



Частное профессиональное образовательное учреждение
«Северо-Кавказский межотраслевой колледж»

ИНН/КПП 0600001944/060001001 ОГРН 1220600000455
386101, Республика Ингушетия, г. Назрань, пр-т. Базоркина, д. 3

ПРИНЯТО
на заседании учебно-методического
совета Протокол
от «13» апреля 2026 г. № 3

**КОМПЛЕКТ КОНТРОЛЬНО-ОЦЕНОЧНЫХ
СРЕДСТВ
МДК 02.01 Технология разработки программного
обеспечения**

Специальность: 09.02.07 Информационные системы и программирование

*Тип образовательной программы: Программа подготовки специалистов
среднего звена*

Квалификация: Программист

Форма обучения: очная

Назрань, 2026

Содержание

1. ОБЩИЕ ПОЛОЖЕНИЯ.....	3
2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МДК	4
3. ФОРМЫ И МЕТОДЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ	6
4. ПАСПОРТ КОМПЛЕКТА КОНТРОЛЬНО-ОЦЕНОЧНЫХ СРЕДСТВ ПО МДК.02.01....	7
5. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ ПО МДК.....	9
6. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО МДК	17
7. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ	18

1. ОБЩИЕ ПОЛОЖЕНИЯ

Комплект контрольно-оценочных средств (КОС) междисциплинарного курса МДК 02.01 Технология разработки программного обеспечения является частью ПМ.02 программы подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование (квалификация «Программист»).

ФОС предназначен для оценки достижения запланированных результатов обучения по междисциплинарному курсу и включает оценочные средства для проведения текущего контроля успеваемости и промежуточной аттестации.

Текущий контроль успеваемости осуществляется в пределах учебного времени, отведённого на изучение МДК. Результаты текущего контроля фиксируются в журнале учебных занятий по пятибалльной системе («5», «4», «3», «2»).

Промежуточная аттестация по МДК проводится в форме дифференцированного зачета. Дифференцированный зачет проводится после завершения освоения МДК в сроки, установленные календарным учебным графиком. Вопросы и задания доводятся до сведения обучающихся в течение первых двух месяцев от начала обучения.

2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ МДК

Общие компетенции (ОК)

Код	Формулировка компетенции
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях
ОК 04	Эффективно взаимодействовать и работать в коллективе и команде
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учётом особенностей социального и культурного контекста
ОК 06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
ОК 07	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях
ОК 08	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 09	Пользоваться профессиональной документацией на государственном и иностранном языках

Профессиональные компетенции

Код	Формулировка компетенции
ПК 2.1	Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

В результате освоения МДК обучающийся должен:

Иметь практический опыт:

- Разрабатывать и оформлять требования к программным модулям по предложенной документации.
- Разрабатывать тестовые наборы (пакеты) для программного модуля.
- Разрабатывать тестовые сценарии программного средства.
- Инспектировать разработанные программные модули на предмет соответствия стандартам кодирования.

Уметь:

Код	Умение
У1	Анализировать проектную и техническую документацию
У2	Использовать специализированные графические средства построения и анализа архитектуры программных продуктов

У3	Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов
У4	Определять источники и приёмники данных
У5	Проводить сравнительный анализ
У6	Выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace)
У7	Оценивать размер минимального набора тестов
У8	Разрабатывать тестовые пакеты и тестовые сценарии
У9	Выявлять ошибки в системных компонентах на основе спецификаций
У10	Использовать выбранную систему контроля версий
У11	Выполнять тестирование интеграции
У12	Организовывать постобработку данных
У13	Использовать приёмы работы в системах контроля версий
У14	Выполнять ручное и автоматизированное тестирование программного модуля
У15	Использовать методы для получения кода с заданной функциональностью и степенью качества
У16	Применять приёмы работы в системах контроля версий

Знать:

