Федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА и ГОСУДАРСТВЕННОЙ СЛУЖБЫ при ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»

ЧЕБОКСАРСКИЙ ФИЛИАЛ РАНХиГС

Ильина Любовь Николаевна

Подписано: Ильина Любовь Николаевна
DN: cn=Ильина Любовь Николаевна, c=RU, o=ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ "РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ", email=ilina@cheb.ranepa.ru
Дата: 2025.03.06 08:28:46 +03'00'

Утверждена Ученым советом
Чебоксарского филиала РАНХиГС
(в составе ООП ПССЗ)
Протокол от «31» марта 2022 г. № 03/2022

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ К ГОСУДАРСТВЕННОЙ ИТОГОВОЙ АТТЕСТАЦИИ ВЫПУСКНИКОВ

для студентов специальности

09.02.07 Информационные системы и программирование

г. Чебоксары
2022

СОДЕРЖАНИЕ

1. **ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ГИА……...……..3**
2. **СТРУКТУРА ПРОЦЕДУР ГИА И ПОРЯДОК ПРОВЕДЕНИЯ……...….5**
3. **ТИПОВЫЕ ЗАДАНИЯ ДЛЯ ДЕМОНСТРАЦИОННОГО ЭКЗАМЕНА……..5**
4. **ПОРЯДОК ОРГАНИЗАЦИИ И ПРОВЕДЕНИЯ ЗАЩИТЫ ДИПЛОМНОЙ РАБОТЫ (ДИПЛОМНОГО ПРОЕКТА)……...…9**
5. **ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ ОСВОЕННОГО ВИДА ДЕЯТЕЛЬНОСТИ ПРИ ПРОВЕДЕНИИ ГИА …….14**

1. ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ГИА

1.1. Особенности образовательной программы

Фонды оценочных средств разработаны для специальности 09.02.07 Информационные системы и программирование.

В рамках специальности СПО 09.02.07 Информационные системы и программирование предусмотрено освоение следующих квалификаций:

- программист

Наименование основных видов деятельности	**Наименование профессиональных модулей**	**Квалификация**
ВД 01 Разработка модулей программного обеспечения для компьютерных систем.	ПМ.01 Разработка модулей программного обеспечения для компьютерных систем	программист
ВД 02 Осуществление интеграции программных модулей.	ПМ.02 Осуществление интеграции программных модулей	программист
ВД 04 Сопровождение и обслуживание программного обеспечения компьютерных систем.	ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем.	программист
ВД 11 Разработка, администрирование и защита баз данных	ПМ.11 Разработка, администрирование и защита баз данных	программист

1.2. Применяемые материалы

Для разработки оценочных заданий по каждому из сочетаний квалификаций рекомендуется применять следующие материалы:

Квалификация (сочетание квалификаций)	Профессиональный стандарт	Компетенция
- программист	06.001 Программист»	- программист

1.3 Перечень результатов, демонстрируемых на ГИА

Для специальности 09.02.07 Информационные системы и программирование

Оцениваемые основные виды деятельности и компетенции по ним	Описание тематики выполняемых в ходе процедур ГИА заданий (*направленных на демонстрацию конкретных освоенных результатов по ФГОС)*
Демонстрационный экзамен	

Уровень ДЭ: ФГОС СПО	Программные решения для бизнеса. КОД 1.2-2023-2025
Защита дипломного проекта (работы)	

Оцениваемые основные виды деятельности и профессиональные компетенции	Описание выполняемых в ходе процедур ГИА заданий (*направленных на демонстрацию конкретных освоенных результатов по ФГОС)*
Демонстрационный экзамен	
A - Системный анализ и проектирование B - Разработка программного обеспечения C - Стандарты разработки программного обеспечения D – Документирование программных решений	**Модуль 1:** Системный анализ и проектирование» Данный модуль предполагает работу по определению требований к информационной системе на основе анализа описания предметной области, создание спецификаций к прецедентам. В рамках модуля должно быть реализовано проектирование диаграммы сущность-связь и создание словаря данных.
	Модуль 2: Разработка программного обеспечения» Вы можете выбрать любую среду разработки и язык программирования из доступных, но должны сделать это обдуманно, придерживаться при их использовании профессиональных стандартов. Обязательным требованием является обеспечение ограниченного доступа к продукту, возможности ввода и хранения данных. Программный продукт должен быть готовым решением. Пользователи не должны устанавливать или настраивать СУБД, вручную переносить хранящиеся данные и т.п. Учтите: компьютер при проверке будет сконфигурирован точно также, как и ваш перед началом знакомства с ним.
	Модуль 3: Стандарты разработки программного обеспечения» Модуль отражает общий профессионализм решения: обратная связь системы с пользователем, стабильная работа всех разработанных программ, стиль кода на протяжении разработки всей системы, организация файловой структуры проекта, соблюдение культуры кодирования, комментарии к коду, умение работать с системой контроля версий.
	Модуль 4: Документирование программных решений» Модуль отражает навык документирования разрабатываемого в ходе экзамена программного решения.

2. СТРУКТУРА ПРОЦЕДУР ГИА И ПОРЯДОК ПРОВЕДЕНИЯ

2.1. Структура задания для процедуры ГИА

Программа государственной итоговой аттестации, задания, критерии их оценивания, продолжительность демонстрационного экзамена утверждаются образовательной организацией и доводятся до сведения студентов не позднее чем за шесть месяцев до начала государственной итоговой аттестации.

Задания для проведения демонстрационного экзамена для каждого обучающегося определяется методом случайного выбора в начале демонстрационного экзамена. Перечень модулей для выбора и возможные сочетания модулей определяются образовательной организацией исходя из возможностей образовательной организации и особенностей образовательной программы. Время, отводимое на выполнение заданий демонстрационного экзамена, определяется образовательной организацией.

2.2. Порядок проведения процедуры

Процедура проведения демонстрационного экзамена предполагает осуществление контрольных мероприятий в течение двух дней.

В первый день проводится организационное собрание, целевой инструктаж по охране труда и безопасному выполнению работ по специальности и проверка теоретических знаний по модулям программы в соответствии с присваиваемой квалификацией и знаний по охране труда и безопасному выполнению работ в профессиональной деятельности.

