

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ
Проректор по учебной работе,
качеству образования – первый
проректор
Хагуров Т.А.
подпись



«30» мая 2025

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.О.05«Технологии автоматизации программирования»

Направление подготовки 02.04.02 Фундаментальная информатика и
информационные технологии

Направленность (профиль) Интеллектуальные системы и технологии

Форма обучения очная

Квалификация магистр

Краснодар 2025

Рабочая программа дисциплины «Технологии автоматизации программирования» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.04.02 Фундаментальная информатика и информационные технологии.

Программу составил: А.Н. Полетайкин, доц. каф. ИТ, к.т.н., доц.


_____ подпись

Рабочая программа дисциплины «Технологии автоматизации программирования» утверждена на заседании кафедры информационных технологий протокол №15 от «14» мая 2025г.

Заведующий кафедрой (разработчика)

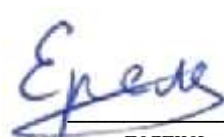
В. В. Подколзин


_____ подпись

Рабочая программа обсуждена на заседании кафедры вычислительных технологий протокол №7 от «07» мая 2025 г.

И.о. заведующего кафедрой (выпускающей)

А.А. Еремин


_____ подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол №4 от «23» мая 2025 г.

Председатель УМК факультета

А. В. Коваленко


_____ подпись

Рецензенты:

Бегларян М. Е., Проректор по учебной работе, Краснодарский кооперативный институт (филиал) АНО ВО Центросоюза РФ «Российский университет кооперации»

Рубцов Сергей Евгеньевич, кандидат физико-математических наук, доцент кафедры математического моделирования ФГБОУ ВО «КубГУ»

1 Цели и задачи изучения дисциплины (модуля)

1.1 Цель освоения дисциплины

Целью курса является формирование у студентов знаний, умений и практических навыков в области создания программных систем разного назначения автоматизированным способом с учетом задач будущей профессиональной деятельности.

Дисциплина рассматривает применение методов, подходов и инструментальных средств автоматизированного программирования и разработки программных систем (ПС).

1.2 Задачи дисциплины

- изучение методологии автоматизированной разработки ПО;
- освоение способов организации автоматизированной разработки ПО;
- углубление знаний по программированию и коллективной разработке приложений с применением современных технологий разработки ПО;
- выработка умений и навыков в области формирования и использования среды автоматизированной разработки ПО;
- освоение современных технологий автоматизированной сборки, упаковки и тестирования приложения, автоматизации развертывания в разных окружениях.

Предметом учебной дисциплины являются методы, подходы и алгоритмы автоматизированной разработки ПО.

1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Технологии автоматизации программирования» относится к «Обязательная часть» Блока 1 «Дисциплины (модули)» учебного плана.

1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

ОПК-1. Способен находить, формулировать и решать актуальные проблемы прикладной математики, фундаментальной информатики и информационных технологий;

ОПК-3. Способен проводить анализ математических моделей, создавать инновационные методы решения прикладных задач профессиональной деятельности в области информатики и математического моделирования,

и выражается следующими индикаторами их достижения:

ОПК-1.1. Обладает фундаментальными знаниями в области математических и естественных наук, теории коммуникаций, а именно знает:

- *современные архитектуры и технологии разработки и отладки ПО;*
- *современные подходы, математические методы, математические модели, алгоритмы, программы, технические и инструментальные средства разработки ПО с современной архитектурой.*

ОПК-1.3. Имеет практический опыт работы с решением математических задач и применяет его в профессиональной деятельности, а именно владеет навыками:

- *организации программного процесса автоматизированной разработки ПО;*
- *выбора современных технологий для автоматизированной разработки ПО;*
- *постановки задачи на создание ПО;*
- *развертывание рабочей среды для автоматизированной разработки ПО;*
- *разработки и тестирования ПО с использованием технологий CI/CD;*
- *сопровождения ПО в условиях реализации версионного контроля;*
- *оформления технической документации к ПО и программному приложению.*

ОПК-3.1. Знает методы теории алгоритмов, методы системного и прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей.

ОПК-3.2. Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем.

