходимы ссылки в тексте ВКР в квадратных скобках ([номер источника]).

Если по результатам выполненной работы студент имеет публикации и/или свидетельства о регистрации программ, их также необходимо включить в список использованных источников. Этот факт следует отметить в заключении ВКР.

2.5.9 Требования к Приложениям и их оформлению

Приложение 1. Глоссарий

Глоссарий является основой, ключом для единообразного понимания описаний требований Заказчиком и Разработчиком.

Глоссарий оформляется, как текст, состоящий из абзацев, каждый из которых определяет значение одного из терминов проблемной области. Термин обычно выделяют полужирным кеглем. Иногда проблемную область целесообразно сегментировать на ряд «подобластей» (subject areas). Тогда каждой из них в глоссарии выделяется отдельный параграф.

Приложение 2. Листинг программного продукта

Печатается шрифтом Times New Roman 12 pt, через 1 интервал.

3. Технические требования к оформлению пояснительной записки к ВКР

Общие технические требования

ВКР печатается на стандартных листах бумаги формата А4. Поля: левое – 30 мм, правое – 10 мм, верхнее и нижнее – 20 мм. Шрифт Times New Roman, размер 12 пт, межстрочный интервал 1,5, абзацный отступ 1,25 см, выравнивание по ширине. Каждая новая глава начинается с новой страницы; это же правило относится к другим основным структурным частям работы (введению, заключению, списку использованных источников, приложениям и т.д.).

Страницы ВКР, включая приложения, должны иметь сквозную нумерацию. Первой страницей является титульный лист, на котором номер страницы не ставится.

Каждый том ВКР должен быть переплетен в твердый переплет.

Правила написания буквенных аббревиатур

В тексте выпускной квалификационной работы, кроме общепринятых буквенных аббревиатур, могут быть использованы вводимые авторами буквенные аббревиатуры, сокращенно обозначающие какие-либо понятия из соответствующих областей знания. При этом первое упоминание таких аббревиатур указывается в круглых скобках после полного наименования, в дальнейшем они употребляются в тексте без расшифровки.

Полный перечень используемых сокращений должен быть приведен в Приложении 1 к

Правила написания формул, символов

Формулы располагают отдельными строками в центре листа или внутри текстовых строк. В тексте рекомендуется помещать формулы короткие, простые, не имеющие самостоятельного значения и не пронумерованные. Наиболее важные формулы, а также длинные и громоздкие формулы располагают на отдельных строках.

Нумеровать следует наиболее важные формулы, на которые имеются ссылки в работе. Порядковые номера формул обозначают арабскими цифрами в круглых скобках у правого края страницы. Нумерация может быть сквозной по всей работе, либо по главам с указанием номера главы, точки, номера формулы.

Пояснение значений, символов и числовых коэффициентов следует приводить непосредственно под формулой в той же последовательности, как и в формуле. Значение каждого символа и числового коэффициента следует давать с новой строки, первую строку пояснения начинают со слова «где» без двоеточия.

Правила оформления таблиц

Таблицу следует располагать в работе непосредственно после текста, в котором она упоминается впервые, или на следующей странице.

По содержанию таблицы делятся на аналитические и неаналитические. Аналитические таблицы являются результатом обработки и анализа цифровых показателей. Как правило, после таких таблиц делается обобщение, например, *"табл. 5 позволяет сделать вывод, что..."*, *"из табл. 2 видно, что..."* и т.п.

В неаналитических таблицах помещаются, как правило, необработанные статистические данные, необходимые лишь для информации или констатации.

Таблица должна иметь порядковый номер, который записывается после слова «Таблица», при этом таблица может иметь содержательный заголовок. Подпись к таблице располагается перед таблицей, с выравниванием по правому краю, например:

Таблица 5. Сравнение технологий тестирования.

Заголовок графы 1	Заголовок графы 2	...	Заголовок графы ...

Если в ВКР одна таблица, то слово "Таблица" не используется. В этом случае в тексте слово "таблица" необходимо писать без сокращения, например: *«как видно из таблицы...», «по результатам анализа (см. таблицу) видно, что...».*

Заголовки граф (столбцов) таблицы должны начинаться с прописных букв, подзаголовки со строчных, если последние подчиняются заголовку. Заголовки граф указываются в единственном числе.

