

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лиодинович
Должность: Ректор
Дата подписания: 03.06.2024 10:36:53
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Министерство науки и высшего образования РФ

**Федеральное государственное бюджетное образовательное учреждение
высшего образования**

«Дагестанский государственный технический университет»

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

дисциплина **МДК.02.01 «Технология разработки программного обеспечения»**

специальность **09.02.07 Информационные системы и программирование**

квалификация **программист**

основное общее образование
уровень образования, на базе которого осваивается ППССЗ

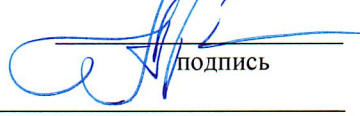
факультет **среднего профессионального образования,**


кафедра **ПОВТиАС**

форма обучения **очная**

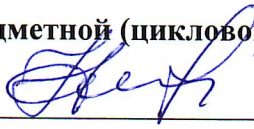
Программа составлена в соответствии с требованиями ФГОС СПО по специальности 09.02.07 Информационные системы и программирование с учетом рекомендаций и ОПОП СПО по специальности

Разработчик  Айгумов Т.Г., к.э.н., доцент
« 1 » _____ 2022 г.
подпись


Зав. кафедрой, за которой закреплена дисциплина  Айгумов Т.Г., к.э.н., доцент
« 1 » 11 2022 г.
подпись

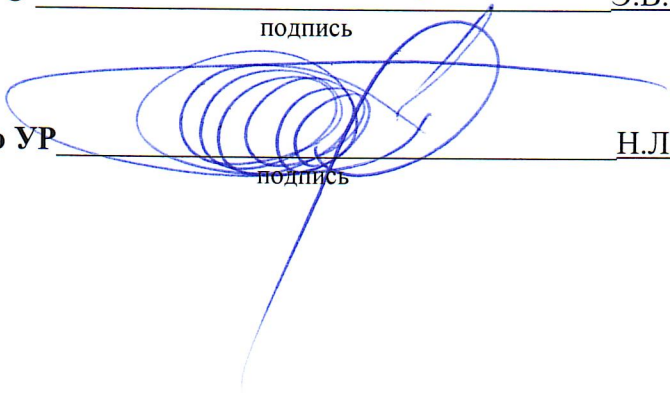
Зав. выпускающей кафедрой по данной специальности  Айгумов Т.Г., к.э.н., доцент
« 1 » 11 2022 г.
подпись

Программа одобрена на заседании предметной (цикловой) комиссии специальности 09.02.07 Информационные системы и программирование от « 30 » 11 2022 г., протокол № 3.

Председатель предметной (цикловой) комиссии  У.А. Мусаева, к.т.н., доцент
подпись
« 30 » 11 2022 г.

Декан факультета  М.М. Абдусаламова
подпись

Начальник УО  Э.В. Магомаева
подпись

Проректор по УР  Н.Л. Баламирзоев
подпись

СОДЕРЖАНИЕ

	стр.
1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	9
4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	12

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ МДК.02.01 «ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»

1.1. Место дисциплины в структуре основной профессиональной образовательной программы

Учебная дисциплина МДК.02.01 «Технология разработки программного обеспечения» является обязательной частью профессионального модуля ПМ. 02 «Осуществление интеграции программных модулей» профессионального цикла программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.07 «Информационные системы и программирование».

Рабочая программа дисциплины МДК.02.01 «Технология разработки и программного обеспечения» является частью основной профессиональной образовательной программы в соответствии с ФГОС по специальности 09.02.07 «Информационные системы и программирование» для очного обучения студентов, имеющих основное общее образование, по программе базовой подготовки.

Учебная дисциплина «Технология разработки программного обеспечения» обеспечивает формирование профессиональных компетенций по виду деятельности Осуществление интеграции программных модулей ФГОС специальности 09.02.07 «Информационные системы и программирование».

Особое значение дисциплина имеет при формировании и развитии профессиональных компетенций:

- 1) ПК 2.1 Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент;
- 2) ПК 2.2 Выполнять интеграцию модулей в программное обеспечение;
- 3) ПК 2.3 Выполнять отладку программного модуля с использованием специализированных программных средств.

