

# Аграрно-экономический техникум



Первый проректор

М.Д. Мукайлов

« 24 » апреля 2025 г.

**«МДК.02.02. Инструментальные средства разработки программного обеспечения»**

### 09.02.07 «Информационные системы и программирование»

*Срок обучения СПО по ППСЗ – 3 г.10 м.*

# Махачкала 2025г

Рабочая программа дисциплины разработана на основе Федерального государственного образовательного стандарта по специальности (профессии) среднего профессионального образования для специальности **09.02.07**

**«Информационные системы и программирование», утвержденного приказом Министерства образования и науки Российской Федерации от 9 декабря 2016 г. № 1547.**

Организация-разработчик: ФГБОУ ВО «Дагестанский государственный аграрный университет имени М.М. Джамбулатова» Аграрно-экономический техникум.

**СОГЛАСОВАНО:**



Директор АЭТ

подпись

Магомедов Д.А.

Одобрено на заседании ПЦК  
Общепрофессиональных и специальных  
дисциплин по специальности 09.02.07  
«Информационные системы и  
программирование»  
«14» апреля 2025г., протокол № 8

Председатель ПЦК



Рабданова З.К.

**СОГЛАСОВАНО:**

Директор Компании Солог- IT, Интернет решения



Салихов А.Б.

Ф.И.О.

## **СОДЕРЖАНИЕ**

<b>1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ МЕЖДИСЦИПЛИНАРНОГО КУРСА</b>	<b>4</b>
<b>2. СТРУКТУРА И СОДЕРЖАНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА</b>	<b>6</b>
<b>3. УСЛОВИЯ РЕАЛИЗАЦИИ МЕЖДИСЦИПЛИНАРНОГО КУРСА</b>	<b>8</b>
<b>4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ МЕЖДИСЦИПЛИНАРНОГО КУРСА</b>	<b>9</b>

**1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ  
МЕЖДИСЦИПЛИНАРНОГО КУРСА**

**«МДК.02.02 Инструментальные средства разработки программного обеспечения»**

**1.1. Цель и планируемые результаты освоения профессионального модуля**

В результате изучения профессионального модуля студент должен освоить основной вид деятельности **Осуществление интеграции программных модулей** и соответствующие ему общие компетенции и профессиональные компетенции:

**1.1.1. Перечень профессиональных компетенций**

Вид деятельности: Осуществление интеграции программных модулей	
ПК 2.1.	Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент
ПК 2.2.	Выполнять интеграцию модулей в программное обеспечение
ПК 2.3.	Выполнять отладку программного модуля с использованием специализированных программных средств
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

1.1.2. В результате освоения профессионального модуля студент должен:

Иметь практический опыт	модели процесса разработки программного обеспечения; основные принципы процесса разработки программного обеспечения; основные подходы к интегрированию программных модулей; основы верификации и аттестации программного обеспечения
уметь	использовать выбранную систему контроля версий; использовать методы для получения кода с заданной функциональностью и степенью качества
знать	модели процесса разработки программного обеспечения; основные принципы процесса разработки программного обеспечения; основные подходы к интегрированию программных модулей; основы верификации и аттестации программного обеспечения

## 1.2. Количество часов, отводимое на освоение профессионального модуля

Вид учебной работы	Объем в часах
Объем образовательной программы	64
в том числе:	
лекции	28
практические занятия	28
Самостоятельная работа	8
Промежуточная аттестация (дифференцированный зачет)	