Код	Знание
31	Модели процесса разработки программного обеспечения
32	Основные принципы процесса разработки программного обеспечения
33	Основные подходы к интегрированию программных модулей
34	Виды и варианты интеграционных решений
35	Современные технологии и инструменты интеграции
36	Основные протоколы доступа к данным
37	Методы и способы идентификации сбоев и ошибок при интеграции приложений
38	Методы отладочных классов
39	Стандарты качества программной документации
310	Основы организации инспектирования и верификации
311	Встроенные и основные специализированные инструменты анализа качества программных продуктов
312	Графические средства проектирования архитектуры программных продуктов
313	Методы организации работы в команде разработчиков
314	Основы верификации и аттестации программного обеспечения
315	Методы и схемы обработки исключительных ситуаций
316	Основные методы и виды тестирования программных продуктов

3. ФОРМЫ И МЕТОДЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ

Формы текущего контроля по МДК:

- устный опрос (фронтальный, индивидуальный, комбинированный);
- тестирование (письменное или компьютерное);
- письменная проверка (ответы на вопросы, решение задач, составление схем, выполнение заданий для самостоятельной работы);
- практическая проверка (при проведении практических и лабораторных занятий);
- самоконтроль и взаимопроверка.

Критерии оценки профессиональных компетенций (фрагмент)

Компетенция	Критерии оценки	Методы оценки
ПК 2.1 Разрабатывать требования к программным модулям на основе анализа документации	«Отлично» – разработан и обоснован вариант интеграционного решения с помощью графических средств, учтены бизнес-процессы, оформление по стандартам, результат сохранён в системе контроля версий.	Экзамен (практическое задание), защита отчётов по практическим и лабораторным работам, наблюдение за деятельностью обучающегося.
ПК 2.4 Осуществлять разработку тестовых наборов и сценариев	«Отлично» – обоснован размер тестового покрытия, разработаны тестовый сценарий и пакеты, выполнено тестирование интеграции, ручное и автоматизированное, заполнены протоколы.	Экзамен (практическое задание), защита отчётов, наблюдение.
ПК 2.5 Производить инспектирование кода на соответствие стандартам кодирования	«Отлично» – знание стандартов кодирования более чем одного языка, выявлены все несоответствия в предложенном коде.	Экзамен (практическое задание по инспектированию кода), защита отчётов, наблюдение.

Оценки по ОК выставляются на основе экспертного наблюдения за выполнением работ (обоснованность постановки цели, эффективность использования информационных технологий, взаимодействие в команде, грамотность речи, соблюдение норм поведения и т.д.).