При переносе таблицы на следующую страницу шапку таблицы следует повторить и над ней поместить слова "продолжение таблицы" с указанием номера таблицы. Допускается не повторять заголовок таблицы. В этом случае пронумеровывают графы и повторяют их нумерацию на следующей странице.

На все таблицы необходимы ссылки в тексте, например: *«данные анализа (табл. 5) показывают, что...»* или *«анализ (см. табл. 5) свидетельствует, что...»*

Все приводимые в таблицах данные должны быть достоверны, однородны и сопоставимы, в основе их группировки должны лежать существенные признаки.

Если в тексте приводятся ранее опубликованные таблицы, необходимо указать ссылку на источник.

Правила оформления иллюстраций

Основными видами иллюстраций в ВКР являются: диаграмма, технический рисунок, схема, фотография, график, скриншот.

Иллюстрации следует располагать в работе непосредственно после текста, в котором они упоминаются впервые, или на следующей странице, если в указанном месте они не помещаются.

Номер следует размещать под иллюстрацией посередине строки после слова «Рисунок». Иллюстрация может иметь наименование, которое располагается после номера, например, «Рисунок 5. Количество обращений пользователей с запросами».

На все иллюстрации должны быть даны ссылки в работе, например, «... в соответствии с рис. 2». Допустима также ссылка на иллюстрацию в скобках, например: (рис. 4).

Все иллюстрации в ВКР должны быть пронумерованы арабскими цифрами, двойной порядковой нумерацией в пределах одной главы работы. Так, например, во второй главе рисунки будут иметь номера 2.1, 2.2, ..., а в третьей – 3.1, 3.2,

Правила оформления ссылок на использованные источники

Библиографические ссылки употребляют [11]:

- * при цитировании;
- * при заимствовании положений, формул, таблиц, иллюстраций;
- * при необходимости отсылки к другому изданию, где более полно изложен вопрос;
- * при анализе в тексте опубликованных работ.

Для связи текста ВКР со списком использованных источников используют отсылки в виде заключенного в квадратные скобки порядкового номера источника в списке. Например, отсылка на несколько источников может иметь вид: [13-17, 19]. При использовании цитат, фактических, статистических и иных сведений в отсылке дополнительно указывается страница: [14, с. 7] или страницы: [19, с. 7-9]. При необходимости отсылки могут содержать имя автора (авторов), название документа, год издания, обозначение и номер тома, главы и т.п.

Если текст цитируется не по первоисточнику, а по другому документу, то в начале ссылки приводят слова: «Цит. по:» (цитируется по), «Приводится по:», с указанием источника заимствования.

Оформление списка использованных источников

Список использованных источников является обязательной составной частью ВКР. В список включаются библиографические сведения об использованных при подготовке работы источниках, а также о цитируемых в тексте документах и источниках фактических или статистических сведений. Список использованных источников (список литературы) следует формировать по мере упоминания, начиная обязательно с номера 1. При нумерации списка используется сплошная нумерация для всего текста документа в целом. При этом произведения одного автора расставляются в списке по алфавиту заглавий или по годам публикации, в прямом хронологическом порядке.

В ВКР список использованных источников размещается после текста работы до приложений.

Правила представления областей (элементов) библиографического описания, применение знаков предписанной пунктуации осуществляются в соответствии с ГОСТ 7.1 [12] и ГОСТ 7.82 [13].

Области библиографического описания:

- заголовок: может содержать имена авторов (от одного до трех), наименование страны и обозначение вида документа (заголовок формы).;
- основное заглавие документа: сведения, относящиеся к заглавию, например: Линейная алгебра : учеб. пособие;
- общее обозначение материала;
- сведения об ответственности: за косой чертой (/) повторяется фамилия первого автора и перечисляются фамилии второго и третьего авторов, инициалы ставятся впереди фамилии; например: / А. И. Абрамов, Е. В. Кораблев ; под ред. А. А. Ирвина или / В. Новак ; пер. с англ. Если авторов больше трех, то указываются инициалы и фамилия первого автора со словами [и др.] ;
- сведения об издании: например . - 2-е изд., испр., доп.;
- выходные данные: город, где издана книга : название издательства, год издания, например: . - М. : Физматлит, 2010;
- физическую характеристику документа - количество страниц в данном документе или на каких страницах расположена данная статья;
- сведения о серии (в круглых скобках);
- обозначение и порядковый номер тома или выпуска (для ссылок на публикации в многочастных или сериальных документах);

- примечания (количество иллюстраций, тираж и др.);
- Международный стандартный номер (ISBN, ISSN) и условий доступности.