## 2. СТРУКТУРА И СОДЕРЖАНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА

Наименование разделов и тем профессионального модуля (ПМ), междисциплинарных курсов (МДК)	Содержание учебного материала, лабораторные работы и практические занятия, внеаудиторная (самостоятельная) учебная работа обучающихся, курсовая работа (проект)	Объем в часах
<b>МДК.2.2 Интегральные средства разработки программного обеспечения</b>		
<b>Тема 2.2.1 Современные технологии и инструменты интеграции.</b>	<b>Содержание</b>	25
	1. Понятие репозитория проекта, структура проекта.	
	2. Виды, цели и уровни интеграции программных модулей. Автоматизация бизнес-процессов.	
	3. Выбор источников и приемников данных, сопоставление объектов данных.	
	4. Транспортные протоколы. Стандарты форматирования сообщений.	
	5. Организация работы команды в системе контроля версий.	
	<b>В том числе практических занятий и лабораторных работ</b>	12
	1. Практическое занятие «Разработка структуры проекта»	
	2. Практическое занятие «Разработка модульной структуры проекта (диаграммы модулей)»	
	3. Практическое занятие «Разработка перечня артефактов и протоколов проекта»	
	4. Практическое занятие «Настройка работы системы контроля версий (типов импортируемых файлов, путей, фильтров и др. параметров импорта в репозиторий)»	
	5. Практическое занятие «Разработка и интеграция модулей проекта (командная работа)»	
	6. Практическое занятие «Отладка отдельных модулей программного проекта»	
	7. Практическое занятие «Организация обработки исключений»	
<b>Тема 2.2.2 Инструментарий тестирования и анализа качества программных средств</b>	<b>Содержание</b>	20
	1. Отладка программных продуктов. Инструменты отладки. Отладочные классы.	
	2. Ручное и автоматизированное тестирование. Методы и средства организации тестирования.	
	3. Инструментарий анализа качества программных продуктов в среде разработки.	
	4. Обработка исключительных ситуаций. Методы и способы идентификации сбоев и ошибок.	
	5. Выявление ошибок системных компонентов.	
	<b>В том числе практических занятий и лабораторных работ</b>	12
	1. Практическое занятие «Применение отладочных классов в проекте»	
	2. Практическое занятие «Отладка проекта»	
	3. Практическое занятие «Инспекция кода модулей проекта»	
	4. Практическое занятие «Тестирование интерфейса пользователя средствами инструментальной среды разработки»	
	5. Практическое занятие «Разработка тестовых модулей проекта для тестирования отдельных модулей»	

	6. Практическое занятие «Выполнение функционального тестирования»	
	7. Практическое занятие «Тестирование интеграции»	
	8. Практическое занятие «Документирование результатов тестирования»	
<b>Итого</b>		64
<b>Промежуточная аттестация</b>	Зачет с оценкой	

### **3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

**3.1. Для реализации программы профессионального модуля должны быть предусмотрены следующие специальные помещения:**

**Лаборатория «Программного обеспечения и сопровождения компьютерных систем»:**

- Автоматизированные рабочие места на 12-15 обучающихся (процессор не ниже Core i3, оперативная память объемом не менее 4 Гб;) или аналоги;
- Автоматизированное рабочее место преподавателя (процессор не ниже Core i3, оперативная память объемом не менее 4 Гб;)или аналоги;
- Проектор и экран;
- Маркерная доска;
- Программное обеспечение общего и профессионального назначения

### **3.2. Информационное обеспечение реализации программы**

Для реализации программы библиотечный фонд образовательной организации имеет печатные и электронные образовательные и информационные ресурсы, для использования в образовательном процессе.

#### **3.2.1. Основные источники**

Практикум и индивидуальные задания по интегральному исчислению функции одной переменной (типовые расчеты) : учебное пособие для спо / В. А. Болотюк, Л. А. Болотюк, Ю. Г. Галич [и др.]. — Санкт-Петербург : Лань, 2022. — 336 с. — ISBN 978-5-507-44050-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/207524>

### **4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ МЕЖДИСЦИПЛИНАРНОГО КУРСА**



Для обучающегося инвалида или обучающегося с ОВЗ форма текущего контроля устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.). При необходимости обучающимся предоставляется дополнительное время для подготовки ответа.

Форма промежуточной аттестации устанавливается с учетом индивидуальных психофизических особенностей обучающегося инвалида или обучающегося с ОВЗ (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.). При необходимости осуществляется увеличение времени на подготовку к промежуточной аттестации, а также предоставляется дополнительное время для подготовки ответа в ходе промежуточной аттестации. Возможно установление индивидуальных графиков прохождения промежуточной аттестации.