Во второй день проводится проверка практических умений и профессиональных компетенций по модулям программы в соответствии с присваиваемой квалификацией и проводится подведение итогов демонстрационного экзамена.

Таблица 1

Общее количество модулей в задании для ДЭ	4 модуля
Количество модулей для проведения демонстрационного экзамена для одного студента	4 модуля
Время выполнения всех модулей задания демонстрационного экзамена	7 академических часов
Введение вариативного модуля на уровне образовательной организации по согласованию с работодателем	возможно
Максимальное время выполнения задания демонстрационного экзамена	7 академических часов
Общее максимальное количество баллов за выполнение задания демонстрационного экзамена одним студентом, распределяемое между модулями	50,0 баллов

3. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ ДЕМОНСТРАЦИОННОГО ЭКЗАМЕНА

3.1. Структура и содержание типового задания

3.1.1. Формулировка типового практического задания (приводится наименование задания для оценки результатов освоения программы СПО)

Описание модуля 1: «Системный анализ и проектирование»

Данный модуль предполагает работу по определению требований к информационной системе на основе анализа описания предметной области, создание спецификаций к прецедентам. В рамках модуля должно быть реализовано проектирование диаграммы сущность-связь и создание словаря данных.

При выполнении модуля 1 ставятся следующие цели:

1. Определение функциональных требований к системе.
2. Проектирование системы с помощью диаграмм UML.
3. Проектирование системы хранения данных.

При выполнении данного модуля 1 ставятся следующие задачи:

1. Изучить описание предметной области.
2. Определить функциональные требования к системе.
3. Разработать диаграмму вариантов использования системы.
4. Разработать ER-диаграмму и словарь данных.

Диаграмма прецедентов Для согласования процесса разработки с заказчиком Вам необходимо ознакомиться с описанием предметной области и заданием экзамена, сделать диаграмму прецедентов (Use Case) для основных пользователей системы. Проектирование базы данных (ERD)

На основе описания предметной области и задания демонстрационного экзамена (все сессии) Вам необходимо спроектировать ER-диаграмму для информационной системы. Обязательна 3 нормальная форма с обеспечением ссылочной целостности. При разработке диаграммы обратите внимание на согласованную осмысленную схему именования, создайте необходимые первичные и внешние ключи, определите ограничения внешних ключей, отражающие характер предметной области. Data Dictionary Для диаграммы ER необходимо создать словарь данных – набор информации, описывающий, какой тип данных хранится в базе данных, их формат, структуру и способы использования данных. Обратите внимание на соответствие вашей диаграммы и словаря данных. Используйте подходящие типы данных, ограничения и форматы. Отразите в документе ограничения, определенные в таблицах, включая первичные ключи, отношения внешнего ключа с другими таблицами и ненулевые ограничения. Не забудьте сделать необходимые пояснения и комментарии к неоднозначным полям.

Описание модуля 2: «Разработка программного обеспечения»

Вы можете выбрать любую среду разработки и язык программирования из доступных, но должны сделать это обдуманно, придерживаться при их использовании профессиональных стандартов.

Обязательным требованием является обеспечение ограниченного доступа к продукту, возможности ввода и хранения данных.

Программный продукт должен быть готовым решением. Пользователи не должны устанавливать или настраивать СУБД, вручную переносить хранящиеся данные и т.п. Учтите: компьютер при проверке будет сконфигурирован точно также, как и ваш перед началом знакомства с ним.

При выполнении модуля 2 ставятся следующие цели:

1. Разработка программного продукта

При выполнении данного модуля 2 ставятся следующие задачи:

1. Выбрать технологический стек для реализации программного продукта.
2. Разработать объекты баз данных, импортировать предоставленные данные, при отсутствии данных для импорта заполнить таблицы тестовыми данными.
3. Разработать программный продукт на основании предоставленных функциональных требований.

Описание модуля 3: «Стандарты разработки программного обеспечения»

Модуль отражает общий профессионализм решения: обратная связь системы с пользователем, стабильная работа всех разработанных программ, стиль кода на протяжении

разработки всей системы, организация файловой структуры проекта, соблюдение культуры кодирования, комментарии к коду, умение работать с системой контроля версий.

При выполнении модуля 3 ставятся следующие цели:

1. Разработка кода программного продукта в соответствии с отраслевыми стандартами.

При выполнении модуля 3 ставятся следующие задачи:

1. Реализовать обратную связь системы с пользователем.
2. Соблюдать культуру кодирования.
3. Результаты работы предоставить в системе контроля версий.

Название приложения Используйте соответствующие названия для ваших приложений и файлов. Так, например, наименование настольного приложения должно обязательно включать название компании- заказчика.

Файловая структура Файловая структура проекта должна отражать логику, заложенную в приложение. Например, все формы содержатся в одной директории, пользовательские визуальные компоненты – в другой, классы сущностей – в третьей. Структура проекта Каждая сущность должна быть представлена в программе как минимум одним отдельным классом. Классы должны быть небольшими, понятными и выполнять одну единственную функцию (Single responsibility principle). Для работы с разными сущностями используйте разные формы, где это уместно.

Макет и технические характеристики

Все компоненты системы должны иметь единый согласованный внешний вид, соответствующий руководству по стилю, а также следующим требованиям:

- разметка и дизайн (предпочтение отдается масштабируемой компоновке;
- должно присутствовать ограничение на минимальный размер окна;
- должна присутствовать возможность изменения размеров окна, где это необходимо;
- увеличение размеров окна должно увеличивать размер контентной части, например, таблицы с данными из БД);
- группировка элементов (в логические категории);
- использование соответствующих элементов управления (например, выпадающих списков для отображения подстановочных значений из базы данных);
- расположение и выравнивание элементов (метки, поля для ввода и т.д.)
- последовательный переход фокуса по элементам интерфейса (по нажатию клавиши TAB);
- общая компоновка логична, понятна и проста в использовании;
- последовательный пользовательский интерфейс, позволяющий перемещаться между существующими окнами в приложении (в том числе обратно, например, с помощью кнопки «Назад»);
- соответствующий заголовок на каждом окне приложения (не должно быть значений по умолчанию типа MainWindow, Form1 и тп). Обратная связь с пользователем.