Примеры оформления библиографических записей использованных источников приведены в Приложении Б.

Список использованных источников

1. Приказ Министерства образования и науки РФ от 12 марта 2015 г. N 229 ОБ УТВЕРЖДЕНИИ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 09.03.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ (УРОВЕНЬ БАКАЛАВРИАТА)
2. Положение о выпускной квалификационной работе студентов, обучающихся по программам бакалавриата, специалитета и магистратуры в КРСУ. Режим доступа: https://krsu.edu.kg/images/informacia/08_info/02_studentu/poloj_vypusk_kval_rabota.pdf
3. Рабочий учебный план по программе бакалавриата, направление 09.03.04 «Программная инженерия». Режим доступа: http://ict.krsu.edu.kg/images/UP_bak_2018.pdf
4. Рабочая программа преддипломной практики. Режим доступа: http://ict.krsu.edu.kg/images/PPr_RP_bak.pdf
5. ГОСТ 7.32-2001. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. - М.: Изд-во стандартов, 2001
6. Липаев В.В. Программная инженерия сложных заказных программных продуктов. Учебное пособие.- М.: МАКС- Пресс, 2014.- 312 с.
7. Липаев В.В. Документирование сложных программных средств. М.:Вузовское образование, 2015.-216 с.
8. Буч Г. Коналлен Д. Максимчук Р.А. Хьюстон К. Энгл М. Янг Б. Объектно-ориентированный анализ и проектирование с примерами приложений. М.: Вильямс, 2008.-720с.
9. Майерс Г. , Баджетт Т., Сандлер К. Искусство тестирования программ.– М.: Вильямс, 2013. -272с.
10. Орлов С.А. Программная инженерия. Учебник для вузов. 5-е издание обновленное и дополненное. Стандарт третьего поколения. – СПб.: Питер, 2016.-640с.
11. ГОСТ 7.0.5–2008. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая ссылка Общие требования и правила составления. М.– Стандартиформ, 2008.
12. ГОСТ 7.1-2003. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления. - М.: Изд-во стандартов, 2003.
13. ГОСТ 7.82-2001. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления. - М.: Изд-во стандартов, 2001.

Приложение А

Образцы содержаний спецификаций требований (SRS)

А.1. Образец Раздела 3 SRS, организованный по режимам: версия 1

- 3. Специфические требования
 - 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
 - 3.2. Функциональные требования
 - 3.2.1. Режим 1
 - 3.2.1.1. Функциональное требование 1.1
 - ...
 - 3.2.1.n. Функциональное требование 1.n
 - 3.2.2. Режим 2
 - ...
 - 3.2.m. Режим m
 - 3.2.m.1. Функциональное требование m.1
 - ...
 - 3.2.m.n. Функциональное требование m.n
 - 3.3. Требования к производительности
 - 3.4. Проектные ограничения
 - 3.5. Атрибуты программной системы
 - 3.6. Прочие требования

А.2. Образец Раздела 3 SRS, организованный по режимам: версия 2

- 3. Специфические требования
 - 3.1. Функциональные требования
 - 3.1.1. Режим 1
 - 3.1.1.1. Внешние интерфейсы
 - 3.1.1.1.1. Пользовательские интерфейсы
 - 3.1.1.1.2. Аппаратные интерфейсы

- 3.1.1.1.3. Программные интерфейсы
 - 3.1.1.1.4. Коммуникационные интерфейсы
 - 3.1.1.2. Функциональные требования
 - 3.1.1.2.1. Функциональное требование 1
 - ...
 - 3.1.1.2.n. Функциональное требование n
 - 3.1.2. Режим 2
 - ...
 - 3.1.m. Режим m
- 3.2. Проектные ограничения
- 3.3. Атрибуты программной системы
- 3.4. Прочие требования

А.3. Образец Раздела 3 SRS, организованный по классам пользователей

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Функциональные требования
 - 3.2.1. Класс пользователей 1
 - 3.2.1.1. Функциональное требование 1.1
 - ...
 - 3.2.1.n. Функциональное требование 1.n
 - 3.2.2. Класс пользователей 2
 - ...
 - 3.2.m. Класс пользователей m
 - 3.2.m.1. Функциональное требование m.1
 - ...
 - 3.2.m.n. Функциональное требование m.n
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы

- 3.6. Прочие требования

А.4. Образец Раздела 3 SRS, организованный по объектам

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Классы/объекты
 - 3.2.1. Класс/объект 1
 - 3.2.1.1. Атрибуты (непосредственные или унаследованные)
 - 3.2.1.1.1. Атрибут 1
 - ...
 - 3.2.1.1.n. Атрибут n
 - 3.2.1.2. Функции (сервисы и методы, непосредственные или унаследованные)
 - 3.2.1.2.1. Функциональное требование 1.1
 - ...
 - 3.2.1.2.m. Функциональное требование 1.m
 - 3.2.1.3. Сообщения (принимаемые или отправляемые)
 - 3.2.2. Класс/объект 2
 - ...
 - 3.2.p. Класс/объект p
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы
- 3.6. Прочие требования

А.5. Образец Раздела 3 SRS, организованный по Функциональным возможностям

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы

- 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Функциональные возможности системы
 - 3.2.1. Системная функция 1
 - 3.2.1.1. Введение (назначение функции)
 - 3.2.1.2. Последовательность «запрос/ответ»
 - 3.2.1.3. Связанные функциональные требования
 - 3.2.1.3.1. Функциональное требование 1
 - ...
 - 3.2.1.3.n. Функциональное требование n
 - 3.2.2. Системная функция 2
 - ...
 - 3.2.m. Системная функция m
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы
- 3.6. Прочие требования

А.6. Образец Раздела 3 SRS, организованный по внешним воздействиям

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Функциональные требования
 - 3.2.1. Воздействие 1
 - 3.2.1.1. Функциональное требование 1.1
 - ...
 - 3.2.1.n. Функциональное требование 1.n
 - 3.2.2. Воздействие 2
 - ...
 - 3.2.m. Воздействие m
 - 3.2.m.1. Функциональное требование m.1

- ...
 - 3.2.m.n. Функциональное требование m.n
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы
- 3.6. Прочие требования

А.7. Образец Раздела 3 SRS, организованный по функциональной иерархии

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Функциональные требования
 - 3.2.1. Информационные потоки
 - 3.2.1.1. Диаграмма потоков данных 1
 - 3.2.1.1.1. Сущности
 - 3.2.1.1.2. Процессы
 - 3.2.1.1.3. Топология
 - 3.2.1.2. Диаграмма потоков данных 2
 - 3.2.1.2.1. Сущности
 - 3.2.1.2.2. Процессы
 - 3.2.1.2.3. Топология
 - ...
 - ...
 - 3.2.1.n. Диаграмма потоков данных n
 - 3.2.1.n.1. Сущности
 - 3.2.1.n.2. Процессы
 - 3.2.1.n.3. Топология
 - 3.2.2. Описание процессов
 - 3.2.2.1. Процесс 1
 - 3.2.2.1.1. Входные данные
 - 3.2.2.1.2. Алгоритм или формула обработки

- 3.2.1.1.3. Результирующие данные
- 3.2.2.2. Процесс 2
 - 3.2.2.2.1. Входные данные
 - 3.2.2.2.2. Алгоритм или формула обработки
 - 3.2.2.2.3. Результирующие данные
- ...
- 3.2.2.m. Процесс m
 - 3.2.2.m.1. Входные данные
 - 3.2.2.m.2. Алгоритм или формула обработки
 - 3.2.2.m.3. Результирующие данные
- 3.2.3. Спецификации структур данных
 - 3.2.3.1. Структура 1
 - 3.2.3.1.1. Тип записи
 - 3.2.1.1.2. Поля записи
 - 3.2.3.2. Структура 2
 - 3.2.3.2.1. Тип записи
 - 3.2.1.2.2. Поля записи
 - ...
 - 3.2.3.p. Структура p
 - 3.2.3.p.1. Тип записи
 - 3.2.1.p.2. Поля записи
- 3.2.4. Словарь данных
 - 3.2.4.1. Элемент данных 1
 - 3.2.4.1.1. Имя
 - 3.2.4.1.2. Представление
 - 3.2.4.1.3. Единицы/Форматы
 - 3.2.4.1.4. Точность
 - 3.2.4.1.5. Диапазон
 - 3.2.4.2. Элемент данных 2
 - 3.2.4.2.1. Имя
 - 3.2.4.2.2. Представление
 - 3.2.4.2.3. Единицы/Форматы
 - 3.2.4.2.4. Точность
 - 3.2.4.2.5. Диапазон