1.2. Цель и планируемые результаты освоения дисциплины

В рамках программы учебной дисциплины обучающимися осваиваются умения, усваиваются знания и практический опыт.

Код и формулировка компетенции	Умения	Знания	Практический опыт
ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент	<ul style="list-style-type: none"> - анализировать проектную и техническую документацию; - использовать специализированные графические средства построения и анализа архитектуры программных продуктов; - организовывать заданную интеграцию модулей 	<ul style="list-style-type: none"> - модели процесса разработки программного обеспечения; - основные принципы процесса разработки программного обеспечения; - виды и варианты интеграционных решений; - основные протоколы доступа к данным; 	<ul style="list-style-type: none"> - разрабатывать и оформлять требования к программным модулям по предложенной документации; - разрабатывать тестовые наборы (пакеты) для программного модуля; - разрабатывать тестовые сценарии программного

	<p>в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов;</p> <ul style="list-style-type: none"> - определять источники и приемники данных; - проводить сравнительный анализ; - выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace); - оценивать размер минимального набора тестов; - разрабатывать тестовые пакеты и тестовые сценарии 	<ul style="list-style-type: none"> - методы отладочных классов; - стандарты качества программной документации; - графические средства проектирования архитектуры программных продуктов 	<p>средства</p>
<p>ПК 2.2. Выполнять интеграцию модулей в программное обеспечение</p>	<ul style="list-style-type: none"> - использовать методы для получения кода с заданной функциональностью и степенью качества; - использовать различные транспортные протоколы и стандарты форматирования сообщений; - создавать классы-исключения на основе базовых классов; - выполнять ручное и автоматизированное тестирование программного модуля 	<ul style="list-style-type: none"> - основные подходы к интегрированию программных модулей; - основы верификации программного обеспечения; - современные технологии и инструменты интеграции; - методы и способы идентификации сбоев и ошибок при интеграции приложений 	<ul style="list-style-type: none"> - интегрировать модули в программное обеспечение; - инспектировать разработанные программные модули на предмет соответствия стандартам кодирования
<p>ПК 2.3. Выполнять отладку программного модуля с</p>	<ul style="list-style-type: none"> - использовать выбранную систему контроля версий; 	<ul style="list-style-type: none"> - основы верификации и аттестации программного 	<ul style="list-style-type: none"> - отлаживать программные модули

использованием специализированных программных средств	<ul style="list-style-type: none"> - использовать инструментальные средства отладки программных продуктов; - выполнять тестирование интеграции; - организовывать постобработку данных; - использовать приемы работы в системах контроля версии; - выполнять отладку, используя методы и инструменты условной компиляции; - выявлять ошибки в системных компонентах на основе спецификаций. 	<ul style="list-style-type: none"> обеспечения; - основные методы отладки; - методы и схемы обработки исключительных ситуаций; - приемы работы с инструментальными средствами тестирования и отладки; - основы организации инспектирования и верификации; - встроенные и основные специализированные инструменты анализа качества программных продуктов; - методы организации работы в команде разработчиков. 	
---	--	--	--

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем в часах
Объем образовательной программы учебной дисциплины	103
Объем работы обучающихся во взаимодействии с преподавателем:	72
в том числе:	
лекции	17
практические занятия	17
лабораторные занятия	34
консультация	4
Самостоятельная работа	13
Промежуточная аттестация в форме экзамена	6 семестр (18 часов)

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
1	2	3	4
Тема 1. Основы разработки ПО.	Содержание учебного материала		ПК 2.1
	Основные понятия и определения. Типы ПО. Жизненный цикл и этапы разработки ПО. Модели ЖЦ ПО. Качество ПО.	2	
	в том числе практических и лабораторных занятий	6	
	Практическая работа №1. Жизненный цикл и этапы разработки ПО.	2	
	Лабораторная работа №1. Анализ предметной области ПО.	4	
	Самостоятельная работа	2	
	Подготовка к практической работе	1	
	Подготовка к лабораторной работе	1	
Тема 2. Разработка и анализ требований к ПО.	Содержание учебного материала		ПК 2.1
	Введение в анализ требований. Описание требований заказчика. Шаги разработки пользовательского интерфейса. Методологии и инструментальные средства для требований заказчика. Детальные требования и их типы. Методы проектирования ПО.	4	
	в том числе практических и лабораторных занятий	8	
	Практическая работа №2. Шаги разработки пользовательского интерфейса.	2	
	Практическая работа №3. Детальные требования и их типы.	2	
	Лабораторная работа №2. Оформление спецификации требований к ПО.	4	
	Самостоятельная работа	3	
	Подготовка к практической работе	1	
Подготовка к лабораторной работе	2		
Тема 3.	Содержание учебного материала		ПК 2.2, ПК 2.3

