Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Шутов Олег Леонтьевич
Должность: Ректор
Дата подписания: 16.11.2026 11:26:31
Уникальный программный ключ:
6892313c2153d214b87fca0fd68c13fa12d41989

**Разработка программно-информационных систем**
**09.03.04 Программная инженерия**
**2025 год набора**

**Приложение В**

к основной профессиональной образовательной программе
по направлению подготовки 09.03.04 Программная инженерия,
утвержденной приказом от 30.08.2025 г. № 47-О
*(в редакции приказа от 02.11.2026 г. № 87-О)*

АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОБРАЗОВАТЕЛЬНАЯ
ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
**«КУБАНСКИЙ ИНСТИТУТ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ»**
(АНОО ВО «КИПО»)

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

## Б1.В.ДВ.01.02 Архитектура и проектирование программных систем

Направление подготовки

**09.03.04 Программная инженерия**

Направленность (профиль)

**Разработка программно-информационных систем**

Уровень высшего образования

**Бакалавриат**

Квалификация

**Бакалавр**

Форма обучения

**очная/заочная**

Год набора

**2025**

Рабочая программа дисциплины (модуля) Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем» предназначена для реализации основной профессиональной образовательной программы высшего образования.

Составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 09.03.04 Программная инженерия (Приказ Минобрнауки России от 19.09.2017 г. № 920 зарегистрирован в Минюсте России от 16.10.2017 г. № 48546).

# СОДЕРЖАНИЕ

1 Цели и задачи изучения дисциплины(модуля)
1.1 Цель освоения дисциплины (модуля)
1.2 Задачи дисциплины (модуля)
1.3 Место дисциплины (модуля) в структуре образовательной программы
1.4 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы
2. Структура и содержание дисциплины (модуля)
2.1 Распределение трудоёмкости дисциплины (модуля) по видам работ
2.2 Содержание дисциплины (модуля)
2.3 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
3. Образовательные технологии, применяемые при освоении дисциплины (модуля)
4. Фонды оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)
4.1. Структура оценочных средств для текущего контроля и промежуточной аттестации по дисциплине (модулю)
4.2. Типовые задания для текущего контроля и вопросы (теоретические и практические) для промежуточной аттестации по дисциплине (модулю)
5. Методические указания для обучающихся по освоению дисциплины (модуля)
6. Материально-техническое и учебно-методическое обеспечение по дисциплине (модулю)

## 1 Цели и задачи изучения дисциплины (модуля)

### 1.1 Цель освоения дисциплины

Целью изучения дисциплины Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем» является получение теоретических знаний о принципах, технологии, методах и средствах проектирования архитектуры программных систем, а также приобретение практических навыков в выполнении действий по различным фазам создания программных продуктов.

### 1.2 Задачи дисциплины

изучение основных понятий, принципов и подходов к проектированию архитектуры программных систем;

освоение современных архитектурных стилей (монолитная, микросервисная, событийно-ориентированная, клиент-серверная, сервис-ориентированная и др.) и паттернов проектирования;

формирование умений анализировать требования к программной системе и выбирать оптимальное архитектурное решение;

освоение нотаций и инструментов для документирования архитектуры (UML, C4 model);

приобретение навыков разработки, анализа и обоснования архитектурных решений;

изучение методов оценки качества архитектуры и работы с архитектурными рисками;

формирование способности представлять и защищать архитектурные решения перед командой и заказчиком.

### 1.3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» учебного плана.

В соответствии с рабочим учебным планом дисциплина изучается на 4 курсе по очной форме и на 4 курсе по заочной форме обучения.

Вид промежуточной аттестации: экзамен.

### 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:

| Код и наименование компетенции | Код и наименование индикаторов достижения результатов обучения по дисциплине | Планируемые результаты обучения |
|---|---|---|
| ПК-10. Способен разрабатывать компиляторы, загрузчики, сборщики | ПК-10.1. Знает архитектуру целевой аппаратной платформы, для которой разрабатывается программное обеспечение<br>ПК-10.2. Умеет применять технологию разработки компиляторов, загрузчики, сборщики<br>ПК-10.3. Владеет навыками сопровождения разработанных компиляторов, загрузчиков, сборщиков | **Знать:** основные архитектурные стили (монолит, микросервисы, клиент-сервер); принципы проектирования программных систем (SOLID, DRY, KISS); основные паттерны проектирования (порождающие, структурные, поведенческие); нотации для описания архитектуры (UML: диаграммы классов, компонентов, последовательностей, развертывания). |

| Код и наименование компетенции | Код и наименование индикаторов достижения результатов обучения по дисциплине | Планируемые результаты обучения |
|---|---|---|
| | | **Уметь:** анализировать требования и выбирать архитектурное решение для программной системы; разрабатывать UML-диаграммы для описания различных аспектов архитектуры системы; применять выбранные паттерны проектирования при разработке архитектуры; проектировать программные интерфейсы (API) и взаимодействие компонентов. **Владеть:** демонстрировать в деятельности навыками выбора и обоснования архитектурного стиля для конкретной программной системы; навыками документирования архитектурных решений с использованием стандартных нотаций (Uml, C4 model); навыками рефакторинга и эволюционного изменения архитектуры; навыками представления и защиты архитектурных решений перед командой. |

Результаты обучения по дисциплине достигаются в рамках осуществления всех видов контактной и самостоятельной работы обучающихся в соответствии с утвержденным учебным планом.