ОПК-3.3. Имеет практический опыт применения разработки программного обеспечения и тестирования программных продуктов, а именно владеет навыками:

- разработки программного приложения на одном или нескольких языках программирования при помощи современных средств разработки и отладки ПО;
- осуществления версионного контроля программного приложения при помощи современных распределенных средств управления версиями;
- осуществления тестирования программного приложения с применением специальных методов и современных средств тестирования и технологий CI/CD.

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 5 зач. ед. (180 часов), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)					
		3					
Контактная работа, в том числе:	54,3	54,3					
Аудиторные занятия (всего):	54	54					
Занятия лекционного типа	18	18					
Лабораторные занятия	36	36					
Занятия семинарского типа (семинары, практические занятия)	–	–					
Иная контактная работа:	0,3	0,3					
Контроль самостоятельной работы (КСР)							
Промежуточная аттестация (ИКР)	0,3	0,3					
Самостоятельная работа, в том числе:	99	99					
Курсовая работа	–	–					
Проработка учебного (теоретического) материала	36	36					
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)	36	36					
Реферат	–	–					
Подготовка к текущему контролю	25	25					
Контроль:	26,7	26,7					
Подготовка к экзамену	26,7	26,7					
Общая трудоемкость	час.	180	180				
	в том числе контактная работа	54,3	54,3				
	зач. ед	5	5				

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины. Разделы (темы) дисциплины, изучаемые в 3 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1.	Методология DevOps	17	2	—	4	11
2.	Жизненный цикл ПО по DevOps	17	2	—	4	11
3.	Методология Scrum по DevOps	17	2	—	4	11
4.	Современные архитектуры ПО	17	2	—	4	11
5.	Особенности микросервисной архитектуры ПО	17	2	—	4	11
6.	Коллективная web-разработка приложений с микросервисной архитектурой	17	2	—	4	11
7.	Использование фреймворка Django для разработки web-приложений	17	2	—	4	11
8.	Введение в Docker. Контейнеризация приложений	17	2	—	4	11
9.	Управление проектными рисками	17	2		4	11
ИТОГО по разделам дисциплины		153	18	—	36	99
Контроль самостоятельной работы (КСР)		—				
Промежуточная аттестация (ИКР)		0,3				
Подготовка к текущему контролю		26,7				
Общая трудоемкость по дисциплине		180				

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов (тем) дисциплины

2.3.1 Занятия лекционного типа

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
1.	Методология DevOps	Ключевые ценности DevOps, архитектура, инструментальная цепочка. История возникновения DevOps. CI/CD: логическая и временная структура, потенциальные проблемы. CI/CD: автоматизация производства ПО, проверка, сборка, мониторинг	ЛР
2.	Жизненный цикл ПО по DevOps	Рабочий процесс DevOps, его преимущества перед обычным рабочим процессом. Процесс документирования ПО в методологии DevOps. Процесс эксплуатации ПО в методологии DevOps. Технологии и инструменты создания ПО в методологии DevOps.	ЛР
3.	Методология Scrum по DevOps	Методология Scrum в контексте воплощения DevOps. Логическая и временная структура, потенциальные трудности. Основные проблемы итеративной разработки ПО	ЛР
4.	Современные архитектуры ПО	Распространенные варианты архитектур ПО. Многослойная архитектура ПО: структура, преимущества, недостатки. Сервис-ориентированная архитектура ПО: типы сервисов, структура, преимущества, недостатки. Микросервисная архитектура ПО: структура, отличия от монолита, преимущества, недостатки	ЛР
5.	Особенности микросервисной архитектуры ПО	Микросервисы: преимущества и недостатки их использования при создании ПО. Основные преимущества микросервисной архитектуры ПО. Основные недостатки микросервисной архитектуры	ЛР