Проектирование и архитектура ПО.	Основы проектирования программных систем. Модульное проектирование. Структурное проектирование. Основы ООП программных систем. Архитектура ПО. Диаграмма UML. История развития технологии программирования и построения архитектуры ПО.	4	
	в том числе практических и лабораторных занятий	12	
	Практическая работа №4 Основы проектирования программных систем.	2	
	Практическая работа №5. Архитектура ПО.	2	
	Лабораторная работа №3. Проектирование модулей ПО.	4	
	Лабораторная работа №4. Разработка модулей ПО.	4	
	Самостоятельная работа	2	
	Подготовка к практической работе	1	
	Подготовка к лабораторной работе	1	
Тема 4. Отладка и тестирование ПО.	Содержание учебного материала		ПК 2.2, ПК 2.3
	Оценка стоимости и причин ошибок в ПО. Основные понятия отладки и тестирования ПО. Тестирование интеграции ПО. Тестированию по методу «белого ящика». Методика тестирования программных систем. Функциональное тестирование ПО.	2	
	в том числе практических и лабораторных занятий	6	
	Практическая работа №6. Методика тестирования программных систем.	2	
	Лабораторная работа №5. Отладка и тестирование программы.	4	
	Самостоятельная работа	2	
	Подготовка к практической работе	1	
	Подготовка к лабораторной работе	1	
Тема 5. Управление программными проектами.	Содержание учебного материала		ПК 2.2, ПК 2.3
	Особенности процесса разработки ПО. Управление приоритетами проектов. Планирование проекта. Меры, метрики и модели. Управление персоналом проекта. Реализация проекта. Управление рисками проекта.	2	
	в том числе практических и лабораторных занятий	10	
	Практическая работа №7. Планирование проекта. Меры, метрики и	2	

	модели.		
	Лабораторная работа №6. Управление персоналом проекта. Типы совместной деятельности	4	
	Лабораторная работа №7. Оценка трудоемкости разработки ПО.	4	
	Самостоятельная работа	2	
	Подготовка к практической работе	1	
	Подготовка к лабораторной работе	1	
Тема 6. Документирование программных средств.	Содержание учебного материала		ПК 2.3
	Документация, создаваемая и используемая в процессе разработки ПО. Стандартизация программной документации. Государственные стандарты РФ (ГОСТЫ Р).	3	
	в том числе практических и лабораторных занятий	9	
	Практическая работа №8. Документация, создаваемая и используемая в процессе разработки ПО.	2	
	Практическая работа №9. Стандартизация программной документации.	1	
	Лабораторная работа №8. Принципы разработки руководства программиста.	2	
	Лабораторная работа №9. Разработка руководства пользователя.	4	
	Самостоятельная работа	2	
	Подготовка к практической работе	1	
	Подготовка к лабораторной работе	1	
Итого:	Лекций	17	
	Практических занятий	17	
	Лабораторных занятий	34	
	Самостоятельная работа	13	
	Консультация	4	
	Промежуточная аттестация в форме экзамена	18	

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Материально-техническое обеспечение

Для реализации программы учебной дисциплины предусмотрены следующие специальные помещения (в соответствии с ФГОС и ОПОП): лекционный кабинет, оснащенный оборудованием: мультимедиа проектор, компьютер с лицензионным программным обеспечением; компьютерные классы с компьютерами по количеству обучающихся и рабочее место преподавателя, техническими средствами обучения: учебные наглядные пособия (таблицы, плакаты), принтер, комплект учебников (учебных пособий) по количеству обучающихся, комплект учебно-методической документации для лабораторных занятий, кабинет для практических занятий.

