

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Ростовский государственный экономический университет (РИНХ)»

УТВЕРЖДАЮ
Директор Таганрогского института
имени А. П. Чехова (филиала)
РГЭУ (РИНХ)
_____ С. А. Петрушенко
«20» мая 2025 г.

Рабочая программа дисциплины
Современные технологии разработки программного обеспечения

Направление подготовки
09.04.03 Прикладная информатика

Направленность (профиль) программы магистратуры
09.04.03.02 Информационные системы и анализ больших данных

Для набора 2025 года

Квалификация
Магистр

КАФЕДРА информатики**Распределение часов дисциплины по семестрам / курсам**

Семестр (<Курс>.<Семестр на курсе>)	2 (1.2)		Итого	
	Неделя			
Вид занятий	УП	РП	УП	РП
Лекции	28	28	28	28
Лабораторные	32	32	32	32
Итого ауд.	60	60	60	60
Контактная работа	60	60	60	60
Сам. работа	120	120	120	120
Часы на контроль	36	36	36	36
Итого	216	216	216	216

ОСНОВАНИЕ

Учебный план утвержден учёным советом вуза от 28.02.2025 протокол № 9.

Программу составил(и): канд. техн. наук, Зав. каф., Тюшнякова И. А.;Преп., Горкунов М.С.

Зав. кафедрой: Тюшнякова И. А.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	предоставление обучаемым знаний и умений в области проектирования, тестирования, отладки, внедрения и сопровождения программного обеспечения (ПО), составления документации на ПО – от этапа сбора и анализа требований заказчика, до этапа внедрения.
-----	--

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ОПК-5:	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;
ОПК-5.1:	Знать современное программное и аппаратное обеспечение информационных и автоматизированных систем
ОПК-5.2:	Уметь модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач;
ОПК-6:	Способен исследовать современные проблемы и методы прикладной информатики и развития информационного общества;
ОПК-6.1:	Знать содержание, объекты и субъекты информационного общества, критерии эффективности его функционирования; структуру интеллектуального капитала, проблемы инвестиций в экономику информатизации и методы оценки эффективности; правовые, экономические, социальные и психологические аспекты информатизации; теоретические проблемы прикладной информатики, в том числе семантической обработки информации, развитие представлений об оценке качества информации в информационных системах; современные методы, средства, стандарты информатики для решения прикладных задач различных классов; правовые, экономические, социальные и психологические аспекты информатизации деятельности организационно-экономических систем;
ОПК-6.2:	Уметь проводить анализ современных методов и средств информатики для решения прикладных задач различных классов;
ПКР-3:	Способен руководить процессами разработки программного обеспечения
ПКР-3.1:	Знать методы и приемы формализации и алгоритмизации задач, технологии программирования, особенности выбранной среды программирования, методы принятия управленческих решений и требования к разработке проектно-технической документации, методы и средства проверки работоспособности программного обеспечения
ПКР-3.2:	Уметь писать программный код на выбранном языке программирования, работать в компьютерных сетях, использовать выбранную среду программирования, применять методы и средства проверки работоспособности программного обеспечения

В результате освоения дисциплины обучающийся должен:

Знать:

современное программное и аппаратное обеспечение информационных и автоматизированных систем, жизненный цикл программ, технологии разработки программных комплексов (соотнесено с индикатором ОПК-5.1)
 содержание, объекты и субъекты информационного общества, критерии эффективности его функционирования; структуру интеллектуального капитала, проблемы инвестиций в экономику информатизации и методы оценки эффективности (соотнесено с индикатором ОПК-6.1)
 методы и приемы формализации и алгоритмизации задач, технологии программирования, особенности выбранной среды программирования, методы принятия управленческих решений и требования к разработке проектно-технической документации (соотнесено с индикатором ПКР-3.1)

Уметь:

модernизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач (соотнесено с индикатором ОПК-5.2)
 проводить анализ современных методов и средств информатики для решения прикладных задач различных классов; составлять отчетную документацию; пользоваться современными системами контроля версий и багтрекинга (соотнесено с индикатором ОПК-6.2)
 писать программный код на выбранном языке программирования, работать в компьютерных сетях, использовать выбранную среду программирования, применять методы и средства проверки работоспособности программного обеспечения; производить начальный сбор требований для обеспечения постановки ТЗ; выявлять дефекты ПО и оформлять запросы на модификацию (соотнесено с индикатором ПКР-3.2)