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
		ПО. Архитектурный паттерн «Микросервисы поверх SOA». Масштабирование ПО с микросервисной архитектурой	
6.	Коллективная web-разработка приложений с микросервисной архитектурой	Современные фреймворки коллективной разработки ПО. Система контроля версий Git: основные понятия и архитектура. Построение работы с репозиториями в Git. Состояние файлов с точки зрения Git. Ветвление и слияние в Git. Работа с удаленными репозиториями. Совместная работа в Git в условиях коллективной разработки ПО. Решение конфликтов в Git в условиях коллективной разработки ПО	ЛР
7.	Использование фреймворка Django для разработки web-приложений	Использование Django для разработки web-приложений на ЯП Python: преимущества и недостатки. Установка Django. Создание и конфигурирование проекта Django. Шаблоны и модели проекта Django	ЛР
8.	Введение в Docker. Контейнеризация приложений	Использование Docker для создания приложений с микросервисной архитектурой. Использование Docker: преимущества, недостатки, отличия от использования виртуальных машин. Установка Docker. Основные команды Docker. Dockerfile. Установка docker-compose и подготовка образа приложения	ЛР
9	Управление проектными рисками	Источники проблем при разработке ПО. Идентификация и оценивание рисков на этапах ЖЦ ПО. Решения по минимизации рисков разработки ПО	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП – выполнение курсового проекта, КР – курсовой работы, РГЗ – расчетно-графического задания, Р – написание реферата, Э – эссе, К – коллоквиум, Т – тестирование, РЗ – решение задач.

2.3.2 Лабораторные занятия

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1.	Методология DevOps	Формирование среды разработки программного приложения	ЛР
2.	Жизненный цикл ПО по DevOps		ЛР
3.	Методология Scrum по DevOps		ЛР
4.	Современные архитектуры ПО		ЛР
5.	Особенности микросервисной архитектуры ПО	Работа с Docker Debian	ЛР
6.	Коллективная web-разработка приложений с микросервисной архитектурой	Работа с Dockerfile и docker-compose	ЛР
7.	Использование фреймворка Django для разработки web-приложений	Тема, сети и переменные окружения Docker	ЛР
8.	Введение в Docker. Контейнеризация приложений	Применение методологии CI/CD	ЛР
9	Управление проектными рисками	Риск-устойчивое автоматизированное программирование	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП – выполнение курсового проекта, КР – курсовой работы, РГЗ – расчетно-графического задания, Р – написание реферата, Э – эссе, К – коллоквиум, Т – тестирование, РЗ – решение задач.

График выполнения лабораторных работ:

№ л.р.	Тема	Объем, часов	Неделя выполнения
1	2	3	4
1	Установка и настройка Linux (Debian 12). Работа с файлами и каталогами. Права доступа в Linux. Работа с текстовыми файлами. Процессы. Bash	4	1, 2
2	Настройка сети в Linux. Работа в терминале через SSH. Установка файлового менеджера MC.	4	3, 4

№ л.р.	Тема	Объем, часов	Неделя выполнения
1	2	3	4
3	Настройка среды разработки программного приложения: Python+Django+PostgreSQL	4	5, 6
4	Работа с Docker Debian. Установка чистой ВМ Debian 12. Настройка сети. Установка Докер и docker-compose	6	7–9
5	Работа с Dockerfile и docker-compose. Создание образа и на его основе контейнера, с использованием простого стартового образа Python: добавление файлов приложения Django в создаваемый образ.	6	10–12
6	Тема, сети и переменные окружения Docker: Использование volumes и bind. Получение данных темплейтов Django в докер-контейнере из каталога на хост-машине	6	12–14
7	Применение методологии CI/CD: автоматизированная сборка, упаковка и тестирование приложения, автоматизация развертывания в разных окружениях. Управление рисками проекта	6	14–16
	Итого:	36	

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

– Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

– Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

– Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

– Информационно-коммуникационные технологии (ИКТ) – расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

– Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) – совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология – индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций – анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
	Л, ЛР	Практические занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	4
Итого			4

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные и методические материалы

4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «название дисциплины».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме заданий к выполнению лабораторных работ и **промежуточной аттестации** в форме вопросов и заданий к экзамену.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