Образовательное учреждение обеспечивает организацию и проведение текущего контроля и промежуточной аттестации по модулю.

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Критерии оценки	Методы оценки
<b>Раздел модуля 2 Средства разработки программного обеспечения</b>		
ПК 2.2 Выполнять интеграцию модулей в программное обеспечение	<p>Оценка «<b>отлично</b>» - в системе контроля версий выбрана верная версия проекта, проанализирована его архитектура, архитектура доработана для интеграции нового модуля; выбраны способы форматирования данных и организована их постобработка, транспортные протоколы и форматы сообщений обновлены (при необходимости); протестирована интеграция модулей проекта и выполнена отладка проекта с применением инструментальных средств среды; выполнена доработка модуля и дополнительная обработка исключительных ситуаций в том числе с созданием классов-исключений (при необходимости); определены качественные показатели полученного проекта; результат интеграции сохранен в системе контроля версий.</p> <p>Оценка «<b>хорошо</b>» - в системе контроля версий выбрана верная версия проекта, его архитектура доработана для интеграции нового модуля; выбраны способы форматирования данных и организована их постобработка, транспортные протоколы и форматы сообщений обновлены (при</p>	<p>Дифференцированный зачет: практическое задание по обеспечению интеграции заданного модуля в предложенный программный проект</p> <p>Защита отчетов по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>

	<p>необходимости); выполнена отладка проекта с применением инструментальных средств среды; выполнена доработка модуля и дополнительная обработка исключительных ситуаций (при необходимости); определены качественные показатели полученного проекта; результат интеграции сохранен в системе контроля версий.</p> <p>Оценка «<b>удовлетворительно</b>» - в системе контроля версий выбрана верная версия проекта, его архитектура доработана для интеграции нового модуля; выбраны способы форматирования данных и организована их постобработка, форматы сообщений обновлены (при необходимости); выполнена отладка проекта с применением инструментальных средств среды; выполнена доработка модуля (при необходимости); результат интеграции сохранен в системе контроля версий.</p>	
<p>ПК 2.3 Выполнять отладку программного модуля с использованием специализированных программных средств</p>	<p>Оценка «<b>отлично</b>» - в системе контроля версий выбрана верная версия проекта; протестирована интеграция модулей проекта и выполнена отладка проекта с применением инструментальных средств среды; проанализирована и сохранена отладочная информация; выполнена условная компиляция проекта в среде разработки; определены качественные показатели полученного проекта в полном объеме; результаты отладки сохранены в системе контроля версий.</p> <p>Оценка «<b>хорошо</b>» - в системе контроля версий выбрана верная версия проекта; протестирована интеграция модулей проекта и выполнена отладка проекта с применением инструментальных средств среды; выполнена условная компиляция проекта в среде разработки; определены качественные показатели полученного проекта в достаточном объеме; результаты отладки сохранены в системе контроля версий.</p> <p>Оценка «<b>удовлетворительно</b>» - в системе контроля версий выбрана верная версия проекта; выполнена отладка проекта с применением инструментальных средств среды; выполнена условная компиляция проекта в среде разработки; определены качественные показатели полученного проекта в достаточном объеме; результаты отладки сохранены в системе контроля версий.</p>	<p>Дифференцированный зачет: практическое задание по выполнению отладки программного модуля.</p> <p>Защита отчетов по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>

ПК 2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования	<p>Оценка <b>«отлично»</b> - продемонстрировано знание стандартов кодирования более чем одного языка программирования, выявлены все имеющиеся несоответствия стандартам в предложенном коде.</p> <p>Оценка <b>«хорошо»</b> - продемонстрировано знание стандартов кодирования более чем одного языка программирования, выявлены существенные имеющиеся несоответствия стандартам в предложенном коде.</p> <p>Оценка <b>«удовлетворительно»</b> - продемонстрировано знание стандартов кодирования языка программирования, выявлены некоторые несоответствия стандартам в предложенном коде.</p>	<p>Дифференцированный зачет: практическое задание по инспектированию программного кода</p> <p>Защита отчетов по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
--	---	--

Формы текущего контроля успеваемости:

**Опрос (О)** - это основной вид устной проверки, может использоваться как фронтальный (на вопросы преподавателя по сравнительно небольшому материалу краткие ответы (как правило, с места) дают многие обучающиеся), так и индивидуальный (проверка знаний отдельных обучающихся).