Уведомляйте пользователя о совершаемых им ошибках или о запрещенных в рамках задания действиях, запрашивайте подтверждение перед удалением, предупреждайте о неотвратимых операциях, информируйте об отсутствии результатов поиска и т.п. Окна сообщений соответствующих типов (например, ошибка, предупреждение, информация) должны отображаться с соответствующим заголовком и пиктограммой. Текст сообщения должен быть полезным и информативным, содержать полную информацию о совершенных ошибках пользователя и порядок действий для их исправления. Также можно использовать визуальные подсказки для пользователя при вводе данных.

Обработка ошибок. Не позволяйте пользователю вводить некорректные значения в текстовые поля сущностей. Например, в случае несоответствия типа данных или размера поля введенному значению. Оповестите пользователя о совершенной им ошибке. При возникновении непредвиденной ошибки приложение не должно аварийно завершать работу. Оформление кода Идентификаторы переменных, методов и классов должны отражать суть и/или цель их использования, в том числе и наименования элементов

управления (например, не должно быть значений по умолчанию типа Form1, button3). Идентификаторы должны соответствовать соглашению об именовании (Code Convention) и стилю CamelCase (для C# и Java) и snake_case (для Python). Допустимо использование не более одной команды в строке. Комментарии Используйте комментарии для пояснения неочевидных фрагментов кода. Запрещено комментирование кода. Хороший код воспринимается как обычный текст. Не используйте комментарии для пояснения очевидных действий. Комментарии должны присутствовать только в местах, которые требуют дополнительного пояснения. Используйте тип комментариев, который в дальнейшем позволит сгенерировать XML- документацию, с соответствующими тегами (например, param, return(s), summary и др.)

Описание модуля 4: «Документирование программных решений»

Модуль отражает навык документирования разрабатываемого в ходе экзамена программного решения.

При выполнении модуля 4 ставятся следующие цели:

1. Разработка технической документации на программное решение.

При выполнении модуля 4 ставятся следующие задачи:

1. Разработать руководство пользователя разработанного программного решения.

Руководство пользователя

Вам необходимо разработать руководство пользователя для вашего настольного приложения, которое описывает последовательность действий для выполнения всех функций вашей системы. При подготовке документации старайтесь использовать живые примеры и скриншоты вашей системы для более наглядного пояснения шагов работы с различным функционалом.

3.2. Критерии оценивания выполнения задания демонстрационного экзамена

3.2.1. Порядок оценки

Общее максимальное количество баллов за выполнение задания демонстрационного экзамена одним студентом, распределяемое между модулями задания дано в таблице 2.

Таблица 2

Раздел	Критерий	Оценки		
		Judgment (если это применимо)	Объективная	Общая
A	Системный анализ и проектирование	0,80	11,20	**12,0**
B	Разработка программного обеспечения	0,00	26,00	**26,00**
C	Стандарты разработки программного обеспечения	1,10	5,90	**7,00**
D	Документирование программных решений	4,00	1,00	**5,00**
Итого		5,90	44,10	**50,00**

Критерии оценки задания демонстрационного экзамена основываются на следующем:

- соблюдение правил безопасного выполнения работ и требований охраны труда;
- подготовка к работе, организация рабочего места;
- качество выполнение работ в соответствии с заданием и техническими требованиями к качеству результатов работ;

- полнота и скорость выполнения работ;
- учитываются навыки:

- разработки программного обеспечения;
- системного анализа и проектирования;
- документирования программного решения;
- мобильной разработки;

3.2.2. Порядок перевода баллов в систему оценивания.

Перевод в оценку баллов, полученных за демонстрационный экзамен, рекомендуется проводить следующим образом:

от 0 % до 19,99 % от максимального количества баллов означает оценку «неудовлетворительно»;

от 20,00 % до 39,99% - означает оценку «удовлетворительно»;

от 40,00% до 69,99% - означает оценку «хорошо»;

от 70,00% до 100 % - означает оценку «отлично».

4. ПОРЯДОК ОРГАНИЗАЦИИ И ПРОВЕДЕНИЯ ЗАЩИТЫ ДИПЛОМНОЙ РАБОТЫ (ДИПЛОМНОГО ПРОЕКТА)

4.1. Общие положения

Государственная итоговая аттестация является частью оценки качества освоения основной профессиональной образовательной программы по специальности 09.02.07 Информационные системы и программирование и является обязательной процедурой для выпускников очной формы обучения, завершающих освоение образовательной программы.

К итоговым аттестационным испытаниям, входящим в состав государственной итоговой аттестации, допускаются обучающиеся, успешно завершившие в полном объеме освоение основной образовательной программы по специальности 09.02.07 Информационные системы и программирование.

Необходимым условием допуска к ГИА является представление документов, подтверждающих освоение выпускниками общих и профессиональных компетенций при изучении теоретического материала и прохождении практики по каждому из основных видов профессиональной деятельности.

4.2. Примерная тематика дипломных работ (проектов) по программе подготовки специалистов среднего звена для специальности 09.02.07 «Информационные системы и программирование» Квалификация: программист

1. Мониторинг и администрирование информационной системы заказов и установки радиокоммуникаций предприятия
2. Мониторинг и администрирование информационной системы заказов установки газового оборудования компании
3. Проектирование, разработка и администрирование модуля «Организация и хранение данных» информационной системы
4. Мониторинг и администрирование информационной системы хранения и продажи продукции на оптовом складе
5. Внедрение и особенности администрирования аналитических хранилищ данных

