

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агабекян Раиса Левонновна

Должность: ректор

Дата подписания: 17.12.2021 15:24:58

Уникальный программный ключ:

4237c7ccb9b9e111bbaf1f4fcd9201d015c4dbaa123ff774747307b9b9fbche

**Негосударственное аккредитованное некоммерческое частное  
образовательное учреждение высшего образования  
«Академия маркетинга и социально-информационных технологий –**

**ИМСИТ»**

**(г. Краснодар)**

**Академический колледж**

**УТВЕРЖАЮ**

Председатель НМС, проректор

По учебной работе, профессор

Н. Н. Павелко

13 апреля 2020 г.



**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

по подготовке и защите дипломных работ

основной профессиональной образовательной программы

по специальности 09.02.03 Программирование в компьютерных системах

**Краснодар, 2020**

Разработчик: Большакова М. В., преподаватель первой категории академического колледжа

Методические указания по подготовке и защите дипломных работ разработаны на основе основной профессиональной образовательной программ среднего профессионального образования программы подготовки специалистов среднего звена, в соответствии с федеральным государственным образовательным по специальности 09.02.03 Программирование в компьютерных системах, утвержденным Приказом Министерства образования и науки РФ от 28.07.2014 г. №804 (Зарегистрировано в Минюсте России 21.08.2014 г. № 33733), Федеральным законом Российской Федерации от 29.12.12 г. № 273- ФЗ Образовании в Российской Федерации (ред. 25.12.2018г.), Положением о подготовке и защите дипломных работ обучающихся, осваивающих основные профессиональные образовательные программы СПО в Академии маркетинга и социально-информационных технологий- ИМСИТ (г Краснодар). Содержат: общие положения, структуру, требования к оформлению, рекомендации по проведению анализа объекта исследования, список рекомендуемых источников.

Методические указания одобрены на заседании Предметно-цикловой комиссией технического профиля Протокол № 9 от 13 апреля 2020 г.

Председатель ПЦК

М. В. Большакова

Методические указания утверждены на заседании Научно-методического совета Академии. Протокол № 8 от 13 апреля 2020г.

Председатель НМС, проректор по учебной работе, профессор

Н.Н. Павелко

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1 Общие положения.....	5
1.1 Общие требования к дипломной работе.....	5
2 Этапы выполнения дипломной работы.....	10
2.1 Выбор темы дипломной работы и назначения научного руководителя....	11
2.2 Разработка задания и календарного плана – графика выполнения дипломной работы.....	14
2.3 Изучение теоретических аспектов темы работы.....	16
2.4 Написание и оформление дипломной работы.....	18
2.5 Технические требования к оформлению дипломной работы .....	58
3 Подготовка и защита дипломной работы.....	64
3.1 Получение отзыва и рецензия на дипломную работу.....	64
3.1.1 Отзыв руководителя.....	64
3.1.2 Рецензирование дипломной работы.....	64
3.2 Прохождение предварительной защиты дипломной работы.....	65
3.3 Подготовка и защита дипломной работы.....	66
3.3.1 Подготовка дипломной работы.....	66
3.3.2 Защита дипломной работы.....	68
3.3.3 Критерии оценки дипломных работ.....	70
4 Учебно-методическое обеспечение дипломной работы.....	73
ПРИЛОЖЕНИЯ.....	75

## ВВЕДЕНИЕ

Государственная итоговая аттестация выпускников является обязательной и осуществляется после освоения программы СПО по специальности 09.02.03 Программирование в компьютерных системах в полном объеме.

Государственная итоговая аттестация включает подготовку и защиту дипломной работы для выпускников, осваивающих программы подготовки специалистов среднего звена.

Подготовка дипломной работы направлена на развитие и закрепление у обучающихся навыков творческого и всестороннего анализа научной, методической и специальной литературы по выбранной проблематике; выработку умений грамотно и убедительно излагать материал, четко формулировать теоретические обобщения, выводы и рекомендации.

В процессе выполнения дипломной работы обучающийся должен не только отразить теоретические знания по дисциплинам, которые формируют его специальность, но и исследовательскую работу в области изучения новейших аппаратных устройств и программных комплексов для организаций.

В результате успешной защиты дипломной работы Государственной экзаменационной комиссией принимается решение о присвоении выпускнику квалификации. На основании этого решения выдается документ о среднем профессиональном образовании Государственного образца по соответствующей специальности.

Методические указания разработаны в соответствии с Государственным образовательным стандартом среднего профессионального образования по специальности 09.02.03 Программирование в компьютерных системах, утвержденным приказом Минобрнауки России от 28 июля 2014 г. № 804 (ред. от 21.10.2019 г.).

В процессе их создания учитывались положения:

ГОСТ 2.105-95 ЕСКД. Общие требования к текстовым документам;

ГОСТ 2.301-68 ЕСКД. Форматы;

ГОСТ 7.0.5-2008 СИБИД. Библиографическая ссылка. Общие требования ;

ГОСТ 7.9-95 СИБИД. Реферат и аннотация. Общие требования;

ГОСТ 7.32-2017 СИБИД. Отчет о научно-исследовательской работе.

Структура и правила оформления (с поправкой);

ГОСТ 7.80-2000 СИБИД. Библиографическая запись. Заголовок. Общие требования и правила составления;

ГОСТ 7.82-2001 СИБИД. Библиографическая запись.

Библиографическое описание электронных ресурсов. Общие требования и правила составления;

ГОСТ 8.417-2002 ГСИ. Единицы величин.

Методическими указаниями закреплена система контроля графика выполнения работы и консультаций, обучающихся на всех этапах их работы над избранной темой со стороны руководителей дипломных работ и выпускающего инженерно - информационного отделения академического колледжа Академии ИМСИТ.

В методических рекомендациях, отражены: цель и задачи дипломной работы, ее тематика, требования к структуре и содержанию, состав и последовательность работ по оформлению и защите, формируемые компетенции.

# **1 Общие положения**

## **1.1 Общие требования к дипломной работе**

Дипломная работа является видом итоговой аттестации обучающихся и выполняется на заключительном этапе обучения: последний год обучения в Академическом колледже в соответствии с учебным планом.

Целью подготовки дипломной работы является систематизация и углубление теоретических и практических знаний, полученных в рамках учебного плана, закрепление навыков самостоятельной исследовательской работы. Работа должна свидетельствовать о степени владения обучающимися профессиональными компетенциями и готовности к практической деятельности. По результатам защиты дипломной работы, Государственная экзаменационная комиссия решает вопрос о присвоении выпускнику квалификации Техник – программист по специальности 09.02.03 Программирование в компьютерных системах.

Дипломная работа направлена на решение следующих задач:

- систематизацию, закрепление и расширение теоретических и практических знаний, полученных в рамках учебного плана и применение этих знаний при решении конкретных научных и производственных задач;
- развитие навыков самостоятельной работы и применения методик исследования и экспериментирования при решении разрабатываемых в выпускной работе проблем и вопросов;
- выяснение подготовленности обучающихся к практической деятельности в различных областях программирования.

Дипломная работа выполняется на основе глубокого изучения нормативно-правовой база, литературы, периодических изданий, Интернет-ресурсов и может основываться на обобщении выполненных курсовых работ, если они выполнялись в рамках соответствующего профессионального модуля.

Область профессиональной деятельности выпускников: совокупность методов и средств для разработки, сопровождения и эксплуатации программного обеспечения компьютерных систем.

Объектами профессиональной деятельности выпускников являются:

- компьютерные системы;
- автоматизированные системы обработки информации и управления;
- программное обеспечение компьютерных систем (программы, программные комплексы и системы);
- математическое, информационное, техническое, эргономическое, организационное и правовое обеспечение компьютерных систем;
- первичные трудовые коллективы.

Техник-программист готовится к следующим видам деятельности:

- Разработка программных модулей программного обеспечения для компьютерных систем.
- Разработка и администрирование баз данных.
- Участие в интеграции программных модулей.
- Выполнение работ по профессии «Оператор электронно-вычислительных и вычислительных машин».

Техник-программист должен обладать общими компетенциями, включающими в себя способность:

ОК-1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК-2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК-3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК-4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального личностного развития.

ОК-5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК-6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.

ОК-7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК-8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК-9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

**Техник-программист должен обладать профессиональными компетенциями, соответствующими основным видам профессиональной деятельности:**

1. Разработка программных модулей программного обеспечения для компьютерных систем.

ПК 1.1. Выполнять разработку спецификаций отдельных компонент.

ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять оптимизацию программного кода модуля.

ПК 1.6. Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

2. Разработка и администрирование баз данных.

ПК 2.1. Разрабатывать объекты базы данных.

ПК 2.2. Реализовывать базу данных в конкретной системе управления базами данных (СУБД).

ПК 2.3. Решать вопросы администрирования базы данных.

ПК 2.4. Реализовывать методы и технологии защиты информации в базах данных.

3. Участие в интеграции программных модулей.

ПК 3.1. Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.

ПК 3.2. Выполнять интеграцию модулей в программную систему.

ПК 3.3. Выполнять отладку программного продукта с использованием специализированных программных средств.

ПК 3.4. Осуществлять разработку тестовых наборов и тестовых сценариев.

ПК 3.5. Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования.

ПК 3.6. Разрабатывать технологическую документацию.

4. Выполнение работ по одной или нескольким профессиям рабочих, должностям служащих.

Государственная итоговая аттестация представляет собой форму оценки степени и уровня освоения обучающимися основной образовательной программы.

Государственная итоговая аттестация проводится на основе принципов объективности и независимости оценки качества подготовки обучающихся.

Государственная итоговая аттестация, завершающая освоение основных профессиональных образовательных программ, является обязательной и проводится в порядке и в форме, которые установлены образовательной организацией, если иное не установлено Федеральным законом. Федеральным государственным образовательным стандартом по специальности 09.02.03 Программирование в компьютерных системах, утвержденным Министерством образования и науки РФ 28 июля 2014г. № 804 предусмотрено выполнение дипломной работы.

Необходимым условием допуска к государственной итоговой аттестации является представление документов, подтверждающих освоение обучающимся компетенций при изучении теоретического материала и прохождении практики по каждому из основных видов профессиональной деятельности. В том числе выпускником могут быть предоставлены отчеты о ранее достигнутых результатах, дополнительные сертификаты, свидетельства



(дипломы) олимпиад, конкурсов, творческие работы по специальности, характеристики с мест прохождения преддипломной практики.

Подготовка и защита дипломной работы являются проверкой качества полученных студентом знаний и умений, практического опыта, сформированности общих и профессиональных компетенций, позволяющих решать профессиональные задачи.

Результаты защиты дипломной работы являются основанием для принятия Государственной экзаменационной комиссией решения о присвоении соответствующей квалификации и выдаче диплома государственного образца.

На выполнение дипломной работы по ФГОС СПО отводится 4 недели, на государственную итоговую аттестацию отводится 2 недели.

Итоговая государственная аттестация проводится согласно утвержденному графику. Государственная итоговая аттестация включает подготовку и защиту дипломной работы. Обязательное требование - соответствие тематики дипломной работы содержанию одного или нескольких профессиональных модулей.

Темы дипломных работ определяются образовательной организацией. Студенту предоставляется право выбора темы дипломной работы, в том числе предложения своей тематики с необходимым обоснованием целесообразности ее разработки для практического применения. При этом тематика дипломной работы должна соответствовать содержанию одного или нескольких профессиональных модулей, входящих в образовательную программу среднего профессионального образования.

Для подготовки дипломной работы студенту назначается руководитель и, при необходимости, консультанты.

Закрепление за студентами тем дипломных работ, назначение руководителей и консультантов осуществляется распорядительным актом образовательной организации.

Тема дипломной работы (ДР) должна иметь практико-ориентированный характер. Перечень примерных тем ДР разрабатывается преподавателями междисциплинарных курсов в рамках профессиональных модулей, рассматривается соответствующими предметными (цикловыми) комиссиями и утверждается НМС Академии ИМСИТ после предварительного положительного заключения работодателей. Тема ДР должна быть предварительно согласована с работодателем.

При определении темы ДР следует учитывать, что содержание ДР может основываться:

- на обобщении выполненной ранее студентом курсовой работы, если она выполнялась в рамках соответствующего профессионального модуля;

– на использовании результатов выполненных компетентностно-ориентированных заданий при подготовке к экзамену по соответствующему профессиональному модулю.

Выбор темы ДР студентом осуществляется на 4 курсе 7 семестре до начала производственной практики, так как при ее прохождении студент должен собрать практический материал для выполнения дипломной работы.

## **2 Этапы выполнения дипломной работы**

Выполнение дипломной работы (ДР) по специальности 09.02.03 Программирование в компьютерных системах осуществляется в несколько этапов:

- выбор темы и назначение научного руководителя;
- разработка задания и календарного плана выполнения дипломной работы;
- составление плана дипломной работы;
- подбор и изучение литературы;
- написание и оформление дипломной работы;
- представление дипломной работы научному руководителю, получение отзыва и устранение указанных в нем замечаний;
- представление дипломной работы на рецензирование;
- защита дипломной работы.

В целях оказания дипломнику теоретической и практической помощи в период подготовки и написания ДР, приказом ректора Академии ИМСИТ каждому студенту назначается руководитель.

К руководству ДР привлекаются высококвалифицированные специалисты учреждений и предприятий в области, касающейся тематики дипломной работы.

Уточнение и изменение (корректировка) темы дипломной работы после подписания приказа производится только в порядке исключения по письменному заявлению студента на имя ректора и утверждается приказом по академии.

К каждому руководителю может быть одновременно прикреплено не более 10 студентов.

В обязанности руководителя ДР входит:

- разработка совместно со студентами плана ДР. План дипломной работы должен быть тщательно продуман и обоснован. Он включает следующие разделы: Введение, Основную часть (состоящую, как правило, из четырёх глав, каждая из которых – из двух-пяти параграфов), Заключение, Список использованных источников и Приложения;
- разработка задания на подготовку ДР с указанием сроков начала и окончания работы.

В нем указываются: тема дипломной работы, краткая характеристика основного содержания работы;

- оказание помощи студенту в разработке индивидуального графика работы на весь период выполнения ДР;
- консультирование студента по вопросам содержания и последовательности выполнения ДР;

- оказание помощи студенту в подборе необходимых источников;
- контроль хода выполнения ДР в соответствии с установленным графиком в форме регулярного обсуждения руководителем и студентом хода работ;
- оказание помощи (консультирование студента) в подготовке презентации и доклада для защиты ДР;
- предоставление письменного отзыва на ДР.

По утвержденным темам руководители дипломных работ разрабатывают индивидуальные задания для каждого студента.

Задания на дипломную работу рассматриваются предметно-цикловыми комиссиями, подписываются руководителем работы и утверждаются деканом факультета.

В отдельных случаях допускается выполнение ДР группой студентов. При этом индивидуальные задания выдаются каждому студенту.

Задание на ДР выдается студенту не позднее, чем за две недели до начала производственной практики (преддипломной).

Задания на дипломную работу сопровождаются консультацией, в ходе которой разъясняются назначение и задачи, структура и объем работы, принципы разработки и оформления, примерное распределение времени на выполнение отдельных частей дипломной работы.

Общее руководство и контроль над ходом выполнения дипломных работ осуществляют заместитель директора по учебной работе, председатель предметно-цикловой комиссии в соответствии с должностными обязанностями.

Детально содержание каждого этапа будет рассмотрено ниже.

## **2.1 Выбор темы дипломной работы и назначение руководителя**

Первый этап выполнения дипломной работы заключается в выборе обучающимся темы ДР.

Типовая тематика ДР подготавливается выпускающим Инженерно-информационным отделением заблаговременно. Тематика разрабатывается ведущими преподавателями профиля с учетом уровня развития теории в области программирования и его отдельных направлений, а также заявок предприятий (фирм), органов государственной власти и местного самоуправления и ежегодно корректируется, утверждается на заседании предметно-цикловой комиссии. Тематика дипломных работ отражает основные сферы и направления деятельности специалистов в области финансов на микро- и мезо- уровнях.

Дипломные работы носят квалификационный и аттестационный характер, в связи с чем, тематика работ должна:

- отвечать требованиям актуальности исследования;
- обеспечивать самостоятельность выполнения работы;
- предусматривать необходимость критической проработки достаточно большого объема научной литературы;
- обеспечивать обязательность использования при подготовке работы знаний, приобретенных при изучении профессиональных дисциплин;
- иметь научную и практическую значимость.

Качество дипломной работы в значительной степени зависит от выбранной темы.

Тема дипломной работы выбирается обучающимся самостоятельно. Выбор темы определяется интересами, склонностями обучающимся и возможностью его трудоустройства, потребностью развития и совершенствования самого объекта исследования, научной специализацией профиля и ее преподавателей.

При выборе темы дипломной работы следует руководствоваться актуальностью проблемы, возможностью получения фактических данных по теме дипломной работы, наличием специальной научной литературы, практической значимостью для конкретного предприятия, организации, учреждения, что может подтверждаться заказом организации на разработку дипломной работы, вместе с тем выбор темы необходимо согласовать с руководителем.

Таким образом, выбранная тема дипломной работы должна быть актуальной, отличаться научной новизной и иметь практическую значимость.

Обучающийся, выбрав тему дипломной работы, представляют председателю ПЦК заполненное **заявление** по форме, представленной в приложении.

Если тема дипломной работы не выбрана, она назначается обучающимся заведующим отделения в соответствии с тематикой.

На основании заявления выбранная тема и руководитель дипломной работы закрепляется приказом ректора Академии. Руководитель назначается из числа преподавателей отделения колледжа.

Важно иметь в виду, что руководитель не является ни соавтором, ни редактором ДР, и Обучающийся не должен рассчитывать на то, что руководитель обязан исправлять имеющиеся в дипломной работы работе орфографические, стилистические и иные ошибки.

В соответствии с заданием законченная и оформленная работа с отзывами руководителя, а так же рецензией представляется выпускником в деканат за 3-5 дней до защиты, в случае успешного прохождения предзащиты.

**Руководитель дипломной работы:**

- оказывает практическую помощь студенту в выборе темы дипломной работы и устанавливает календарный план-график ее выполнения (Приложение);

- выдает задание на ДР, которым Обучающийся должен руководствоваться в процессе ее выполнения. Данный документ подписывается руководителем, председателем ПЦК и принимается для исполнения (Приложение);

- оказывает помощь в выборе методики проведения исследования;

- дает консультации по подбору литературных источников и фактических материалов, необходимых для выполнения ДР;

- проводит регулярные консультации со студентами по согласованному графику или по мере необходимости с целью оказания методической помощи и проверки хода выполнения ДР;

- после выполнения дипломной работы дает письменный отзыв, содержащий оценку дипломной работы в целом и по каждому ее разделу, с точки зрения уровня выполнения задания и соответствия установленным требованиям;

- подписывает готовую дипломную работу.

Обучающийся выпускник должен регулярно давать отчет о полученных результатах в сроки, установленные руководителем и зафиксированные в Календарном плане-графике подготовки дипломной работы.

Невыполнение обучающимся - выпускником указаний руководителя или неявка на консультацию без уважительных причин дают последнему основание отказать от руководства ДР.

Обучающийся, не подготовивший дипломную работу в срок, отчисляется из Академии ИМСИТ с предоставлением последующего права защиты согласно нормативным требованиям.

Рекомендации и замечания руководителя Обучающийся выпускник должен воспринимать критически. Он может учитывать их или отклонить по своему усмотрению, так как теоретически и методически правильная разработка и освещение темы, а также качество содержания и оформление дипломной работы целиком и полностью лежат на ответственности обучающегося - выпускника, а не руководителя.

В процессе проведения исследования руководитель не реже одного раза в неделю проводит необходимые консультации, ведет постоянный контроль за ходом выполнения работы и информирует об этом деканат.

Подписывая ДР на титульном листе, руководитель дает гарантию, что ДР написана и оформлена в соответствии с требованиями Академии ИМСИТ. Если представленная выпускником работа, по мнению руководителя, не соответствует необходимым требованиям, то он вправе не ставить свою подпись на титульном листе и как следствие не допускать работу к защите.

## 2.2 Разработка задания и календарного плана-графика выполнения дипломной работы

С целью организации дальнейшей работы студентом совместно с руководителем после утверждения темы работы и закрепления руководителя разрабатывает **задание** на выполнение дипломной работы (приложение) с указанием сроков начала и окончания работы. В нем указываются: тема дипломной работы, целевая установка основные вопросы, подлежащие разработке (исследованию).

В задании должна быть сформулирована целевая установка, определяющая научное и практическое содержание дипломной работы. Задание подписывается руководителем и обучающимся.

Работу по разработке и выдаче заданий организует и контролирует председатель ПЦК отделения. При утверждении задания председатель ПЦК проверяет соответствие темы дипломной работы теме, утвержденной приказом ректора Академии, при необходимости уточняет с руководителем содержание основных пунктов задания. Для контроля за ходом выполнения дипломной работы предусмотрен календарный **план-график** выполнения дипломной работы (Приложение), разработанный на отделении.

Календарный план-график включает следующие этапы выполнения и защиты дипломных работ:

- разработка плана дипломной работы ;
- подбор литературных источников и сбор фактологического материала;
- консультации с руководителем дипломной работы;
- выполнение литературного обзора по теме;
- обоснование проведения экспериментальной или расчетной части работы, целей, выбор или разработка методик и условий проведения, эксперимента (при необходимости);
- анализ полученных результатов или фактологического материала;
- разработку разделов дипломной работы ;
- представление дипломной работы на отзыв руководителю, а так же на рецензирование;
- подготовку к защите: разработку слайдов, презентаций, иллюстрационного материалов;
- разработку содержания доклада, подготовку к возможным вопросам, возникающим в период защиты;
- предварительное заслушивание доклада по содержанию дипломной работы на предзащите;
- защиту дипломной работы.

Обучающийся, ознакомившись с графиком, обязаны четко выполнять его.

Состояние дипломной работы обсуждается на заседании предметно-цикловой комиссии. Студентам, допустившим отставание в работе, оказывается учебно-методическая помощь, в необходимых случаях применяются меры воспитательного и административного характера. Следующим этапом является разработка плана дипломной работы. Он разрабатывается студентом при активной помощи руководителя. План дипломной работы должен быть тщательно продуман и обоснован. Он включает следующие **разделы**:

- *Введение,*
- *Основную часть,*
- *Заключение,*
- *Список использованных источников,*
- *Приложения.*

Количество разделов и подразделов в основной части во многом зависит от характера и сложности темы.