Индикаторы достижения компетенций считаются сформированными при достижении соответствующих им результатов обучения.

## 2. Структура и содержание дисциплины

### 2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 6 зачетных единиц (216 час.), их распределение по видам работ представлено в таблице

| Виды работ | | Всего часов | | |
|---|---|---|---|---|
| | | ОФО | ОЗФО | ЗФО |
| **Контактная работа, в том числе:** | | **72** | **-** | **14** |
| **Аудиторные занятия (всего):** | | **72** | **-** | **14** |
| занятия лекционного типа | | 18 | - | 4 |
| практические занятия | | 54 | - | 10 |
| **Иная контактная работа:** | | **-** | **-** | **-** |
| Контрольная работа | | - | - | - |
| Курсовая работа | | - | - | - |
| **Самостоятельная работа, в том числе:** | | **108** | **-** | **193** |
| Самоподготовка по темам (разделам) дисциплины | | 98 | - | 173 |
| Подготовка к текущему контролю | | 10 | - | 20 |
| **Контроль:** | | **36** | **-** | **9** |
| Промежуточная аттестация (экзамен) | | 36 | - | 9 |
| **Общая трудоемкость** | **час.** | **216** | **-** | **216** |
| | **в том числе контактная работа** | **72** | **-** | **14** |
| | **зач. ед** | **6** | **-** | **6** |

## 2.2 Содержание дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые на 4 курсе (очная форма обучения)

| № | Наименование темы/раздела | Количество часов | | | | |
|---|---|---|---|---|---|---|
| | | Всего | В том числе в виде практической подготовки | Аудиторная работа | | Внеаудиторная работа (СР) |
| | | | | Л | ПЗ | |
| 1. | 1. **Введение в дисциплину**.<br>Основные понятия и определения.<br>Индустрия промышленной разработки программных систем.<br>Становление и развитие программной инженерии.<br>Определение понятия программный инженер. | 24 | - | 2 | 7 | 15 |
| 2. | 2. **Архитектура программных систем.** Архитектурные структуры и представления<br>Модульные структуры. Структуры распределения. Отношения между структурами<br>Варианты архитектур программных систем.<br>Архитектуры: основанная на уровнях абстракций, на портах, на потоках данных<br>Макетирование. Стратегии конструирования ПС | 22 | - | 2 | 8 | 12 |
| 3. | 3. **Проектирование программных систем**. Постановка требований к ПС<br>Методология решения задач проектирования по Майерсу. Уровни требований к программным системам.<br>Процесс определения целей продукта и проекта. | 22 | - | 2 | 7 | 13 |
| 4. | 4. **Разработка технического задания на проектирование программных систем**<br>Проектирование программных систем.<br>Анализ требований и разработка внешних спецификаций | 24 | - | 2 | 6 | 16 |
| 5. | 5. **Разработка предварительного внешнего проекта**.<br>Процесс внешнего проектирования | 20 | - | 2 | 6 | 12 |
| 6. | 6. **Методы формального функционального проектирования**.<br>Анализ требований и определение спецификаций при объектном подходе к проектированию.<br>Методология проектирования архитектуры программных систем. Методы проектирования компонентных и модульных архитектур ПС | 23 | - | 4 | 8 | 11 |
| 7. | 7. **Структурное проектирование**.<br>Оценка сложности модульных иерархических структур | 23 | - | 2 | 6 | 15 |
| 8. | 8. **Проектирование и программирование подсистем**<br>Пример проектирования структуры программной системы. | 22 | - | 2 | 6 | 14 |
| | *ИТОГО по разделам дисциплины* | 180 | - | 18 | 54 | 108 |
| | Контрольная работа | - | - | - | - | - |
| | Курсовая работа | - | - | - | - | - |
| | Промежуточная аттестация (экзамен) | 36 | - | - | - | - |
| | Общая трудоемкость по дисциплине | 216 | - | 18 | 54 | 108 |

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы (темы) дисциплины, изучаемые на 4 курсе (заочная форма обучения)