компании
6. Мониторинг и администрирование информационной системы модуля «Информационная безопасность»
7. Проектирование, разработка и администрирование информационной системы отдела разработки программного обеспечения летательных аппаратов
8. Разработка и администрирование модуля системы управления реестром конструкторской документации предприятия
9. Разработка автоматизированной системы информирования персонала (для конкретной организации).
10. Разработка системы разграничения доступа к сетевым ресурсам локальной вычислительной сети на базе Windows Server.
11. Разработка системы разграничения доступа к сетевым ресурсам локальной вычислительной сети на базе CentOS Server.
12. Разработка автоматизированной информационной системы «Комплекс автоматизированного контроля текущей успеваемости студентов» (для образовательной организации).
13. Разработка автоматизированной информационной системы ведения учета земельных участков
14. Разработка программного обеспечения представления информации базы данных радиоэлектронного мониторинга земной поверхности
15. Мониторинг и администрирование отдела тестирования научно-производственного предприятия
16. Мониторинг и администрирование информационной системы «Сеть заправочных станций компании
17. Разработка и особенности администрирование системы «Документооборот предприятия
18. Разработка и администрирование модуля «Закупки запчастей» информационной системы «Гамма»
19. Разработка прикладного программного решения сервисного центра (для конкретной организации
20. Разработка прикладного программного решения «Учета трудоустройства выпускников техникума» (для образовательной организации)
21. Разработка прикладного программного решения информирования персонала (для конкретной организации)
22. Разработка прикладного программного решения по продаже и аренде недвижимости на основе веб-технологий (на примере…)
23. Разработка автоматизированной системы выявления типичных неисправностей компьютера и способов их устранения
24. Разработка клиент-серверной информационной системы учета, контроля и перемещения измерительных приборов производства
25. Разработка программного обеспечения нейросети для классификации радиостанций
26. Разработка клиент-серверной информационной системы учета, контроля и перемещения измерительных приборов производства
27. Разработка OPC сервера для учёта и контроля движения нефтепродуктов в топливопроводящей сети
28. Разработка CRM системы
29. Реализация библиотеки для работы с платами мультиплексного канала информационного обмена (указать название технологии или платы)
30. Разработка программного обеспечения для системы регулирования пропускного движения
31. Разработка компьютерной игры
32. Разработка веб-сайта системы контроля версий на основе сервисной технологии

(SAAS, PAAS и т.д.)
33. Разработка программного обеспечения для системы регулирования пропускного движения
34. Разработка веб-сайта системы контроля версий
35. Разработка модуля web-сайта колледжа (СПО) для специальности 09.02.07 «Информационные системы и программирование» (для образовательной организации)
36. Разработка прикладного программного решения онлайн магазина продаж на платформе «1С:Предприятие 8.3» (на примере…)
37. Разработка модуля web -сайта колледжа (СПО) для специальности Информационные системы и программирование (для образовательной организации)
38. Разработка модуля web -сайта колледжа (СПО) для специальности Технология машиностроения (для образовательной организации)
39. Модификация автоматизированной информационной Системы «Учета трудоустройства выпускников техникума» (для образовательной организации)
40. Модификация автоматизированной информационной системы «Формирование междисциплинарных тестовых онлайн заданий» (для образовательной организации)

4.3. Структура и содержание дипломной работы

Дипломный проект (работа) - завершающий этап обучения, который аккумулирует знания и умения, приобретенные в процессе обучения, и позволяет студентам продемонстрировать профессиональную компетентность.

ФГОС СПО определяет следующее требования к выпускнику по итогам освоения образовательно программы: овладение основными видами профессиональной деятельности (ВПД), общими (ОК) и профессиональными (ПК) компетенциями в соответствии с квалификационной характеристикой.

Выпускник должен быть готов к профессиональной деятельности как будущий специалист, который сможет применить полученные теоретические знания и практические умения для выполнения производственных задач на предприятиях соответствующей отрасли.

Дипломные работы должны быть выполнены в строгом соответствии с требованиями к выполнению текстовых документов, подписаны в соответствии с требованиями, содержать приложения, раскрывающие и дополняющие тему дипломного проекта.

Дипломный проект (работа) представляет собой законченную квалификационную работу, содержащую результаты самостоятельной деятельности студента в период производственной (преддипломной) практики и дипломного проектирования в соответствии с утвержденной тематикой.

Требования к дипломному проекту:

- соответствие тематики выпускной квалификационной работы содержанию одного или нескольких профессиональных модулей;
- обоснование выбора темы исследования, её актуальности,
- обзор опубликованной литературы по выбранной теме,
- изложение полученных результатов, их анализ, обсуждение и выводы,
- список использованной литературы и содержание;
- оформление в соответствии со стандартами ЕСПД.

Дипломная работа состоит из пояснительной записки и графической части. В пояснительной записке дается теоретическое и расчетное обоснование принятых в проекте решений и включает в себя расчётно-технологическую и экономическую аргументацию

актуальности данного проекта. В графической части принятое решение представлено в виде чертежей, схем, графиков, диаграмм.

Структура дипломного проекта состоит из последовательно расположенных следующих элементов:

1. титульный лист
2. индивидуальный план задание выполнения проекта
3. качественные характеристики деятельности студента при выполнении ДР (отзыв, рецензия)
4. содержание пояснительной записки дипломного проекта включает:

- введение
- основная часть
- заключение
- список использованных источников,
- приложения и (или) мультимедийная презентация.

На титульном листе указывается наименование вышестоящей организации, наименование образовательного учреждения, специальность, тема дипломного проекта, фамилия, имя отчество, подпись научного руководителя, консультанта (ов), фамилия, имя, отчество, подпись студента.

В содержании документа по разделам последовательно излагаются названия разделов, подразделов, названия пунктов и подпунктов пояснительной записки. Заголовки содержания должны точно повторять заголовки в тексте. Все заголовки указываются без точки в конце. Напротив, последнего слова каждого заголовка у правой границы листа содержания указывается номер страницы, с которых начинается каждый раздел, пункт или подпункт.

Во введении дается краткое обоснование исследуемой проблемы, целей, задач, методов и направлений разработки избранной темы. В нем содержится оценка современного состояния решаемой проблемы, основание и исходные данные для разработки темы. Также, во введении должна быть показана актуальность и новизна темы.

В основной части приводятся данные, отражающие сущность, методику и основные результаты выполнения дипломного проекта.

Основная часть содержит несколько разделов, каждый из которых в свою очередь делится на 2-3 пункта (параграфа). По содержанию основная часть корректируются в зависимости от профиля специальности, и включает в себя специальную, расчетную, технологическую и экономическую часть.