- при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;
- при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;
- при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	Методология DevOps	ОПК-1.1 ОПК-1.3	Лабораторная работа 1, 2	Вопрос на экзамене 1-6
2	Жизненный цикл ПО по DevOps	ОПК-3	Лабораторная работа 3	Вопрос на экзамене 3-10
3	Методология Scrum по DevOps	ОПК-3.2 ОПК-3.3	Лабораторная работа 3	
4	Современные архитектуры ПО	ОПК-3.1	Лабораторная работа 4	Вопрос на экзамене 11-15
5	Особенности микросервисной архитектуры ПО	ОПК-1.1 ОПК-3.2	Лабораторная работа 4	Вопрос на экзамене 16-19
6	Коллективная web-разработка приложений с микросервисной архитектурой	ОПК-3	Лабораторная работа 5	Вопрос на экзамене 20-25
7	Использование фреймворка Django для разработки web-приложений	ОПК-1.1 ОПК-1.3	Лабораторная работа 5, 6	Вопрос на экзамене 26-28
8	Введение в Docker. Контейнеризация приложений	ОПК-1.1 ОПК-1.3 ОПК-3.3	Лабораторная работа 6, 7	Вопрос на экзамене 29-31
9	Управление проектными рисками	ОПК-1	Лабораторная работа 7	Вопрос на экзамене 32-40

Показатели, критерии и шкала оценки сформированных компетенций

Соответствие **пороговому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **удовлетворительно**):

Знания общие, слабо конкретизированы и фрагментарны. Студен в общем раскрывает суть вопроса и/или допускает несколько концептуальных ошибок.

Умения соотносить знания в области программирования с реальными задачами разработки ПО демонстрируются слабо и не позволяют успешно реализовать программный проект.

Умения определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем не выявлены.

Навыки сформированы слабо, что не позволяет студенту успешно осуществлять разработку и тестирование программного обеспечения.

Соответствие **базовому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **хорошо**):

Знания систематизированы и демонстрируются на уровне, достаточном для успешной работы в коллективе разработчиков ПО. Студен в конкретных понятиях раскрывает суть вопроса и допускает несколько мелких ошибок.

Умения соотносить знания в области программирования с реальными задачами разработки ПО демонстрируются уверенно и позволяют успешно реализовать программный проект.

Умения определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем имеют место.

Навыки сформированы в достаточной степени, что позволяет студенту успешно осуществлять разработку и тестирование программного обеспечения.

Соответствие **продвинутому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **отлично**):

Знания систематизированы и достаточно глубокие. Их уровень достаточен для успешной работы в коллективе разработчиков ПО и руководства программным процессом в соответствии с методологией DevOps.

Умения соотносить знания в области программирования с реальными задачами разработки ПО демонстрируются уверенно и позволяют успешно реализовать программный проект в любых ролях гибкой команды в соответствии с методологией DevOps.

Умения определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем имеют место и позволяют гибко планировать и применять информационные и интеллектуальные ресурсы программного процесса в соответствии с методологией DevOps.

Навыки сформированы в достаточной степени, что позволяет студенту успешно осуществлять разработку и тестирование программного обеспечения, а также руководить программным процессом в соответствии с методологией DevOps.

На всех уровнях освоения компетенций проявляются признаки исследовательской деятельности. При управлении программным процессом на всех этапах жизненного цикла ПО студент применяет научный подход.

Шкала оценивания экзамена

5-балльная шкала	Критерии оценки
Отлично «5»	Все индикаторы достижения компетенций на высоком уровне
Хорошо «4»	Все индикаторы ОПК-1 – высокий уровень, остальные индикаторы не ниже среднего
Удовлетворительно «3»	Все индикаторы ОПК-1 – не ниже среднего уровня, остальные индикаторы – не ниже порогового
Неудовлетворительно «2»	Хотя бы 1 индикатор достижения компетенции не выражен

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Материалы для промежуточной аттестации (экзамен):