Основной задачей плана является формулировка заголовков разделов и подразделов дипломной работы. Предварительная формулировка цели и задач исследования облегчает составление плана. Принципиально в дипломной работе должно быть столько глав, сколько изложено в задании основных вопросов, подлежащих разработке (исследованию). Соответственно, и заголовки глав должны быть созвучны содержанию этих вопросов. Детализация и конкретизация задач осуществляется в параграфах глав. Подобный подход обеспечивает выполнение требования к дипломным работам в соответствии ее содержания теме.

Примерный план дипломной работы приведён в приложении.

Практика показывает, что наиболее характерными **ошибками при разработке плана** являются:

1. Совпадение названия глав (параграфов) с темой дипломной работы (главы).
2. Названия глав (параграфов) не раскрывают реального содержания темы дипломной работы (главы) и относятся к другой области знаний.

Обе ошибки не допустимы, особенно вторая, поскольку она приводит к несоответствию содержания дипломной работы ее теме.

При разработке плана необходимо помнить о требованиях научности и практической значимости дипломной работы.

Все главы работы, предусмотренные планом, должны быть связаны друг с другом в логической последовательности и обеспечивать выполнение основных требований, предъявляемых к ней.

Логическая последовательность пунктов плана имеет первостепенную важность. Этапы исследовательской работы должны следовать один за другим так, чтобы прохождение каждого предыдущего этапа необходимым образом



подводило к последующему. Нарушение логической последовательности плана деятельности может завести в тупик и даже сделать работу невыполнимой.

План должен обладать динамическим характером и допускать различные тактические изменения в ходе исследования при сохранении его основной стратегии.

Работа над предварительным планом необходима, поскольку она дает возможность еще до начала написания дипломной работы выявить логические неточности, информационные накладки, повторы, неверную последовательность глав и параграфов, неудачные формулировки выделенных частей или даже ДР в целом. План должен быть разработан и утвержден не позднее чем через 10-12 дней после получения Обучающимся задания на дипломную работу. Необходимо заметить, что в процессе написания работы происходит неоднократное уточнение формулировок названий глав и параграфов работы. На заключительной стадии написания дипломной работы план окончательно корректируется руководителем и на его основе формируется содержание.

Следует иметь в виду, что любая тема может иметь несколько вариантов изложения. В одних случаях будет доминировать теоретическая часть, в других – практическая. Направление и характер работы могут видоизменяться также в зависимости от того, в какой мере будет привлечен фактический материал.

Основной задачей плана является формулировка заголовков разделов и подразделов дипломной работы. Предварительная формулировка цели и задач исследования облегчает составление плана. Детализация и конкретизация задач осуществляется в подразделах и пунктах работы.

3. Совпадение названия разделов (подразделов, пунктов) с темой дипломной работы (раздела).

4. Названия разделов (подразделов, пунктов) не раскрывают реального содержания темы дипломной работы (раздела) и относятся к другой области знаний.

*Обе ошибки не допустимы, особенно вторая, поскольку она приводит к несоответствию содержания дипломной работы ее теме.*

При разработке плана необходимо помнить о требованиях научности и практической значимости дипломной работы.

План должен обладать динамическим характером и допускать различные тактические изменения в ходе исследования при сохранении его основной стратегии. Работа над предварительным планом необходима, поскольку она дает возможность еще до начала написания дипломной работы выявить логические неточности, информационные накладки, повторы, неверную последовательность разделов и подразделов, неудачные формулировки

выделенных частей или даже всей работы в целом. Необходимо заметить, что в процессе написания работы происходит неоднократное уточнение формулировок названий разделов и подразделов работы.

### **2.3 Изучение теоретических аспектов темы работы**

Основной задачей данного этапа является подбор и составление списка литературных источников к каждой главе дипломной работы. Список используемых источников позволяет в значительной степени оценить качество проделанного исследования. Так, при отсутствии в перечне новейших материалов или признанных в научной среде трудов по избранной теме можно сделать вывод, что ДР отличается требуемой глубиной исследования и не основывается на последних достижениях научной мысли.

Знакомство с опубликованной по теме дипломной работы литературой начинается с разработки идеи, то есть замысла предлагаемого исследования. При этом исследуются материалы, опубликованные в различных отечественных и зарубежных изданиях, официальные материалы, практические документы организации.

Подбор литературы осуществляется в учебных, публичных и научных библиотеках при помощи каталогов и библиографов, а также во Всемирной сети Интернет.

При подборе литературы необходимо ориентироваться на следующие типы:

1. Научные специализированные журналы по тематике выбранной специальности.
2. Информационные издания по определенным тематическим направлениям, цель выпуска которых - оперативная информация, как о самих изданиях, так и о существенных сторонах их содержания.
3. Периодические издания (научно-популярные, специализированные газеты и журналы).
4. Учебники и учебные пособия, рекомендованные к использованию Министерством образования Российской Федерации или отраслевыми министерствами.
5. Монографии известных ученых и практиков.
6. Справочники.
7. Научные отчеты научно-исследовательских и других учреждений.
8. Государственные, отраслевые и территориальные нормативные документы (законы, кодексы, постановления правительства, приказы и инструкции отраслевых министерств).

Количество источников, использованных при написании работы, должно быть не менее 30, из них 5-10 законодательных и нормативных актов, монографии, научные сборники и статьи.

Их соотношение зависит от избранной темы. При небольшом количестве литературы непосредственно по исследуемой теме необходимо использовать литературу по смежным темам. Допустимо и приветствуется использование литературы на иностранных языках, экспертных исследований. Учебники и учебные пособия могут быть использованы при написании теоретической части работы.

При написании дипломной работы должна быть использована литература последних 3-5 лет. Исключение составляют основополагающие теоретические труды по избранной теме; историческая тематика, а так же отсутствие в последние два-три года радикальных изменений в той области, по теме которой пишется дипломная работа, либо в смежных областях. На последнее нужно обратить особое внимание, т.к. изменения, например, в налоговом законодательстве, касаются всех групп экономических отношений, и, соответственно, предметных областей и содержания дипломных работ.

## **2.4 Написание и оформление дипломной работы**

Дипломная работа должна представлять собой законченную разработку актуальной экономической проблемы и обязательно включать в себя как теоретическую часть, в которой показаны знания основ теории по разрабатываемой проблеме, так и практическую часть, в которой необходимо показать уровень сформированных компетенций, предусмотренных соответствующим ФГОС СПО, профессиональных знаний выпускника, его умений и навыков по осуществлению практической и / или научной деятельности.

ДР разрабатывается на конкретном материале предприятий, организаций, органов управления и должна содержать решение актуальных задач.

Дипломная работа обязательно должна как теоретический, так и практический материал.

ДР, содержащая только теоретический материал (или изложение действующих инструкций, методик и т.п.) без практических рекомендаций к защите не допускается.

Особое внимание следует уделить **логике изложения материала.**

Основные ее **требования:**

- соответствие выводов и предложений результатам анализа;
- отсутствие повторений и дублирования по разделам;
- точное соответствие текста дипломной работы поставленным в плане вопросам;
- корректность и ясность формулировок.

Не допускается дословное переписывание литературных источников. **Язык и стиль** дипломной работы должны соответствовать нормам письменной

научной речи. Прежде всего, необходимо соблюдать формально-логическую последовательность, целостность и связность изложения материала. Также должен использоваться терминологический аппарат данной предметной области, без применения профессиональной лексики (жаргона) и лексики средств массовой информации. В этой связи необходимо обратить внимание на юридически правильные названия учреждений и организаций, упоминаемых в работе. Сокращения этих названий должны соответствовать требованиям ГОСТ или нормативных актов.

Дипломная работа должна содержать следующие **структурные элементы**:

- Титульный лист;
- Бланк задания;
- **РЕФЕРАТ**;
- **СОДЕРЖАНИЕ**;
- **ВВЕДЕНИЕ**;
- Основная часть;
- **ЗАКЛЮЧЕНИЕ**;
- **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**;
- **ПРИЛОЖЕНИЯ**.

Структурные элементы перечислены в порядке размещения их в документе.

**Титульный лист** является первой страницей дипломной работы, заполняется по строго определенным правилам, пример оформления титульного листа приведен в (приложении).

**Реферат** должен кратко отражать основное содержание дипломной работы и содержать следующие структурные элементы:

- сведения об объеме дипломной работы, количестве иллюстраций, таблиц, приложений, количестве частей дипломной работы, количестве использованных источников;
- перечень ключевых слов и словосочетаний, включающий от 5 до 10 слов или словосочетаний из текста дипломной работы, которые в наибольшей степени характеризуют её содержание и обеспечивают возможность информационного поиска. Ключевые слова приводятся в именительном падеже и печатаются прописными буквами в строку через запятые;
- текст реферата, который должен отражать объект исследования, цель работы, метод или методологию проведения работы, основные результаты работы, рекомендации по внедрению или итоги внедрения результатов дипломной работы, область применения, экономическую эффективность или значимость работы.

Оптимальный объем реферата – 0,75 страницы текста.

В **Содержании** последовательно перечисляют все заголовки дипломной работы с указанием страниц, с которых они начинаются. Заголовки должны точно повторяться в тексте дипломной работы. Сокращать заголовки в содержании, давать их в иной редакции по сравнению с заголовками в тексте не допускается. Пример оформления содержания приведен в Приложении.

В состав дипломной работы входит иллюстративная графическая часть, которая представляет собой набор слайдов, отражающих основное содержание работы.

Рекомендуемый объем текстовой части дипломной работы 55 - 70 листов формата А4 без учета приложений и 8 – 12 слайдов.

Содержание проектной (основной) части зависит от темы дипломной работы.

Если дипломная работа выполняется по направлению «Разработка базы данных», то 2.1 Построение модели данных.

Если ДР выполняется по направлению «Разработка программного обеспечения информационной системы», то

2.1 Построение модели данных

2.2 Функционально-структурная модель информационной системы

2.3 Определение архитектурных представлений.

Если ДР выполняется по направлению «Системы, основанные на web-технологии», то 2.1 Построение модели данных

2.2 Определение архитектурных представлений

2.3 Выбор платформы.

Если ДР выполняется по направлению «Создание электронных учебников (обучающих систем)», то

2.1 Подготовка проекта гипертекста для компьютерной реализации

2.2 Проектирование интеллектуального ядра

2.3 Разработка компьютерной поддержки

2.4 Озвучивание, визуализация с помощью компьютерных средств. Если ДР выполняется по направлению «Программные комплексы», то 2.1 Построение алгоритма.

При необходимости ДР может содержать обоснование экономической эффективности проекта.

**Введение** (общим объемом не более 3 стр.) должно содержать общие сведения о ДР, краткую характеристику, резюме. В нем необходимо отразить актуальность и новизна выбранной темы (1-3 абзаца), указать объект и предмет исследования, цель работы, задачи которые нужно решить для достижения цели, предполагаемые пути решения и используемые методики (исследовать, проанализировать, выбрать, построить, разработать), практическую

значимость полученных результатов. Введение не должно содержать общих фраз, а должно указывать на конкретные решения для данной работы.

Целью работы может быть создание программного продукта в виде программного модуля, законченной программы, автоматизированной системы для решения задач пользователя конкретной предметной области (предприятия). Дополнительно может достигаться совершенствование информационной базы, применение новых технических средств сбора, передачи, обработки и выдачи информации. Во введении необходимо также перечислить вопросы, которые будут рассмотрены в работе, выделив вопросы, которые предполагается решить практически. Необходимо указать структуру ДР (Введение, количество глав, заключение, список использованных источников, приложения).

Целью **аналитической части** является рассмотрение существующего состояния предметной области, характеристик объекта и системы управления и обоснование предложений по устранению выявленных недостатков, внедрению новых подходов, новых технологий и т. д.

Раздел является первым в работе и содержит следующие подразделы:

- 1.1 Исследование предметной области
- 1.2 Модель предметной области
- 1.3 Анализ требований к программному обеспечению
- 1.4 Техническое задание на разработку
- 1.5 Выбор метода и среды разработки программного обеспечения

Название раздела может быть изменено автором и соответствовать тематике дипломной работы.

При написании раздела учитываются результаты преддипломной практики, обзор литературы, информация в сети Internet. В этом разделе должна быть приведена технико-экономическая характеристика предметной области, определены цели и задачи проектирования и разработки программного продукта, сделан обзор, анализ и выбор среды разработки.

В качестве **предметной области** могут выступать предприятия, фирмы, объединения и т.д. или их подразделения. Технико-экономическая характеристика предметной области должна включать наименование, юридический статус и направление деятельности предприятия, его организационную структуру и краткую характеристику технико-экономических аспектов объекта проектирования. Такими аспектами являются:

- объект проектирования;
- тип производства (услуг);
- количество передаваемой по сети информации.

Характеризуя предприятие, необходимо акцентировать внимание на тех его структурных компонентах, в которых будут использованы результаты

(наработки) данного выполнения дипломной работы, давая подробное описание предметной области, учитывая количество персонала, количество компьютеров в сети, этажность здания, где будет располагаться компьютерная сеть, наличие филиалов, виды и объем выпуска продукции или оказания услуг. Необходимо указать, если есть, головную организацию и дочерние организации.

Надо описать основные тенденции развития предприятия и отрасли, сильные и слабые стороны предприятия.

Любая экономическая система это множество ресурсов и процессов, связанных между собой. Обычно их описывают в терминах предметной области.

Под предметной областью понимают элементы материальной системы, информация о которых обрабатывается и хранится в экономической информационной системе.

При проектировании программного обеспечения выясняют запросы и пожелания клиента и определяют возможный подход к решению задачи.

Общая схема исследования предметной области может быть представлена в виде:

- Получение общего представления.
- Проведение интервью с ключевыми лицами по проблемам предметной области и текущих решений по управлению данной предметной областью.
- Сбор законодательной, нормативной и регламентирующей документации по предметной области.
- Анализ стратегий (концепций, политик) и других нормативных документов предметной области.
- Ранжирование списка проблем (решение которых требует минимальных усилий; решение которых дает максимальный эффект и т.п.), выявление первоочередных проблем, выявление системных проблем, выявление обусловленности проблем.
- Выявление узких мест, проблем, симптомов. Составление сводного перечня проблем предметной области.

На основе проведенного исследования осуществляют постановку задачи проектирования (цель разрабатываемого программного продукта и круг его пользователей, варианты использования, функциональные и нефункциональные требования), собирают исходные данные, необходимые для проектирования, разрабатывают модель предметной области, то есть выполняют структуризацию в виде документов.

Необходимо иметь в виду, что к нефункциональным требованиям относятся такие свойства системы, как ограничения среды и реализации, производительность, зависимость от платформы, расширяемость и надежность. Под надежностью понимают точность, среднюю наработку на

отказ, число ошибок на тысячу строк программы, класс и т.д. Требования по производительности налагают специфические условия на скорость, пропускную способность, время отклика и используемую память. Большинство требований, связанных с производительностью, относятся лишь к нескольким вариантам использования и должны быть приписаны к ним как именованные значения

Пример. Специальные требования к варианту использования «Оплата счета». Требования по производительности. Когда покупатель подает счет к оплате, система должна выдать результат проверки запроса не медленнее чем за 1,0 секунду в 90 % случаев. Время проверки никогда не должно превышать 10,0 секунд.

Цель моделирования **предметной области** состоит в том, чтобы понять и описать наиболее важные классы контекста предметной области. Небольшие предметные области обычно содержат от 10 до 50 основных классов. В более обширной предметной области классов может быть гораздо больше. *Классом объектов* называют совокупность объектов, обладающих одинаковым набором свойств. Например, если в качестве предметной области рассмотреть образовательную организацию, то в ней можно выделить следующие классы объектов: обучающиеся, преподаватели, аудитории и т. д.

Модель предметной области определяет наиболее важные типы объектов контекста системы. Объекты предметной области представляют собой «предметы», которые существуют, или события, происходящие в среде, где работает система.

Многие из объектов предметной области или классов предметной области можно определить в ходе опроса специалистов исследуемой организации.

Классы предметной области разбивают на три типовых категории:

- бизнес-объекты, описывающие сущности, используемые в бизнесе, например, заявки, счета, контракты;
- объекты и понятия реального мира, которые система должна отслеживать, такие как вражеские самолеты, ракеты и траектории;
- события, которые произойдут или произошли, например прибытие самолета, отлет самолета или перерыв на обед.

Модель предметной области описывают диаграммами UML, главным образом диаграммами классов, которые предоставляют разработчикам информацию о классах предметной области и их ассоциативных связях.

Например, классы предметной области: Заказ, Счет, Предмет и Банковский счет. Система будет использовать Интернет для пересылки заказов, счетов и платежей между покупателями и продавцами. Система должна помогать покупателю готовить заказы, продавцу рассчитывать стоимость заказов и рассылать счета и покупателю проверять правильность



выписанных счетов и совершать платеж продавцу со своего банковского. Отметим, что Заказ— это запрос покупателя продавцу на поставку изделий. Каждое изделие занимает «одну строку» в заказе. Заказ имеет такие атрибуты, как дата выписки и адрес поставки.

Счет — это запрос на оплату, посылаемый продавцом покупателю в ответ на заказ товаров или услуг. Счет имеет такие атрибуты, как сумма к оплате, дата выписки и максимальная отсрочка оплаты. На несколько заказов может быть выслан один счет. Счет считается оплаченным, когда деньги с банковского счета покупателя перешли на банковский счет продавца.

Банковский счет имеет такие атрибуты, как баланс и владелец. Атрибут владелец идентифицирует лицо, которому принадлежит банковский счет

Диаграмма классов изображена на рисунке 1.

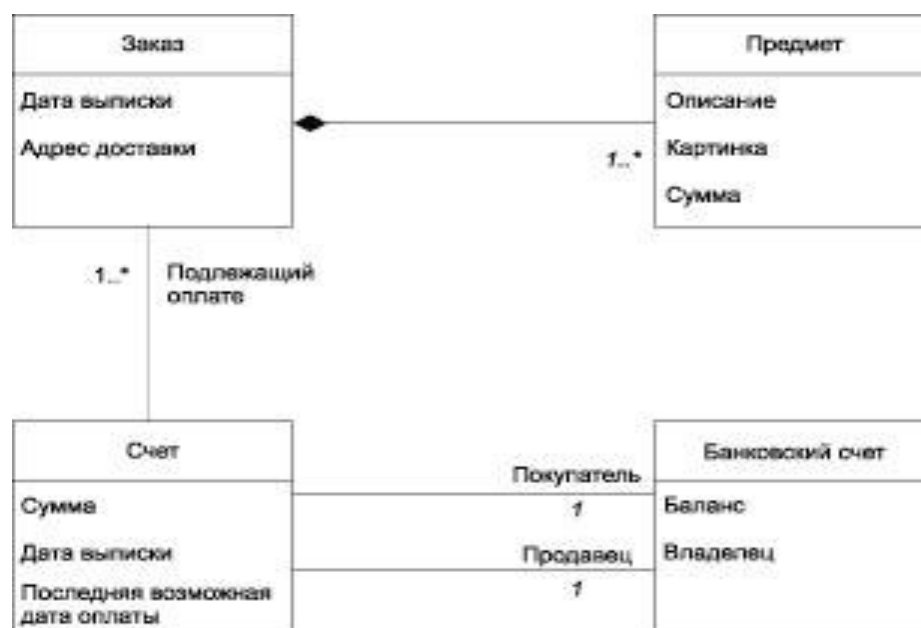


Рисунок 1 - Диаграмма классов в модели предметной области

Для обоснования актуальности предполагаемой разработки необходимо привести данные обзора использованных источников и литературы, касающихся рассматриваемой проблемы и состояния ее разрешения в настоящее время и сформулировать задачи, которые следует решить в ДР.

В разделе **анализа требований к программному обеспечению** проходит структуризация уже собранных ранее требований. Цель этапа — предоставить четкий список не дублируемых требований к системе, которые должны быть выделены из избыточных и частично дублирующих сценариев и пользовательских историй, которые были полученных на предыдущем этапе. Правильно сгруппированные требования помогут обойтись минимальным количеством функционала для удовлетворения максимально большего количества целей.

Наиболее эффективным способом анализа требований является определение сценариев работы пользователей с будущим продуктом.

Сценарий представляет совокупность всех процессов с их окружением, в которых продукт будет участвовать. Ценность сценария состоит в том, что он описывает способы взаимодействия с продуктом всех его пользователей одновременно на протяжении всего цикла эксплуатации продукта и, следовательно, гарантирует отсутствие к нему взаимоисключающих требований. Проверяют сценарий путем анализа его выполнения всеми заинтересованными лицами.

Каждый сценарий должен содержать: информацию обо всех типах пользователей, взаимодействующих с ним, описание процессов, которые будут затрагивать продукт, требования к дизайну (операционная система; приложения, с которыми интегрируется программный продукт, форматы ввода вывода) и приоритет.

В разделе **«Выбор метода и среды разработки программного обеспечения»** производят описание метода программирования и (или) среды разработки и обосновывают их выбор.

Известны следующие методы разработки программного обеспечения: модульное, объектно-ориентированное программирование, компонентно-ориентированное программирование и параллельное программирование.

Модульная программа такая, в которой любую часть логической структуры можно изменить, не вызывая изменений в остальных частях программы. Одним из наиболее абстрактных понятий, связанных с модульностью, является понятие независимости: в модульной программе каждый модуль не зависит от других, то есть его можно изменить или модифицировать без последствий в других модулях. Достоинства модульных программ следующие:

- модульные программы легко составлять и отлаживать. Функциональные компоненты такой программы могут быть написаны и отлажены порознь;

- такую программу легче сопровождать и модифицировать. Функциональные компоненты могут быть изменены, переписаны или заменены без изменений в остальных частях.

Недостатки модульности:

- Модульный подход иногда требует большего времени ЦП. Эта проблема возникает, прежде всего, в тех случаях, когда программа отличается, наличием большого числа подпрограмм, написанных на языках высокого уровня;

- В модульном подходе может потребоваться несколько больший объем памяти. Если каждой подпрограмме отводится отдельная часть рабочей памяти, то всей программе может потребоваться несколько больший объем памяти.