Владеть:

модернизации программного и аппаратного обеспечения информационных и автоматизированных систем (соотнесено с индикатором ОПК-5.2)

проведения анализа современных методов и средств информатики для решения прикладных задач различных классов; навыками поиска и анализа информации, с применением телекоммуникационных технологий (соотнесено с индикатором ОПК-6.2)

написания программного кода на выбранном языке программирования, навыками работы в компьютерных сетях, использования выбранной среды программирования, применения методов и средств проверки работоспособности программного обеспечения (соотнесено с индикатором ПКР-3.2)

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ**Раздел 1. Современные технологии разработки ПО**

№	Наименование темы, краткое содержание	Вид занятия / работы / форма ПА	Семестр / Курс	Количество часов	Компетенции
1.1	Жизненный цикл ПО. Модели жизненного цикла: каскадный.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.2	Модели жизненного цикла: v-образный, спиральный, extreme.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.3	ООП в языке Python.	Лабораторные занятия	2	4	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.4	Программные требования: основы, процессы, извлечение.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.5	Программные требования: анализ, спецификация, утверждение.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.6	Принципы Solid.	Лабораторные занятия	2	4	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.7	Проектирование ПО: основы, ключевые вопросы.	Лекционные занятия	2	4	ПКР-3 ОПК-6

					ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.8	Проектирование ПО: структура и архитектура, нотация.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.9	Принцип единственной ответственности	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.10	Принцип открытости / закрытости.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.11	Проектирование ПО: анализ качества и оценка ПО.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.12	Проектирование ПО: стратегии и методы проектирования.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.13	Конструирование ПО: основы конструирования.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.14	Конструирование ПО: управление конструированием.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.15	Принцип подстановки Барбары Лисков.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1

					ОПК-6.2 ОПК-5.1 ОПК-5.2
1.16	Принцип разделения интерфейсов.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.17	Принцип построения ПО: keep it short and simple (KISS).	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.18	Принцип построения ПО: Don't repeat yourself (DRY).	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.19	Принцип построения ПО: You aren't need it (YAGNI).	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.20	Элементы ФП в Python.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.21	Контроль качества ПО: тестирование и верификация.	Лекционные занятия	2	4	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.22	Контроль качества ПО: аудит пакета документов.	Лекционные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.23	CODE REVIEW (инспекция кода).	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.24	Метрики покрытия кода.	Лабораторные	2	2	ПКР-3

		занятия			ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.25	Методика работы по ISO 9000.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.26	Фреймворк Selenium.	Лабораторные занятия	2	2	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.27	Курсовая работа	Самостоятельная работа	2	60	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.28	Подготовка к лабораторным работам.	Самостоятельная работа	2	60	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2
1.29	Подготовка к промежуточной аттестации	Экзамен	2	36	ПКР-3 ОПК-6 ОПК-5 ПКР-3.1 ПКР-3.2 ОПК-6.1 ОПК-6.2 ОПК-5.1 ОПК-5.2

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Учебные, научные и методические издания

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
1	Мейер Б.	Объектно-ориентированное программирование и программная инженерия	Москва: Национальный Открытый Университет «ИНТУИТ», 2016	http://biblioclub.ru/index.php?page=book&id=429034
2	Билл Любанович	Простой Python. Современный стиль программирования. 2-е изд.	Санкт-Петербург: Питер, 2021	https://ibooks.ru/reading.php?short=1&productid=373510

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
3	Мякишев, Д. В.	Разработка программного обеспечения АСУ ТП на основе объектно-ориентированного подхода: методическое пособие	Москва: Инфра-Инженерия, 2019	http://www.iprbookshop.ru/86635.html
4	Суханов, М. Б.	Программная инженерия: учебное пособие	Санкт-Петербург: Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2018	http://www.iprbookshop.ru/102465.html