1. Список вопросов к экзамену

1. Методология DevOps: ключевые ценности, архитектура, инструментальная цепочка.
2. Основные проблемы итеративной разработки ПО. История возникновения DevOps.
3. Методология Scrum: логическая и временная структура, потенциальные проблемы.
4. CI/CD: логическая и временная структура, потенциальные проблемы.
5. CI/CD: автоматизация производства ПО, проверка, сборка, мониторинг.
6. Рабочий процесс DevOps, его преимущества перед обычным рабочим процессом.
7. Процесс документирования ПО в методологии DevOps.
8. Процесс эксплуатации ПО в методологии DevOps.
9. Технологии и инструменты создания ПО в методологии DevOps.
10. Облачная технология IaC. Виртуализация и контейнеризация в IaC.
11. Распространенные варианты архитектур ИС.
12. Многослойная архитектура ИС: структура, преимущества, недостатки.
13. Сервис-ориентированная архитектура ИС: типы сервисов, структура, преимущества, недостатки.
14. Микросервисная архитектура ИС: структура, отличия от монолита, преимущества, недостатки.
15. Микросервисы: преимущества и недостатки их использования при создании ИС.
16. Основные преимущества микросервисной архитектуры ИС.
17. Основные недостатки микросервисной архитектуры ИС.
18. Архитектурный паттерн «Микросервисы поверх SOA».
19. Масштабирование ИС с микросервисной архитектурой.
20. Современные фреймворки коллективной разработки ПО.
21. Система контроля версий Git: основные понятия и архитектура. Построение работы с репозиториями в Git. Состояние файлов с точки зрения Git.
22. Ветвление и слияние в Git. Работа с удаленными репозиториями.
23. Совместная работа в Git в условиях коллективной разработки ПО. Решение конфликтов в Git в условиях коллективной разработки ПО.

24. Использование Django для разработки web-приложений на ЯП Python: преимущества и недостатки.
25. Установка Django. Создание и конфигурирование проекта Django.
26. Шаблоны и модели проекта Django.
27. Использование Docker для создания приложений с микросервисной архитектурой.
28. Использование Docker: преимущества, недостатки, отличия от использования виртуальных машин.
29. Установка Docker. Основные команды Docker
30. Dockerfile. Установка docker-compose и подготовка образа приложения.
31. Неопределенность проекта ИС. Факторы неопределенности и её учет посредством управления рисками.
32. Понятие и характеристики риска. Примеры рисков в проектах ИС. Понятие проектного риска.
33. Технология управления проектными рисками. Процесс управления проектными рисками и его планирование.
34. Реализация этапа идентификации проектных рисков: источники данных, процедуры, результаты.
35. Реализация этапа анализа проектных рисков: цель и задачи, процедуры. Ведение реестра рисков. Результаты анализа рисков.
36. Реализация этапа анализа проектных рисков: риск-матрицы: разновидности и особенности построения, анализ риск-матриц и его влияние на риски.
37. Реализация этапа обработки рисков: основные методы и мероприятия, стратегии и методы. Эффективность методов обработки рисков. Результаты обработки рисков.
38. Стратегии и методы обработки рисков. Методика принятия решений по результатам анализа проектных рисков.
39. Особенности планирования рисков проектной деятельности. Модель динамики вероятности риска и величины потерь.
40. Ролевая модель риск-менеджмента и её связь с ролевой моделью командной разработки ИС.

2. Практическое задание на экзамен

1. Составить докерфайл и докер-композицию для развертывания web-сервера Apache. Директорию с html-страницей примонтировать с хоста. Директорию на хосте, монтируемую внутрь контейнера, указать как /settings[двузначный номер билета].
2. Выполнить схематичное описание процессов CI/CD для задачи внесения изменений на web-страницу.

Примеры типовых заданных изменений:

- добавление/удаление визуальных компонентов;
- изменение свойств визуальных компонентов;
- реализация атомарной функциональности.

Перечень компетенций (части компетенции), проверяемых оценочным средством

ОПК-1. Способен находить, формулировать и решать актуальные проблемы прикладной математики, фундаментальной информатики и информационных технологий;

ОПК-3. Способен проводить анализ математических моделей, создавать инновационные методы решения прикладных задач профессиональной деятельности в области информатики и математического моделирования,

4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Методические рекомендации, определяющие процедуры оценивания на экзамене:

Критерии оценивания теоретического задания:

- Для получения оценки «удовлетворительно» необходимо в общем раскрыть суть вопроса.
- Для получения оценки «хорошо» необходимо полностью раскрыть суть вопроса и отразить конкретное его содержание.
- Для получения оценки «отлично» необходимо показать глубокое представление по данному вопросу, привести примеры, демонстрирующие суть вопроса.