Объектно-ориентированное программирование (ООП) позволяет моделировать объекты определённой предметной области путем программирования их содержания и поведения в пределах класса. Конструкция «класс» обеспечивает механизм инкапсуляции для реализации абстрактных типов данных. Инкапсуляция как бы скрывает и подробности внутренней реализации типов, и внешние операции и функции, допустимые для выполнения над объектами этого типа.

Разработка объектно-ориентированных программ состоит из следующих последовательных работ:

- определение основных объектов, необходимых для решения данной задачи;
- определение закрытых данных (данных состояния) для выбранных объектов;
- определение второстепенных объектов и их закрытых данных;
- определение иерархической системы классов, представляющих выбранные объекты;
- определение ключевых сообщений, которые должны обрабатывать объекты каждого класса;
- разработка последовательности выражений, которые позволяют решить поставленную задачу;
- разработка методов, обрабатывающих каждое сообщение;
- очистка проекта, то есть устранение всех вспомогательных промежуточных материалов, использовавшихся при проектировании;
- кодирование, отладка, компоновка и тестирование.

Компонентно-ориентированное программирование (КОП) можно описать примерно такой формулой: КОП = ООП+ модульность (включая сокрытие информации и позднее связывание модулей, то есть возможность подгружать необходимые модули в процессе выполнения программы, а не заранее, как в старых системах программирования). В КОП запрещено наследование от типов, реализованных в других модулях; наследовать можно только абстрактным, чисто интерфейсным типам.

Важным практическим следствием реализации концепции компонентного подхода для экономики программирования является снижение стоимости проектирования и реализации программного обеспечения. Еще одно очевидное достоинство компонентного программирования – возможность усовершенствования стратегии повторного использования кода. Код с более высоким уровнем абстракции не требует существенной модификации при адаптации к изменившимся условиям задачи или новым типам данных.

Кроме того, к преимуществам концепции компонентного программирования следует отнести унификацию обработки объектов различной природы. Абстрактные классы и методы позволяют единообразно

оперировать гетерогенными данными, причем для адаптации к новым классам и типам данных не требуется реализации дополнительного программного кода. Важно также отметить, что идеология компонентного программирования основана на строгом математическом фундаменте, что обеспечивает интуитивную прозрачность исходного текста для математически мыслящего программиста, а также верифицируемость программного кода.

Концепция компонентного программирования универсальна и в равной степени применима для различных подходов к программированию, включая функциональный и объектно-ориентированный.

Всегда существуют «большие» задачи, которые не по силам одному компьютеру. Такие задачи приходится решать на параллельных вычислительных системах.

Параллельное программирование включает в себя все черты более традиционного, последовательного программирования, но в нем есть три дополнительных, четко определенных этапа.

- определение параллелизма: анализ задачи с целью выделить подзадачи, которые могут выполняться одновременно;

- выявление параллелизма, то есть изменение структуры задачи таким образом, чтобы было эффективно выполнять подзадачи. Для этого обычно находят зависимости между подзадачами и организуют соответствующим образом исходный код;

- реализация параллельного алгоритма в исходном коде с помощью системы обозначений параллельного программирования.

Выбор среды разработки обусловлен темой дипломной работы и задачами, которые необходимо решить, чтобы достичь цель дипломной работы.

**Техническое задание на разработку программного обеспечения** составляют в соответствии с требованиями ГОСТ 19.201-78. Оно содержит разделы:

- введение;
- назначение разработки;
- требования к программе или программному изделию;
- требования к программной документации;
- технико-экономические показатели;
- порядок контроля и приемки;

Например, техническое задание на программный продукт «Деятельность библиотеки» содержит следующие разделы:

- цель проекта;
- рамки проекта;
- цель работы системы;
- структура автоматизируемого объекта;

- описание подразделений;
- описание ролей;
- макеты документов;
- краткая характеристика отдельных подсистем;
- описание основных решаемых задач;
- описание структуры данных для создания одноименных форм;
- нефункциональные требования (практичность, надежность, производительность, возможность сопровождения);
- требования к интерактивной пользовательской документации и системе подсказок (руководство администратора системы, руководство пользователя системы);
- ограничения проектирования (требования к системному программному обеспечению, требования к языкам программирования, требования к информационному обеспечению (клиентские операционные системы, серверные операционные системы, системы управления базами данных));
- требования к техническому обеспечению;
- интерфейсы (требования к элементам пользовательского интерфейса, общие требования к пользовательским интерфейсам);
- применяемые стандарты;
- *основная часть.*

**Проектная часть** дипломной работы является описанием решений, принятых по всей вертикали проектирования. Глава должна быть основана на информации, представленной в аналитической части, обобщать ее. По сути, проектная часть является решением проблематики, изложенной в аналитической части, на языке информационных технологий. Поэтому недопустимо, если при проектировании используется информация об объекте управления, не описанная в первой главе.

Проектная часть является второй главой и может состоять из следующих разделов и подразделов:

1. Функционально-структурная модель ИС.
2. Архитектурное представление программного продукта.
3. Логическая модель базы данных

Содержание проектной (основной) части зависит от темы дипломной работы.

1) Содержание проектной части по направлению «Разработка базы данных» Если дипломная работа выполняется по направлению «Разработка базы данных», то необходимо рассмотреть построение модели данных.

Данный раздел ДР должен содержать подробное описание модели данных на языке диаграмм *сущность-связь* (ER-диаграмм). Использование ER-диаграмм является наиболее естественным и удобным средством отражения модели базы данных. Основные понятия ER-диаграмм (сущность, связь,

экземпляр сущности, атрибут и ключ сущности) полностью соответствуют основным атрибутам реляционной модели данных.

На рисунке 3 перечислены элементы ER-диаграмм.

Сущность обозначается в виде прямоугольника с наименованием:

а – без атрибутов,

б – с указанием атрибутов,

в – с ключевым атрибутом.

Обозначения связей между сущностями показаны на рисунке 3.

При этом каждая связь может иметь одну из двух модальностей связи: ---  
- Может; Должен. У связи может быть разная модальность с разных концов и может быть прочитана как слева направо, так и справа налево.

Например, в базе данных библиотечных книг, находящихся на руках у читателей библиотеки, присутствуют две сущности: «Читатель» и «Книга». При этом один читатель может иметь на руках более одной книги, но каждая книга может быть на руках только у одного из читателей.

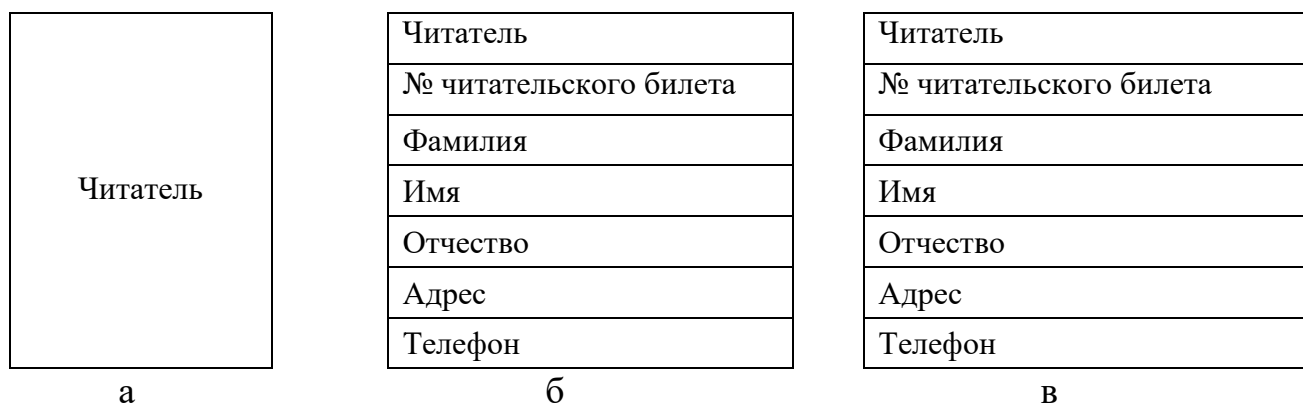


Рисунок 3 – Обозначение сущности

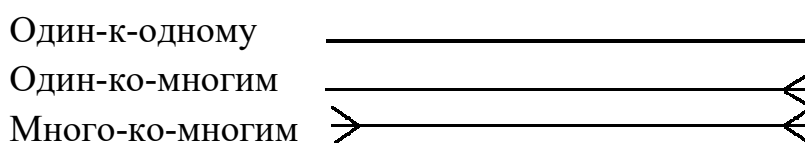


Рисунок 4 – Обозначение связей между сущностями

В связи может быть разная модальность с разных концов и может быть прочитана как слева направо, так и справа налево.

Например, в базе данных библиотечных книг находящихся на руках у читателей библиотеки, присутствуют две сущности: «Читатель» и «Книга». При этом один читатель может иметь на руках более одной книги, но каждая книга может быть на руках только у одного из читателей.

Следуя логике задачи, разумно сделать вывод, что «Читатель» “может” иметь несколько «Книг». «Книга» одного наименования на руках «Читателя» “должна” быть только одна. Поэтому диаграмма, описывающая модель данной базы, имеет вид, представленный на рисунке 4, где связь между таблицами

читается слева направо: «Читатель может иметь несколько книг»; справа налево: «Книга должна принадлежать только одному читателю».

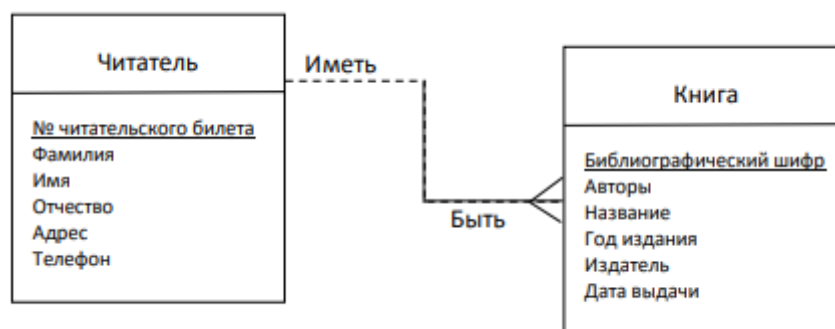


Рисунок 4 – Диаграмма модели данных

Содержание проектной части по направлению «Системы, основанные на web технологиях»

Выбор типа проекта. Прежде, чем приступить к проектированию WEB-ресурса, необходимо решить, каким он будет. Это – обязательный этап, так как без него велик риск того, что проект будет незавершен и этому есть веские причины. Заказчик, как правило, не имеет четкого представления о том, из чего состоит WEB-дизайн. Отсюда появляются ошибочные входящие требования, что приводит к очередным переделываниям оригинал макетов.

Решить проблему поможет выбор пунктов из нижеследующих тем, которые затем будут воплощены в проекте. Во избежание невыполнимых сочетаний требований, из каждой темы выбирается только один пункт.

Тема 1: Web-проект по восприятию

- спокойный;
- vip, представительный, официальный, строгий;
- вызывающий, достающий, сдувающий;
- легкомысленный;
- шокирующий;
- по теме, но очень необычный и очень интересный;
- отражающий специфику фирмы;
- с уклоном в детство (визуалы, формы, цвета, композиция);
- что-то другое (что? Объяснить или показать аналог).

Тема 2: Категории Web-проектов

– корпоративный информационный (строгий, официальный сайт компании, простые и понятные формы, процентное соотношение текст/графика = 90/10)(предпочтителен для часто обновляемых информационных и новостных, проектов с большим объемом информации);

– корпоративный имиджевый (эксклюзивный оригинальный дизайн выгодно представляющий компанию, использование нестандартных идей и

решений в оформлении, процентное соотношение текст/графика = 50.50) (создается для рекламы и продвижения компании, товаров или услуги);

- визитная карточка (небольшой по объему статичный проект, содержащий от 1 до 5 информационных страниц, дизайн может быть любой сложности). Рекомендуются как предварительный этап освоения интернет для людей сомневающих в целесообразности создания сайта и не готовых к большим денежным вложениям, недорогой (~500\$) и быстрый в изготовлении (~14 дней);

- интернет-магазин (дорогой и сложный целевой проект). Люди, заказывающие такой проект знают, что делают;

- информационный портал (текстовый дизайн, сборка новостей также и с других информационных источников, графика в основном в виде конкретных фотографий к новостям);

- игровой портал (сложный развлекательный проект, предусматривающий большую посещаемость и ресурсоемкость);

- персональный проект (личный проект содержит все, что угодно и проектируется, обычно самим автором);

- спец-проекты (галереи фотографий, почта, счетчики, рубрикаторы, баннерные крутилки и тд.);

- другое (показать или толково описать пример проекта).

Тема 3: Использование бренда (торговая марка, имиджевый образ)

- бренд должен быть основным элементом главной страницы (внутренних страниц);

- бренд носит фоновую нагрузку;

- на усмотрение дизайнера;

- бренд не нужен.

Тема 4: Цветовая гамма и насыщенность

- только цвета фирменного стиля;

- цвета фирменного стиля и любые сочетаемые с ними;

- цвета, подходящие к стилю дизайна и тематике сайта (на усмотрение дизайнера);

- цвета, подходящие к стилю дизайна и тематике сайта (перечислить, какие нужны);

- только кричащие, динамичные локальные цвета (красный, синий, желтый, зеленый);

- только мягкие пастельные цвета;

- использовать монохромные темы (черный, белый, серый тона);

- другие приемлемые сочетания (какие?);

- на усмотрение дизайнера;

- кислотные флюорисцентные цвета.

Тема 5: Визуалы (можно указать несколько пунктов)



- на усмотрение дизайнера;
- не нужно;
- абстракции;
- люди;
- животные;
- растения;
- техника;
- бытовые предметы;
- неживая природа (что?);
- растровые и орнаментальные элементы;
- другое (что?).

#### Тема 6: Фон страницы

- белый (нет фона);
- однотонный подходящий к дизайну (на усмотрение дизайнера);
- однотонный подходящий к имиджу (пример фона);
- абстрактный или тематический орнамент-узор;
- на усмотрение дизайнера.

#### Тема 7: Вёрстка (общая композиция)

- жесткая левосторонняя/правосторонняя;
- жесткая по центру;
- резиновая (растягивающаяся по размеру экрана монитора).

#### Тема 8: Размер web-страницы по горизонтали (для жесткой верстки)

- 800x600;
- 1024x768;
- на усмотрение дизайнера.

#### Тема 9: Внутренние и внешние страницы

- структура и оформление главной и внутренних страниц – одинакова;
- структура и оформление главной и внутренних страниц – разные;
- присутствие сплеш-страницы (плохая оптимизация);
- на усмотрение дизайнера.

#### Тема 10: Особые пожелания заказчика

Пожелания заказчика.

**Определение архитектурных представлений.** Данная стадия разработки Интернет-ресурса заключается в построении его схемы. Обычно схема представляет собой чертеж структуры сайта, состоящий из двух основных графических элементов: *прямоугольника* и *линии*. Прямоугольник соответствует странице сайта, внутри которого обычно находится текст названия или содержания страницы. Линия обозначает связь одной страницы с другой, которая на практике реализуется с помощью текстовой гиперссылки.

Для построения схемы можно воспользоваться либо *графическим* либо *программным* методом.

Графический метод заключается в привлечении средств чертежной графики: бумага, чертежные инструменты. Можно также воспользоваться программным обеспечением (ПО) компьютерной графики.

Программный метод состоит в использовании готового ПО, предназначенного для визуального проектирования WEB-сайтов (Microsoft FrontPage, Namo WebEditor и т. П.).

В качестве примера приведем общую схему сайта мебельной фирмы (рисунок 5).



Рисунок 5 — Схема сайта

Независимо от выбора метода, рекомендуется использовать готовые шаблоны WEB-ресурсов, изготовленные фирмами–разработчиками ПО, что не только экономит время на разработку, но и обеспечивает полное соответствие современному дизайну. Существуют как коммерческие, так и свободно распространяемые шаблоны, которые можно найти в сети Интернет. В крайнем случае, если ни один из найденных шаблонов не соответствует требованиям к WEB-ресурсу, всегда можно отредактировать наиболее близкий по своей форме шаблон к нужному виду.

**Выбор платформы.** Следующий этап разработки состоит в выборе платформы, на которой будет реализован WEB-ресурс.

Если будущий ресурс является обычной страницей в сети Интернет, цель которого – информировать посетителей и не более, то для его реализации вполне достаточно использовать язык разметки HTML и какой-нибудь язык сценариев, выполняемых на стороне клиента (JavaScript, VBScript и т. П.). Создавать такие сайты можно и «вручную», т. Е. набирать текст и разметку в простом текстовом редакторе, или с использованием специализированного ПО, например Namo WebEditor или FrontPage. Данный способ реализации определяет и способ размещения ресурса. Так как вся его функциональность

будет полностью выполняться браузером, то размещать подобные ресурсы можно на бесплатных хостах.

Если предполагается использовать сайт для коммерческих целей, т. е. с целью заработка, то возможны варианты:

- в случае использования рекламы на сайте в качестве основного (или единственного) источника дохода, то стоит остановить свой выбор на платных хостах, так как рекламные скрипты нередко выполняются на стороне сервера, а владельцы бесплатных хостов обычно ограничивают такую услугу или исключают её совсем. Кроме того фирмы-рекламодатели стараются избегать бесплатных хостов;

- если будущий WEB-ресурс должен обладать большей функциональностью, нежели только информировать посетителей, а, например, выполнять функции Интернет-магазина, то использование только HTML и сценариев для обозревателя явно недостаточно. Необходимо, чтобы на хосте было установлено специализированное ПО, которое включает как минимум WEB-сервер, например Apache, и транслятор языка программирования сценариев для WEB-сервера, например PHP.

Если предполагается использовать ресурс для доступа к хранилищу данных, то необходимо наличие на хосте WEB-сервера, языка сценариев для него и СУБД (Oracle, MySQL, SQL Server и т. П.)

**Содержание проектной части по направлению «Создание электронных учебников (Обучающих систем)».** Принципы создания электронного учебника включают в себя:

1. Принцип квантования: разбиение материала на разделы, состоящие из модулей, минимальных по объему, но замкнутых по содержанию.

2. Принцип полноты: каждый модуль должен иметь следующие компоненты

- теоретическое ядро,
- контрольные вопросы по теории,
- примеры,
- задачи и упражнения для самостоятельного решения,
- контрольные вопросы по всему модулю с ответами,
- контрольная работа,
- контекстная справка (Help),
- исторический комментарий.

3. Принцип наглядности: каждый модуль должен состоять из коллекции кадров с минимумом текста и визуализацией, облегчающей понимание и запоминание новых понятий, утверждений и методов.

4. Принцип ветвления: каждый модуль должен быть связан гипертекстными ссылками с другими модулями так, чтобы у пользователя был выбор перехода в любой другой модуль. Принцип ветвления не исключает, а

даже предполагает наличие рекомендуемых переходов, реализующих последовательное изучение предмета.

5. Принцип регулирования: студент самостоятельно управляет сменой кадров, имеет возможность вызвать на экран любое количество примеров (понятие «пример» имеет широкий смысл: это и примеры, иллюстрирующие изучаемые понятия и утверждения, и примеры решения конкретных задач, а также контр-примеры), решить необходимое ему количество задач, задаваемого им самим или определяемого преподавателем уровня сложности, а также проверить себя, ответив на контрольные вопросы и выполнив контрольную работу заданного уровня сложности.

6. Принцип адаптивности: электронный учебник должен допускать адаптацию к нуждам конкретного пользователя в процессе учебы, позволять варьировать глубину и сложность изучаемого материала и его прикладную направленность в зависимости от будущей специальности студента. Применительно к нуждам пользователя генерировать дополнительный иллюстративный материал, предоставлять графические и геометрические интерпретации изучаемых понятий и полученных учащимся решений задач.

7. Принцип компьютерной поддержки: в любой момент работы студент может получить компьютерную поддержку, освобождающую его от рутинной работы и позволяющую сосредоточиться на сути изучаемого в данный момент материала, рассмотреть большее количество примеров и решить больше задач.

Причем компьютер не только выполняет громоздкие преобразования, разнообразные вычисления и графические построения, но и совершает математические операции любого уровня сложности, если они уже изучены ранее, а также проверяет полученные результаты на любом этапе, а не только на уровне ответа.

8. Принцип собираемости: электронные учебники должны быть выполнены в форматах, позволяющих компоновать их в единые электронные комплексы, расширять и дополнять их новыми разделами и темами, а также формировать электронные библиотеки по отдельным дисциплинам или личные электронные библиотеки студента, преподавателя или исследователя.

Достоинства электронного учебника (ЭУ) заключаются в том, что он:

- облегчает понимание изучаемого материала за счет иных нежели в печатной учебной литературе, способов подачи материала;
- допускает адаптацию в соответствии с потребностями обучающегося, уровнем его подготовки, интеллектуальными возможностями и амбициями;
- предоставляет широчайшие возможности для самопроверки на всех этапах работы;
- содержит практически неограниченное количество разъяснений, повторений, подсказок и проч.;

- позволяет использовать компьютерную поддержку для решения большего количества задач, освобождает время для анализа полученных решений и их графической интерпретации;
- дает возможность преподавателю проводить занятие в форме самостоятельной работы за компьютерами, оставляя за собой роль руководителя и консультанта;
- позволяет преподавателю быстро и эффективно контролировать знания студентов, задавать содержание и уровень сложности контрольного мероприятия.

На первом этапе разработки ЭУ целесообразно подобрать в качестве источников такие печатные и электронные издания, которые наиболее полно соответствуют стандартной программе, лаконичны и удобны для создания гипертекстов, содержат большое количество примеров и задач, имеются в удобных форматах (принцип собираемости).

На втором этапе разрабатывают оглавление, то есть разбивают материал на разделы, состоящие из модулей, минимальных по объему, но замкнутых по содержанию, а также составляют перечень понятий, которые необходимы и достаточны для овладения предметом (двух– или трехуровневый индекс).