В специальной части приводится информация о технических требованиях, конструкции и назначении исследуемого объекта. В расчётной части выполняются расчёты, подтверждающие гипотезу данного исследования. Технологическая часть характеризует выбор методов, материалов, этапов и операций, необходимых для решения технологических аспектам данной проблемы. Экономическая часть обосновывает ожидаемые технико-экономическими показателями изготовления объекта исследования.

Каждая из частей занимает пропорциональную часть по отношению ко всему объему пояснительной записки.

В разделе «Заключение» пояснительной записки содержатся итоги работы, важнейшие выводы, к которым пришел автор характеризующие итоги работы в решении поставленных во введении задач, формулируются общие выводы по всем разделам, рассматривается и оценивается выполнение поставленных задач и достигнутые при этом результаты.

Список использованной литературы составляется в алфавитном порядке фамилий авторов или названий произведений (при отсутствии фамилий автора).

Графическая часть выпускных квалификационных работ должна быть выполнена на компьютере в программах Adobe Photoshop, CorelDRAW, Microsoft Word, Libre Office.

Пояснительная записка и графическая часть оформляются в единую папку, жестко брошюруются, имеют твердый переплет.

4.4. Порядок оценки результатов дипломного проектирования

Критерии оценки дипломной работы (дипломного проекта):

- соответствие названия работы ее содержанию, четкая целевая направленность;
- логическая последовательность изложения материала;
- глубина исследования и убедительность аргументации;
- конкретность представления практических результатов работы;
- соответствие оформления выпускной квалификационной работы методическим рекомендациям по оформлению квалификационной работы (дипломного проекта);

4.5. Порядок оценки защиты дипломного проекта (работы)

Результаты государственной итоговой аттестации определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и объявляются в тот же день после оформления в установленном порядке протоколов заседаний государственных экзаменационных комиссий.

Критерии оценки защиты выпускной квалификационной работы (дипломного проекта):

- грамотность и четкость доклада;
- точность ответов на вопросы;
- использование технических средств для сопровождения доклада.

При определении окончательной оценки за защиту дипломного проекта учитываются:

- доклад выпускника по каждому разделу дипломного проекта;
- ответы на вопросы;
- оценка руководителя;
- оценка рецензента.

Оценка «отлично» ставится за доклад, в котором в полном объеме освещены все разделы проекта, самостоятельно и уверенно сформулировано и доведено до сведения ГЭК содержание проекта, доклад построен последовательно и технически грамотно, четко и правильно даны ответы на все заданные вопросы ГЭК.

Оценка «хорошо» ставится за доклад, в котором не в полном объеме раскрыты разделы проекта, доклад самостоятелен и построен достаточно уверенно и грамотно, однако, допущены неточности при формулировке определений и неуверенность в ответах по заданным вопросам ГЭК.

Оценка «удовлетворительно» ставится за доклад, в котором не в полном объеме освещены все разделы проекта, последовательность нарушена, формулировки и определения доводятся недостаточно четко, допускаются ошибки и неточности в использовании технической терминологии, на заданные вопросы ГЭК не даны ответы.

Оценка «неудовлетворительно» ставится за доклад, в котором не раскрыты разделы проекта, не даны формулировки определений и понятий, допущены грубые ошибки при использовании технической терминологии, не сформулированы ответы на вопросы ГЭК.

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ ОСВОЕННОГО ВИДА ДЕЯТЕЛЬНОСТИ ПРИ ПРОВЕДЕНИИ ГИА

Таблица 1 – Оценочные средства для проверки освоенного вида деятельности при проведении ГИА

Профессиональные модули	**Подготовка дипломной работы (ДР)**	**Защита ДР**	**ДЭ**
Разработка модулей программного обеспечения для компьютерных систем	+	+	+
Оценочные средства (теоретические вопросы)			
Вопрос	**Эталон ответа**		
Графические задание алгоритма	Способ представления алгоритма с помощью геометрических фигур,		
Объектно-ориентированная программа	Совокупность объектов и способов их взаимодействия		
Назначение главного меню	Осуществление функций управления при разработке программ		
Назначение панели инструментов	Меню команд быстрого доступа к командам, содержащее набор кнопок, функции которых эквивалентны наиболее часто употребляемым командам Главного меню		
Назначение палитры компонентов	Содержит пиктограммы, которые представляют компоненты VCL , которые можно включить в приложение		
Окно редактора кода	Окно содержащее текст программы на языке Object Pascal, связанный с каждой формой в приложении:		
Метод Application.Run	Подключает автоматический цикл обработки сообщений Windows к приложению		
Свойство Align	Определяет способ выравнивания компонента внутри контейнера:		
Объект Memo1	Многострочный редактор, содержащий несколько строк текста		
Свойство ScrollBars	Задает в поле редактирования полосы прокрутки		
Невизуальные компоненты	Таймеры, компоненты для работы с базами данных, списки изображений		
Файлы описания форм	Текстовые файлы с расширением DFM, в котором сохраняются значения свойств формы и ее компонентов		
Программный модуль (Unit)	Содержит Все относящиеся к форме объявления и методы обработки событий, написанные на языке Object Pascal		
Интерфейсная секция модуля (interface)	Содержит: Список стандартных модулей библиотеки VCL, Раздел описания типов, Раздел описания переменных (объявление собственно объекта формы)		
Программный файл проекта	Главная часть приложения, с которой начинается выполнение программы и которая обеспечивает инициализацию других модулей		
Осуществление интеграции программных модулей	+	+	+
Характеристика программы и программного обеспечения	Программа - это последовательность команд компьютера, приводящая к решению задачи. Основными характеристиками программ являются: • алгоритмическая сложность (логика алгоритмов обработки информации); • состав и глубина проработки реализованных функций обработки; • полнота и системность функций обработки; • объём файлов программ; • требования к операционной системе и техническим средствам обработки со стороны программного средства;		

- объём дисковой памяти;
- размер оперативной памяти для запуска программ;
- тип процессора;
- версия операционной системы;
- наличие вычислительной сети и др.