**Лабораторная работа (ЛР)** - письменная работа по теме. Состоит из нескольких задач различной степени сложности.

Критерии оценивания

*Оценки «отлично»* заслуживает студент, обнаруживший глубокое знание материала, умение свободно выполнять задания, понимающий взаимосвязь основных понятий работы;

*Оценки «хорошо»* заслуживает студент, обнаруживший полное знание материала; успешно выполняющий предусмотренные задания; и допустивший незначительные ошибки в коде;

*Оценки «удовлетворительно»* заслуживает студент, обнаруживший знания основного материала в объеме, необходимом для дальнейшего изучения междисциплинарного курса. Справляющийся с выполнением заданий; допустивший погрешности в написании программы, но обладающий необходимыми знаниями для их устранения под руководством преподавателя;

*Оценки «неудовлетворительно»* заслуживает студент, обнаруживший существенные пробелы в знании основного материала; не справляющийся с выполнением заданий, допустивший серьезные недостатки в написании кода, нуждающийся в повторении основных разделов курса под руководством преподавателя.

Формы текущего контроля

№	Название темы	Формы текущего контроля успеваемости, промежуточной аттестации
---	---------------	--

1	Тема 1. Современные технологии и инструменты интеграции.	О, ЛБ
2	Тема 2. Инструментарий тестирования и анализа качества программных средств	О, ЛБ

Примечание. Формы текущего контроля успеваемости: опрос (О), Лабораторная работа (ЛР).

Образовательные результаты (знания)	Показатели оценки результата
- основные принципы процесса разработки программного обеспечения;	Формулирование основных принципов процесса разработки программного обеспечения.
- методы отладочных классов;	Выбор необходимого метода отладки.
- основные протоколы доступа к данным;	Формулирование основных протоколов доступа.
- виды и варианты интеграционных решений.	Определение видов и вариантов интеграционных решений.
- использовать методы для получения кода с заданной функциональностью и степенью качества;	Выбор методов для получения кода с заданной функциональностью и степенью качества.
- определять источники и приемники данных;	Определение источников и приемников данных. Выбор способов форматирования данных и постобработка.
- организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов;	Интегрирование модуля в программное обеспечение. Тестирование интеграции модулей проекта и отладка проекта с применением инструментальных средств среды.
- выполнять отладку, используя методы и инструменты условной компиляции;	Отладка модуля с помощью методов и инструментов условной компиляции (классы debug и trace).
- разрабатывать тестовые пакеты и тестовые сценарии;	Разработка тестовых сценариев и тестовых пакетов в соответствии с этими сценариями в соответствии с минимальным размером тестового покрытия. Выполнение тестирования интеграции и ручное тестирование и с применением инструментальных средств, заполнены протоколы тестирования.
- оценивать размер минимального набора тестов;	Расчет размера набора и подбор тестов. Обоснование размера тестового покрытия.
- разрабатывать элементы программного модуля в соответствии с требованиями;	Разработка дополнительных элементов для имеющихся модулей. Обоснование необходимости использования элементов.
- выявлять ошибки в системных компонентах на основе	Выявление ошибок системных компонент (при наличии), заполнены протоколы тестирования.