5.1. Учебные, научные и методические издания

	Авторы, составители	Заглавие	Издательство, год	Библиотека / Количество
1	Вендров, Александр Михайлович	Проектирование программного обеспечения экономических информационных систем: учеб. для студентов эконом. высш. учеб. заведений	М.: Финансы и статистика, 2005	3 экз.
2	Мякишев Д. В.	Разработка программного обеспечения АСУ ТП на основе объектно-ориентированного подхода: теория, модели, методы: методическое пособие	Москва Вологда: Инфра-Инженерия, 2019	http://biblioclub.ru/index.php?page=book&id=564227
3	Прохоренко Н.А., Дронов В.А.	Python 3 и PyQt 5. Разработка приложений	Санкт-Петербург: БХВ-Петербург, 2016	https://ibooks.ru/reading.php?short=1&productid=353570

5.2. Профессиональные базы данных и информационные справочные системы

Научная электронная библиотека – <https://www.elibrary.ru/defaultx.asp>
 Электронная библиотека по техническим наукам – <http://techlibrary.ru>

5.3. Перечень программного обеспечения

Python
OpenOffice

5.4. Учебно-методические материалы для обучающихся с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения:

- столы, стулья;
- персональный компьютер / ноутбук (переносной);
- проектор;
- экран / интерактивная доска.

Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет, и/или в специализированных лабораториях, предусмотренных образовательной программой.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1 Показатели и критерии оценивания компетенций:

1.2

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
ОПК-5: Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем			
<i>Знать:</i> современное программное и аппаратное обеспечение информационных и автоматизированных систем, жизненный цикл программ, технологии разработки программных комплексов	Осуществление поиска и сбора необходимой литературы, изучение основной и дополнительной литературы, подготовка доклада	соответствие проблеме исследования; полнота и содержательность ответа; умение приводить примеры; умение отстаивать свою позицию; умение пользоваться дополнительной литературой; соответствие представленной информации материалам учебной литературы, сведениям из информационных ресурсов Интернет	ВЭ – вопросы к экзамену (вопросы 1-30), Д – доклад (1-25)
<i>Уметь:</i> модernизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Выполнение индивидуального практического задания, подготовка доклада, использование современных инструментальных средств	достоверность решения заданий с помощью программных средств, умение пользоваться инструментальными средствами и электронными ресурсами	ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)
<i>Владеть:</i> модernизации программного и аппаратного обеспечения информационных и автоматизированных систем	Выполнение индивидуального домашнего задания, подготовка доклада, использование современных инструментальных средств	достоверность решения заданий с помощью программных средств, соответствие проблеме исследования, умение пользоваться инструментальными средствами и электронными ресурсами	ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)
ОПК-6: Способен исследовать современные проблемы и методы прикладной информатики и развития информационного общества			
<i>Знать:</i> содержание, объекты и субъекты информационного общества, критерии эффективности его функционирования; структуру интеллектуального капитала, проблемы инвестиций в экономику информатизации и методы оценки эффективности	Осуществление поиска и сбора необходимой литературы, изучение основной и дополнительной литературы, подготовка доклада	соответствие проблеме исследования; полнота и содержательность ответа; умение приводить примеры; умение отстаивать свою позицию; умение пользоваться дополнительной литературой; соответствие представленной информации материалам учебной литературы, сведениям из информационных ресурсов Интернет	ВЭ – вопросы к экзамену (вопросы 1-30), Д – доклад (1-25)

<i>Уметь:</i> проводить анализ современных методов и средств информатики для решения прикладных задач различных классов; составлять отчетную документацию; пользоваться современными системами контроля версий и багтрекинга	Выполнение индивидуального практического задания, подготовка доклада, использование современных инструментальных средств	достоверность решения заданий с помощью программных средств, умение пользоваться инструментальными средствами и электронными ресурсами	ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)
<i>Владеть:</i> проведения анализа современных методов и средств информатики для решения прикладных задач различных классов; навыками поиска и анализа информации, с применением телекоммуникационных технологий	Выполнение индивидуального домашнего задания, подготовка доклада, использование современных инструментальных средств	достоверность решения заданий с помощью программных средств, соответствие проблеме исследования, умение пользоваться инструментальными средствами и электронными ресурсами	ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)
ПКР-3: Способен руководить процессами разработки программного обеспечения			
<i>Знать:</i> методы и приемы формализации и алгоритмизации задач, технологии программирования, особенности выбранной среды программирования, методы принятия управленческих решений и требования к разработке проектно-технической документации	Осуществление поиска и сбора необходимой литературы, изучение основной и дополнительной литературы, подготовка доклада	соответствие проблеме исследования; полнота и содержательность ответа; умение приводить примеры; умение отстаивать свою позицию; умение пользоваться дополнительной литературой; соответствие представленной информации материалам учебной литературы, сведениям из информационных ресурсов Интернет	ВЭ – вопросы к экзамену (вопросы 1-30), Д – доклад (1-25)
<i>Уметь:</i> писать программный код на выбранном языке программирования, работать в компьютерных сетях, использовать выбранную среду программирования, применять методы и средства проверки работоспособности программного обеспечения; производить начальный сбор требований для обеспечения постановки ТЗ; выявлять дефекты ПО и оформлять запросы на модификацию	Выполнение индивидуального практического задания, подготовка доклада, использование современных инструментальных средств	достоверность решения заданий с помощью программных средств, умение пользоваться инструментальными средствами и электронными ресурсами	ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)