На третьем этапе перерабатывают тексты источников в соответствии с оглавлением, индексом и структурой модулей; исключают те из них, которые не вошли в перечни, и пишут те, которых нет в источниках; разрабатывают систему контекстных справок (Help); определяют связи между модулями и другие гипертекстные связи.

Таким образом, подготавливают проект гипертекста для компьютерной реализации. На четвертом этапе гипертекст реализуют в электронной форме.

На пятом этапе разрабатывают компьютерную поддержку: определяют математические действия в каждом конкретном случае поручаются компьютеру и в какой форме должен быть представлен его ответ.

Здесь же проектируют и реализуют интеллектуальное ядро (ИЯ); разрабатывают инструкции для пользователей по применению интеллектуального ядра ЭУ для решения математических задач (правила набора математических выражений и взаимодействия с ИЯ).

ИЯ целесообразно сделать так, чтобы его можно было заменять на более мощный компьютерный пакет типа DERIVE, Reduce, MuPAD, Maple V и т.п.

Теперь электронный учебник готов к дальнейшему совершенствованию (озвучиванию и визуализации) с помощью мультимедийных средств.

На шестом этапе изменяют способы объяснения отдельных понятий и утверждений и отбирают тексты для замены мультимедийными материалами.

На седьмом этапе разрабатывают тексты звукового сопровождения отдельных модулей с целью разгрузки экрана от текстовой информации и

использования слуховой памяти студента для облегчения понимания и запоминания изучаемого материала.

На восьмом этапе разработанные тексты звукового сопровождения записывают на диктофон и реализуют на компьютере.

На девятом этапе разрабатывают сценарии визуализации модулей для достижения наибольшей наглядности, максимальной разгрузки экрана от текстовой информации и использования эмоциональной памяти студента для облегчения понимания и запоминания изучаемого материала.

На десятом этапе производят визуализацию текстов, разработанных сценариев с использованием рисунков, графиков и, возможно, анимации.

На этом заканчивают разработку ЭУ и начинают его подготовку к эксплуатации. Следует отметить, что подготовка к эксплуатации ЭУ может предполагать некоторые коррекции его содержательной и мультимедийной компонент.

### **Содержание проектной части по направлению «Программные комплексы».**

Программные комплексы разрабатывают при решении прикладных задач из различных областей техники, экономики и т.д. При этом основные трудности после выполнения этапов постановки задачи и математической формализации заключаются в построении алгоритмов.

Алгоритм должен обладать следующими свойствами:

- Дискретность – алгоритм должен представлять собой последовательность отдельных шагов, четко отделенных друг от друга.

- Конечность – описания алгоритма должно содержать конечное число шагов.

- Конечность ввода – для работы алгоритма требуется ограниченное множество входных данных.

- Определенность (детерминированность) – каждый шаг алгоритма должен быть четко определен и понятен исполнителю алгоритма. При исполнении алгоритма не должно возникать неоднозначности, требующей волевых решений от исполнения алгоритма.

- Результативность (иногда это свойство называют выполнимостью) – алгоритм должен приводить к решению задачи за конечное число шагов. Но это возможно только, если входные данные алгоритма соответствуют его области определения.

- Эффективность – для решения задачи должны использоваться ограниченные ресурсы компьютера.

- Массовость – алгоритм должен быть пригоден для решения не одной задачи, а целого класса задач. Например, алгоритм для решения квадратных уравнений позволяет решить любое уравнение, а не только какое-то конкретное.

– Для того, чтобы построить качественный алгоритм необходимо выполнить следующие этапы:

- Постановка задачи.
- Построение модели.
- Разработка алгоритма.
- Проверка правильности алгоритма.
- Реализация.
- Анализ алгоритма и его сложности.
- Программная реализация алгоритма.
- Проверка программы.
- Составление документации.

Существует 4 основных способа построения алгоритмов:

- Природным языком (описываем что, как и когда надо сделать)
- С помощью графических схем
- Псевдокод (полуформализованное описание на условном алгоритмическом языке, который включает в себя, как элементы языка программирования, так и фразы естественного языка, арифметические действия и прочее).
- Программный.

После разработки алгоритма и доказательства его правильности приступают к реализации алгоритма. В целом, сложность этого этапа довольно высокая, поскольку не всякий шаг алгоритма может быть напрямую выражен в конструкциях некоторого языка программирования.

Иногда один из шагов алгоритма требует написания отдельной программы, в других случаях необходимо создать сложную структуру вспомогательных данных, исходя из выбранной модели (раздел 3 настоящего пособия).

Далее следует определение архитектурных представлений.

**Содержание проектной части по направлению «Разработка информационной системы (ИС)».**

В процессе проектирования строят структурную, функциональную и информационную модели информационной системы.

**Структурная модель ИС.** Структурную модель системы составляют на первом этапе построения системы управления организацией. При этом изучают и формализуют представление внешних воздействий, структуру взаимоотношений между элементами и их особенности (характерные параметры, стратегии, ограничения, критерии).

При анализе последствий внешних воздействий особое внимание уделяют характеристике среды функционирования и взаимоотношениям с другими объектами и системами (вышестоящими организациями, общественными структурами и т.д.) с содержанием двух типов

взаимоотношений координации: вертикальной, имеющей иерархическую структуру соподчиненности (субординации), и горизонтальной, соответствующей межличностным и межгрупповым отношениям.

Структурная модель организации строится по правилам символического изображения системы путем классификации элементов. Каждому элементу каждого класса соответствует определенный символ.

Наиболее важным моментом здесь являются определение значимости элементов системы на основании наличия между ними определенной совокупности отношений.

Такое ранжирование позволяет перевести качественное понимание значимости в количественную форму, соотнеся ее с соответствующей квалификацией элементов анализируемой или создаваемой структуры.

В качестве примера рассмотрим структурную модель системы, которой является компьютер.

Структурную модель системы еще называют структурной схемой. На структурной схеме отражается состав системы и ее внутренние связи. Наряду с термином «связь» нередко употребляют термин «отношение».

Наглядным способом описания структурной модели системы являются графы. На рисунке 6 в виде ориентированного графа приведена структурная модель компьютера.

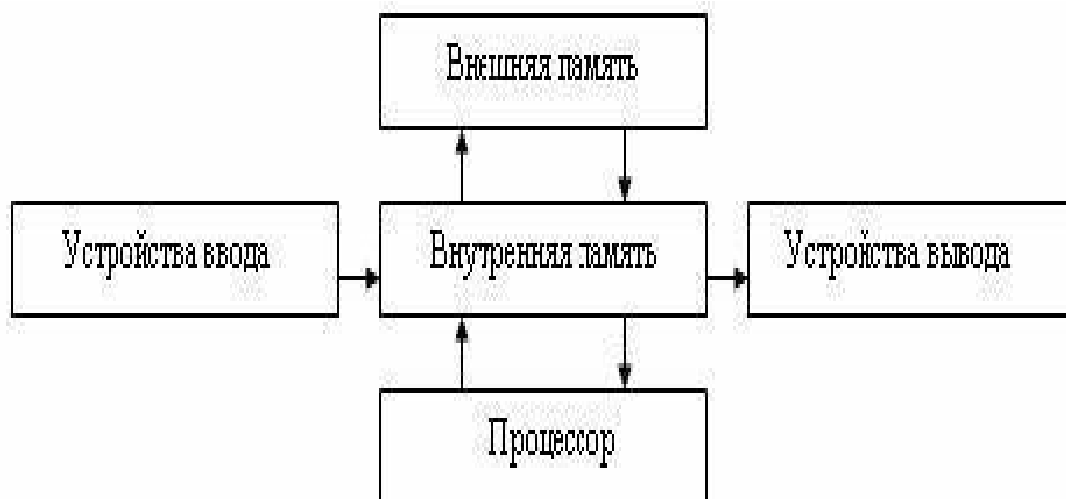


Рисунок 6 – Структурная модель компьютера

Здесь стрелки обозначают информационные связи между элементами системы.

Направление стрелок указывает на направление передачи информации.

Однако, если проектировщика интересуют связи по управлению, то граф-модель компьютера будет иметь вид (рисунок 7).



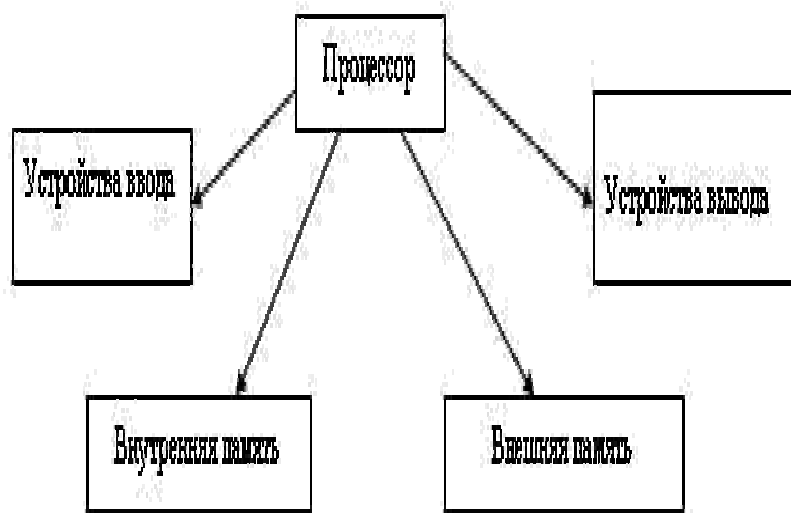


Рисунок 7 – Структурная модель компьютера

Здесь стрелка обозначает направление управляющего воздействия. Смысл схемы заключается в том, что процессор управляет работой всех остальных устройств компьютера.

Следовательно, структурная модель одной и той же системы может быть разной в зависимости от целей моделирования.

**Функциональная модель ИС.** Функциональную модель системы составляют в соответствии с разработанной на первом этапе структурной моделью по предварительному описанию целей функционирования и тех задач, решение которых находится в компетенции соответствующих элементов структуры.

Функциональная модель являются определяющей в составлении должностных инструкций элементов системы и их групп (компонент).

Функциональная модель – это набор регламентов бизнес процессов, подлежащих автоматизации в рамках разрабатываемой системы.

Обычно для функционального моделирования используют IDEF0 методологию, в соответствии с которой составляют диаграмму бизнес-процесса.

Диаграмма представляют собой совокупность функциональных блоков и интерфейсных дуг (входящей, исходящей или управляющей). Пример функциональной модели дан на рисунке 8.

Часто функциональную модель дополняют бизнес-процедурами, отражающими поведение должностных лиц в каждой конкретной ситуации при работе с системой.

В результате построения функциональной модели заказчик и исполнитель четко представляют последовательность действий при использовании информационной системы должностными лицами.

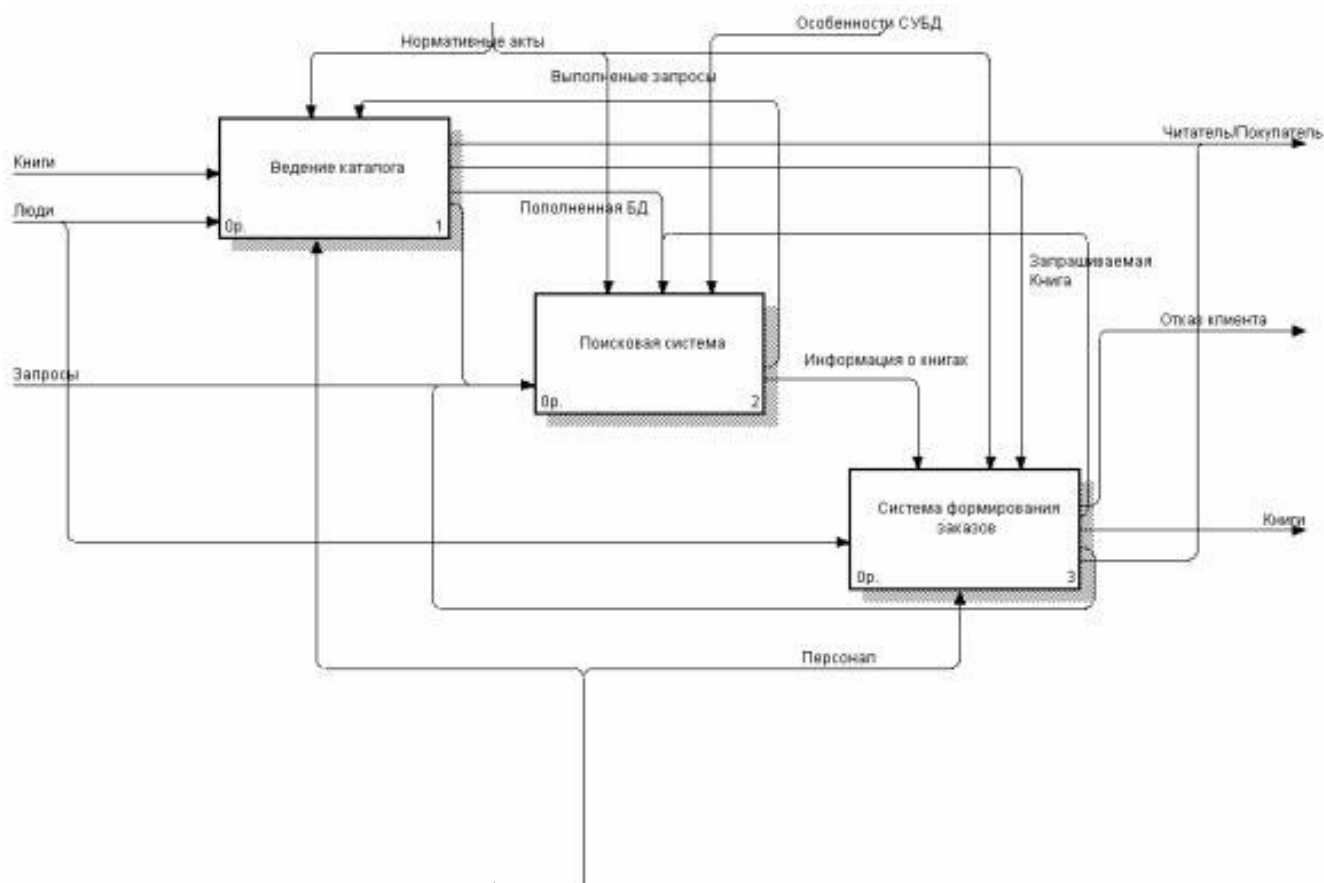


Рисунок 8 – Пример функциональной модели по технологии IDEF0

**Информационная модель ИС.** Постановка задачи создания информационной модели тесно увязана с целями моделирования. В самом общем приближении можно выделить следующие цели:

- хранение информации;
- возможность упорядочения данных по некоторым признакам;
- возможность создания различных критериев выбора данных;
- представление информации в удобном для пользователя виде.

Информационная модель — совокупность информации, характеризующая свойства и состояния объекта, процесса, явления, а также взаимосвязь с внешним миром. Ее представляют в виде схемы данных, являющейся структурным представлением движения информационных потоков (путь данных) с момента поступления входной информации до момента выдачи выходных форм.

Информационная модель включает в себя совокупность входных и выходных документов, файлов входной оперативной, постоянной, промежуточной и результирующей информации.

Методика разработки информационной модели предполагает моделирование:

- взаимосвязей входных, промежуточных и результирующих информационных потоков и функций предметной области (структурно-

функциональная диаграмма функциональная диаграмма или диаграмма потоков данных).

В описании информационной модели необходимо пояснить, на основе каких входных документов и нормативно-справочной информации выполняются функции по обработке данных и формируются конкретные выходные документы;

– данных информационной базы (диаграмма «сущность-связь» - инфологической модели и диаграмма взаимосвязей файлов – даталогической модели), необходимых для функционирования информационной системы.

Структуры потоков данных и определения их компонентов хранятся и анализируются в словаре данных.

Каждая логическая функция (процесс) может быть детализирована с помощью DFD нижнего уровня.

Когда дальнейшая детализация перестает быть полезной, переходят к выражению логики функции при помощи спецификации процесса.

Содержимое каждого хранилища сохраняют в словаре данных. Модель данных хранилища раскрывают посредством ER-диаграмм.

Заключительным этапом исследований организации является оценка возможности использования в системе управления новых информационных технологий и необходимого для этого технического и программного обеспечения, выявления потребностей в адаптации имеющихся инструментальных средств и разработке проблемно ориентированных пакетов прикладных программ.

**Определение архитектурных представлений.** Под *архитектурой программного обеспечения (ПО)* понимают набор его внутренних структур, которые состоят из *компонентов*, их связей и возможных взаимодействий между *компонентами*, а также доступных извне свойств этих *компонентов*.

Под *компонентом* понимают достаточно произвольный структурный элемент ПО, который можно выделить, определив интерфейс взаимодействия между ним и всем, что его окружает. Обычно при разработке ПО термин «компонент» имеет более узкий смысл как самая маленькая часть системы, которую можно включать или не включать в ее состав. Такой компонент также имеет определенный интерфейс и удовлетворяет некоторому набору правил, называемому компонентной моделью.

Использование архитектуры системы необходимо для решения следующих задач:

- Анализ альтернативных проектов системы.
- Планирование модификации системы, внесения изменений в ее организацию.
- Выработка критериев приемки системы при ее сдаче в эксплуатацию.

– Разработка документации по ее использованию и сопровождению, включая обучающие и маркетинговые материалы.

На рисунке 9 показана архитектура авиасимулятора, который должен моделировать определенные условия полета и создавать некоторые события, а именно:

- Скоростной и высотный режим полета, запас горючего, их изменения со временем.
- Модель самолета и ее характеристики по управляемости, возможным режимам полета и скорости реакции на различные команды.
- Погода за бортом и ее изменения со временем.
- Рельеф и другие особенности местности в текущий момент, их изменения со временем.
- Исходный и конечный пункты полета, расстояние и основные характеристики рельефа между ними.
- Исправность или неисправность элементов системы контроля полета и управления самолетом, показатели системы мониторинга и их изменение со временем.
- Наличие пролетающих вблизи курса самолета других самолетов, их геометрические и скоростные характеристики.
- Чрезвычайные ситуации, например, террористы на борту, нарушение герметичности корпуса, внезапные заболевания и «смерть» отдельных членов экипажа.

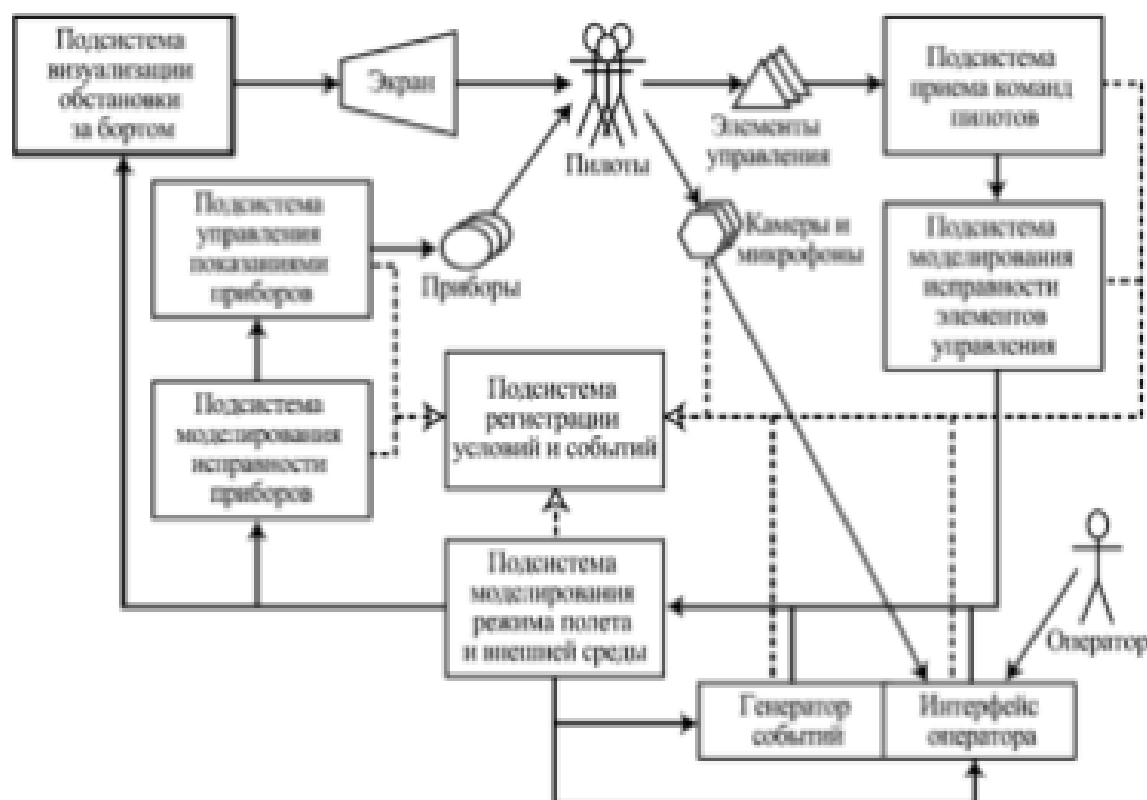


Рисунок 9 – Архитектура авиасимулятора

При проектировании архитектуры системы на основе требований, зафиксированных в виде вариантов использования, первые возможные шаги состоят в следующем:

#### 1.Выделение компонентов

Выбирают набор «основных» сценариев использования — наиболее существенных и выполняемых чаще других. Каждый сценарий использования системы представляют в виде последовательности обмена сообщениями между полученными компонентами. При возникновении дополнительных хорошо выделенных подзадач добавляют новые компоненты, и сценарии уточняют.

#### 2. Определение интерфейсов компонентов

Для каждого компонента в результате выделяют его интерфейс — набор сообщений, которые он принимает от других компонентов и посылает им. Рассматривают «неосновные» сценарии, которые так же разбивают на последовательности обмена сообщениями с использованием, по возможности, уже определенных интерфейсов. Если интерфейсы недостаточны, их расширяют. Если интерфейс компонента слишком велик, или компонент отвечает за слишком многое, его разбивают на более мелкие.

#### 3.Уточнение набора компонентов

Там, где это необходимо в силу требований эффективности, надежности или удобства сопровождения, несколько компонентов можно объединить в один либо разделить на несколько.