<p>использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>пределять источники и приемники данных;</p> <p>организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнеспроцессов;</p> <p>выполнять отладку, используя методы и инструменты условной компиляции (классы <code>debug</code> и <code>trace</code>).</p> <p>разрабатывать тестовые пакеты и тестовые сценарии;</p> <p>оценивать размер минимального набора тестов;</p> <p>разрабатывать элементы программного модуля в соответствии с требованиями;</p> <p>- выявлять ошибки в системных компонентах на основе спецификаций.</p>	<ol style="list-style-type: none"> <li>1. Выполнить сценарий события, когда текст заголовка становится красным, когда пользователь наводит на него курсор и цвет возвращается, когда курсор отводится. Для этого нужно воспользоваться CSS и JavaScript. Преобразуйте страницу с заголовком «Добро пожаловать на нашу страницу!» - и текстом «Здесь много интересной информации. Здесь много интересной информации. Здесь много интересной информации. Здесь много интересной информации.». Где такой сценарий можно еще использовать на практике, при каких ситуациях?</li> <li>2. На Web-странице применить анимацию. Выполнить сценарий ситуации, когда текст «Текст, шагом марш!» должен перемещаться слева направо. Для этого нужно воспользоваться тэгом, ограничивающим текст, идентификатором <code>id</code>, CSS, функцией <code>moveTextO</code>, оператором <code>if</code>, атрибутом CSS <code>pixelLeft</code>, атрибутом <code>pixelLeft</code>, метода <code>setTimeout</code>, событием <code>onLoad</code>. Условие: чтобы запустить сценарий на выполнение, если текст находится менее чем в 500 пикселях от левой границы экрана. Каждый раз текст будет перемещаться вправо на два пикселя. Установить интервал до повторного запуска функции <code>moveText()</code>, равным 50 мс. Где такой сценарий можно еще использовать на практике, при каких ситуациях?</li> <li>3. Выполнить сценарий ситуации проверки, содержится ли на странице элемент <code>h1</code> - Первый заголовок? Можно воспользоваться страницей с одним элементом <code>h1</code>. Где такой сценарий можно еще использовать на практике, при каких ситуациях?</li> <li>4. Способность отыскивать новые тэги позволяет сделать активным любой, даже самый незначительный элемент Web-страницы. Выполнить сценарий ситуации, когда имя тэга выясняется с помощью <code>window.event.srcElement.tagName</code> и указывается в строке состояния. Объект <code>SrcElement</code> обращается к исходному элементу, то есть к элементу, генерируемому событием. Подобный элемент можно легко обнаружить. Где такой сценарий можно еще использовать на практике, при каких ситуациях?</li> <li>5. Выполнить сценарий ситуации, используя событие <code>onContextmenu</code>, когда пользователь щелкает по полю документа правой кнопкой мыши, чтобы открыть контекстное меню. Это событие должно позволить запустить сценарий до того, как меню возникнет на экране, или вовсе предотвратить появление контекстного меню. Последнее можно отменить, воспользовавшись свойством <code>event.returnValue</code> и указав значение <code>false</code>. Тем самым отменяется событие, которое должно произойти по умолчанию.</li> <li>6. Создать в Visual Studio DLL (динамическую библиотеку) логина и пароля для идентификации пользователя при загрузке приложения. Где такое задание может быть использовано на практике, при каких ситуациях?</li> <li>7. Составить тесты для программного продукта методом «черного ящика». Результаты оформить в таблице. Проанализировать полученный результат</li> </ol>
---	---

Вопросы к дифференцированному зачету:

1. Понятие репозитория проекта, структура проекта.
2. Виды, цели и уровни интеграции программных модулей.
3. Автоматизация бизнес-процессов.
4. Выбор источников и приемников данных, сопоставление объектов данных.
5. Транспортные протоколы.
6. Стандарты форматирования сообщений.
7. Организация работы команды в системе контроля версий.
  8. Отладка программных продуктов.
9. Инструменты отладки.
10. Отладочные классы.
11. Ручное и автоматизированное тестирование.
12. Методы и средства организации тестирования.
13. Инструментарии анализа качества программных продуктов в среде разработке.
14. Обработка исключительных ситуаций.
15. Методы и способы идентификации.
16. Выявление ошибок системных компонентов.