4. ПАСПОРТ КОМПЛЕКТА КОНТРОЛЬНО-ОЦЕНОЧНЫХ СРЕДСТВ ПО МДК.02.01

№	Наименование темы	Результаты обучения (умения, знания)	ПК, ОК	Текущий контроль успеваемости
Тема 1.1. Основные понятия и стандартизация требований к ПО				
1	Понятия требований, уровни требований	32, 39	ОК 01-09, ПК 2.1	Устный опрос, тестирование
2	Классификация требований: FURPS+. Методологии и стандарты	31, 34	ОК 01-09, ПК 2.1	Устный опрос, тестирование
3	Методы организации работы в команде разработчиков	31, 32, 313	ОК 01-09, ПК 2.1	Устный опрос, тестирование
4	Современные принципы и методы разработки ПО. Подходы к интеграции	31–37, 315	ОК 01-09, ПК 2.1	Устный опрос, тестирование
5	Системы контроля версий. Стандарты кодирования	39, 313	ОК 01-09, ПК 2.1	Устный опрос, тестирование
6	Анализ предметной области. Разработка и оформление технического задания	У1, У4, У5	ОК 01-09, ПК 2.1	Отчёт по практическому занятию
7	Построение архитектуры ПО. Изучение работы в системе контроля версий	У2, У10, У13, У15, У16	ОК 01-09, ПК 2.1	Отчёт по практическому занятию
Тема 1.2. Описание и анализ требований. Диаграммы IDEF				
8	Унифицированный язык моделирования UML. Назначение	31, 32, 39, 312, 313	ОК 01-09, ПК 2.1	Устный опрос, тестирование
9	Модель UML и её элементы	31, 32, 39, 312, 313	ОК 01-09, ПК 2.1	Устный опрос, тестирование
10	Диаграммы UML: общие и специальные	31, 32, 39, 312, 313	ОК 01-09, ПК 2.1	Устный опрос, тестирование
11	Описание и оформление требований (спецификация). Анализ требований	У1, У2, У4, У5, У15	ОК 01-09, ПК 2.1	Устный опрос, тестирование
12	Методологии моделирования IDEF. Синтаксис и семантика IDEF0	У1, У2, У4, У5, У15	ОК 01-09, ПК 2.1	Устный опрос, тестирование
13	Построение диаграмм вариантов использования и последовательности	У1, У2, У4, У5, У15	ОК 01-09, ПК 2.1	Отчёт по лабораторной работе
14	Построение диаграмм кооперации, развёртывания, деятельности, состояний, классов	У1, У2, У4, У5, У15	ОК 01-09, ПК 2.1	Отчёт по лабораторной работе
15	Построение диаграммы компонентов и диаграммы потоков данных	У1, У2, У4, У5, У15	ОК 01-09, ПК 2.1	Отчёт по лабораторной работе
Тема 1.3. Оценка качества программных средств				
16	Цели, задачи и виды тестирования. Стандарты качества. Меры и метрики	31–35, 37–39, 315	ОК 01-09, ПК 2.4	Устный опрос, тестирование

17	Тестовое покрытие. Тестовый сценарий, тестовый пакет	31–35, 37–39, 315	ОК 01-09, ПК 2.4	Устный опрос, тестирование
18	Анализ спецификаций. Верификация и аттестация ПО	311, 314	ОК 01-09, ПК 2.5	Устный опрос, тестирование
19	Оценка необходимого количества тестов. Разработка артефактов тестирования	У6–У9, У11, У12, У14	ОК 01-09, ПК 2.4	Отчёт по лабораторной работе
20	Оценка ПО с помощью метрик и инспекция кода на соответствие стандартам	У1, У5, У6, У9, У12	ОК 01-09, ПК 2.5	Отчёт по лабораторной работе

Самостоятельная работа обучающихся (темы для конспектирования):

- Анализ требований и их формализация (32, 39)
- Примеры систем контроля версий (39, 313)
- Стандарт кодирования C# (39)
- Основные этапы развития технологии разработки ПО (31, 32)
- Эволюция моделей жизненного цикла ПО (31, 32)
- Стандарты, регламентирующие процесс разработки ПО (31, 32, 39)
- Анализ проблемы и постановка задачи (312)
- Техническое задание – ГОСТ 34.602-89 (32, 39)

5. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ ПО МДК

5.1. Типовые задания для оценки знаний и умений

Тема 1.1. Основные понятия и стандартизация требований к ПО

Вопросы для устного опроса:

1. Приёмы формулирования и выявления требований.
2. Методы анализа требований.
3. Понятие требований, уровни требований. Нефункциональные требования.
4. Классификация требований: модель FURPS+.
5. Современные принципы и методы разработки ПО (нисходящее, модульное, структурное программирование, CASE, RAD).
6. Организация работы в команде разработчиков. Модель проектной группы MSF.
7. Ролевая структура проекта (IBM, объектно-ориентированные технологии).
8. Системы контроля версий: локальные, централизованные, децентрализованные (достоинства, недостатки, примеры).
9. Стандарты кодирования. Принципы разработки ПО.

Тестовые задания:

1. Отметьте определения понятия «требование»:

- a) исходные данные для проектирования АИС
- b) совокупность средств, методов и персонала для обработки информации
- c) условия и возможности, которым должна удовлетворять система

Правильный ответ: c

2. К какому уровню относится требование «сокращение срока выполнения заказов на предприятии»?