Программное обеспечение — это совокупность программ, обеспечивающих функционирование компьютеров и решение с их помощью задач предметных областей. Программное обеспечение представляет собой неотъемлемую часть компьютерной системы, является логическим продолжением технических средств и определяет сферу применения компьютера.

Характеристика программного продукта и его специфика

Программный продукт — комплекс взаимосвязанных программ для решения определённой проблемы (задачи) массового спроса, подготовленный к реализации как любой вид промышленной продукции.

Некоторые характеристики программных продуктов:

- Алгоритмическая сложность (логика алгоритмов обработки информации).
- Состав и глубина проработки реализованных функций обработки.
- Полнота и системность функций обработки.
- Объём файлов программ.
- Требования к операционной системе и техническим средствам обработки со стороны программного средства.
- Объём дисковой памяти.
- Размер оперативной памяти для запуска программ.
- Тип процессора.
- Версия операционной системы.
- Наличие вычислительной сети и другие.

Специфика программных продуктов заключается в уникальности процесса разработки алгоритмов и программ, зависящего от характера обработки информации и используемых инструментальных средств. Также специфика заключается в том, что их эксплуатация должна выполняться на правовой основе — лицензионные соглашения между разработчиком и пользователями с соблюдением вторских прав разработчиков программных продуктов.

Качество программного продукта

Качество программного продукта — это комплекс характеристик, определяющих способность выполнять возложенные на него функции.

Некоторые критерии качества программного продукта:

- Функциональность. Продукт признаётся функциональным, если выполняет возложенные на него задачи и отвечает заданным потребностям пользователей.
- Надёжность. Под надёжностью понимают бесперебойное выполнение возлагаемых на продукт задач в заданных условиях в течение установленного времени.
- Юзабилити (удобство использования). Этот параметр характеризует степень удобства продукта для пользователей, его наглядность, лёгкость эксплуатации и изучения.
- Эффективность. Параметру соответствует степень обеспечения продуктом необходимой производительности при заданных условиях.
- Удобство сопровождения. Этот показатель характеризует простоту анализа, тестирования, коррекции компонентов продукта, его обслуживания, а также степень адаптации к новым условиям.
- Портативность. Степень лёгкости переноса продукта на другую платформу.
- Совместимость. Способность программных компонентов взаимодействовать друг с другом.
- Защищённость. Минимизация угроз, связанных с несанкционированным чтением, изменением информации и т. д..

Аттестация программных систем

Процесс, целью которого является доказательство того, что в результате разработки системы достигли тех целей, которые планировали достичь благодаря её использованию. Иными словами, аттестация — это проверка соответствия системы ожиданиям заказчика.

Принципы системного проектирования	• Качество и эффективность. Главенствующая роль отводится эффективности при обеспечении заданного качества. • Комплексность. Учитываются все значимые факторы и характеристики. • Анализ всех стадий жизненного цикла объекта проектирования. Включает в себя проектирование, опытную отработку, производство, наземную и лётную эксплуатацию, включая снятие с эксплуатации. • Единство составных частей. Эффективность решения задачи зависит от того, насколько полно учтены все связи между частями рассматриваемого объекта и с взаимодействующими с ним другими объектами. • Учёт социально-экономических и экологических последствий. Системное проектирование основывается на тщательном совместном рассмотрении объекта проектирования и процесса проектирования.
Модульное программирование	Технология разработки компьютерных программ, основанная на представлении программируемой задачи в виде структуры подзадач, для каждой из которых разрабатывается программный модуль — поименованная программная компонента с заданным интерфейсом
Программная ошибка	Ошибка в программе или в системе, приводящая к неожиданному поведению программы и, как следствие, выдаче некорректного результата
Структурное тестирование	Стратегия тестирования, которая предполагает создание тестов на основе структуры системы и её реализации. Такой подход иногда называют тестированием методом «белого ящика».
Отладка программ	Процесс поиска, локализации и исправления ошибок в компьютерной программе.
Сопровождение программ	Процесс улучшения, оптимизации и устранения дефектов ПО после передачи его в эксплуатацию.
Инструментальные средства программирования	Программные инструменты, предназначенные для обеспечения полного цикла проектирования программного продукта (написание текста программы, компиляция, компоновка, отладка, тестирование, сопровождение и др.).
Оболочки экспертных систем	Готовая программная среда, которая может быть приспособлена к решению определённой проблемы путём создания соответствующей базы знаний.
Компоненты инструментальных систем технологий программирования	1. База данных разработки (репозиторий). Центральное компьютерное хранилище информации, связанной с проектом (разработкой) программного средства в течение всего его жизненного цикла. 2. Инструментарий. Набор инструментов, определяющий возможности, предоставляемые системой коллективу разработчиков. Обычно этот набор является открытым: помимо минимального набора (встроенные инструменты), он содержит средства своего расширения (импортированными инструментами). 3. Интерфейсы. Разделяются на пользовательский и системные. Пользовательский интерфейс обеспечивает доступ разработчикам к инструментарию (командный язык и т. п.), реализуется оболочкой системы. Системные интерфейсы обеспечивают взаимодействие между инструментами и их общими частями.
Имитационное моделирование	Распространенная разновидность аналогового моделирования, реализуемого с помощью набора математических средств, специальных компьютерных программ-симуляторов и особых IT, позволяющих создавать в памяти компьютера процессы-аналоги, с помощью которых можно провести целенаправленное исследование структуры и функций реальной системы в режиме ее «имитации», осуществить оптимизацию некоторых ее параметров.
Проверка достоверности модели	Проверка достоверности модели заключается в определении степени соответствия модели реальности. Выделяют несколько видов проверок достоверности модели: 1. Валидация модели. Подтверждение того, что построенная модель достаточно хорошо представляет объект исследования. 2. Верификация модели. Подтверждение того, что модель реализована в среде моделирования именно так, как спроектирована, без искажений. 3. Тестирование модели. Нахождение неточностей и ошибок модели. При тестировании на вход модели подаются заранее подготовленные наборы входных данных, а на выходе проверяется соответствие выходных данных ожидаемым результатам.