<p><i>Владеть:</i> навыками написания программного кода на выбранном языке программирования, навыками работы в компьютерных сетях, использования выбранной среды программирования, применения методов и средств проверки работоспособности программного обеспечения</p>	<p>Выполнение индивидуального домашнего задания, подготовка доклада, использование современных инструментальных средств</p>	<p>достоверность решения заданий с помощью программных средств, соответствие проблеме исследования, умение пользоваться инструментальными средствами и электронными ресурсами</p>	<p>ЛЗ- лабораторные задания (1-16), ИЗ- индивидуальное задание(1-4), Д – доклад (1-25)</p>
--	---	---	--

1.2 Шкалы оценивания:

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале:

- 84-100 баллов (оценка «отлично»);
- 67-83 баллов (оценка «хорошо»);
- 50-66 баллов (оценка «удовлетворительно»);
- 0-49 баллов (оценка «неудовлетворительно»).

2 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

Вопросы к экзамену

по дисциплине «Современные технологии разработки программного обеспечения»

1. Жизненный цикл программного обеспечения: классические и гибкие модели, их преимущества и недостатки.
2. Объектно-ориентированный подход в разработке программного обеспечения: основные принципы и методика проектирования.
3. Анализ требований к программному обеспечению: основные этапы, инструменты и документация.
4. Архитектурный дизайн программного обеспечения: уровни абстракции, слои и паттерны проектирования.
5. Разработка технического задания: структура, содержание и особенности написания.
6. Современные языки программирования: сравнительный анализ и сфера применения.
7. Паттерны проектирования: основные виды и примеры их использования в реальных проектах.
8. Тестирование программного обеспечения: стратегии, виды и методики тестирования.
9. Управление качеством программного обеспечения: сертификация, стандартизация и оценка качества.
10. Конфигурационное управление: цели, задачи и инструменты конфигурационного менеджмента.
11. Инструментальные средства поддержки разработки программного обеспечения: IDE, системы контроля версий, bug-tracking.
12. Безопасность программного обеспечения: угрозы, уязвимости и методы защиты.
13. Производительность и масштабируемость программного обеспечения: методы оценки и улучшения.
14. Непрерывная интеграция и доставка: CI/CD пайплайны, DevOps-подходы.
15. Front-end разработка: современные технологии и библиотеки (React, Angular, Vue.js).
16. Back-end разработка: серверные технологии и фреймворки (Django, Flask, Spring Boot).
17. Mobile-разработка: Android и iOS, основные технологии и инструменты.

18. Web-сервисы и API: RESTful API, GraphQL, SOAP.
19. Высоконагруженные системы: архитектура, кэширование, горизонтальное масштабирование.
20. Микросервисная архитектура: принципы, преимущества и случаи применения.
21. Машинное обучение и большие данные: интеграция ML-моделей в ПО, конвейеры данных.
22. Agile-методы разработки: Scrum, Kanban, XP.
23. Continuous Integration и Delivery: Jenkins, GitLab CI, Travis CI.
24. Docker и контейнеризация: цели, принципы и инструменты.
25. Kubernetes и оркестровка контейнеров: принципы работы и преимущества.
26. Методы оценки трудозатрат и рисков в разработке ПО: PERT, СОСОМО II.
27. Документирование программного обеспечения: стандарты и лучшие практики.
28. Cross-platform разработка: Electron, Flutter, Xamarin.
29. Удалённая разработка: инструменты и подходы к удалённому сотрудничеству.
30. Современные тренды и направления в разработке программного обеспечения: IoT, Blockchain, AI-first approach.