- a) бизнес-требования
- b) требования пользователей
- c) функциональные требования

Правильный ответ: a

3. Какие требования регламентируют поведение системы?

- a) системные требования
- b) функциональные требования
- c) нефункциональные требования

Правильный ответ: b

4. Что из перечисленного относится к нефункциональным требованиям?

- a) Система должна отправлять уведомления по email
- b) Время отклика системы не должно превышать 2 секунд
- c) Пользователь должен ввести логин и пароль
- d) Система должна формировать отчёт в формате PDF

Правильный ответ: b

5. Какой стандарт в России регламентирует содержание технического задания на разработку АС?

- a) ГОСТ 19.201-78
- b) ГОСТ 34.602-89
- c) ISO 9001
- d) IEEE 829

Правильный ответ: b

6. Что из перечисленного не является принципом разработки ПО?

- a) Модульность
- b) Абстрагирование
- c) Случайное тестирование

d) Инкапсуляция

Правильный ответ: с

7. Какая модель жизненного цикла ПО предполагает возврат на предыдущие этапы при обнаружении ошибок?

- a) Каскадная
- b) Итеративная
- c) Спиральная
- d) V-образная

Правильный ответ: с

8. Какие требования описывают ограничения на процесс разработки (инструменты, стандарты, бюджет)?

- a) Функциональные
- b) Ограничения (Constraints)
- c) Атрибуты качества
- d) Бизнес-правила

Правильный ответ: b

9. Что такое «золочение» (gold plating) продукта?

- a) Удаление лишних функций
- b) Добавление функций, не указанных в спецификации
- c) Оптимизация производительности
- d) Документирование кода

Правильный ответ: b

10. Какой метод сбора требований предполагает наблюдение за работой пользователя без активного вмешательства?

- a) Интервью
- b) Анкетирование
- c) Этнографический метод
- d) Мозговой штурм

Правильный ответ: с

11. Что из перечисленного является артефактом этапа анализа требований?

- a) Исходный код
- b) Спецификация требований (SRS)
- c) Тестовый сценарий
- d) Инсталляционный пакет

Правильный ответ: b

12. Какой тип требований определяет удобство использования системы?

- a) Надёжность
- b) Производительность
- c) Удобство использования (Usability)
- d) Поддерживаемость

Правильный ответ: с

13. Что такое «верификация требований»?

- a) Проверка соответствия требований потребностям пользователя
- b) Проверка требований на полноту, непротиворечивость, корректность
- c) Оценка стоимости реализации требований
- d) Приоритизация требований

Правильный ответ: b

Практические занятия № 1–4:

- Анализ предметной области. Разработка и оформление технического задания.
- Построение архитектуры программного средства. Изучение работы в системе контроля версий.

Тема 1.2. Описание и анализ требований. Диаграммы IDEF

Вопросы для устного опроса:

1. Унифицированный язык моделирования UML: назначение.
2. Виды сущностей и типы отношений UML.
3. Диаграммы UML: классификация, требования к оформлению.
4. Диаграммы вариантов использования, классов, автомата, деятельности, последовательности, коммуникации, компонентов, размещения.
5. Синтаксис и семантика языка IDEF0. Диаграммы IDEF0 (виды, текст, глоссарий). Отношения блоков.

Тестовые задания:

1. **Что означает аббревиатура UML?**
 - a) Unified Modeling Language
 - b) Universal Modeling Language
 - c) Unified Module Language
 - d) User Modeling Language*Правильный ответ: a*
2. **Какая диаграмма UML показывает взаимодействие объектов во времени?**
 - a) Диаграмма классов
 - b) Диаграмма последовательности
 - c) Диаграмма компонентов
 - d) Диаграмма деятельности*Правильный ответ: b*
3. **Какой элемент диаграммы вариантов использования обозначает внешнего по отношению к системе пользователя или систему?**
 - a) Вариант использования
 - b) Актор
 - c) Граница системы
 - d) Отношение включения*Правильный ответ: b*
4. **Отношение «extend» на диаграмме вариантов использования означает:**
 - a) Один вариант использования включает другой
 - b) Вариант использования расширяет другой при определённых условиях
 - c) Обобщение акторов
 - d) Зависимость между классами*Правильный ответ: b*
5. **Какой тип диаграммы UML предназначен для моделирования бизнес-процессов и алгоритмов?**
 - a) Диаграмма состояний
 - b) Диаграмма деятельности
 - c) Диаграмма пакетов
 - d) Диаграмма объектов*Правильный ответ: b*
6. **Какая диаграмма UML показывает физическую структуру развёртывания системы на узлах?**
 - a) Диаграмма компонентов
 - b) Диаграмма развёртывания
 - c) Диаграмма пакетов
 - d) Диаграмма объектов*Правильный ответ: b*
7. **Какой вид отношений между классами означает «часть-целое» (сильная форма)?**
 - a) Ассоциация

- b) Агрегация
- c) Композиция
- d) Наследование

Правильный ответ: c

8. **Что такое «вариант использования» (use case)?**

- a) Описание последовательности действий, которые система выполняет для достижения результата
- b) Графическое изображение классов
- c) Способ хранения данных
- d) Отчёт о тестировании

Правильный ответ: a

9. **Какая диаграмма UML моделирует поведение системы как конечного автомата?**

- a) Диаграмма последовательности
- b) Диаграмма состояний
- c) Диаграмма деятельности
- d) Диаграмма вариантов использования

Правильный ответ: b

10. **Какая диаграмма UML показывает зависимости между модулями (файлами, библиотеками)?**

- a) Диаграмма компонентов
- b) Диаграмма классов
- c) Диаграмма деятельности
- d) Диаграмма развёртывания

Правильный ответ: a

11. **Что обозначает кратность (multiplicity) на диаграмме классов?**

- a) Количество атрибутов класса
- b) Количество экземпляров одного класса, связанных с другим
- c) Количество методов класса
- d) Сложность класса

Правильный ответ: b

12. **Какой из перечисленных элементов не является сущностью UML?**

- a) Класс
- b) Интерфейс
- c) Процесс
- d) Пакет

Правильный ответ: c

13. **Отношение «зависимость» в UML обозначается:**

- a) Сплошной линией с ромбом
- b) Пунктирной линией со стрелкой
- c) Сплошной линией с треугольником
- d) Пунктирной линией без стрелки

Правильный ответ: b

14. **Для чего используется диаграмма потоков данных (DFD)?**

- a) Для моделирования бизнес-процессов
- b) Для описания движения данных между процессами, хранилищами и внешними сущностями
- c) Для проектирования базы данных
- d) Для тестирования интеграции

Правильный ответ: b

Лабораторные работы № 1–5:

1. Построение диаграммы вариантов использования и диаграммы

- последовательности.
2. Построение диаграммы кооперации и диаграммы развёртывания.
 3. Построение диаграммы деятельности, диаграммы состояний и диаграммы классов.
 4. Построение диаграммы компонентов.
 5. Построение диаграмм потоков данных.

Тема 1.3. Оценка качества программных средств

Вопросы для устного опроса:

1. Тестирование: организация процесса, оптимальная стратегия, основные определения, принципы.
2. Методы тестирования «белого ящика» и «чёрного ящика».
3. Тестовое покрытие.Arteфакты тестирования.
4. Верификация и аттестация программного обеспечения.

Тестовые задания:

1. **Что такое верификация ПО?**
 - a) Проверка соответствия продукта реальным потребностям пользователя
 - b) Проверка соответствия продукта формальным требованиям и стандартам
 - c) Измерение производительности системы
 - d) Оценка стоимости разработки

Правильный ответ: b
2. **Что такое валидация ПО?**
 - a) Проверка правильности синтаксиса кода
 - b) Подтверждение того, что продукт удовлетворяет потребностям заказчика
 - c) Проверка на наличие утечек памяти
 - d) Тестирование безопасности