Сопровождение и обслуживание программного обеспечения компьютерных систем	+	+	+

Стандарты в области информационных систем.	• ГОСТ Р 52294-2004 «Информационная технология. Управление организацией. Электронный регламент административной и служебной деятельности. Основные положения». Стандарт распространяется на автоматизированные системы обработки информации и управления учреждений, предприятий и организаций. • ГОСТ Р 52919-2008 «Информационная технология. Методы и средства физической защиты. Классификация и методы испытаний на огнестойкость. Комнаты и контейнеры данных». Стандарт распространяется на средства физической защиты от воздействия огня оборудования и данных информационно-коммуникационных технологий. • ГОСТ Р 53114-2008 «Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения». Стандарт устанавливает основные термины, применяемые при проведении работ по стандартизации в области обеспечения информационной безопасности в организации. • ГОСТ Р МЭК 60950-2002 «Безопасность оборудования информационных технологий». Стандарт устанавливает требования безопасности, а также общие нормы, правила и методы испытания оборудования информационных технологий. • ГОСТ Р 59793-2021 «Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания». Стандарт распространяется на автоматизированные системы, используемые в различных видах деятельности, и устанавливает стадии и этапы их создания.
Методы предотвращения угроз надежности.	• Предотвращение ошибок проектирования в CASE-технологиях. Например, за счёт структурного программирования. • Систематическое тестирование. Это процесс выполнения программ с намерением найти ошибки. • Обязательная сертификация.
Способы оперативного повышения надежности ПО.	• Временная, информационная и программная избыточность. • Резервирование программ. • N-версионное программирование. • Верификация и валидация программ с последующей коррекцией. • Программирование с выявлением ошибок проекта. Цель — обнаружить остаточные ошибки во время выполнения программы, чтобы предотвратить критически опасные отказы системы. • Метод контрольных функций как экономный способ повышения надёжности ПО.
Понятия «компьютерный вирус» и «программа - антивирус».	Компьютерный вирус — это специально написанная программа, как правило, небольшая по размерам, способная самопроизвольно присоединяться к другим программам, создавать свои копии и внедрять их в файлы, системные области дисков и в вычислительные сети. Цель вируса — нарушить работу программ, испортить файлы и каталоги, создать помехи в работе на компьютере. Антивирусная программа — это специализированная программа для обнаружения компьютерных вирусов, а также нежелательных программ и восстановления заражённых (модифицированных) такими программами файлов, а также для профилактики — предотвращения заражения (модификации) файлов или операционной системы вредоносным кодом.

Режимы проверки антивирусных программ.	1. Проверка в режиме реального времени (постоянная проверка). Обеспечивает непрерывность работы антивирусной защиты. Проверяются все действия, совершаемые другими программами и самим пользователем, на предмет вредоносности, вне зависимости от их исходного расположения. 2. Проверка по требованию. Пользователь указывает, какие файлы, каталоги или области диска необходимо проверить, и время, когда нужно произвести такую проверку — в виде расписания или разового запуска вручную. 3. Сканирование при обращении к файлам. При открытии файла пользователь и операционная система автоматически сканируют его, чтобы выявить, что файл содержит вредоносный программный код. 4. Сканирование файлов по расписанию. Пользователь указывает, на каких дисках и каталогах следует проверять вредоносный программный код и составляет расписание для выполнения такой проверки. 5. Сканирование сетевого трафика. Антивирус способен сканировать сетевой трафик, поступающий на компьютер из интернета, и уходящий из компьютера.
Средства шифрования.	Программы и устройства, которые шифруют и дешифруют информацию, а также проверяют, вносились ли в неё изменения. Основные виды средств шифрования: 1. Симметричное. Для кодирования и расшифровки информации используется один и тот же открытый ключ, доступ к которому может получить любой пользователь. Такой способ достаточно уязвим с точки зрения безопасности данных, поэтому он чаще применяется не для передачи, а для хранения информации. 2. Асимметричное. Более сложный вид шифрования данных, при котором для кодирования и дешифровки используются разные ключи. При этом ключ, который нужен для разгадывания кода, — закрытый, то есть им владеет только нужный получатель. Такой вид шифрования информации считается более надёжным, и его алгоритмы применяются, например, в системе цифровых подписей и блокчейне. 3. Хеш-функция. Особенность этого вида шифрования в том, что он не имеет обратной силы, то есть хеш-функцию невозможно раскодировать. Исходные данные можно преобразовать миллион раз, и результат всегда будет одинаковый. Однако, если внести изменение в первоначальную информацию, изменится и хеш-функция. Благодаря этому свойству использование этого вида шифрования чаще всего встречается при хранении паролей на сайтах. Средства шифрования бывают: • Программные. Устанавливаются на устройство отдельно и работают в оперативной памяти компьютера. • Программно-аппаратные. Вшиты в специальное устройство (обычно токен). Все операции происходят на этом устройстве и скрыты от оперативной памяти компьютера, к которому он подключён. Такой вид средств шифрования считается более безопасным, чем программный.
Разновидности атак на протоколы.	• Межсетевая атака (Man-in-the-middle). Злоумышленник встраивается в канал общения между участниками, внося изменения в передаваемые сообщения или перенаправляя их. • Атака повтором сообщения (replay attack). Повторное использование ранее переданного сообщения или какой-либо его части в текущем сеансе протокола. • Атака подмены типа (type flaw attack). Похожа на атаку повтором, но сообщение передаётся на другом раунде протокола, тем самым изменяется его значение в протоколе. • Атака с параллельными сеансами (parallel-session attack). Злоумышленник инициирует несколько параллельных сеансов с участниками и передаёт сообщения из одного сеанса в другой. • Атака открытого ключа (Public Key). Направлена на компрометацию ключей шифрования для доступа к защищенным данным. • Атака переполнения буфера (Buffer Overflow). Направлена на заполнение памяти уязвимой программы, которая может привести к сбою системы или внедрению вредоносного кода.