#### 4.Достижение нужных свойств.

На основе возможных сценариев использования системы выполняют анализ характеристик архитектуры и оценку ее пригодности для решения поставленных задач или сравнительный анализ нескольких архитектур. Это, так называемый, метод анализа архитектуры ПО (Software Architecture Analysis Method, SAAM).

Алгоритм метода:

- Определить набор сценариев действий пользователей или внешних систем на основе возможностей, которые планируются для реализации в системе или являются новыми. Чем полнее набор сценариев, тем выше качество анализа.

- Определить архитектуру (или несколько сравниваемых архитектур).

- Классифицировать сценарии. Для каждого сценария из набора следует определить, поддерживается ли он данной архитектурой или нужно вносить в нее изменения. Поддержка сценария означает, что лицо, заинтересованное в его выполнении, считает ее степень достаточной, а необходимые при этом действия удобными.

- Оценить сценарии. Определить, какие из них полностью поддерживаются рассматриваемыми архитектурами. Для каждого неподдерживаемого сценария надо определить необходимые изменения в

архитектуре. К ним относятся внесение новых компонентов, изменения в существующих, а также изменения связей и способов взаимодействия. При наличии возможности стоит оценить трудоемкость внесения таких изменений.

- Выявить взаимодействие сценариев.
- Оценить архитектуру в целом (или сравнить несколько заданных архитектур), используя оценки важности сценариев и степень их поддержки архитектурой.

Рассмотрим сравнительный анализ двух архитектур на примере индексатора — программы для построения индекса некоторого текста, то есть упорядоченного по алфавиту списка его слов без повторений.

1. Выделим следующие сценарии работы или модификации программы:

- Реализация работы индексатора в инкрементальном режиме, то есть чтении на входе одной фразы за другой и пополнение получаемого в процессе работы индекса.
- Игнорирование индексатором предлогов, союзов, местоимений, междометий, частиц и других служебных слов.
- Возможность обработки индексатором текстов, подаваемых ему на вход в виде архивов.
- Размещение в индексе только слов в основной грамматической форме — существительных в единственном числе и именительном падеже, глаголов в неопределенной форме и т.д.

2. Определим две возможных архитектуры индексатора для сравнительного анализа. В качестве первой архитектуры рассмотрим разбиение индексатора на два компонента.

Один компонент принимает на свой вход входной текст, полностью прочитывает его и выдает на выходе список слов, из которых он состоит. Второй компонент принимает на вход список слов, а на выходе выдает его упорядоченный вариант без повторений. Этот вариант архитектуры построен в стиле «каналы и фильтры» (рисунок 10).



Рисунок 10 - Архитектура индексатора в стиле «каналы и фильтры»

В дополнение к этим данным имеются следующие компоненты:

- Первый читает очередной символ на входе и передает его на обработку одному из остальных. Если это разделитель слов (пробел, табуляция, перевод строки), управление получает второй компонент. Если это буква — третий. Если входной текст кончается — четвертый.

– Второй компонент осуществляет ввод последнего слова — оно помещается в список перед тем местом, на которое указывает ссылка, после чего последнее слово становится пустым, а ссылка начинает указывать на первое слово в списке.

– Третий компонент добавляет прочитанную букву в конец последнего слова, после чего перемещает ссылку на следующее за полученным слово в списке.

– Четвертый компонент выдает полученный индекс на выход.

Эта архитектура построена в стиле «репозиторий» (рисунок 11).

Определим поддерживаемые сценарии из выделенного набора.

Первый сценарий прямо поддерживается второй архитектурой. Чтобы поддержать его в первой, необходимо внести изменения в оба компонента так, чтобы первый компонент мог пополнять промежуточный список, читая входной текст фразой за фразой, а второй — аналогичным способом пополнять результирующий упорядоченный список, вставляя туда поступающие ему на вход слова.



Рисунок 11 - Архитектура индексатора в стиле репозитория

Обе архитектуры не поддерживают второй сценарий. Для его поддержки в первой архитектуре необходимо изменить первый компонент или вставить после него дополнительный фильтр, отбрасывающий вспомогательные части речи. Для поддержки этого сценария второй архитектурой нужно ввести дополнительный компонент, который перехватывает буквы, выдаваемые модулем их обработки (соответственно, этот модуль больше не должен перемещать указатель по итоговому списку) и сигналы о конце слова от первого компонента, после чего он должен отсеивать служебные слова.

Третий сценарий также требует изменений в обеих архитектурах. Однако в обоих случаях эти изменения одинаковы — достаточно добавить дополнительный компонент, декодирующий архивы, если они подаются на вход.

Четвертый сценарий также не поддерживается обеими архитектурами. Требуемые им изменения аналогичны требованиям второго сценария, только в этом случае дополнительный компонент-фильтр должен еще и преобразовывать слова в их основную форму и только после этого пытаться добавить результат к итоговому индексу. Таким образом, требуется, как и во втором случае, изменить или добавить один компонент в первой архитектуре и изменить один и добавить новый — во второй.

В целом первая архитектура на предложенных сценариях выглядит лучше второй. Единственный ее недостаток — отсутствие возможности инкрементально поставлять данные на вход компонентам. Если его устранить, сделав компоненты способными потреблять данные постепенно, эта архитектура станет почти идеальным вариантом, поскольку она легко расширяется — для решения многих дополнительных задач потребуется только добавлять компоненты в общий конвейер.

Вторая архитектура, несмотря на выигрыш в инкрементальности, проигрывает в целом. Основная ее проблема — слишком специфически построенный компонент — обработчик букв. Необходимость изменить его в нескольких сценариях показывает, что нужно объединить обработчик букв и обработчик конца слов в единый компонент, выдающий слова целиком, после чего полученная архитектура не будет ничем уступать исправленной первой.

Для представления архитектуры, удобно использовать графические языки. На настоящий момент наиболее проработанным и широко используемым из них является UML. UML предлагает использовать для описания архитектуры 8 видов диаграмм: диаграммы классов, диаграммы объектов, диаграммы компонентов, диаграммы развертывания, диаграммы деятельности, сценариев, диаграммы взаимодействия и диаграммы состояний.

**Реализация разработки.** В основе третьей главы лежит описание реализации разработанного программного продукта. Рассмотрим структуру раздела:

- 1 Разработка программных модулей
- 2 Разработка интерфейса пользователя
- 3 Реализация программных интерфейсов
- 4 Тестирование программного продукта
- 5.Руководство пользователя (при необходимости программиста, системного администратора)
- 6 Руководство по установке

**Реализация программных модулей.** В данном разделе необходимо описать все программные модули, реализованные в ДР. Приступая созданию модуля необходимо иметь о нем четкое представление.

Под программным модулем понимается программа или функционально заверченный фрагмент программы, который характеризуют:



- обработку и возвращает один набор выходных данных;
- функциональная завершенность. Модуль выполняет набор определенных операций для реализации каждой функции, достаточных для завершения начатой обработки данных;
- логическая независимость. Результат работы данного фрагмента программы не зависит от работы других модулей;
- слабые информационные связи с другими модулями. Обмен данными между отдельными модулями должен быть минимален;
- размер и сложность модуля должны быть в разумных пределах.

В ДР в целях презентации должны присутствовать листинги (списки операторов на языке программирования) программных модулей. Для ясности листинги рекомендуется сопровождать комментариями.

Допускается размещение только модулей, созданных самим студентом. Те модули, которые сгенерированы средой программирования (Microsoft Visual Studio, Borland C++ Builder, Interface Builder и т. П.), размещать не рекомендуется, так как это не является признаком высокой квалификации студента в области программирования и создает обманчивое впечатление о размере проекта. Листинги могут присутствовать как в тексте проекта, так и в приложениях к нему. В любом случае они должны служить дополнением к излагаемому материалу и обязательно сопровождаться пояснениями и ссылками (если расположены в приложении).

**Разработка интерфейса пользователя.** Этот раздел ДР должен содержать описание интерфейса пользователя. Интерфейс пользователя или UI (User Interface) является неотъемлемой частью программы. С помощью UI пользователь «общается» с программой, производя ввод исходных данных и просматривая результаты. В зависимости от типа программного продукта и его назначения, UI может принимать различные формы, но при этом должен быть дружелюбным к пользователю. Этого можно достигнуть, если следовать следующим принципам:

- интерфейс должен быть легким для освоения и не создавать перед пользователем преграду, которую он должен будет преодолеть, чтобы приступить к работе;
- разработчик программного обеспечения не должен недооценивать умственные способности пользователей. Другими словами не стоит давать программе слишком большие полномочия и право указывать пользователю, что ему нужно делать. Тем самым пользователь загоняется в тесные рамки данного программного продукта, и ему навязывается определенный стиль работы. Например, излишняя «болтливость» (вывод лишних сообщений) программы или отказ выполнить естественную с точки зрения пользователя команду и заменить ее другой («ты должен сделать, так как я хочу, а не то, что тебе нужно»);

– выполняемое приложение не должно вызывать своими сообщениями у пользователя чувство недоумения или непонимания чего от него хотят. Например, при открытии файла, приложение корректно распознает его тип, но, тем, ни менее, запрашивает у пользователя формат файла. При этом пользователь сам может его не знать.

Интерфейс современного прикладного программного обеспечения, в основном, является оконным. При его разработке нужно стремиться к тому, чтобы основные элементы оконного интерфейса удовлетворяли следующим позициям:

– не использовать нестандартные элементы интерфейса. Например, командные кнопки не только с текстом, но и с рисунком из-за обычно невысокого качества графики выглядят непрофессионально. «Чем стандартнее компоненты, тем лучше и профессиональнее вид»;

– палитра инструментов должна быть небольшой, то есть не содержать много значков, а только самые используемые;

– одинаковое расстояние между элементами управления – признак профессионализма, так как способствует лучшему восприятию UI.

– использовать TabOrder («правильный порядок»). Курсор должен перемещаться к элементам управления на форме при нажатии клавиши Tab на клавиатуре компьютера в том порядке, в каком они визуальны расположены на форме;

– при выборе шрифтов, оставить их такими, какими они определены по умолчанию. Можно добиться того, чтобы смена пользователем стандартных шрифтов в графической оболочке операционной системы (ОС) отразилась и на UI приложения. Для этого, в настройках проекта приложения указать в свойствах шрифт соответствующие глобальные переменные. В Windows – это WindowText, MenuText и т. Д.;

– при выборе цветов, оставить их принятыми в ОС по умолчанию. Пусть программа выглядит так, как того хочет пользователь;

– программа должна хорошо управляться как с помощью «мыши», так и с помощью клавиатуры. Не должно быть функций, которые выполняются только с помощью «мыши», за исключением тех, которые могут выполняться только с помощью «мыши» (рисование в графическом редакторе и т. П.). Наиболее популярные операции нужно снабдить «горячими» клавишами для их быстрого вызова;

– в заголовке окна приложения с мультидокументным интерфейсом должен соблюдаться следующий порядок следования – вначале имя открытого файла, а затем имя приложения.

В пояснительной записке ДР описание интерфейса следует сопровождать рисунками его элементов (рисунок 12).

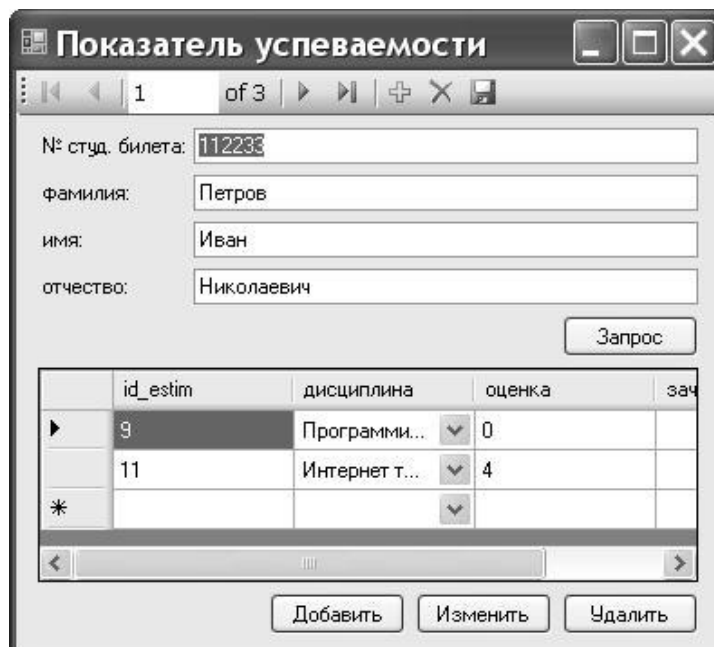


Рисунок 12 – Пользовательский интерфейс приложения «Показатель успеваемости студентов»

**Реализация программных интерфейсов.** Под программным интерфейсом понимают функциональность, которую некоторый программный компонент предоставляет другим программным компонентам. Различают два вида такой функциональности:

- функциональность, используемая при создании прикладных программ, которая называется интерфейсом программирования приложений или интерфейсом прикладного программирования (API). Более конкретно, это описание способа, который позволяет какому-либо фрагменту программного обеспечения обращаться к другой программе за получением сервиса. Этим сервисом может быть предоставление доступа к данным или выполнение конкретной функции;
- функциональность, используемая при создании системных компонентов, называемая интерфейсом системного программирования или SPI.

Кроме того, программные интерфейсы могут быть разноуровневыми, относясь либо к уровню программного кода (API и SPI), либо к уровню кода машинного (ABI и SBI).

При разработке прикладного программного обеспечения обычно приходится иметь дело с API. Примерами применения API могут служить динамические библиотеки (DLL) и базы данных. В частности, технология ODBC дает возможность программисту использовать стандартный интерфейс прикладного программирования для доступа к базам данных и, при этом, осуществляет перевод вызовов подпрограмм библиотеки API в команды конкретного сервера.

В ДР необходимо указать вид интерфейса, реализованного в программе и составляющие его программные модули.

Если использованы динамические библиотеки, то необходимо указать из каких модулей они состоят и обосновать их включение в библиотеки с точки зрения функциональности всего проекта. Например, так: «В приложении, для определения оптимальных параметров управления предприятием, используется численное решение системы линейных уравнений. Для этих целей был отдельно разработан программный модуль `slq.ch`. Для экономии размеров загрузочного модуля программы и, как следствие, ресурсов вычислительной системы, модуль `slq.ch` был включен в состав динамической сборки `lineq.dll`».

Если проект является базой данных, то нужно указать посредством какого интерфейса устанавливается связь приложения-клиента с данными. Примерами таких сервисов могут быть ODBC, OLE DB, ADO и ADO.NET.

**Тестирование программного продукта.** Испытание (тестирование) является обязательным заключительным этапом создания программы. Цель тестирования программы состоит в том, чтобы убедиться, что программа удовлетворяет следующим требованиям:

- решает ту задачу, для которой предназначена;
- обладает живучестью или робастностью (*robustness*), то есть выдает правильный результат при любых входных данных.

Данный раздел ДР должен содержать полный отчет о произведенных испытаниях программного обеспечения. Отчет должен содержать программу испытаний, наборы исходных данных для испытаний, результаты проведенных испытаний.

Испытания конкретного программного проекта начинаются с составления программы испытаний, которую можно представить в виде последовательности этапов, каждый из которых должен гарантировать получение верных результатов при правильных входных данных и выдачу сообщений об ошибках при ошибочных данных:

- Начальная стадия испытания. Начинается на этапе написания программы, на котором нужно решить вопрос о способе тестирования. Если данный вопрос неясен, программа должна быть переписана заново, либо разбита на модули. В противном случае программу вообще не удастся проверить до конца и рано или поздно при очередном прогоне она даст сбой.

- Проверка в нормальных условиях. Тестирование осуществляется на основе данных, характерных для реальных условий функционирования программы. Имеют место ограничения на область изменения данных, в которой программа должна осуществлять свою работоспособность. Проверка должна показать, что программа выдает правильные результаты для этих совокупностей данных. Например, при нахождении корней квадратного

уравнения  $ax^2+bx+c=0$  входными данными являются значения коэффициентов  $a$ ,  $b$  и  $c$ . Проверкой в нормальных условиях будут их значения 1, 0 и 0,25, соответственно.

– Проверка в экстремальных условиях. Следует сразу после второго этапа. Главная цель состоит в установлении того факта, что поля данных промежуточных результатов имеют размеры, достаточные для проведения требуемых вычислений. Тестовые данные на этом этапе включают граничные значения области изменения входных переменных и значения, приводящие к экстремальным значениям выходных переменных. Каждая программа характеризуется своими собственными экстремальными данными, которые должны подбираться программистом. Например, при испытаниях программы, выполняющей действие  $A+B+C+D$ , экстремальными значениями одноразрядных положительных переменных  $A$ ,  $B$ ,  $C$  и  $D$  будут 9, 9, 9 и 9, соответственно. В тоже время, эти значения, не являются таковыми, если выполняется действие  $[(A+B)/C]^D$ . Экстремальными в данном случае будут 9, 9, 1 и 9. Для сравнения:  $[(9+9)/1]^9=18^9$ , а  $[(9+9)/9]^9=2^9$ .

Наиболее часто используют следующие типы экстремальных испытаний:

– граничные испытания. Состоит в использовании экстремальных значений переменных в качестве тестовых данных. Данный вид испытаний зачастую предоставляет наилучшие возможности для выявления ошибок. Если программа работает правильно в экстремальных условиях, обычно это означает, что она будет нормально работать и в любой другой области значений переменных;

– нулевые примеры. Имитируют состояние данных, которое имеет место в реальных условиях эксплуатации программ. Для цифрового ввода, это – нули. Для символьных данных – цепочка пробелов. Для указателей – нулевые значения указателя;

– граничные объемы данных. Цель данного испытания состоит в том, чтобы установить, что произойдет с программой, если ей на обработку не поступит ни одного элемента данных или только один и сохранит ли она при этом свою работоспособность.

– Проверка в исключительных ситуациях. Ошибочно введенные данные могут вызвать ошибки времени выполнения либо могут быть восприняты программой как приемлемые, что в итоге приведет к неверным результатам. Цель – выяснить, что произойдет, если программе придется иметь дело со значениями переменной, на работу с которыми она не рассчитана. Например,  $a=0$  для программы, вычисляющей  $[-b+\sqrt{D}]/2a$ . Поэтому тестовыми данными должны быть входные значения переменных, которые лежат за пределами допустимой для конкретной программы области изменений.

В сформированных в ДР тестовых наборах данных должны быть учтены следующие рекомендации:

1. подбирать тестовые данные так, чтобы каждая команда программы была бы использована хотя бы один раз;
2. тестовые данные должны быть трех типов:
  - тестовые данные, создаваемые программистом. Используются в редко встречающихся ситуациях, и призваны для проверки общей работоспособности программы;
  - реальные модифицированные тестовые данные;
  - реальные тестовые данные в полном объеме.
3. каждый новый тест должен содержать определенный класс данных. Например, в программе начисления заработной платы за неделю учтены следующие виды отработанного времени:  $< 40$  часов;  $= 40$  часов и  $> 40$  часов. Для неё могут быть сформированы следующие классы тестовых данных, учитывающие указанные интервалы реальных данных, экстремальные значения отработанного времени и ошибочные значения:  $-1$ ;  $0$ ;  $35$ ;  $40$ ;  $50$ ;  $110$ .

После прогона каждого теста необходимо проверять правильность машинных результатов. Для этого существует несколько способов:

- вычисление результатов вручную;
- получение результатов из регистрационной книги, документации или таблиц;
- получение результатов из другой программы.

Испытания проводятся путем прогона каждого теста из набора тестов с последующей проверкой полученных машинных результатов. Например, результаты тестирования программы для нахождения корней квадратного уравнения можно представить в виде таблицы (Таблица 1):

Таблица 1 – Результаты тестирования программы

Коэффициенты			Машинный результат	Контрольный результат	Комментарий
1	1	2	1,00; -2,00	1; -2	начальный тест
1	0	0,25	(0,00; 0,50) (0,00; - 0,50)	$0+i1/2$ ; $0-i1/2$ ;	проверка в нормальных условиях
0	0	0	Любое число	Любое число	нулевой пример
0	2	1	-0,50	-1/2	экстремальное значение
0	0	2	Решений нет	Решений нет	исключительная ситуация
0	2	0	0,00	0	экстремальное значение
2	1	0	0,00; -0,50	0; -1/2	проверка в нормальных условиях
1	1	0,5	(-0,50; 0,50) (-0,50; -0,50)	$-1/2+i1/2$ ; $-1/2-i1/2$ ;	проверка в нормальных условиях
2	0	0	0,00; 0,00	0; 0	проверка в нормальных условиях

**Руководство пользователя (при необходимости программиста, системного администратора)**

Руководство пользователя — один из основных программных документов, основная задача которого состоит в обеспечении пользователям возможностью самостоятельно решать основные задачи, на которые нацелена программа.

Руководство пользователя содержит полное описание программы с точки зрения целевого применения последней, а именно:

- назначение программы;
- основные задачи и возможности;
- способ отражения предметной области в программе;
- пользовательский интерфейс программы;
- порядок решения основных пользовательских задач;
- все функции программы и порядок их применения;
- пользовательская настройка программы;
- проблемы при использовании и способы их решения.

При документировании небольших программ в руководство пользователя часто включают инструкции по установке, настройке, администрированию, обновлению и прочему обслуживанию программы.

Руководство пользователя составляют на основании ГОСТ 19.504-79. Руководство программиста разрабатывают в трех случаях:

- программный продукт по своему основному назначению является средой разработки или библиотекой (как Delphi или Qt);
- комплекс или программный продукт служит платформой для разработки программ или систем определенного типа (как 1С или Ахapta);
- программа распространяется вместе с исходным кодом или постоянно модифицируется самими разработчиками.

Очевидная задача руководства программиста — снабдить разработчика информацией, которой ему будет достаточно для создания на базе разработанного программного продукта собственных программ или систем.

Руководство программиста должно объяснять:

- С какими объектами программист имеет дело, где они находятся, сколько времени существуют и как они взаимодействуют между собой. Какие из них он создает сам, а какие предоставлены ему изначально средой, фреймворком, библиотекой.

– Какие еще средства разработки (кроме разработанного программного продукта) необходимы для того, чтобы создать приложение или систему. Например, если разработанный программный продукт — библиотека, программисту потребуются компилятор, какая-то среда разработки и прочий инструментарий.