Правильный ответ: b
3. **Какое тестирование выполняется разработчиком для проверки отдельных функций/методов?**
 - a) Системное тестирование
 - b) Интеграционное тестирование
 - c) Модульное (юнит) тестирование
 - d) Приёмочное тестирование

Правильный ответ: c
4. **Что такое «тестовое покрытие» (test coverage)?**
 - a) Количество написанных тестов
 - b) Мера, показывающая, какая часть кода или функциональности проверена тестами
 - c) Количество найденных дефектов
 - d) Время выполнения всех тестов

Правильный ответ: b
5. **Метод «белого ящика» основан на:**
 - a) Знании внутренней структуры и кода программы
 - b) Только спецификации требований
 - c) Случайном вводе данных
 - d) Тестировании через графический интерфейс

Правильный ответ: a
6. **Какая техника тестирования относится к «чёрному ящику»?**
 - a) Покрытие операторов
 - b) Покрытие решений
 - c) Эквивалентное разбиение

- d) Покрытие путей
Правильный ответ: с
7. **Что такое регрессионное тестирование?**
a) Тестирование нового функционала
b) Повторное тестирование ранее работавших функций после изменений
c) Тестирование производительности под нагрузкой
d) Тестирование удобства интерфейса
Правильный ответ: b
8. **Какой артефакт тестирования содержит пошаговое описание проверки?**
a) Чек-лист
b) Тест-кейс (тестовый случай)
c) Баг-репорт
d) План тестирования
Правильный ответ: b
9. **Что из перечисленного обычно входит в баг-репорт?**
a) Шаги воспроизведения
b) Ожидаемый и фактический результат
c) Серьёзность (severity)
d) Все перечисленное
Правильный ответ: d
10. **Какое тестирование проверяет взаимодействие между модулями?**
a) Модульное
b) Интеграционное
c) Системное
d) Приёмочное
Правильный ответ: b
11. **Что такое «стресс-тестирование»?**
a) Тестирование работы системы в условиях, превышающих нормальные нагрузки
b) Тестирование безопасности
c) Тестирование интерфейса
d) Тестирование документации
Правильный ответ: a
12. **Кто обычно проводит альфа-тестирование?**
a) Внешние пользователи
b) Внутренние сотрудники компании-разработчика
c) Сторонняя тестирующая организация
d) Заказчик
Правильный ответ: b
13. **Что такое «инспекция кода»?**
a) Автоматический анализ кода на наличие ошибок
b) Формальная проверка кода группой разработчиков на соответствие стандартам и выявление дефектов
c) Выполнение модульных тестов
d) Сбор метрик сложности кода
Правильный ответ: b
14. **Какой вид тестирования направлен на проверку удобства и понятности интерфейса?**
a) Функциональное тестирование
b) Юзабилити-тестирование
c) Нагрузочное тестирование
d) Тестирование безопасности
Правильный ответ: b

15. Какой метод тестирования использует таблицы решений?

- a) Белый ящик
- b) Чёрный ящик
- c) Серый ящик
- d) Мутационное тестирование

Правильный ответ: b

16. Что такое «дымовое тестирование» (smoke testing)?

- a) Тестирование производительности
- b) Минимальный набор тестов для проверки критической функциональности
- c) Тестирование безопасности
- d) Тестирование совместимости

Правильный ответ: b

Лабораторные работы № 6–7:

Разработка тестового сценария.

Оценка необходимого количества тестов. Разработка тестовых пакетов.

Оценка программных средств с помощью метрик.

Инспекция программного кода на предмет соответствия стандартам кодирования.

5.2. Критерии оценивания

5.2.1. Критерии оценивания устного ответа

Оценка	Характеристика ответа
«5»	Ответ правильный, полный, логически выстроен, литературным языком.
«4»	Ответ правильный, полный, но есть отдельные затруднения в формулировке выводов.
«3»	Ответ в основном правильный, но схематичный или с нарушениями последовательности, неполный.
«2»	Непонимание основного содержания, грубые ошибки, отсутствие логики и обобщений.