Экзаменационное задание (билет) включает 2 теоретических вопроса (формируются из представленных вопросов к экзамену) и 1 практическое задание (формируются из перечня лабораторных заданий, представленных в разделе «Лабораторные задания»).

Критерии оценивания:

Максимальное количество баллов за экзаменационное задание – 100 (50 баллов максимально за теоретические вопросы, 50 баллов максимально за практическое задание).

Критерии оценивания одного теоретического вопроса.

Критерии оценивания теоретического вопроса	Баллы
Изложенный материал фактически верен, наличие глубоких исчерпывающих знаний; правильные, уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе	21-25
Наличие твердых и достаточно полных знаний, правильные действия по применению знаний на практике, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, неуверенность и неточность ответов на дополнительные и наводящие вопросы	17-20
Неполный ответ на вопросы; затрудняется ответить на дополнительные вопросы	1-16
Ответ не связан с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы	0
<i>Максимальный балл за ответ на теоретический вопрос</i>	<i>25</i>

Критерии оценивания одного практического задания.

Критерии оценивания практического задания	Баллы
Задание выполнено в полном объеме, в представленном решении обоснованно получены правильные ответы, проведен анализ, дана грамотная интерпретация полученных результатов, сделаны выводы	41-50
Задание выполнено в полном объеме, но при анализе и интерпретации полученных результатов допущены незначительные ошибки, выводы – достаточно обоснованы, но неполны	26-40
Задание выполнено не в полном объеме, при анализе и интерпретации полученных результатов допущены ошибки, выводы – но неполные или отсутствуют	1-25
Задание выполнено полностью неверно или отсутствует решение	0
<i>Максимальный балл за выполнение практического задания</i>	<i>50</i>

Итоговая оценка формируется из суммы набранных баллов за выполнение экзаменационного задания (2 теоретических вопроса и 1 практическое задание) и соответствует шкале:

- 84-100 баллов (оценка «отлично»);
- 67-83 баллов (оценка «хорошо»);
- 50-66 баллов (оценка «удовлетворительно»);
- 0-49 баллов (оценка «неудовлетворительно»).

Лабораторные задания

по дисциплине «Современные технологии разработки программного обеспечения»

Тематика лабораторных работ

1. Модели жизненного цикла программного обеспечения: Waterfall, V-model, Spiral, Extreme Programming.
2. Объектно-ориентированное программирование на Python: классы, наследование, инкапсуляция.
3. Анализ и спецификация требований к программному обеспечению: use cases, user stories, wireframes.
4. Архитектура программного обеспечения: MVC, MVVM, Microservices.
5. Принципы SOLID в объектно-ориентированном программировании.
6. Проектирование программного обеспечения: UML-диаграммы, DFD-диаграммы.
7. Принцип единственной ответственности (Single Responsibility Principle).
8. Принцип открытости-закрытости (Open-Closed Principle).
9. Принцип подстановки Барбары Лисков (Liskov Substitution Principle).
10. Принцип разделения интерфейсов (Interface Segregation Principle).
11. Принцип инверсии зависимостей (Dependency Inversion Principle).
12. Методологии разработки программного обеспечения: Scrum, Kanban, Lean Development.
13. Рефакторинг кода: выявление дублирующего кода, удаление неиспользуемых конструкций.
14. Принципы построения программного обеспечения: Keep It Simple Stupid (KISS), Don't Repeat Yourself (DRY), You Ain't Gonna Need It (YAGNI).
15. Тестирование программного обеспечения: юнит-тесты, интеграционные тесты, нагрузочное тестирование.
16. Continuous Integration и Continuous Deployment: Jenkins, GitLab CI, Travis CI.

Критерии оценивания:

Максимальное количество баллов – 50 (за 16 лабораторных заданий).

Критерии оценивания лабораторных заданий задания.