– В какой среде функционирует приложение или система. Какими будут его минимальные требования к системе. Понадобятся ли для его запуска какие-

либо дополнительные программные средства: фреймворки, рантаймы, интерпретаторы.

– Что представляет собой минимальное работоспособное приложение или минимальная работоспособная система. Какие объекты и в какой последовательности необходимо создать, как соединить их друг с другом, чтобы приложение осуществило какой-то свой минимальный вывод.

– Как (по шагам) скомпилировать работоспособное приложение или развернуть работоспособную систему.

Кроме того руководство программиста должно содержать полные описания всех предусмотренных в программном продукте объектов. Если это функции, то должны быть приведены их синопсисы, если классы, то описания их интерфейсов и т. Д.

Если программный продукт предполагает использование оригинального языка программирования и снабжен собственным компилятором или интерпретатором, в руководство программиста необходимо включить его описание.

Структура руководства программиста:

1. Назначение и условия применения программы.
2. Характеристика программы.
3. Обращение к программе.
4. Входные и выходные данные.
5. Сообщения.

Руководство программиста может комплектоваться различными схемами, например, схемами базы данных, диаграммами классов, графами вызова

Руководство системного администратора — программный документ, предоставляющий специалисту информацию, необходимую для выполнения работы по установке программ и поддержанию работоспособности всего программного обеспечения, используемого в рамках тех или иных систем.

Руководство системного администратора — вспомогательный документ для прикладных программных продуктов и основной для серверных и системных, которые не имеют непосредственных пользователей.

В руководстве системного администратора должны быть изложены:

- назначение и область применения программы;
- состав программы, основные принципы ее функционирования;
- комплект поставки (если он не указан в отдельном документе);
- системные требования для программы или ее компонентов;
- предпочтительная очередность установки компонентов;
- процедура установки программы или каждого ее компонента;
- порядок обязательной первоначальной настройки программы;
- способы интеграции установленных копий компонентов между собой;



– интегрирование программы со сторонним ПО, например, с сервером БД;

- способы и периодичность контроля правильности работы программы;
- порядок текущего обслуживания работающих копий программы;
- порядок решения всевозможных вспомогательных задач;
- аварийные ситуации и способы их устранения.

Дополнительно в руководстве системного администратора можно описать:

- пользовательский интерфейс административной консоли;
- утилиты командной строки и синтаксис их запуска;
- конфигурационные файлы и правила их написания;
- язык для составления управляющих скриптов.

Структура руководства системного программиста имеет вид:

1. Общие сведения о программе.
2. Архитектура и принципы функционирования.
3. Системные требования.
4. Установка программы.
5. Административная консоль и работа с ней.
6. Файл конфигурации. Составление и правка.
7. Обязательная начальная настройка программы.
8. Проверка правильности функционирования программы.
9. Мероприятия по текущему обслуживанию программы.
10. Оптимизация работы программы.
11. Аварийные ситуации и способы их устранения.

**Руководство по установке.** Этот раздел ДР должен содержать подробное описание действий по установке программного продукта и сообщений, которые при этом могут быть получены.

Как известно, выделяют четыре вида поставки программного обеспечения:

- набор файлов для обычного копирования на жёсткий диск;
- дистрибутив (с инсталлятором);
- инсталлятор;
- менеджеры пакетов и инсталляционные пакеты.

При наборе файлов программу можно считать установленной после копирования их со сменного носителя на жёсткий диск.

Дистрибутив имеет в своём наборе собственную программу-установщик, которая выполняет фактическое копирование файлов. Она вносит в файлы конфигурации необходимые значения параметров и создаёт нужные для работы базы данных, файлы и т. П. Применение дистрибутивов в качестве способа поставки программного обеспечения наиболее полезно и оправдано в случае установки комплексных программный продуктов с большим

количеством внутренних зависимостей и использованием недостаточно распространённых динамических библиотек. Также использование дистрибутивов дает значительную выгоду при поставке приложений, экспортируемых с одной операционной системы в другую. В любом дистрибутиве можно выделить отдельную часть (инсталлятор). Инсталлятор фактически и занимается установкой программного продукта, его настройкой, проверкой корректности установки, а также деинсталляцией в случае необходимости. Таким образом, дистрибутив логически можно разделить на две составляющие: установочные файлы и инсталлятор.

Также существует вариант поставки программного обеспечения в виде одного лишь инсталлятора. В этом случае часть или все установочные файлы берутся из другого места, например, из Интернета. Из преимуществ подобной схемы установки ПО можно выделить отсутствие необходимости загружать из Сети ненужные компоненты. То есть, в самом начале установки необходимо определить, что нужно устанавливать, а что нет. После чего инсталлятор загрузит все необходимые файлы и автоматически установит и сконфигурирует программный продукт

Четвёртый способ распространения программного обеспечения— установочные пакеты. В них имеются только установочные файлы в виде архива, содержащего в себе файлы продукта и инструкции для инсталлятора. К достоинствам данного метода относится возможность установки программного обеспечения в неинтерактивном режиме (quiet mode), которая важна для системных администраторов, обслуживающих достаточно объёмный парк машин, где возможность устанавливать ПО на компьютеры при помощи скриптовых техник не просто важна, а крайне необходима.

**Мероприятия по охране труда, безопасности жизнедеятельности.** Под охраной труда понимается система сохранения жизни и здоровья работников в процессе трудовой деятельности, включающая правовые, социально-экономические, организационно-технические, санитарно-гигиенические мероприятия (ст. 1 ФЗ «Об основах охраны труда в РФ»). Конституция РФ в главе «Основы конституционного строя» провозглашает, что в Российской Федерации охраняется труд и здоровье людей.

Поскольку пользователям программного обеспечения приходится проводить много времени за компьютером, охрана труда имеет для них актуальное значение и по этой причине данный раздел должен содержать основные сведения о мероприятиях по охране труда, касающиеся данной сферы трудовой деятельности.

В ДР должны быть раскрыты меры, направленные на сохранение здоровья и повышение трудоспособности программистов, а именно:

Поиск удобного рабочего положения

– выбор сиденья;

- регулировка высоты сиденья и расположения его спинки;
- расположение предплечий пользователя;
- расположение бедер;
- расположение нижнего отдела спины;
- расположение ступней;
- обеспечение свободного пространства под рабочим столом. Забота о зрении:
  - установка монитора компьютера с целью избежать бликов и отражения света от световых приборов или из окон;
  - оптимальное расположение верхней части экрана монитора относительно уровня глаз;
  - регулировка экрана во избежание отсвечивания;
  - обеспечение оптимального расстояния от экрана дисплея до глаз пользователя;
  - обеспечение отдыха глазам;
  - максимальная продолжительность работы за компьютером;
  - способы расслабления глазных мышц;
  - средства борьбы с пересыханием глаз;
  - подбор оптимального освещения помещения;
  - обеспечение минимального отражения поверхности рабочего стола.
  - уход за экраном дисплея (грязная поверхность затрудняет восприятие изображения и приводит к перенапряжению зрения и к быстрой утомляемости);
- оптимальная настройка параметров дисплея (улучшает восприятие изображения и снимает излишнее напряжение зрения, замедляя утомляемость).

## **2.5 Технические требования к оформлению ДР**

Дипломная работа представляет собой достаточно объемный документ, имеющий сложную структуру. Не только содержание, но и форма его представления должна соответствовать определенным требованиям. Степень соответствия ДР требованиям к ее оформлению определяет научный консультант – сотрудник отделения, на которого возложена эта обязанность.

Рассмотрим более детально правила оформления основных структурных элементов дипломной работы приведены в приложении.

В настоящих указаниях использованы ссылки на следующие документы:

ГОСТ 7.32-2017 СИБИД. Отчет о научно-исследовательской работе. Структура и правила оформления (с поправкой);

ГОСТ Р 7.0.5-2008 Система стандартов по информации, библиотечному и издательскому делу. Библиографическая ссылка. Общие требования и правила составления.

Весь текст работы печатается на белой стандартной бумаге на одной стороне листа формата А4; текст печатается шрифтом TimesNewRoman (черный) через 1,5 интервал, в таблицах интервал должен быть 1,0. Размер шрифта 14, в таблицах – не менее 12.

Размер левого поля (для подшивки) – 30 мм, правого – 15 мм, верхнего и нижнего – 20 мм; количество знаков в строке 60-70, количество строк 28-30.

Отступ абзаца 1,25 см, автоматическая расстановка переносов запрещена, выравнивание текста по ширине.

Страницы нумеруются арабскими цифрами по центру в нижней части листа без точки. Нумерация – сквозная, начинается с титульного листа. Титульный лист включается в общую нумерацию, но не нумеруется.

В содержании указываются **«ВВЕДЕНИЕ»**, номера и названия разделов, подразделов, пунктов, **«ЗАКЛЮЧЕНИЕ»**, **«СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ»** и **«ПРИЛОЖЕНИЯ»**, а также страницы, на которых эти названия размещены. Промежутки от последней буквы названия рубрики до номера страницы заполняют отточием. Слово «стр.» не пишут. Рекомендуем авто-собираемое содержание.

Текст работы должен быть кратким, четким и не допускать различных толкований.

Сокращения слов в тексте не допускаются, кроме общепринятых: *т.е.* (то есть); делается после перечисления: *и т.д.* (и так далее), *и т.п.* (и тому подобное), *и др.* (и другие), *и пр.* (и прочие); при ссылках: *см.* (смотри); при обозначении цифрами годов: *г.* (год), *гг.* (годы). Вместе с тем не допускается внутри предложения сокращение слов: *и другие, и тому подобное, и прочие, например, так называемый.*

Текст основной части работы делят на разделы, подразделы, пункты и подпункты.

Разделы должны иметь сквозную нумерацию в пределах всей работы и обозначаться арабскими цифрами. Каждый раздел следует начинать с нового листа (страницы). После номера раздела точку не ставят.

Подразделы следует нумеровать арабскими цифрами в пределах каждого раздела. Номер подраздела состоит из номера раздела, в который входит данный подраздел, и порядкового номера этого подраздела, разделенных точкой (например: первый подраздел второго раздела – 2.1). После номера подраздела точку не ставят.

Пункты нумеруют в пределах каждого подраздела. Номер пункта состоит из номера раздела, подраздела и пункта, разделенных точками (например:

второй пункт первого подраздела первого раздела – 1.1.2). После номера пункта точку не ставят.

Все структурные элементы работы выделяться **полужирным** начертанием. Разделы, подразделы, пункты и подпункты должны иметь заголовки, отражающие их содержание. Заголовки разделов, подразделов, пунктов печатаются с абзаца. «СОДЕРЖАНИЕ», «ВВЕДЕНИЕ», заголовки разделов работы, «СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ», «ЗАКЛЮЧЕНИЕ» печатаются прописными буквами. Заголовки подразделов, пунктов и подпунктов печатаются строчными буквами (кроме первой прописной). Подчеркивать заголовки и переносить в заголовках слова не допускается. В конце заголовка точка не ставится. Если заголовок состоит из двух предложений, их разделяют точкой.

Слова, напечатанные на отдельной странице («СОДЕРЖАНИЕ», «ВВЕДЕНИЕ», «СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ», «ЗАКЛЮЧЕНИЕ»), служат заголовками соответствующих структурных частей работы, не нумеруются и выравниваются по центру.

Расстояние между заголовками раздела и подраздела должно быть 1 интервала, заголовком и текстом – 1 интервал, между текстом и заголовком – 1 интервала.

*Таблицы.* Цифровой материал рекомендуется помещать в работе в виде таблиц. Таблицу следует располагать в работе непосредственно после текста, в котором упоминается впервые, или на следующей странице. Таблицы слева, справа, сверху и снизу ограничиваются линиями. На все таблицы должны быть даны ссылки в тексте.

Таблицы должны быть пронумерованы арабскими цифрами. Нумерация – сквозная в пределах всей работы. Допускается нумеровать таблицы в пределах раздела. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой. Слово «Таблица» и заголовок начинаются с прописной буквы без абзаца, точка в конце заголовка не ставится.

Заголовки граф таблицы должны начинаться с прописных букв, подзаголовки ее строчных, если последние подчиняются заголовку. Заголовки граф указываются в единственном числе. Графу «№ п/п» в таблицу включать не следует.

Таблицу следует размещать так, чтобы можно было читать ее без поворота работы, если такое размещение невозможно, таблицу располагают так, чтобы ее можно было читать, поворачивая работу по часовой стрелке.

При переносе таблицы шапку таблицы повторяют, с левой стороны таблицы и над ней размещают слова «Продолжение таблицы» с указанием ее номера. Если шапка таблицы велика, допускается ее не повторять: в этом

случае следует пронумеровать графы и повторить их нумерацию на следующей странице. Заголовок таблицы не повторяют.

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

Уравнения и формулы следует выделять из текста в отдельную строку. Выше и ниже каждой формулы или уравнения должно быть оставлено не менее одной свободной строки. Если уравнение не уместится в одну строку, оно должно быть перенесено.

Формулы и уравнения в работе следует нумеровать порядковой нумерацией в пределах всей работы арабскими цифрами в круглых скобках в крайнем правом положении напротив формулы. Допускается нумерация формул в пределах раздела. Если в работе только одна формула или уравнение, то их не нумеруют.

*Ссылки.* При ссылке на источник после упоминания о нем в тексте работы проставляют номер (в квадратных скобках), под которым он значится в библиографическом списке. В необходимых случаях (обычно при использовании цифровых данных или цитаты) указываются и страницы, на которых помещаются используемые данные, например: [12]. Ссылки на таблицы, рисунки, приложения берутся в круглые скобки. При ссылках следует писать: «в соответствии с данными таблицы 5», (таблица 5), «по данным рисунка 3» (рисунок 3), «в соответствии с приложением А», (приложение А), «... по формуле (3)».

*Рисунки.* Все иллюстрации (графики, схемы, диаграммы, чертежи, фотографии и пр.), помещенные в работе с целью придания излагаемому тексту ясности и конкретности, именуются рисунками. Каждый рисунок должен иметь название, которое помещают под ним. Рисунки нумеруются последовательно в пределах всей работы арабскими цифрами. Допускается нумерация в пределах раздела.

*Список использованных источников,* имеющих порядковую нумерацию, приводится автором в конце работы.

Список должен быть составлен в соответствии с указанием источника по тексту.

Сведения об источниках следует нумеровать арабскими цифрами и печатать с абзаца. Библиография оформляется на том языке, на котором опубликован или написан сам источник. Сведения об источниках должны содержать фамилию и инициалы автора (если таковой имеется); полное название книги, после косой черты – данные о редакторе (если книга написана группой авторов); после тире – название города, в котором издана книга; после

двоеточия – название издательства (без кавычек), после запятой – год издания, а также объем. Для некоторых городов приняты специальные сокращения: М. (Москва), СПб. (Санкт-Петербург).

При использовании статьи (главы, раздела) из книги или периодического издания необходимо указать фамилию, инициалы автора (если фамилия автора указана на титульном листе книги), название статьи (главы, раздела), после двух косых линий – название издания, где она помещена; после тире – год; после тире – номер, а также номера страниц, на которых находится данная статья.

*Образцы оформления нормативных документов:*

1. Конституция Российской Федерации. Принята на всенародном голосовании 12.12.93. – М.: ООО «Фирма «Издательство АСТ»», 2017. – 64 с.
2. Трудовой кодекс Российской Федерации. Принят Государственной Думой 21.12.01. Одобрен Советом Федерации 26.12.01. – М.: ИКФ «ЭКМОС», 2015. – 207 с.

*Примечание. Если в нормативный документ вносились изменения, дополнения, то в скобках перед источником указывается последняя редакция.*

*Образец оформления книги одного автора:*

Александр М. Excel 2016: профессиональное программирование на VBA: учебник / М. Александр. - М.: Диалектика, 2018. - 784 с.

*Образец оформления книги двух авторов:*

Фролов А.В. Локальные сети персональных компьютеров. Работа с сервером Novell NetWare: учебное пособие / А.В. Фролов, Г.В. Фролов. - М.: Диалог-Мифи, 2017. - 168 с.

*Образец оформления книги трех авторов:*

Дорогов В.Г. Основы программирования на языке С: Учебное пособие / В.Г. Дорогов, Е.Г. Дорогова, Л.Г. Гагарин. - М.: ИД Форум, НИЦ Инфра-М, 2018. - 224 с.

*Образец оформления книги авторского коллектива «Под редакцией»:*

Баженова И.Ю. Языки программирования: Учебник для студентов / И.Ю. Баженова; Под ред. В.А. Сухомлин. — М.: ИЦ Академия, 2018. — 368 с.

*Образец оформления журнальной статьи:*

Тимохин П.Ю. Метод извлечения поверхностей уровня на гри с помощью программируемой тесселяции // Программирование. – 2020. – № 3. – С. 66–72.

*Образцы оформления статей из сборника научных статей:*

Спиридонов, Э. А. Разработка системы качества вуза [Текст] / Э. А. Спиридонов, Е. В. Савина // Стандарты и качество. – 2012. – № 4. – С. 46 – 47.

*Образец оформления газетной статьи:*

Спиридонов, Э. А. Разработка системы качества вуза // Стандарты и качество. – 2012. – № 4. – 25 марта.

*Образец оформления главы из книги:*

Пшуков, Ю. Г. О нормировании качества жидких электролитов / Ю. Г. Пшуков // Научно-технический прогресс и оптимизация технологических процессов: тез. докл. Всесоюз. науч. конф. , 21 – 22 мая 1987 г. – Тамбов, 1987. – Гл. 14– С. 200-205.

*Образец оформления автореферата:*

Шамгунов Н.Н.Разработка методов проектирования и реализации поведения программных систем на основе автоматного подхода: автореф. дис. ... канд. техн. наук. – М., 2017. – 19 с.

*Образец оформления диссертации:*

Дробышевский М. Д. Методы и программные средства моделирования и генерации сложных сетей с сохранением графовых свойств: дис. ф.-м. н., . – М., 2019. – 451 с.

*Образец оформления статьи Интернет-ресурса:*

Тимохин П.Ю.Калякин А.А. Многоязычное программирование: создание систем с использованием нескольких языков // [Электронный ресурс]. – Режим доступа: [http:// https://infostart.ru/public/975701/](http://https://infostart.ru/public/975701/) (дата обращения 24.04.2020 г.)

**ПРИЛОЖЕНИЯ** оформляют как продолжение содержания работы на последующих ее страницах, располагая их в порядке появления ссылок в работе. Каждое приложение следует начинать с нового листа (страницы) с указанием наверху по центру страницы слова «Приложение», его обозначения и степени. Приложение должно иметь содержательный заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой. Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключение букв Ё, З, Й, О, Ч, Ъ, Ы, Ь.



### **3 Подготовка и защита дипломной работы**

#### **3.1 Получение отзыва и рецензии на дипломную работу**

По завершению дипломная работа представляется на отзыв руководителю и рецензирование.

##### **3.1.1 Отзыв руководителя**

После получения окончательного варианта дипломной работы руководитель в течение 3 дней составляет письменный отзыв.

В отзыве должны быть отражены следующие моменты:

- актуальность темы;
- основные проблемы, рассмотренные в ДР;
- наиболее интересно изложенные вопросы;
- научная и практическая значимость исследования;
- степень самостоятельности при написании дипломной работы, уровень теоретической подготовки студента, его знание основных концепций и научной литературы по избранной теме;
- использованные методы и приемы анализа;
- обоснованность выводов;
- грамотность изложения материала;
- иллюстративность работы.

Особое внимание обращается на имеющиеся в работе и отмеченные ранее недостатки, не устраненные. Руководитель мотивирует возможность или нецелесообразность представления дипломной работы к защите. При этом руководитель не выставляет оценку ДР, а только дает ей качественную характеристику и рекомендует или не рекомендует ее к защите. Само содержание отзыва предполагает обоснованное мнение руководителя о качестве дипломной работы .

В отзыве может быть указано отношение студента к работе, его дисциплинированность или, наоборот, безответственность, регулярность обращения к руководителю. Отзыв руководителя допускается в рукописном виде. Форма отзыва и пример его оформления приведены в ПРИЛОЖЕНИИ Т.

##### **3.1.2 Рецензирование дипломной работы**

С целью получения дополнительной независимой оценки проводится рецензирование дипломной работы.

В качестве рецензентов привлекаются специалисты it отдела организации, системные администраторы, программисты..

Оформленная Дипломная работа вместе с отзывом руководителя предоставляется студентом рецензенту не позднее, чем за 7 дней до предварительной защиты. Рецензент должен составить письменную рецензию в течение 3 дней.

Содержание рецензии излагается в свободной форме с выделением следующих основных моментов:

- актуальность темы дипломной работы;
- основные вопросы (проблемы), рассмотренные в работе для наиболее полного раскрытия темы, при этом можно отметить наиболее удачно изложенные положения, новые выводы и предложения, сделанные автором;
- информационная основа, на которой базируется исследование, отметив ее полноту или недостаточность;
- способности автора работы к рациональному сбору, комплектованию, группировке и анализу экономической информации к проведению аналитических исследований в области экономической безопасности возможности делать обоснованные выводы и предложения на основании информации собранной в период прохождения преддипломной практики, а также общеэкономической информации, собранной в период прохождения практики а также общеэкономической информации о состоянии современной экономики и финансов и перспектив их развития;
- недостатки присущие дипломной работе (неполнота изложения некоторых вопросов, замеченные ошибки в структуре и содержании работы неверно сделанные или недостаточно обоснованные выводы и предложения);
- оценка всей работы с учетом ее соответствия (полного или частичного) требованиям, предъявляемым к ДР. Если по каким-то параметрам такого соответствия нет, то следует обратить на них внимание и показать их влияние на общую оценку работы.

Рецензия может быть отпечатана на бланке организации, в которой работает рецензент или на обычном листе бумаги, подписана и заверена печатью. Рецензия подписывается рецензентом с указанием его ученой степени, ученого звания, должности и места работы.

### **3.2 Прохождение предварительной защиты дипломной работы**

Предварительная защита дипломной работы направлена на повышение качества работы студента, устранения отмеченных замечаний по оформлению и содержанию. Предварительная защита дипломной работы проводится председателями ПЦК в соответствии с графиком. Для этого председатель ПЦК утверждает состав комиссии из профессорско-преподавательского состава

колледжа и сроки предварительной защиты, которые доводятся до сведения студентов Академического колледжа.