Критерии оценивания практического задания:

- Для получения оценки «удовлетворительно» необходимо решить задание 1 с мелкими недоработками.
- Для получения оценки «хорошо» необходимо решить задания 1 и 2 с мелкими недоработками.
- Для получения оценки «отлично» необходимо полностью корректно решить задачи задания 1 и 2.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

- при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;
- при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;
- при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

5.1 Основная литература:

1. Методология и технология разработки программных систем: методы и модели программной инженерии: учебное пособие / А.Н. Полетаikin, Н.Ю. Добровольская;

Министерство науки и высшего образования Российской Федерации, Кубанский государственный университет. – Краснодар: Кубанский гос. ун-т, 2025. – 229 с.

2. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин. — Ростов-на-Дону, Таганрог : Издательство ЮФУ, 2022. — 241 с. — ISBN 978 - 5 - 9275 - 4108 - 9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125718.html> (дата обращения: 09.11.2022). — Режим доступа: для авторизир. пользователей.

3. Шениг, Г. -Ю. PostgreSQL 11. Мастерство разработки / Г. -Ю. Шениг ; перевод А. А. Слинкин. — Москва : ДМК Пресс, 2019. — 352 с. — ISBN 978-5-97060-671-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125100.html>. — Режим доступа: для авторизир. пользователей.

4. Меле, А. Django 2 в примерах / А. Меле ; перевод Д. В. Плотникова. — Москва : ДМК Пресс, 2019. — 408 с. — ISBN 978-5-97060-746-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/126199.html>. — Режим доступа: для авторизир. пользователей.

5. Митра Р., Надареишвили И. Микросервисы. От архитектуры до релиза. — СПб.: Питер, 2023. — 336 с.: ил. — (Серия «Бестселлеры O'Reilly»).

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечных системах *«Лань» и «Юрайт»*.

5.2 Дополнительная литература:

1. Ленц, М. Python: непрерывная интеграция и доставка / М. Ленц ; перевод А. Е. Мамонов, Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 168 с. — ISBN 978-5-97060-797-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/126206.html> (дата обращения: 30.11.2022). — Режим доступа: для авторизир. пользователей.

2. Методы программирования : учебно-методическое пособие / авторы В. В. Подколзин, А. Н. Полетайкин, Е. П. Лукашик [и др.] ; Министерство науки и высшего образования Российской Федерации, Кубанский государственный университет. - Краснодар : Кубанский государственный университет, 2020. - 174 с.

3. Полетайкин, А. Н. Социальные и экономические информационные системы. Законы функционирования и принципы построения [Электронный ресурс] : учебное пособие / А. Н. Полетайкин. — Электрон. текстовые данные. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2016. — 241 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/54800.html>.

4. Мостовой Я.А. Управление программными проектами [Электронный ресурс]: учебное пособие/ Мостовой Я.А.— Электрон. текстовые данные.— Самара: Поволжский государственный университет телекоммуникаций и информатики, 2016.— 103 с.— Режим доступа: <http://www.iprbookshop.ru/71894.html>.— ЭБС «IPRbooks».

5.3. Периодические издания:

1. Базы данных компании «Ист Вью» <http://dlib.eastview.com>
2. Электронная библиотека GREBENNIKON.RU <https://grebennikon.ru/>

5.4. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы

Электронно-библиотечные системы (ЭБС):

1. ЭБС «ЮРАЙТ» <https://urait.ru/>
2. ЭБС «УНИВЕРСИТЕТСКАЯ БИБЛИОТЕКА ОНЛАЙН» <http://www.biblioclub.ru/>

3. ЭБС «BOOK.ru» <https://www.book.ru>
4. ЭБС «ZNANIUM.COM» www.znanium.com
5. ЭБС «ЛАНЬ» <https://e.lanbook.com>