- ...
- 3.2.4.q. Элемент данных q
 - 3.2.4.q.1. Имя
 - 3.2.4.q.2. Представление
 - 3.2.4.q.3. Единицы/Форматы
 - 3.2.4.q.4. Точность
 - 3.2.4.q.5. Диапазон
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы
- 3.6. Прочие требования

А.8. Образец Раздела 3 SRS, демонстрирующий множественную организацию

3. Специфические требования

- 3.1. Внешние требования к интерфейсу
 - 3.1.1. Пользовательские интерфейсы
 - 3.1.2. Аппаратные интерфейсы
 - 3.1.3. Программные интерфейсы
 - 3.1.4. Коммуникационные интерфейсы
- 3.2. Функциональные требования
 - 3.2.1. Класс пользователей 1
 - 3.2.1.1. Функция 1.1
 - 3.2.1.1.1. Введение (назначение функции)
 - 3.2.1.1.2. Последовательность «запрос/ответ»
 - 3.2.1.1.3. Связанные функциональные требования.
 - 3.2.1.2. Функция 1.2
 - 3.2.1.2.1. Введение (назначение функции)
 - 3.2.1.2.2. Последовательность «запрос/ответ»
 - 3.2.1.2.3. Связанные функциональные требования.
 - ...
 - 3.2.1.m. Функция 1.m
 - 3.2.1.m.1. Введение (назначение функции)
 - 3.2.1.m.2. Последовательность «запрос/ответ»
 - 3.2.1.m.3. Связанные функциональные требования.

- 3.2.2. Класс пользователей 2
- ...
- 3.2.n. Класс пользователей n
- 3.3. Требования к производительности
- 3.4. Проектные ограничения
- 3.5. Атрибуты программной системы
- 3.6. Прочие требования

Приложение Б

Примеры библиографических записей

Книги

Однотомное издание

Автор. Заглавие: сведения, относящиеся к заглавию (см. на титуле) / сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Сведения об издании (информация о переиздании, номер издания). - Место издания: Издательство, Год издания. - Объем. - (Серия).

При описании необходимо учитывать следующее:

1. Если у издания один автор, то описание начинается с фамилии и инициалов автора. Далее через точку "." пишется Заглавие. За косой чертой "/" после заглавия Имя автора повторяется, как сведение об ответственности.

2. Если у издания два или три автора, то описание начинается с фамилии и инициалов первого автора. За косой чертой "/" после заглавия сначала указывается первый автор, а потом через запятую - второй (и третий) авторы.

3. Если у издания четыре автора и более, то описание начинается с заглавия. За косой чертой указываются все авторы.

4. Если у издания есть один или несколько авторов, и также указаны редакторы, составители, переводчики и т.п., то информация о них указывается в сведении об ответственности, после всех авторов перед точкой с запятой ";".

5. Если у издания нет автора, но указаны редакторы, составители, переводчики и т.п., то описание начинается с заглавия. За косой чертой после заглавия сразу пишутся редакторы, составители и т.п. с указанием функции.

6. Если у издания нет автора, редакторов и т.п., то после заглавия сразу идет информация об издании после точки и тире ". - ".

Примеры:

Лейбин В.М. Психоанализ: учебник/ В.М.Лейбин. - СПб. : Питер, 2002. -576 с. - (Учебник нового века).

С.Л. Марпл-мл. Цифровой спектральный анализ и его приложения/ С.Л. Марпл-мл.; пер. с англ. О.И.Хабарова и Г.А. Сидоровой ред. И.С. Рыжак. - М. : Мир, 2001. - 584 с.

Современная информатика: наука, технология, деятельность / Р.С. Гиляревский, Г.З. Залаев, И.И. Родионов, В.А. Цветкова; под ред. Ю.М. Арского. - М.: Информатика, 1997. - 211 с.

Логопедия: учебник для студ. дефектолог. фак. пед. вузов/ ред. Л.С. Волкова, С.Н. Шаховская. - 3-е изд., перераб. и доп. - М. : Гуманит. изд. центр ВЛАДОС, 2002. - 680 с. - (Коррекционная педагогика).

Странный ветер: современная японская поэзия: антология совр. яп. лит.: пер. с яп. - М.: Иностранка, 2003. - 476 с. - (Антология современной японской литературы).

Иллюстрированный словарь английского и русского языка с указателями.
- М. : Живой язык, 2003. - 1000 с.