Критерии оценивания практического задания	Баллы
Все задания выполнены в полном объеме, в представленных решениях обоснованно получены правильные ответы, проведен анализ, дана грамотная интерпретация полученных результатов, сделаны выводы	41-50
Задания выполнены в полном объеме, но при анализе и интерпретации полученных результатов допущены ошибки, выводы по результатам выполнения заданий достаточно обоснованы, но неполны	26-40
Задания выполнены не в полном объеме, при анализе и интерпретации полученных результатов допущены ошибки, выводы – но неполные или отсутствуют	1-25
Задания выполнены полностью неверно или отсутствует решения	0
<i>Максимальный балл за все лабораторные задания</i>	
	50

Перечень тем для докладов

по дисциплине «Современные технологии разработки программного обеспечения»

1. Жизненный цикл разработки программного обеспечения: традиционные и гибкие методологии.
2. Object-Oriented Programming (ООП) в Python: основные принципы и их применение.
3. Паттерны проектирования в программировании: их классификация и примеры использования.
4. UML-диаграммы и их роль в проектировании программного обеспечения.
5. Scrum и Kanban: сравнение и выбор подходящей методологии разработки.
6. Microservice Architecture: принципы, преимущества и примеры реализации.
7. Agile и DevOps: философия и принципы современного подхода к разработке ПО.
8. RESTful API: проектирование и реализация.
9. Continuous Integration & Continuous Deployment (CI/CD): инструменты и практики.
10. Docker и контейнеризация: основы и примеры использования.
11. Frontend-разработка: современные фреймворки (React, Angular, Vue.js).
12. Backend-разработка: Django, Flask, Node.js.
13. NoSQL базы данных: MongoDB, Cassandra, Redis.
14. Тестирование программного обеспечения: юнит-тесты, интеграционные тесты, end-to-end тестирование.
15. Code review и best practices инспектирования кода.
16. Безопасность программного обеспечения: защита данных и предотвращение угроз.
17. Cloud computing и инфраструктура как услуга (IaaS, PaaS, SaaS).
18. Производительность и масштабируемость web-приложений.
19. Data Science и Machine Learning в разработке программного обеспечения.
20. Анализ и спецификация требований к программному обеспечению.
21. Современные подходы к написанию и сопровождению технической документации.
22. Low-code и no-code платформы: их применение и ограничения.
23. Serverless архитектура: примеры использования и преимущества.
24. GraphQL: альтернатива REST и будущее API.
25. Domain-Driven Design (DDD): теория и практика применения.

Критерии оценивания:

- 15-25 баллов - выставляется обучающемуся, если тема соответствует содержанию доклада; основные понятия проблемы изложены верно; сделаны обобщения и сопоставления различных точек зрения по рассматриваемому вопросу; сделаны и аргументированы основные выводы, доклад сопровождается разработанной мультимедийной презентацией;

- 0-14 баллов - выставляется обучающемуся, если содержание не соответствует теме; нет ссылок на использованные источники; тема не полностью раскрыта; нет выводов.

Темы индивидуального задания

по дисциплине «Современные технологии разработки программного обеспечения»

ИЗ 1. Разработка простого веб-приложения на Python/Django.

Цель - получить навыки разработки backend-части веб-приложения с использованием Django.

Задачи:

Создать проект Django с базой данных SQLite.

Разработать CRUD-операции для модели данных (например, таблица пользователей).

Настроить аутентификацию и авторизацию.

Реализовать REST API с использованием Django Rest Framework.

Провести тестирование с помощью pytest.

ИЗ 2. Создание frontend-приложения на React.

Цель - освоить разработку клиентской части веб-приложения с использованием React.

Задачи:

Создать SPA-приложение на React с использованием create-react-app.

Реализовать роутинг с помощью react-router-dom.

Интегрировать запросы к REST API, созданному в первом задании.

Создать кастомные компоненты UI.

Настроить unit-тестирование с Jest.

ИЗ 3. Работа с контейнерами Docker.

Цель - освоить контейнеризацию приложений с помощью Docker.

Задачи:

Создать Dockerfile для сборки веб-приложения (Django + PostgreSQL).

Создать docker-compose.yml для одновременного запуска нескольких контейнеров.

Скрестить локальную копию Docker-образа и запустить приложение.

Провести деплой контейнера на удалённый хост (например, DigitalOcean).