Функция и роль шифрования.	Функция шифрования — обратимое преобразование информации в целях сокрытия от неавторизованных лиц с предоставлением в это же время авторизованным пользователям доступа к ней. Роль шифрования заключается в обеспечении трёх состояний безопасности информации: 1. Конфиденциальность. Шифрование используется для скрытия информации от неавторизованных пользователей при передаче или при хранении. 2. Целостность. Шифрование предотвращает изменение информации при передаче или хранении. 3. Идентифицируемость. Шифрование используется для аутентификации источника информации и предотвращения отказа отправителя информации от того факта, что данные были отправлены именно им.
Методы и средства защиты компьютерных систем	1. Организационные (административные). Меры, которые связаны с порядком доступа к информационной системе и правилами работы. Например, парольная политика, которая регламентирует сложность используемых паролей. 2. Физические. Реальные «преграды» на пути нарушителя или реализации угрозы. К ним относятся системы сигнализаций и видеонаблюдения, контроля и управления доступом. Аппаратные и программно-аппаратные (технические). Интегрированные или самостоятельные электронные устройства, которые защищают внутренние элементы информационных систем за счёт разграничения и управления доступом к этим ресурсам. 4. Программные. Решения, которые повышают защищённость данных за счёт обнаружения вредоносного кода, мониторинга трафика и сигнализирования об аномалиях. Например, антивирусы, брандмауэры, средства обнаружения вторжений. 5. Криптографические. Средства защиты, которые преобразуют и шифруют информацию криптографическими методами, используют токены, электронную подпись.
Защита на основе вирусных сигнатур	Защита на основе вирусных сигнатур — это метод работы антивирусов, при котором программа, просматривая файл или пакет, обращается к базе данных с известными вирусами. Сигнатурный анализ базируется на сканировании и анализе уникальной последовательности байтов вредоносного программного обеспечения. В случае обнаружения сходства с вирусными сигнатурами программа идентифицирует вирус и сигнализирует об этом.
Вирусы по способу заражения	1. Резидентные вирусы при заражении компьютера оставляют в оперативной памяти свою резидентную часть, которая затем перехватывает обращение операционной системы к другим объектам заражения, внедряется в них и выполняет свои разрушительные действия вплоть до выключения или перезагрузки компьютера. 2. Нерезидентные вирусы не заражают оперативную память персонального компьютера и являются активными ограниченное время.
Приемы выявления уязвимостей	1. Анализ исходных текстов. Предполагает экспертный анализ исходных текстов программы с целью поиска и исправления ошибок, которые были допущены на этапе её разработки. 2. Использование специализированных программных средств. Сканеры безопасности или системы анализа защищённости позволяют обнаруживать уязвимости на основе активного и пассивного методов. При помощи пассивного метода осуществляется сбор информации о настройках ПО, и на основе этих данных делается вывод о наличии или отсутствии в системе уязвимостей. Активные методы имитируют информационные атаки, и затем на основе анализа результатов делается вывод о наличии уязвимостей в системе. 3. Статический анализ. Предполагает поиск потенциально опасных конструкций в исходном коде программного обеспечения. Этот вид анализа позволяет выявлять технологические уязвимости, которые возникают на этапе разработки ПО. 4. Динамический анализ. Позволяет выявлять не только технологические, но и эксплуатационные уязвимости, связанные с неправильной настройкой ПО. К таким уязвимостям можно отнести использование слабых и нестойких к угадыванию паролей доступа, наличие неиспользуемых учётных записей, неправильно настроенные функции безопасности ПО и др. Фаззинг. Это процесс передачи на вход программе намеренно некорректных

	данных с целью вызова ситуации сбоя или ошибки.
Способы аутентификации в информационных системах	• Парольная аутентификация. Пользователь вводит уникальный код (пароль), который проверяется системой на соответствие сохранённым данным. • Биометрическая аутентификация. Основана на уникальных физических или поведенческих характеристиках пользователя, например, отпечаток пальца, голос или сетчатка глаза. • Токенная аутентификация. Пользователь использует физическое устройство (токен), содержащее уникальный идентификатор, для получения доступа. Например, карты доступа или электронные ключи. • Многофакторная аутентификация. Комбинация двух или более различных методов аутентификации для повышения уровня безопасности. Например, пароль в сочетании с биометрическими данными.
Разработка, администрирование и защита баз данных	+ + +
Банк данных	Разновидность информационной системы, в которой реализованы функции централизованного хранения и накопления обработанной информации организованной в одну или несколько баз данных
База данных	Совокупность специальным образом организованных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области
СУБД	Комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования БД многими пользователями
Словарь данных	Подсистема банка данных, предназначенная для централизованного хранения информации о структурах данных, взаимосвязях файлов БД друг с другом, типах данных и форматах их представления, принадлежности данных пользователям, кодах защиты и разграничения доступа и т.п
Администратор базы данных	Лицо или группа лиц, отвечающих за выработку требований к БД, ее проектирование, создание, эффективное использование и сопровождение
Модель представления данных	Логическая структура данных, хранимых в базе данных
Сервер базы данных	Собственно СУБД и управление хранением данных, доступом, защитой, резервным копированием, отслеживанием целостности данных, выполнением запросов клиентов
Ссылочная целостность данных	Система и набор специальных правил, обеспечивающих единство связанных данных в базе данных
Хранимые процедуры	Набор основных действий и манипуляций с данными, хранятся на сервере, Программы "клиенты" способны их выполнять
Первичный ключ	Один атрибут или минимальный набор из нескольких атрибутов, значения которых в одно и тоже время не бывают одинаковыми, то есть однозначно определяют запись таблицы
Реляционная база данных	Набор отношений, связанных между собой, что обеспечивает возможность поиска одних кортежей по значению других
Внешний ключ	Столбец или группа столбцов таблицы, значения которых совпадают со значениями первичного ключа другой таблицы
Частичная функциональная зависимость	Если между A и B существует функциональная зависимость не ключевого атрибута от части составного ключа, то говорят, что между A и B она существует
Диаграммы "сущность-связь"	Разновидность блок-схемы, где показано, как разные «сущности» (люди, объекты, концепции и так далее) связаны между собой внутри системы.
Оптимизация обработки запросов	Процесс поиска наиболее эффективного способа выполнения запроса в системе управления реляционными базами данных (СУБД).