Система стандартов безопасности труда: сборник. – М.: Изд-во стандартов, 2002. – 102 с.: ил. – (Межгосударственные стандарты).

Многотомные издания Многотомное издание в целом

Автор. Заглавие издания: сведения, относящиеся к заглавию (см. на титуле) / Сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Город издания: Издательство, Год начала издания - год окончания издания. - (Серия). Обозначение и номер тома: Заглавие тома: сведения, относящиеся к заглавию. - Год издания тома. - Объем. Обозначение и номер тома: Заглавие тома: сведения, относящиеся к заглавию. - Год издания тома. - Объем. и т.д.

или:

Автор. Заглавие издания: сведения, относящиеся к заглавию (см. на титуле) / Сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Город издания: Издательство, Год начала издания - год окончания издания. - Кол-во томов. -(Серия).

При описании необходимо учитывать, что в качестве многотомного рассматривают документ, состоящий из заранее определенного количества томов (частей), представляющий собой единое целое по содержанию и оформлению.

Примеры:

Гиппиус, З. Н. Сочинения: в 2 т. / Зинаида Гиппиус. – М.: Лаком-книга, 2001. – (Золотая проза серебряного века).
Т. 1 : Романы. – 367 с. – Содерж.: Без талисмана; Победители; Сумерки духа. Т. 2 : Романы. – 415 с. – Содерж.: Чертова кукла; Жизнеописание в 33 гл.; Роман-царевич: история одного начинания; Чужая любовь.

или

Гиппиус, З. Н. Сочинения: в 2 т. / Зинаида Гиппиус. – М.: Лаком-книга, 2001. – 2 т. – (Золотая проза серебряного века).

Отдельный том многотомного издания

Автор. Заглавие издания: сведения, относящиеся к заглавию (см. на титуле). Обозначение и номер тома: Заглавие тома / Сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Город издания: Издательство, Год

издания - Объем. - (Серия).

или

Автор. Заглавие тома / Сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Город издания: Издательство, Год издания - Объем. - (Заглавие издания: сведения относящиеся к заглавию / Сведения об ответственности (авторы); обозначение и номер тома).

При описании необходимо учитывать, что под томом понимается отдельная физическая единица, входящая в состав многочастного документа, обозначаемая как том, часть, выпуск, сборник, альбом, тетрадь и т. п.

Примеры:

Фихтенгольц Г.М. Курс дифференциального и интегрального исчисления: в 3 т. Т. 3./ Г.М. Фихтенгольц. – М.: Наука, 2002. – 656 с.: ил.

или

Фихтенгольц Г.М. Курс дифференциального и интегрального исчисления / Г.М. Фихтенгольц. – М.: М.: Наука, 2002. – 656 с.: ил. АСТ: Астрель, 2002. – 503 с.: ил.: в 3 т. / Фихтенгольц Г.М.: т. 3).

Неопубликованные документы

Диссертации

Автор. Заглавие: сведения, относящиеся к заглавию (см. на титуле): шифр номенклатуры специальностей научных работников: дата защиты: дата утверждения / сведения об ответственности (автор); последующие сведения об ответственности (коллектив). - Место написания, Дата написания. - Объем.

При описании необходимо учитывать, что в сведениях, относящихся к заглавию, приводят сведения о том, что данная работа представлена в качестве диссертации, а также сведения об ученой степени, на соискание которой представлена диссертация. Сведения приводят в сокращенном виде, например: дис. ... канд. пед. наук или дис. ... д-ра техн. наук

Примеры:

Белозеров, И. В. Религиозная политика Золотой Орды на Руси в XIII–XIV вв.: дис. ... канд. ист. наук : 07.00.02: защищена 22.01.02: утв. 15.07.02 / Белозеров И.В. – М., 2002. – 215 с.

Тевлина, В.В. Исторический опыт подготовки специалистов в области социальной работы в России. Вторая половина XIX-XX вв.: дис. ... д-ра ист. наук: 07.00.02: защищена 04.10.04/ В.В. Тевлина; Поморский государственный университет им. М.В. Ломоносова. - Архангельск, 2004. - 409 с.

Автореферат диссертации

Автор. Заглавие: сведения, относящиеся к заглавию (см. на титуле): шифр номенклатуры специальностей научных работников: дата защиты: дата утверждения / сведения об

ответственности (автор); последующие сведения об ответственности (коллектив). - Место написания, Дата написания. - Объем.