ИЗ 4. Внедрение CI/CD пайплайна.

Цель - реализовать непрерывную интеграцию и доставку с использованием GitLab CI/CD.

Задачи:

Настроить репозиторий на GitLab с исходным кодом приложения.

Создать .gitlab-ci.yml для сборки и тестирования приложения.

Настроить деплой приложения на удалённом сервере.

Автоматизировать процесс ручного тестирования с помощью Selenium Grid.

Критерии оценивания:

- 21-25 баллов - выставляется обучающему, если все пункты задания выполнены безошибочно;
- 17-20 баллов - выставляется обучающему, если задание, большей частью, выполнено или выполнено с ошибками.
- 0-16 баллов - выставляется обучающему, если задание выполнено с существенными ошибками, выполнено наполовину или не выполнено содержание не соответствует теме; нет ссылок на использованные источники; тема не полностью раскрыта; нет выводов.

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

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения обучающихся до промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена, это аттестация в период сессии, которая проводится в соответствии с действующим в РГЭУ (РИНХ) Положением о курсовых, экзаменах и зачётах.

Экзамен проводится по расписанию промежуточной аттестации в компьютерном классе. Количество вопросов в экзаменационном задании – 3 (2 теоретических вопроса и 1 практическое задание). Объявление результатов производится в день экзамена. Результаты аттестации заносятся в ведомость и зачетную книжку обучающегося.

Обучающиеся, не прошедшие промежуточную аттестацию по графику промежуточной аттестации, должны ликвидировать задолженность в установленном порядке.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:

- лекции;
- лабораторные работы.

Важным условием успешного освоения дисциплины «Современные технологии разработки программного обеспечения» является создание системы правильной организации труда, позволяющей распределить учебную нагрузку равномерно в соответствии с графиком образовательного процесса. В ходе лекционных занятий рассматриваются теоретические основы, понятия, методы и проблемы разработки программного обеспечения. Знакомство с дисциплиной происходит уже на первой лекции, где от обучающегося требуется не просто внимание, но и самостоятельное оформление конспекта. При работе с конспектом лекций необходимо учитывать тот фактор, что одни лекции дают ответы на конкретные вопросы темы, другие – лишь выявляют взаимосвязи между явлениями, помогая обучающемуся понять глубинные процессы развития изучаемого предмета.

Изучение дисциплины проходит с акцентом на лабораторные работы. В ходе лабораторных занятий развиваются умения и навыки использования различных инструментов и технологии разработки программных продуктов. Все задания к лабораторным занятиям рекомендуется выполнять непосредственно после соответствующей темы лекционного курса, что способствует лучшему усвоению материала, позволяет своевременно выявить и устранить «пробелы» в знаниях, систематизировать ранее пройденный материал, на его основе приступить к овладению новыми знаниями и навыками.

В процессе подготовки к лабораторным занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме.

Лабораторные занятия проводятся в компьютерных классах с применением специально разработанных учебно-методических материалов, в которых изложены подробные методические рекомендации по изучению каждой темы и выполнению заданий. Наличие таких учебно-методических и дидактических материалов позволяет каждому обучающемуся работать в своем индивидуальном темпе, а также дополнительно прорабатывать изучаемый материал во время самостоятельных занятий.

Перед выполнением лабораторной работы требуется получить вариант задания. Далее необходимо ознакомиться с заданием. Выполнение лабораторной работы следует начать с изучения теоретических сведений, которые приводятся в соответствующих методических указаниях. Результаты работы необходимо оформить в виде отчета. Лабораторная работа считается выполненной, если предоставлен отчет о результатах выполнения задания; проведена защита проделанной работы. Защита проводится в

два этапа: демонстрация результатов выполнения задания; ответы на вопросы из перечня контрольных вопросов, который приводится в задании на лабораторную работу.

Для успешного овладения предлагаемым курсом обучающийся должен обладать определённой информационной культурой: навыками работы с литературой, умением определять и находить информационные ресурсы, соответствующие целям и задачам образовательного процесса, получать к ним доступ и использовать в целях повышения эффективности своей профессиональной деятельности. При изучении данного курса необходимо использовать компьютер, изучать дополнительные информационные ресурсы.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации обучающиеся могут воспользоваться электронно-библиотечными системами.