На предзащите в обязательном порядке должны присутствовать обучающиеся, их руководители и комиссия по предзащите, назначаемая председателем ПЦК и заведующим отделением. Число членов комиссии по предзащите должно быть не менее 2-х. Председателем комиссии по предзащите является председатель ПЦК. Результаты предзащиты оформляются протоколом комиссии. Для прохождения предварительной защиты Обучающийся должен предоставить дипломную работу (не сброшюрованную), оформленную в установленном порядке, а также отзыв руководителя, рецензию, подготовить доклад и иллюстрационный материал (1 экземпляр) или презентацию в электронном виде.

По итогам предварительной защиты председатель ПЦК на основе предоставленных преподавателями-членами комиссии сведений выносит решение о возможности получения допуска соответствующих дипломных работ к защите или необходимости их доработки.

В случае, если Обучающийся не явился на заседание комиссии по предзащите по неуважительной причине или по результатам предзащиты получил отрицательное решение комиссии по предзащите, он не допускается к защите дипломной работы. В случае если Обучающийся не явился на заседание комиссии по предзащите по уважительной причине, подтвержденной документально, ПЦК дополнительно назначаются сроки проведения предзащиты для этого студента.

### **3.3 Подготовка и защита дипломной работы**

#### **3.3.1 Подготовка дипломной работы**

Обучающийся, получив положительный отзыв о ДР от руководителя, рецензию внешнего или внутреннего рецензента и разрешение председателя ПЦК о допуске к защите, должен подготовить доклад. В нем следует отразить, чем Обучающийся руководствовался при выборе и изучении темы, что является предметом исследования, какие методы в нем использованы, какие новые результаты достигнуты, что сделано лично автором. Содержание доклада определяется студентом совместно с руководителем.

Председатель ПЦК на основании внешней рецензии и отзыва руководителя дает заключение о допуске выпускника к защите дипломной работы в ГЭК (приложение Ф) и делает соответствующую запись на титульном листе дипломной работы. Решение о допуске к защите принимается на основании следующих формальных признаков:

- соответствие заголовка дипломной работы теме, закреплённой за студентом, приказом ректора Академии;

- содержание работы раскрывает содержание утвержденной темы;
- работа оформлена в соответствии с настоящими методическими указаниями;
- имеется положительный отзыв руководителя;
- имеется рецензия.

Если же председатель ПЦК считает невозможным допустить выпускника к защите дипломной работы, этот вопрос рассматривается на заседании предметно-цикловой комиссии в присутствии руководителя и выпускника.

Подготовив дипломную работу, Обучающийся-выпускник готовится к ее защите. Подготовка к защите представляет собой важный этап, поскольку следует не только написать высококачественную работу, но и квалифицированно ее защитить. Высокая оценка, данная руководителем и рецензентом, может быть снижена комиссией из-за плохой защиты.

*Для успешной защиты дипломной работы необходимо:*

1. Подготовить доклад, в котором четко излагаются основные результаты дипломной работы .

Доклад может быть подготовлен письменно и использован во время защиты, но выступать на защите следует свободно и уверенно, излагая содержание доклада «своими словами», не зачитывая текст, но лишь опираясь на него глазами. Выступление на защите не должно превышать 10 минут.

В своем докладе выпускник должен:

- кратко раскрыть актуальность и цель исследования, на базе каких материалов оно было проведено;
- раскрыть основные выводы и предложения по результатам исследования;
- продемонстрировать научную, научно-практическую или практическую новизну работы.

Конкретное содержание доклада определяется студентом совместно с руководителем.

Доклад не должен быть перегружен цифровыми данными, они приводятся только в том случае, если они необходимы для доказательства или иллюстрации того или иного вывода.

2. Подготовить иллюстративный материал, который используется во время защиты для большей наглядности. При этом можно применять как технические средства, так и подготовить иллюстративный раздаточный материал для членов ГЭК.

Есть 2 способа представления иллюстративного материала.

*Первый* – распечатка комплекта таблиц, схем, диаграмм и др. наглядной информации, которая отражает основные самостоятельные разработки и задача ее членам ГЭК при защите. Раздаточный материал служит для пояснения содержания работы при ее защите. В раздаточный материал следует

выносить информацию, которая используется в докладе для доказательства обоснованности принятых автором решений и выводов – формулы, таблицы, диаграммы, графики, схемы и т. п.

По содержанию в раздаточном материале обычно повторяют отдельные материалы, помещаемые в основной части работы. Содержание листов раздаточного материала должно быть предельно лаконичным.

Каждый лист раздаточного материала должен иметь содержательный заголовок, который размещают по центру.

Все листы раздаточного материала должны быть выполнены в одном стиле.

Раздаточный (иллюстрационный) материал брошюруется. К раздаточному материалу выполняется титульный лист (приложение ).

*Второй* – подготовка презентации своей работы с использованием компьютерной программы Microsoft Power Point. Компьютерная презентация (КП) дает ряд преимуществ перед обычной бумажно - плакатной. Для полного использования программы подготовки КП необходимо хорошо знать ее особенности. Компьютерная презентация позволяет использовать ее выпускнику как легальную шпаргалку, а с другой стороны, позволяет члену ГЭК одновременно изучать дипломную работу и контролировать выступление выпускника. Поэтому желательно сопровождать выступление презентацией с использованием 15-20 слайдов.

Основными принципами при составлении КП являются лаконичность, ясность, уместность, сдержанность, наглядность (подчеркивание ключевых моментов), запоминаемость (разумное использование ярких эффектов).

Основное требование – каждый слайд должен иметь заголовок, количество слов в слайде не должно превышать 20.

Презентация легко поможет сделать доклад, но она не должна его заменить. Желательно подготовить к каждому слайду заметки по докладу, которые можно распечатать и использовать при подготовке и, в крайнем случае, на самой презентации. Можно также печатать некоторые ключевые слайды в качестве раздаточного материала.

### **3.3.2 Защита дипломной работы**

Защита дипломной работы происходит на открытом заседании Государственной экзаменационной комиссии (ГЭК), на которой могут присутствовать, задавать вопросы и обсуждать дипломную работу все желающие.

График защиты помещается на стенде отделения. Явка студента на защиту в установленный срок строго обязательна.

Задачей ГЭК является определение уровня теоретической подготовки студента, его подготовленности к профессиональной деятельности и принятие решения о возможности выдачи студенту соответствующей квалификации.

Задачей студента при защите является не пересказ того, что изложено в литературе, а изложение и обоснование того, что сделано им самим при изучении и раскрытии темы исследования.

На заседание ГЭК представляются следующие документы:

- сводная ведомость итоговых оценок по учебным дисциплинам, полученных студентом за весь период обучения;

- Дипломная работа с отзывами руководителя и рецензента;

На защиту дипломной работы отводится до одного академического часа на одного обучающегося.

Процесс защиты строится следующим образом:

1. Секретарь ГЭК оглашает тему дипломной работы и предоставляет выпускнику слово для выступления.

2. Доклад (не более 10-15 минут).

Доклад автора дипломной работы должен носить презентационный характер. Для этого необходимо подготовить иллюстрационный материал к дипломной работе и раздать его перед началом защиты своей работы каждому члену ГЭК.

Доклад должен быть увязан с иллюстративным материалом. В ходе доклада обязательна ссылка на все демонстрационные листы, представленные ГЭК.

3. В процессе защиты члены ГЭК имеют возможность ознакомиться с содержанием дипломной работы. После выступления выпускника члены комиссии и другие присутствующие имеют право задавать вопросы по разрабатываемой теме. Вопросы фиксируются в протоколе заседания ГЭК.

Обучающийся должен быть готов к ответу на любой вопрос в пределах изучаемых в Академическом колледже дисциплин по своей специальности. При ответах на вопросы Обучающийся имеет право пользоваться ДР.

4. После ответов на вопросы слово предоставляется руководителю (если он присутствует на защите), либо зачитывается отзыв руководителя на дипломную работу. Также вслух зачитывается рецензия на дипломную работу.

5. После оглашения отзыва и рецензии предоставляется время выпускнику для заключительного слова. В нем выпускник аргументирует свое согласие или несогласие с высказанными замечаниями и благодарит государственную комиссию.

6. После защиты выпускник остается в зале заседания ГЭК и ждет окончания процесса защиты дипломных работ. При этом не разрешается входить и выходить из аудитории.

7. По завершении процесса защиты все выпускники покидают аудиторию и ждут результатов. Обсуждение результатов защиты и выставление оценок проводится на закрытом заседании ГЭК по завершении защиты всех работ, намеченных на данное заседание. Особое внимание при оценке работы обращается на степень самостоятельности, проявленной обучающимся при написании дипломной работы, умение анализировать и критически оценивать действующую практику, защищать положения, обоснованные в работе.

Члены ГЭК оценивают выступление каждого выпускника, оценка за защиту выставляется по итогам обсуждения. (20-30 мин.). Применяются следующие параметры оценки дипломной работы при ее защите:

- соответствие работы утвержденной теме и плану;
- степень разработки конкретных вопросов темы, обоснованность, значимость и полнота сделанных выводов и предложений;
- качество выступления выпускника (содержание и логика изложения речи, полнота ответов на вопросы).

Решение комиссии об оценке дипломной работы принимается с учетом:

- отзыва руководителя;
- оценки внешнего рецензента;
- выступления при защите;
- ответов на поставленные вопросы.

Решения ГЭК об оценке защиты дипломных работ оформляются протоколом и объявляются в тот же день.

После завершения обсуждения выпускники приглашаются в зал для объявления оценок. Оценка за дипломную работу выставляется по четырех балльной системе: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

В случае, если защита дипломной работы признана неудовлетворительной, Обучающийся отчисляется из Академии. Повторная защита допускается после доработки и внесения исправлений, но не более одного раза и не ранее следующего периода работы Государственной аттестационной комиссии.

При неявке студента на защиту по уважительной (подтвержденной документально) причине, защита переносится до следующего периода работы ГЭК (не более чем на один год).

### **3.3.3 Критерии оценки дипломных работ**

Члены ГЭК оценивают выступление каждого выпускника, оценка за защиту выставляется по итогам обсуждения. Применяются следующие параметры оценки дипломной работы при ее защите:

- соответствие работы утвержденной теме и плану;
- степень разработки конкретных вопросов темы, обоснованность, значимость и полнота сделанных выводов и предложений;
- качество выступления выпускника (содержание и логика изложения речи, полнота ответов на вопросы).

Решение комиссии об оценке дипломной работы принимается с учетом:

- отзыва научного руководителя;
- оценки внешнего рецензента;
- выступления при защите;
- ответов на поставленные вопросы.

Защита дипломной работы заканчивается выставлением оценки.

**«Отлично»** выставляется за следующую дипломную работу:

- работа носит исследовательский характер, содержит грамотно изложенную теоретическую базу, глубокий анализ, характеризуется логичным, последовательным изложением материала с соответствующими выводами и обоснованными предложениями;
- имеет положительные отзывы научного руководителя и рецензента;
- при защите работы студент показывает глубокие знания вопросов темы, свободно оперирует данными исследования, вносит обоснованные предложения по изучению деятельности предприятия (организации), во время доклада использует наглядные пособия (таблицы, схемы, графики, слайды) или раздаточный материал, легко отвечает на поставленные вопросы;
- не имеет замечаний по оформлению дипломной работы.

**«Хорошо»** выставляется за следующую дипломную работу:

- проект носит исследовательский характер, содержит грамотно изложенную теоретическую базу, достаточно подробный анализ, характеризуется последовательным изложением материала с соответствующими выводами, однако с не вполне обоснованными предложениями;
- имеет положительные отзывы научного руководителя и рецензента;
- при защите работы студент показывает знания вопросов темы, оперирует данными исследования, вносит обоснованные предложения по изучению деятельности предприятия (организации), эффективному использованию финансовых ресурсов, во время доклада использует наглядные пособия (таблицы, схемы, графики, слайды) или раздаточный материал, без особых затруднений отвечает на поставленные вопросы;
- не имеет замечаний по оформлению дипломной работы.

**«Удовлетворительно»** выставляется за следующую дипломную работу:

- дипломная работа носит исследовательский характер, содержит теоретическую главу, базируется на практическом материале, но отличается поверхностным анализом и недостаточно критическим разбором, в нем



просматривается непоследовательность изложения материала, предоставлены необоснованные предложения;

- в отзывах руководителя и рецензента имеются замечания по содержанию работы и методике анализа;

- при защите студент проявляет неуверенность, показывает слабое знание вопросов темы, не дает полного, аргументированного ответа на заданные вопросы;

- имеет замечания по оформлению дипломной работы.

**«Неудовлетворительно»** выставляется за следующую дипломную работу:

- работа не носит исследовательский характер, не содержит анализа и практического разбора, не отвечает требованиям, изложенным в методических указаниях;

- не имеет выводов, либо носит декларативный характер;

- в отзывах руководителя и рецензента имеются критические замечания;

- при защите работы студент затрудняется отвечать на поставленные вопросы по теме, не знает теории вопроса, при ответе допускает существенные ошибки, к защите не подготовлены наглядные пособия и раздаточный материал;

- имеются серьезные замечания к оформлению дипломной работы.

Решения ГЭК об оценке защиты дипломных работ оформляются протоколом и объявляются в тот же день.

#### **4. Учебно-методическое обеспечение дипломной работы**

Библиотека Академии ИМСИТ имеет читальный зал и абонемент для самостоятельной работы студентов с учебно-методической и научной литературой.

**Рекомендуемые источники и литература** (имеются в библиотеке Академии ИМСИТ)

##### **Основные источники:**

1. Разработка. Внедрение и адаптация программного обеспечения отраслевой направленности: Учебное пособие. / Федорова Г.Н. - М.:КУРС, НИЦ ИНФРА-М, 2016. - 336 с.
2. Разработка и эксплуатация автоматизированных информационных систем: Учебное пособие / Гагарина Л.Г. - М.:ИД ФОРУМ, НИЦ ИНФРА-М, 2017. - 384 с.
3. Голицина, О.Л. И др. Программное обеспечение [Текст]: учеб. Пособие.- М.: Форум, 2018.-448с.
4. Программное обеспечение компьютерных сетей: Учебное пособие / О.В. Исаченко. - М.: НИЦ ИНФРА-М, 2017. - 117 с.
5. Основы проектирования баз данных: Учебное пособие / Голицына О.Л., Партыка Т.Л., Попов И.И., - 2-е изд. - М.:Форум, НИЦ ИНФРА-М, 2016. - 416 с.
6. Алгоритмизация и программирование: Учебное пособие / С.А. Канцедаль. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2017. - 352 с.
7. Основы метрологии, стандартизации и сертификации: Учебное пособие / Н.Д. Дубовой, Е.М. Портнов. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2016. - 256 с.
8. Введение в специальность программиста: Учебник / В.А. Гвоздева. - 2-е изд., испр. и доп.- М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2015. - 208 с.

##### **Дополнительная литература:**

1. Орлов С.А., Цилькер Б.Я. Технология разработки программного обеспечения [Текст].- СПб.: Питер, 2017.-608с.
2. Молчанов, А.Ю. Системное программное обеспечение [Текст]: учебник для СПО.-3-е изд.-СПб: ПИТЕР, 2020
3. Голицина, О.Л., Попов И.И. Программирование на языках высокого уровня [Текст]: учеб. пособие для СПО.-М.: Форум, 2011.-496с.
4. Павловская Т.А. С/С++. Программирование на языке высокого уровня [Текст]: Учебник.-СПб: ПИТЕР, 2012 – 320 с.
5. Иванова Г.С. Технология программирования [Текст]: учебник.- 3-е изд.-М.: КНОРУС, 2017.-336с.

6. Архитектура и проектирование программных систем: Монография / С.В. Назаров. - М.: НИЦ Инфра-М, 2013. - 351 с.
7. Кузнецов, А. С. Многоэтапный анализ архитектурной надежности и синтез отказоустойчивого программного обеспечения сложных систем [Электронный ресурс] : монография / А. С. Кузнецов, С. В. Ченцов, Р. Ю. Царев. - Красноярск: Сиб. федер. ун-т, 2018. - 143 с.
8. Базы данных. В 2-х кн. Кн. 2. Распределенные и удаленные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 272 с.
9. Балугев, Д. Секреты приложений Google [Электронный ресурс] / Денис Балугев. — М.: Альпина Паблишерз, 2016. — 287 с. - ISBN 978-5-9614-1274-1 - Режим доступа: <http://znanium.com/catalog.php?bookinfo=519902>
10. Web-ресурс «Электронная образовательная среда» <http://185.18.111.102/moodle/index.php>
11. «Электронно-библиотечная система ZNANIUM.COM» <http://znanium.com>
12. ЭБС «Айбукс.py/ibooks.ru» <http://ibooks.ru/>
13. Электронные Периодические издания <http://elibrary.ru>
14. Справочно-правовая база «Консультант Плюс»
15. Локальная сеть Академии ИМСИТ
16. Web-ресурс «Официальный сайт Академии ИМСИТ». <http://imsit.ru>

Студентам Академии ИМСИТ обеспечена возможность свободного доступа к фондам учебно-методической документации и интернет-ресурсам.

## ПРИЛОЖЕНИЕ А

### Примерная тематика дипломных работ

#### по специальности 09.02.03 Программирование в компьютерных системах

1. Разработка электронного учебного пособия на примере конкретного курса.
2. Разработка и создание автоматизированного рабочего места «Отдел кадров» для организации.
3. Разработка и создание автоматизированного рабочего места «Учет сведений об имуществе» для организации.
4. Разработка приложения для предметной области «Организация учебного процесса в учебном заведении».
5. Разработка приложения для предметной области «Комплектующие персонального компьютера и периферийные устройства» (с функционалом организации по месту прохождения практики).
6. Разработка приложения для предметной области «Учет товаров и материалов на складе предприятия».
7. Разработка и создание интерактивного Web-сайта с учётом особенностей конкретного предприятия.
8. Разработка и создание сайта-визитки для организации.
9. Разработка приложения для предметной области «Учет товаров в магазине» (с функционалом организации по месту прохождения практики).
10. Разработка приложения для предметной области «Организация учебного процесса в колледже».
11. Разработка приложения для предметной области «Учет средств вычислительной техники в организации».
12. Разработка приложения для предметной области «Учет основных средств предприятия».
13. Разработка приложения для предметной области «Учет транспортных средств и их владельцев» в организации.
14. Разработка автоматизированной системы управления проектами для малых предприятий.
15. Разработка интернет - магазина (на примере ...).
16. Разработка базы данных «Абитуриент».
17. Проектирование и разработка автоматизированного рабочего места специалиста предприятия.
18. Разработка web-сайта предприятия.
19. Разработка автоматизированной системы управления проектами для малых предприятий.
20. Разработка интернет - магазина торговой компании.

21. Разработка клиент-серверной аналитической распределенной информационной системы управления рабочим временем.
22. Разработка программного обеспечения поддержки процессов закупа и учёта хранения товаров на складе.
23. Разработка программного продукта, позволяющего автоматизировать процесс составления учебного расписания в ВУЗе.
24. Разработка программы для автоматизации учета и предварительных заказов на предприятиях различных сфер деятельности.
25. Система контроля и управления доступом сотрудников в помещения компании (на примере ...).
26. Разработка электронного учебника по дисциплине «Базы данных».
27. Автоматизация процесса регистрации и обработки данных (на примере ...).
28. Модернизация торговых процессов в организации, на основе Web-технологий (на примере ...).
29. Разработка web – сайта частной компании (на примере ...).
30. Разработка автоматизированного рабочего места менеджера (на примере ...).
31. Разработка автоматизированного рабочего места коллектора (на примере ...).
32. Разработка мобильного приложения для предоставления доступа студентам к расписанию занятий (на примере ...).
33. Разработка мобильного приложения для предоставления доступа студентам к результатам обучения (по материалам .....).
34. Разработка автоматизированной информационно-аналитической системы для транспортной компании (на примере ...).
35. Разработка автоматизированной рабочей системы учета компьютеров и оргтехники на предприятии (на примере ...).
36. Разработка автоматизированной системы оценки качества товаров и услуг (на примере ...).
37. Разработка автоматизированной системы проверки знаний сотрудников (на примере ...).
38. Разработка автоматизированной системы учета реализации товара (на примере ...).
39. Разработка мобильного приложения для предоставления доступа студентам к электронной образовательной среде (на примере ...).
40. Разработка базы данных для информатизации деятельности предприятия (на примере ...).
41. Разработка базы данных организации (на примере ...).
42. Разработка защищенного модуля доступа к локальной сети (на примере ...).

43. Разработка и создание автоматизированного рабочего места сотрудника отдела кадров (на примере ...).
44. Разработка интерактивного учебного пособия на основе web-технологий (на примере ...).
45. Разработка интернет - магазина торговой организации (на примере ...).
46. Разработка информационной системы для автоматизации учета клиентов предприятия сферы услуг (на примере ...).
47. Разработка мобильного приложения для предоставления доступа студентам к базе данных предприятия (на примере ...).
48. Разработка информационной системы на основе методов Web-ориентированного программирования (на примере ...).
49. Разработка информационной системы по учету контингента учащихся (на примере ...).
50. Разработка информационной системы поддержки принятия решений для персонала торговой компании (на примере ...).
51. Разработка информационной системы работы с клиентами организации (на примере ...).
52. Разработка информационной системы склада предприятия (на примере ...).
53. Разработка информационной системы управления оборудованием (на примере ...).
54. Разработка информационной системы учета оказанных услуг (на примере ...).
55. Разработка информационной системы учета продукции предприятия (на примере ...).
56. Разработка комплекса программ автоматизации процесса регистрации и обработки данных (на примере ...).
57. Разработка модуля проверки базы данных информационной системы (на примере ...).
58. Разработка мобильного приложения для предоставления доступа студентам к товарам и услугам (на примере ...).
59. Разработка программного обеспечения Web-ориентированной информационной системы риэлтерской фирмы (на примере ...).
60. Разработка программного обеспечения для автоматизации работы с клиентами (на примере ...).
61. Разработка программного обеспечения для автоматизированного рабочего места заведующего отделением (на примере ...).
62. Разработка программного обеспечения информационной системы работы с клиентами торговой организации (на примере ...).