При описании необходимо учитывать, что в сведениях, относящихся к заглавию, приводят сведения о том, что данная работа представлена в качестве автореферата диссертации на соискание ученой степени. Сведения приводят в сокращенном виде, например:

автореф. дис. ... канд. физ.-мат. наук

автореф. дис. ... д-ра пед. наук

Примеры:

Бесстрашнова, Я. К. Модели и методы управления надежностью коммерческого кредитования: автореф. дис. ... канд. экон. наук: 08.00.13: защищена 30.10.03 / Я.К. Бесстрашнова; СПбГПУ . — СПб., 2003. - 20 с.

Столярова, Е.В. Коммуникативная направленность текстов политической рекламы: автореф. дис. ... канд. филолог. наук: 10.02.01: защищена 16.12.05/ Е.В. Столярова; Поморский государственный университет им. М.В. Ломоносова. - Архангельск, 2005. - 22 с.

Электронные ресурсы

Существует два вида электронных ресурсов – локального и удаленного доступа. При их описании необходимо учитывать следующее:

1. Описание электронного ресурса в области "Автор" и "Сведения об ответственности" осуществляется по правилам описания книжного издания.

2. Обозначение материала приводят сразу после заглавия в квадратных скобках: [Электронный ресурс].

Электронный ресурс локального доступа

Автор. Заглавие [Электронный ресурс] : сведения, относящиеся к заглавию / сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). — Обозначение вида ресурса ("электрон. дан." и/или "электрон. прогр"). — Место издания: Издательство, Год издания. — Обозначение материала и количество физических единиц. — (Серия).

Примеры:

Даль, Владимир Иванович. Толковый словарь живого великорусского языка Владимира Даля [Электронный ресурс] : подгот. по 2-му печ. изд. 1880—

1882 гг. — Электрон. дан. — М. : АСТ, 1998. — 1 электрон. опт. диск (CD-ROM)

— (Электронная книга).

Атлас-98 [Электронный ресурс] : 3D : самый подроб. полностью трехмер. атлас мира. — Электрон. дан. и прогр. — [Б. м.], 1998. — 1 электрон, опт. диск (CD-ROM). — (Весь мир в 3D).

Электронный ресурс удаленного доступа

Автор. Заглавие [Электронный ресурс] : сведения, относящиеся к заглавию / сведения об ответственности (авторы); последующие сведения об ответственности (редакторы, переводчики, коллективы). — Обозначение вида ресурса ("электрон. текст. дан."). — Место издания: Издательство, Дата издания.

— Режим доступа: URL. Дата обращения - Примечания ("Электрон. версия печ. публикации").

Допускается вместо слов «Режим доступа» писать «URL»

Примеры:

Исследовано в России [Электронный ресурс] : многопредмет. науч. журн. / Моск. физ.-техн. ин-т. — Электрон. журн. — Долгопрудный : МФТИ, 1998 — .

— Режим доступа к журн.: <http://zhurnal.mipt.rssi.ru>. (дата обращения: 15.03.12)

Российская государственная библиотека [Электронный ресурс] / Центр информ. технологий РГБ ; ред. Власенко Т.В. ; Web-мастер Козлова Н.В. — Электрон. дан. — М. : Рос. гос. б-ка, 1997— . Режим доступа: <http://www.rsl.ru>, свободный. (дата обращения: 15.04.12)

История книги [Электронный ресурс]: учебник для вузов / ред. А.А.Говоров, Т.Г.Куприянова; Московский гос. ун-т печати, Лаборатория компьютеризации ФИДиКТ. - Электрон. текстовые дан. и граф. дан. - М.: Издательство МГУП "Мир книги", 1998. - 348 с.: цв. - Режим доступа: <http://www.hi-edu.ru/e-books/HB/>, свободный. - Электрон. версия печ. публикации. (дата обращения: 25.02.12)

Составные части документов

Общая схема описания:

Сведения о статье // Сведения об источнике статьи. - Сведения о местоположении статьи в документе.

Статья из книги

Автор. Заглавие статьи: сведения, относящиеся к заглавию / сведения об ответственности (авторы статьи) // Заглавие книги: сведения, относящиеся к заглавию / сведения об ответственности (авторы книги); последующие сведения об ответственности (редакторы, переводчики, коллективы). - Место издания: Издательство, Год издания. - Местоположение статьи (страницы).

При описании необходимо учитывать, что сведения об издательстве в области выходных данных книги можно опустить.