3.2. Информационное обеспечение реализации программы

Для реализации программы библиотечный фонд образовательной организации имеет печатные и/или электронные образовательные и информационные ресурсы, для использования в образовательном процессе. При формировании библиотечного фонда образовательной организации выбирается не менее одного издания из перечисленных ниже печатных изданий и (или) электронных изданий в качестве основного, при этом список, может быть дополнен новыми изданиями.

3.2.1. Печатные издания

Нормативно - правовые документы:

1. Федеральный закон от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации»

Основная литература:

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, 2023. — 400 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0812-9. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1895679>;

2. Современные технологии разработки программного обеспечения: учебно-методическое пособие / составитель Н. А. Федькова. — Брянск: Брянский ГАУ, 2022. — 58 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/305087>;

3. Китайцева, Е. Х. Алгоритмизация. Технология разработки программного обеспечения: учебно-методическое пособие / Е. Х. Китайцева. — Москва: МИСИ – МГСУ, 2021. — 51 с. — ISBN 978-5-7264-2905-2. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/249011>;

4. Кривоносова, Н. В. Технология WPF. Разработка модулей программного обеспечения: практикум: учебное пособие / Н. В. Кривоносова. — Санкт-Петербург: СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 132 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/279719>;

5. Гусев, К. В. Технология разработки программных приложений: учебное пособие / К. В. Гусев, М. Б. Туманова, Е. А. Чернов. — Москва: РТУ МИРЭА, 2023. — 146 с. — ISBN 978-5-7339-1938-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/382706> (дата обращения: 01.01.2024). — Режим доступа: для авториз. пользователей.

6. Зубкова, Т. М. Технология разработки программного обеспечения / Т. М. Зубкова. — 3-е изд., стер. — Санкт-Петербург: Лань, 2023. — 252 с. — ISBN 978-5-507-45571-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/276419>.

3.2.3. Дополнительные источники:

1. Гагарина, Л. Г. Технология разработки программного обеспечения: учеб. пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул; под ред. Л.Г. Гагариной. — Москва: ИД «ФОРУМ»: ИНФРА-М, 2019. — 400 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-8199-0707-8. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1011120>;

2. Конструирование программного обеспечения: учебное пособие / под ред. Л.Г. Гагариной. — Москва: ИНФРА-М, 2024. — 319 с. — (Высшее образование). — DOI 10.12737/1893880. - ISBN 978-5-16-017861-5. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1893880>;

3. Машкин, А. В. Технология разработки программного обеспечения: учебное пособие / А. В. Машкин. — Вологда: ВоГУ, 2014. — 75 с. — ISBN 978-5-87851-526-9. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/93087>;

4. Андреев, А. Е. Адаптивные технологии разработки программного обеспечения: учебное пособие / А. Е. Андреев, С. И. Кириносенко. — Волгоград: ВолгГТУ, 2015. — 96 с. — ISBN 978-5-9948-1979-1. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/157223>;

5. Зубкова, Т. М. Технология разработки программного обеспечения: учебное пособие / Т. М. Зубкова. — Оренбург: ОГУ, 2017. — 468 с. — ISBN 978-5-7410-1785-2. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/110632>.

3.2.2. Интернет-ресурсы:

1. www.informika.ru/text/index.htm - Информика - государственный научно-исследовательский институт информационных технологий и телекоммуникаций;

2. www.infojournal.ru – научно-методический журнал «ИНФОРМАТИКА И ОБРАЗОВАНИЕ»;

3. <http://school-db.informika.ru> - единая коллекция цифровых образовательных ресурсов;

4. <http://www.osp.ru/pcworld> – журнал «Мир ПК». Компьютерная пресса.

5. www.edu.ru – федеральный портал российского образования;

6. www.elibrary.ru – научная электронная библиотека;

7. www.nehudlit.ru - электронная библиотека учебных материалов.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, тестирования, а также выполнения обучающимися индивидуальных заданий.