Профессиональные базы данных

1. Scopus <http://www.scopus.com/>
2. ScienceDirect <https://www.sciencedirect.com/>
3. Журналы издательства Wiley <https://onlinelibrary.wiley.com/>
4. Научная электронная библиотека (НЭБ) <http://www.elibrary.ru/>
5. Полнотекстовые архивы ведущих западных научных журналов на Российской платформе научных журналов НЭИКОН <http://archive.neicon.ru>
6. Национальная электронная библиотека (доступ к Электронной библиотеке диссертаций Российской государственной библиотеки (РГБ) <https://rusneb.ru/>
7. Президентская библиотека им. Б.Н. Ельцина <https://www.prilib.ru/>
8. База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
9. Springer Journals: <https://link.springer.com/>
10. Springer Journals Archive: <https://link.springer.com/>
11. Nature Journals: <https://www.nature.com/>
12. Springer Nature Protocols and Methods: <https://experiments.springernature.com/sources/springer-protocols>
13. Springer Materials: <http://materials.springer.com/>
14. Nano Database: <https://nano.nature.com/>
15. Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
16. "Лекториум ТВ" <http://www.lektorium.tv/>
17. Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

Информационные справочные системы

1. Консультант Плюс - справочная правовая система (доступ по локальной сети с компьютеров библиотеки)

Ресурсы свободного доступа

1. КиберЛенинка <http://cyberleninka.ru/>;
2. Американская патентная база данных <http://www.uspto.gov/patft/>
3. Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
4. Федеральный портал "Российское образование" <http://www.edu.ru/>;
5. Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
6. Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
7. Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
8. Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
9. Служба тематических толковых словарей <http://www.glossary.ru/>;
10. Словари и энциклопедии <http://dic.academic.ru/>;
11. Образовательный портал "Учеба" <http://www.ucheba.com/>;
12. Законопроект "Об образовании в Российской Федерации". Вопросы и ответы http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety
13. <https://www.djangoproject.com>
14. <https://tproger.ru/articles/pochemu-vam-stoit-vybrat-frejmwork-django-dlja-svoego-sledujushhego-proekta/>

15. <https://medium.com/nuances-of-programming/python-и-веб-разработка-краткое-руководство-858bf8987691>

Собственные электронные образовательные и информационные ресурсы КубГУ

1. Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
2. Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
3. Среда модульного динамического обучения <http://moodle.kubsu.ru>
4. База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>
5. Библиотека информационных ресурсов кафедры информационных образовательных технологий <http://mschool.kubsu.ru;>
6. Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
7. Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

6. Методические указания для обучающихся по освоению дисциплины (модуля)

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю)

7.1 Перечень информационно-коммуникационных технологий

- Проверка домашних заданий и консультирование посредством электронной почты.
- Использование электронных презентаций при проведении лекционных занятий
- Система MOODLE
- Проверка домашних заданий и консультирование посредством ЭОИС КубГУ

7.2 Перечень лицензионного и свободно распространяемого программного обеспечения

OpenOffice
Компилятор C++
Oracle VirtualBox 6
VMware Workstation 16
Putty 0.76 или Kitty 0.76
FileZilla 3.57.0
WinSCP 5.19
Advanced port scanner 2.5
Python 3 (3.7 И 3.9)
numpy 1.22.0
opencv 4.5.5

Keras 2.7.0
 Tensor flow 2.7.0
 matplotlib 3.5.1
 PyCharm 2021
 Cuda Toolkit 11.6
 Фреймворк Django
 Firefox, любая версия
 Putty, любая версия
 Visual Studio Code, версия 1.52+
 Eclipse PHP Development Tools, версия 2020-06+
 Плагин Remote System Explorer (RSE) для Eclipse PDT
 JetBrains PHP Storm
 GIT
 Java Version 8 Update 311
 Clojure 1.10.3.1029.ps1
 SWI Prolog 8.4
 IntelliJ Idea IDE 2021
 Mozilla Firefox 96
 Google Chrome 97
 GitHub Desktop 2.9
 PHP Storm 2021
 FileZilla 3.57.0
 Putty 0.76

8. Материально-техническое обеспечение по дисциплине (модулю)

№	Вид работ	Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения
1.	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2.	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3.	Практические занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
4.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
5.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
6.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.