Примеры:

Двинаянинова, Г. С. Комплимент: коммуникативный статус или стратегия в дискурсе / Г. С. Двинаянинова // Социальная власть языка : сб. науч. тр. / Воронеж. межрегион. ин-т обществ. наук. – Воронеж, 2001. – С. 101–106.

Майо-Знак, Э.О. Статистика печати / Э.О. Майо-Знак // Книговедение : энцикл. словарь. - М. : Сов. энцикл., 1982. - С. 517-519.

Воронцова, М.В. Федеральный и региональный компоненты содержания экологического образования/ М. В. Воронцова //XV международные Ломоносовские чтения: сб. научных трудов / ПГУ. - Архангельск: Изд-во ПГУ,

Статья из журнала

Автор. Заглавие статьи: сведения, относящиеся к заглавию / сведения об ответственности (авторы статьи) // Название журнала. - Год выпуска. - Номер выпуска. - Местоположение статьи (страницы).

При описании необходимо учитывать, что если статья размещены в двух и более журналах, то сведения о ее местоположении в каждом из номеров отделяют точкой с запятой.

Примеры:

Ладынин, И. А. Дафны в библейской и египетско-христианской традиции о финале царствования Априя: конец 570-х - начало 560-х годов до н. э./ И. А. Ладынин //Вестник древней истории. - 2004. - N 3. - С. 3-13.

Казаков, Н. А. Запоздалое признание / Н.Казаков // На боевом посту. – 2000. – № 9. – С. 64–76; № 10. – С. 58–71.

Статья из продолжающихся изданий

Автор. Заглавие статьи: сведения, относящиеся к заглавию / сведения об ответственности (авторы статьи) // Заглавие издания. Название серии - Год издания. - Номер выпуска: Заглавие выпуска. - Местоположение статьи (страницы).

Примеры:

Боголюбов, А. Н. О вещественных резонансах в волноводе с неоднородным заполнением / А. Н. Боголюбов, А. Л. Делицын, М. Д. Малых // Вестник Моск. ун-та. Сер. 3, Физика. Астрономия. – 2001. – № 5. – С. 23–25.

Белова, Г. Д. Некоторые вопросы уголовной ответственности за нарушение налогового законодательства / Г. Д. Белова // Актуал. проблемы прокурор. надзора / Ин-т повышения квалификации рук. кадров Генер. прокуратуры Рос. Федерации. – 2001. – Вып. 5 : Прокурорский надзор за исполнением уголовного и уголовно-процессуального законодательства. Организация деятельности прокуратуры. – С. 46–49.

Бебякова, Н.А. Особенности физического развития и минерального обмена школьников-пользователей персонального компьютера/ Н. А. Бебякова, Т. П. Экономова

//Вестник Поморского университета. Сер. "Физиологические и психолого-педагогические науки". - 2005. - № 1(7). - С. 29-33. - (Педагогика и психология).

ВНИМАНИЕ!!! Список источников не должен содержать никаких подзаголовков (разделов), например, «Статья из журнала», «Электронный ресурс удаленного доступа» и т.п.

Приложение В
Пример титульного листа ВКР

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ
Государственное образовательное учреждение высшего профессионально образования
КЫРГЫЗСКО-РОССИЙСКИЙ СЛАВЯНСКИЙ УНИВЕРСИТЕТ

Естественно-технический факультет
Кафедра информационных и вычислительных технологий

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему:

Разработка программных средств мониторинга геофизических параметров окружающей среды

Выполнил студент группы ЕПИ-1-15 _____ Иванов И.И.
(подпись)

Научный руководитель ВКР,
уч. степень, уч. звание, должность _____ Петров П.П.
(подпись)

«К защите» Заведующий
кафедрой ИВТ
_____ Десятков Г.А.
« ____ » _____ 2018г.

Бишкек 2018г.

Приложение Г
Форма заявления на утверждение темы ВКР

Утверждаю

Заведующему кафедрой ИВТ

Зав. кафедрой _____
(подпись)

Г.А.Десяткову
от студента группы

« ____ » _____ 201__ г.

ЗАЯВЛЕНИЕ

Прошу утвердить мне тему выпускной квалификационной работы бакалавра «Разработка программных средств мониторинга геофизических параметров окружающей среды».

Подпись студента

Дата

Научный руководитель:

Должность, уч. степень и звание

Фамилия И.О.

Подпись

Дата

Утверждено приказом КРСУ № _____ от _____