Результаты обучения	Критерии оценки	Методы оценки
<p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> - модели процесса разработки программного обеспечения; - основные принципы процесса разработки программного обеспечения; - виды и варианты интеграционных решений; - основные протоколы доступа к данным; - методы отладочных классов; - стандарты качества программной документации; - графические средства проектирования архитектуры программных продуктов; - основные подходы к интегрированию программных модулей; - основы верификации программного обеспечения; - современные технологии и инструменты интеграции; - методы и способы идентификации сбоя и ошибок при интеграции приложений; - основы верификации и аттестации программного обеспечения; - основные методы отладки; - методы и схемы обработки исключительных ситуаций; - приемы работы с инструментальными средствами тестирования и отладки; - основы организации инспектирования и 	<p><i>Шкала оценивания для экзамена</i></p> <p><i>«Отлично»</i> Показывает высокий уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> - демонстрирует высокое и прочное освоение материала; - исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; - правильно формирует определения; - демонстрирует умения самостоятельной работы с нормативно-правовой литературой; - умеет делать выводы по излагаемому материалу. <p><i>«Хорошо»</i> Показывает достаточный уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> - демонстрирует достаточно полное знание материала, основных теоретических положений; - достаточно последовательно, грамотно и логически стройно излагает теоретический 	<p>Текущий контроль при проведении:</p> <ul style="list-style-type: none"> - тестирования; - письменного/устного опроса; - защита отчетов по практическим и лабораторным занятиям; - оценка результатов самостоятельной работы (рефератов, докладов, теоретической части проектов, учебных исследований и т.д.): - экспертная оценка демонстрируемых умений, выполняемых действий в процессе практических и лабораторных занятий; <p>Промежуточная аттестация в форме:</p> <ul style="list-style-type: none"> - экзамена.

<p>верификации;</p> <ul style="list-style-type: none"> - встроенные и основные специализированные инструменты анализа качества программных продуктов; - методы организации работы в команде разработчиков. 	<p>материал;</p> <ul style="list-style-type: none"> - демонстрирует умения ориентироваться в нормативно-правовой литературе; - умеет делать достаточно обоснованные выводы по излагаемому материалу. 	
<p><i>Перечень умений, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> - анализировать проектную и техническую документацию; использовать специализированные графические средства построения и анализа архитектуры программных продуктов; - организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов; определять источники и приемники данных; - проводить сравнительный анализ; - выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace); - оценивать размер минимального набора тестов; - разрабатывать тестовые пакеты и тестовые сценарии; - использовать методы для получения кода с заданной функциональностью и степенью качества; - использовать различные транспортные протоколы и стандарты форматирования сообщений; - создавать классы-исключения на основе базовых классов; - выполнять ручное и автоматизированное 	<p>«Удовлетворительно» Показывает пороговый уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> - демонстрирует общее знание изучаемого материала; - испытывает затруднения при ответах на дополнительные вопросы; - знает основную рекомендуемую литературу; - умеет строить ответ в соответствии со структурой излагаемого материала. <p>«Неудовлетворительно» Ставится в случае:</p> <ul style="list-style-type: none"> - незнания значительной части программного материала; - невладения понятийным аппаратом дисциплины; - допущения существенных ошибок при изложении учебного материала; - неумения строить ответ в соответствии со структурой излагаемого вопроса; - неумения делать выводы по излагаемому материалу. 	

<p>тестирование программного модуля;</p> <ul style="list-style-type: none"> - использовать выбранную систему контроля версий; - использовать инструментальные средства отладки программных продуктов; - выполнять тестирование интеграции; - организовывать постобработку данных; - использовать приемы работы в системах контроля версии; - выполнять отладку, используя методы и инструменты условной компиляции; - выявлять ошибки в системных компонентах на основе спецификаций. 		
<p><i>Практический опыт:</i></p> <ul style="list-style-type: none"> - разрабатывать и оформлять требования к программным модулям по предложенной документации; - разрабатывать тестовые наборы (пакеты) для программного модуля; - разрабатывать тестовые сценарии программного средства; - интегрировать модули в программное обеспечение; - инспектировать разработанные программные модули на предмет соответствия стандартам кодирования. - отлаживать программные модули 		