63. Разработка программного обеспечения процесса регистрации и обработки данных для предприятия (на примере ...).
64. Разработка программного обеспечения системы учета работ по сервисному обслуживанию (на примере ...).
65. Разработка программного обеспечения учета продаж (на примере ...).
66. Разработка программного обеспечения учета работ системного администратора организации (на примере ...).
67. Разработка программы приема и передачи сообщений в локальной сети (на примере ...).
68. Проектирование информационной системы автоматизации управления выполнением курсовых и выпускных квалификационных работ.
69. Проектирование информационной системы планирования мероприятий Академического колледжа.
70. Разработка интернет – магазина.
71. Разработка информационно-справочной системы нормативно-правовых документов Академии ИМСИТ.
72. Автоматизация бизнес-процессов компании.
73. Проектирование программного модуля «Библиотека Академического колледжа».
74. Проектирование информационной системы управления учебным процессом.
75. Автоматизация учета рабочего времени сотрудников компании.
76. Разработка программного обеспечения для деловой игры.
77. Разработка имитационных моделей.
78. Разработка электронных учебных курсов (учебников, методических указаний, практических и лабораторных работ).

## ПРИЛОЖЕНИЕ

**Негосударственное аккредитованное некоммерческое частное  
образовательное учреждение высшего образования АКАДЕМИЯ  
МАРКЕТИНГА И СОЦИАЛЬНО-ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИЙ – ИМСИТ (г. Краснодар)**

**Академический колледж**

УТВЕРЖДАЮ

Председатель ПЦК

\_\_\_\_\_ М. В. Большакова

"\_\_" \_\_\_\_\_ 20\_\_ г.

### ЗАДАНИЕ

**на дипломную работу**

студенту **Иванову Ивану Ивановичу**

Тема дипломной работы: «Совершенствование организационных и технических мер по управлению информационными системами» (по материалам ООО «Кот и Кит», г. Краснодар), закреплена приказом ректора № \_\_\_\_\_ от "\_\_\_" \_\_\_\_\_ 20\_\_ г.

Целевая установка: совершенствование организационных и технических мер по управлению информационными системами ООО «Кот и Кит», г. Краснодар.

Основные вопросы, подлежащие разработке (исследованию):

1. Общая характеристика электронных учебников.
2. Проектирование автоматизированного учебного курса.
3. Реализация разработки проекта.
4. Мероприятия по охране труда и безопасности жизнедеятельности.

Основная литература (согласно приложению).

Срок представления законченной работы "\_\_\_" \_\_\_\_\_ 20\_\_ г.

Дата выдачи задания "\_\_\_" \_\_\_\_\_ 20\_\_ г.

Руководитель \_\_\_\_\_ Петров П.П.

Задание получил "\_\_\_" \_\_\_\_\_ 20\_\_ г.

Студент \_\_\_\_\_ Иванов И.И.



## Образец заполнения Календарного плана-графика

Негосударственное аккредитованное некоммерческое частное образовательное  
учреждение высшего образования  
«Академия маркетинга и социально-информационных технологий – ИМСИТ»  
(г.Краснодар)

Академический колледж

### КАЛЕНДАРНЫЙ ПЛАН-ГРАФИК

выполнения дипломной работы

Обучающийся \_\_\_\_\_

Тема \_\_\_\_\_

№ п\п	Наименование этапа	Срок выполнения	Отметка руководителя о выполнении
1	Выбор темы, подбор литературы	до 1 ноября	
2	Составление плана работы и утверждение его руководителем	до 15 ноября	
3	Изучение литературы	в течение всего периода работы	
4	Сбор и обработка практического материала	в течение всего периода работы	
5	Представление руководителю: первой главы работы второй главы третьей главы всей работы	до 10 декабря до 15 февраля до 1 марта до 15 апреля	
6	Доработка глав с учетом указаний руководителя и оформление работы	до 1 мая	
7	Представление работы на отзыв руководителю	1-2 мая	
8	Передача работы на рецензию	6-7 мая	
9	Представление работы председателю ПЦК профиля	до 10 мая	
10	Представление работы на отделение в переплетенном виде с отзывом руководителя и рецензией	до 5 июня	

Обучающийся- \_\_\_\_\_

Руководитель работы \_\_\_\_\_

Негосударственное аккредитованное некоммерческое частное  
образовательное учреждение высшего образования  
АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ – ИМСИТ (г.  
Краснодар)

Академический колледж

«ДОПУСТИТЬ К ЗАЩИТЕ»  
Председатель предметно-  
цикловой комиссии

\_\_\_\_\_ М. В. Большакова

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

ДИПЛОМНАЯ РАБОТА

на тему: «СОВЕРШЕНСТВОВАНИЕ ОРГАНИЗАЦИОННЫХ И ТЕХНИЧЕСКИХ  
МЕР ПО УПРАВЛЕНИЮ ИНФОРМАЦИОННЫМИ СИСТЕМАМИ»  
(по материалам ООО «Кот и Кит», г. Краснодар)

Специальность 09.02.03 Программирование в компьютерных  
системах

Работу выполнил  
студент 4 курса  
очной формы обучения  
группы 16-СПО-ПКС- 01  
Иванов Иван Иванович

Руководитель,  
преподаватель

\_\_\_\_\_ П.П. Петров

Краснодар 2020

## ПРИЛОЖЕНИЕ

*Образец оформления*

### РЕФЕРАТ

Работа 57 с., 12 рис., 5 табл., 35 источн., 5 прил.

АВТОМАТИЗИРОВАННЫЙ УЧЕБНЫЙ КУРС, ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ, WEB-ПРОГРАММИРОВАНИЕ, КЛИЕНТСКИЕ ЯЗЫКИ WEB-ПРОГРАММИРОВАНИЯ, HTML, JAVASCRIPT, ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, БРАУЗЕР, WEB-ИНТЕРФЕЙС.

Цель дипломной работы - разработка автоматизированного учебного курса по дисциплине «Web программирование».

Методы исследования – экспериментальный, методы системного анализа, моделирования, теоретический.

Объектом исследования являются автоматизированные учебные курсы.

Предмет исследования - программные средства разработки автоматизированных учебных курсов

Основные результаты – разработан автоматизированный учебный курс, позволяющий повысить эффективность учебного процесса.

Область применения системы – учебный процесс Академии маркетинга и социально-информационных технологий.

## ПРИЛОЖЕНИЕ

### *Образец оформления*

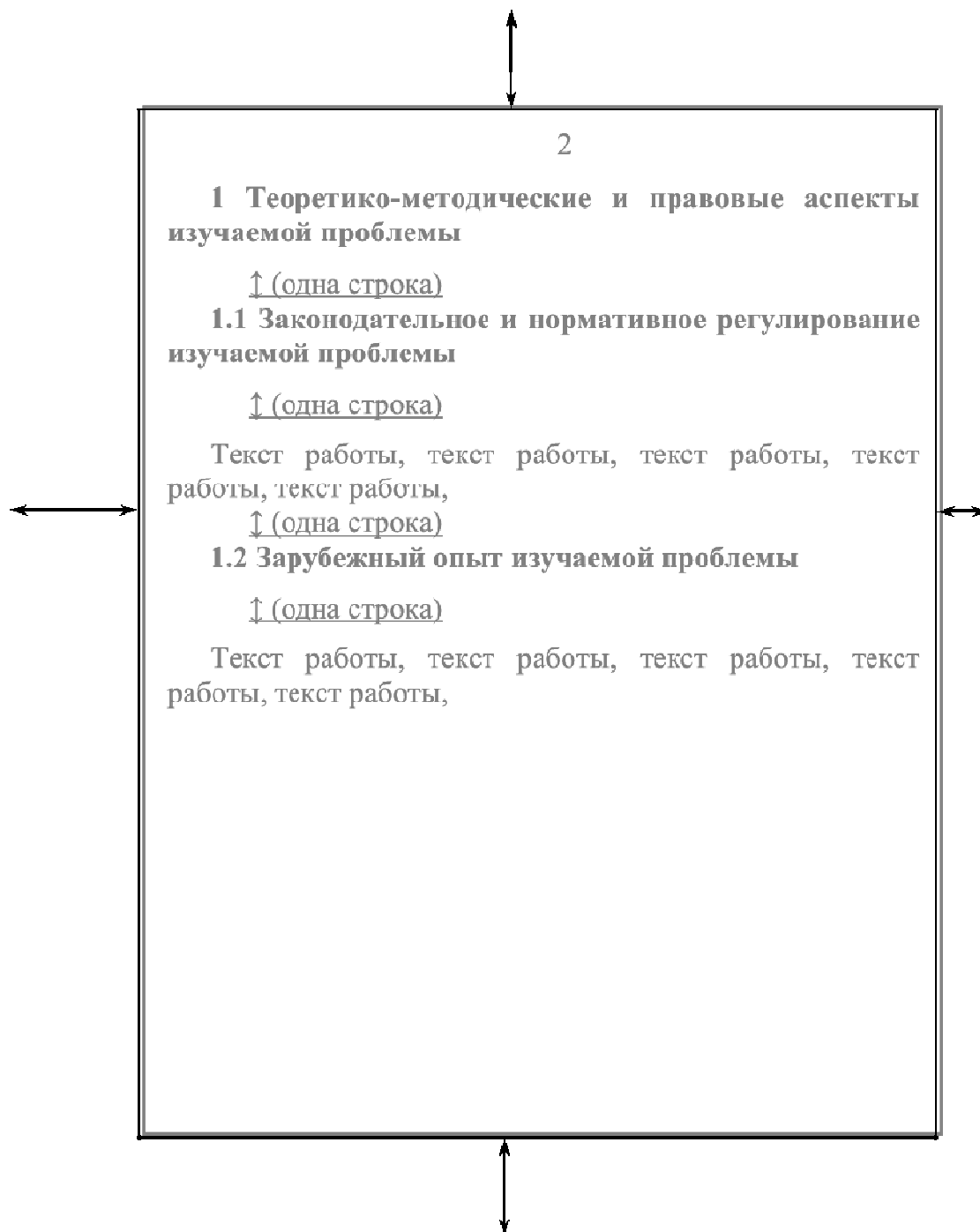
## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	5
1 Общая характеристика электронных учебников.....	7
1.1 Исследование предметной области автоматизированных учебных курсов.....	7
1.2 Концептуальная модель автоматизированных учебных курсов.....	12
1.3 Требования к программному обеспечению автоматизированных учебных курсов.....	18
1.4 Техническое задание на разработку автоматизированного учебного курса.....	20
1.5 Выбор метода и среды разработки программного обеспечения.....	23
2 Проектирование автоматизированного учебного курса.....	25
2.1 Подготовка проекта гипертекста.....	25
2.2 Проектирование интеллектуального ядра.....	29
3 Реализация разработки.....	33
3.1 Разработка программных модулей.....	33
3.2 Разработка интерфейса пользователя.....	36
3.3 Разработка руководства пользователя.....	39
3.4 Разработка руководства программиста.....	42
4 Мероприятия по охране труда и безопасности жизнедеятельности.....	44
4.1 Вредные воздействия производственных факторов при работе с ЭВМ...	44
4.2 Факторы, влияющие на опасность поражения электрическим током.....	46
4.3 Методы повышения безопасности.....	49
ЗАКЛЮЧЕНИЕ.....	50
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	53
ПРИЛОЖЕНИЕ А Исходный код.....	55

# ПРИЛОЖЕНИЕ

## ФОРМА И ОБРАЗЕЦ оформления структурных частей дипломной работы работы

### Форма основной части работы



## ВВЕДЕНИЕ

Текст введения. Текст введения. Текст введения. Текст введения. Текст введения.

Текст введения. Текст введения.

----- Разрыв страницы -----

### 1 Название первого раздела

#### 1.1 Название первого подраздела первого раздела

Текст работы, текст работы, текст работы, текст работы, текст работы , текст работы, текст работы, текст работы (таблица 1).

Таблица 1 – Название таблицы


Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы.

#### 1.2 Название второго подраздела первого раздела

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы (рисунок 1).

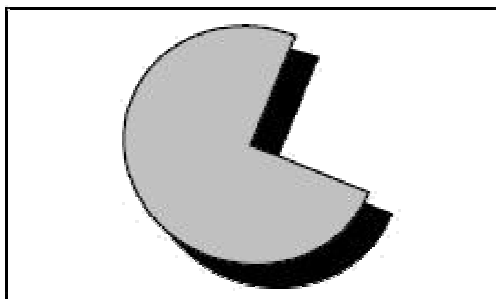


Рисунок 1 – Название рисунка

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы.

----- Разрыв страницы -----

## 2 Название второго раздела

### 2.1 Название первого подраздела второго раздела

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы. Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы (рисунок 2).

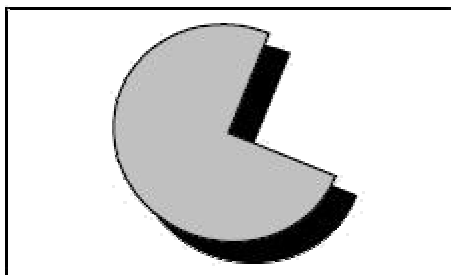


Рисунок 2 – Название рисунка

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы.

### 2.2 Название второго подраздела второго раздела

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы (таблица 2).

Таблица 2 – Название таблицы


Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы.

----- Разрыв страницы -----

## 3 Название третьего раздела

### 3.1 Название первого подраздела третьего раздела

Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы. Текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы, текст работы (рисунок 3).

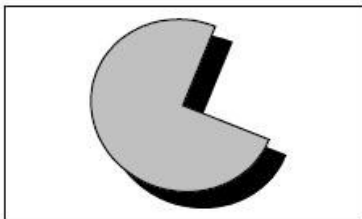


Рисунок 3 – Название рисунка

Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы (таблица 3).

Таблица 3 – Название таблицы


----- Разрыв страницы -----

Продолжение таблицы 3


Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы.

### 3.2 Название второго подраздела третьего раздела

Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы.

----- Разрыв страницы -----

## 4 Название четвёртого раздела

### 4.1 Название первого подраздела четвёртого раздела

Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы. Текст работы, текст работы, текст работы, текст  
работы, текст работы, текст работы, текст работы, текст работы (рисунок 4).

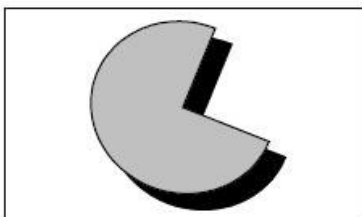


Рисунок 4 – Название рисунка



Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы (таблица 4).

Таблица 4 – Название таблицы


----- Разрыв страницы -----

Продолжение таблицы 4


Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы.

**4.2 Название второго подраздела четвёртого раздела**

Текст работы, текст работы, текст работы, текст работы, текст работы,  
текст работы, текст работы, текст работы.

----- Разрыв страницы -----

**ЗАКЛЮЧЕНИЕ**

Текст заключения. Текст заключения. Текст заключения. Текст  
заключения. Текст заключения. Текст заключения. Текст заключения.

----- Разрыв страницы -----

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

ОБРАЗЕЦ  
оформления таблиц, рисунков, формул

Головка


Заголовки граф

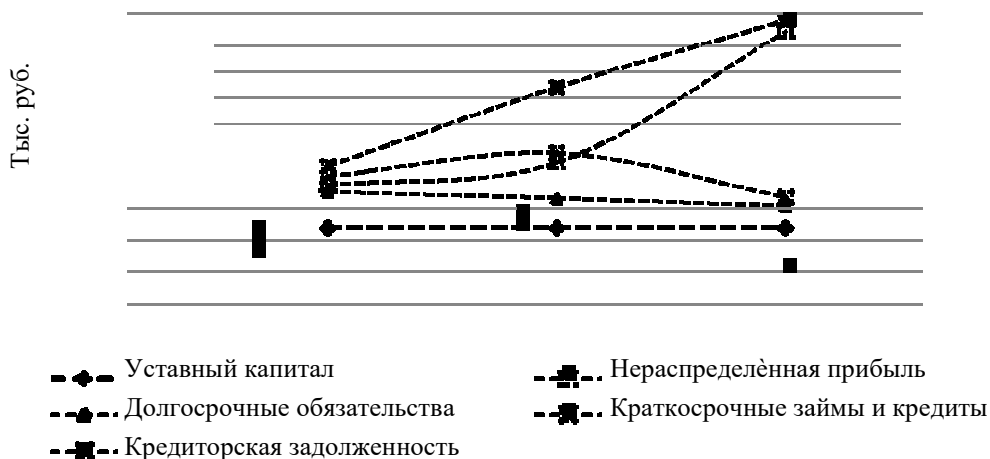
Подзаголовки граф

Строки  
(горизонтальные  
ряды)

Боковик

Графы

Коэффициенты			Машинный результат	Контрольный результат	Комментарий
1	1	2	1,00; -2,00	1; -2	начальный тест
1	0	0,25	(0,00; 0,50) (0,00; - 0,50)	0+i1/2; 0-i1/2;	проверка в нормальных условиях
0	0	0	Любое число	Любое число	нулевой пример
0	2	1	-0,50	-1/2	экстремальное значение
0	0	2	Решений нет	Решений нет	исключительная ситуация
0	2	0	0,00	0	экстремальное значение
2	1	0	0,00; -0,50	0; -1/2	проверка в нормальных условиях
1	1	0,5	(-0,50; 0,50) (-0,50; -0,50)	-1/2+ i1/2; -1/2-i1/2;	проверка в нормальных условиях
2	0	0	0,00; 0,00	0; 0	проверка в нормальных условиях


$$O_{T3}(OCc) = OCc_1 - OCc_2 \times T_p(B), \quad (1)$$

ОСс<sub>2</sub> – средняя стоимость оборотных средств предыдущего периода.

## **ПРИЛОЖЕНИЕ**

**Негосударственное аккредитованное некоммерческое частное образовательное  
учреждение высшего образования  
«Академия маркетинга и социально-информационных технологий – ИМСИТ»  
(г.Краснодар)**

**Академический колледж**

### **ИЛЛЮСТРАЦИОННЫЙ МАТЕРИАЛ**

**на тему: «СОВЕРШЕНСТВОВАНИЕ ОРГАНИЗАЦИОННЫХ И ТЕХНИЧЕСКИХ  
МЕР ПО УПРАВЛЕНИЮ ИНФОРМАЦИОННЫМИ СИСТЕМАМИ»  
(по материалам ООО «Кот и Кит», г. Краснодар)**

**Специальность 09.02.03 Программирование в компьютерных  
системах**

Работу выполнил  
студент 4 курса  
очной формы обучения  
группы 16-СПО-ПКС- 01  
Иванов Иван Иванович

Краснодар  
2020

## ПРИЛОЖЕНИЕ

**Негосударственное аккредитованное некоммерческое  
частное образовательное учреждение высшего  
образования**

**АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ – ИМСИТ (г.  
Краснодар)**

## ОТЗЫВ

**на дипломную работу**

студента колледжа \_\_\_\_\_

\_\_\_\_\_  
(фамилия, имя, отчество)

выполненной на тему: \_\_\_\_\_

Дипломная работа \_\_\_\_\_

\_\_\_\_\_  
(фамилия, имя, отчество)

содержит: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Руководитель дипломной работы оценивает выполненную работу студента \_\_\_\_\_  
(фамилия, имя, отчество)

как \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Руководитель дипломной работы \_\_\_\_\_

" \_\_\_\_\_ " \_\_\_\_\_ 20 \_\_\_\_\_ г. \_\_\_\_\_  
подпись (ФИО)

## ПРИЛОЖЕНИЕ

(обязательное)

### РЕЦЕНЗИЯ

На дипломную работу обучающегося  
Академического колледжа группы 16-СПО-ПКС-01  
Академии маркетинга и социально-информационных технологий ИМСИТ,  
г. Краснодар

**Иванова Ивана Ивановича**

Выполненную на тему: «Совершенствование организационных и технических  
мер по управлению информационными системами»  
(по материалам ООО «Кот и Кит», г. Краснодар)

Настоящая дипломная работа соответствует заявленному заданию. Актуальность данной работы обусловлена тем, что информатизация различных сфер деятельности человека, в том числе и образования, дает возможность применять современные средства программирования и разработки электронных приложений для повышения уровня образовательных технологий.

В первой главе исследованы теоретические основы предметной области автоматизированных учебных курсов, приводятся основные требования к программному обеспечению автоматизированных учебных курсов.

Во второй главе проводится проектирование автоматизированного учебного курса путём подготовки проекта гипертекста и проектирования интеллектуального ядра.

В третьей главе дипломной работы рассмотрены пути реализации разработки проекта – приводятся разработка программных модулей, интерфейса пользователя, руководства пользователя и программиста.

В четвёртой главе разработаны мероприятия по охране труда и безопасности жизнедеятельности.

В заключении приведены краткие выводы и обобщения по всем основным разделам работы.

Структура и содержание работы раскрывают тему исследования. Материал дипломной работы изложен грамотно, последовательно, убедительно.

Дипломная работа допускается к защите в ГЭК и заслуживает положительной оценки.

Рецензент

МП \_\_\_\_\_ / \_\_\_\_\_  
(подпись) (ФИО)

«\_\_\_» \_\_\_\_\_ 2020 г.

## ПРИЛОЖЕНИЕ

(обязательное)

**Негосударственное аккредитованное некоммерческое частное  
образовательное учреждение высшего образования  
АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ – ИМСИТ (г.  
Краснодар)**

**Академический колледж**

### ЗАКЛЮЧЕНИЕ

председателя предметно-цикловой комиссии социально-экономического профиля

Большаковой Марии Викторовны

о допуске к защите дипломной работы, выполненной обучающимся

---

(фамилия, имя, отчество)

Дипломная работа соответствует требованиям ФГОС СПО специальности, представляет собой законченный результат исследования, в котором раскрыты теоретические и практические аспекты темы исследования - в полном объеме, что определяет сформированность общих и профессиональных компетенций.

Председатель ПЦК \_\_\_\_\_ Большакова М.В.  
подпись

" \_\_\_\_ " \_\_\_\_\_ 2020 г.