5.2.2. Критерии оценивания тестовых заданий

Оценка	Процент выполнения
«5»	90–100 %
«4»	70–89,9 %
«3»	50–69,9 %
«2»	0–49,9 %

5.2.3. Критерии оценивания выполнения заданий на лабораторных и практических занятиях

Оценка	Характеристика выполнения
«5»	Работа выполнена полностью и правильно, сделаны верные выводы.
«4»	Работа выполнена правильно с 1–2 несущественными ошибками, исправленными по требованию преподавателя.
«3»	Работа выполнена не менее чем наполовину или допущены 3–4 существенные ошибки.
«2»	Допущено 5 и более существенных ошибок, которые обучающийся не может исправить.

5.2.4. Общая классификация ошибок

Грубые ошибки:

- незнание основных понятий, законов;
- неумение выделить главное, обобщить;
- неумение применить знания для решения задач;
- неумение пользоваться справочной литературой;
- нарушение техники безопасности.

Негрубые ошибки:

- неточность формулировок, определений;
- недостаточно продуманный план ответа;
- нерациональные методы работы с литературой.

6. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО МДК

Вопросы к зачету по МДК.02.01

1. Приёмы формулирования и выявление требований.
2. Методы анализа требований.
3. Понятие требований, уровни требований. Нефункциональные требования.
4. Требования к системе: классификация FURPS+.
5. Современные принципы и методы разработки программных приложений (нисходящее, модульное, структурное программирование, CASE, RAD).
6. Организация работы в команде разработчиков. Модель проектной группы MSF.
7. Ролевая структура проекта (IBM).
8. Системы контроля версий: локальные, централизованные, децентрализованные.
9. Стандарты кодирования. Принципы разработки ПО.
10. Унифицированный язык моделирования UML: назначение.
11. Виды сущностей UML.
12. Типы отношений UML.
13. Диаграммы UML: определение, классификация, требования по оформлению.
14. Диаграммы UML: вариантов использования, классов, автомата, деятельности.
15. Диаграммы UML: последовательности, коммуникации, компонентов, размещения.
16. Специальные диаграммы UML.
17. Синтаксис и семантика языка IDEF0.
18. Диаграммы IDEF0 (виды диаграмм, текст, глоссарий).
19. Отношения блоков на диаграммах IDEF0.
20. Тестирование: организация процесса, оптимальная стратегия, основные определения, принципы.
21. Методы тестирования «белого ящика» (покрытие операторов, решений, условий и др.).
22. Методы тестирования «чёрного ящика».
23. Тестовое покрытие.
24. Артефакты тестирования (стратегия, план, чек-лист, баг-репорт, тестовый сценарий, тестовый случай, матрица соответствия, тестовый набор).
25. Верификация и аттестация программного обеспечения.

Критерии:

- «5» – задание выполнено полностью, код рабочий, использованы все необходимые конструкции, отладка проведена, отчёт оформлен.
- «4» – задание выполнено, но есть мелкие недочёты.
- «3» – задание выполнено частично, есть ошибки.
- «2» – задание не выполнено.

7. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Основные печатные издания

1. Рудаков А. В. Технология разработки программных продуктов: учебник для студентов среднего профессионального образования. – М.: Издательский центр «Академия», 2018.

Основные электронные издания

1. Гагарина, Л. Г. Технология разработки программного обеспечения: учебное пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул; под ред. Л.Г. Гагариной. — Москва: ФОРУМ: ИНФРА-М, 2022. — 400 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0812-9. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1794453> – Режим доступа: по подписке.

Дополнительные источники

1. Зубкова Т.М. Технология разработки программного обеспечения. – Санкт-Петербург: Лань, 2021. – 252 с.

Интернет-ресурсы

- <http://www.intuit.ru> – Интернет-университет информационных технологий (ИНТУИТ.ру)
- <http://programm.ws/index.php> – Учебники по программированию
- <https://metanit.com> – Сайт по программированию