| № | Наименование темы/раздела | Количество часов | | | | |
|---|---|---|---|---|---|---|
| | | Всего | В том числе в виде практической подготовки | Аудиторная работа | | Внеаудиторная работа (СР) |
| | | | | Л | ПЗ | |
| 1. | 1. **Введение в дисциплину**.<br>Основные понятия и определения.<br>Индустрия промышленной разработки программных систем.<br>Становление и развитие программной инженерии.<br>Определение понятия программный инженер. | 29 | - | 1 | 2 | 26 |
| 2. | 2. **Архитектура программных систем.** Архитектурные структуры и представления<br>Модульные структуры. Структуры распределения.<br>Отношения между структурами<br>Варианты архитектур программных систем.<br>Архитектуры: основанная на уровнях абстракций, на портах, на потоках данных<br>Макетирование. Стратегии конструирования ПС | 30 | - | 1 | 1 | 28 |
| 3. | 3. **Проектирование программных систем**. Постановка требований к ПС<br>Методология решения задач проектирования по Майерсу. Уровни требований к программным системам.<br>Процесс определения целей продукта и проекта. | 26 | - | - | 2 | 24 |
| 4. | 4. **Разработка технического задания на проектирование программных систем**<br>Проектирование программных систем.<br>Анализ требований и разработка внешних спецификаций | 24 | - | 1 | 1 | 22 |
| 5. | 5. **Разработка предварительного внешнего проекта**.<br>Процесс внешнего проектирования | 27 | - | - | 1 | 26 |
| 6. | 6. **Методы формального функционального проектирования**.<br>Анализ требований и определение спецификаций при объектном подходе к проектированию.<br>Методология проектирования архитектуры программных систем. Методы проектирования компонентных и модульных архитектур ПС | 26 | - | 1 | 1 | 24 |
| 7. | 7. **Структурное проектирование**.<br>Оценка сложности модульных иерархических структур | 22 | - | - | 1 | 21 |
| 8. | 8. **Проектирование и программирование подсистем**<br>Пример проектирования структуры программной системы. | 23 | - | - | 1 | 22 |
| | *ИТОГО по разделам дисциплины* | 207 | - | 4 | 10 | 193 |
| | Контрольная работа | - | - | - | - | - |
| | Курсовая работа | - | - | - | - | - |
| | Промежуточная аттестация (экзамен) | 9 | - | - | - | - |
| | Общая трудоемкость по дисциплине | 216 | - | 4 | 10 | 193 |

Примечание: Л – лекции, ПЗ – практические занятия / семинары, СР – самостоятельная работа обучающегося

При изучении дисциплины могут применятся электронное обучение, дистанционные образовательные технологии в соответствии с ФГОС ВО.

**2.3 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине)**

Самостоятельная работа – это индивидуальная познавательная деятельность обучающегося как на аудиторных занятиях, так и во внеаудиторное время. Самостоятельная работа должна быть многогранной и иметь четко выраженную направленность на формирование конкретных компетенций.

Цель самостоятельной работы – овладение знаниями, профессиональными умениями и навыками, опытом исследовательской деятельности и обеспечение формирования профессиональных компетенций, воспитание потребности в самообразовании, ответственности и организованности, творческого подхода к решению проблем.

Самостоятельная работа обучающихся направлена на углубленное изучение разделов и тем рабочей программы. Самостоятельная работа предполагает изучение литературных источников, выполнение контрольных заданий и работ, проведение исследований разного характера. Работа основывается на анализе литературных источников и других материалов, а также реальных фактов, личных наблюдений.

Самостоятельная работа включает разнообразный комплекс видов и форм работы обучающихся:

- работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы;
- поиск (подбор) и обзор литературы, электронных источников информации по заданной проблеме курса, написание реферата (доклада, эссе), исследовательской работы по заданной проблеме;
- выполнение задания по пропущенной или плохо усвоенной теме;
- изучение материала, вынесенного на самостоятельную проработку (отдельные темы, параграфы);
- подготовка к практическим занятиям;
- подготовка к промежуточной аттестации.

| № п/п | Вид учебно-методического обеспечения |
|---|---|
| 1. | Методические рекомендации по самостоятельной работе обучающихся. |
| 2. | Методические рекомендации по изучению дисциплины. |
| 3. | Вопросы для письменного/устного собеседования, реферат, сообщение, доклад, эссе, практико-ориентированные задания, мини-кейсы, задания в виде расчетных задач, ситуационные задачи. |

Задания для самостоятельной работы обучающихся по дисциплине Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем» представлены в учебно-методическом отделе.

Контроль результатов самостоятельной работы обучающихся может осуществляться в пределах времени, отведенного на обязательные учебные занятия и внеаудиторную самостоятельную работу обучающихся по дисциплине, может проходить в письменной, устной или смешанной форме.

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) при изучении данной дисциплины предоставлена возможность выбора технологий обучения в зависимости от степени заболевания и осознания своей деятельности. При этом содержание программы дисциплины не изменяется, изменяются, как правило, форма обучения и образовательные технологии. Также обучающимся, имеющим инвалидность, и

лицам с ограниченными возможностями здоровья созданы условия комфортного психологического климата в процессе обучения и возможности оказания помощи в установлении полноценных межличностных отношений с другими обучающимися.

## 3. Образовательные технологии, применяемые при освоении дисциплины (модуля)

В ходе изучения дисциплины предусмотрено использование следующих образовательных технологий: лекции, практические занятия, проблемное обучение, модульная технология, подготовка письменных аналитических работ, самостоятельная работа обучающихся.

Компетентностный подход в рамках преподавания дисциплины реализуется в использовании интерактивных технологий и активных методов (проектных методик, мозгового штурма, разбора конкретных ситуаций, анализа педагогических задач, педагогического эксперимента, иных форм) в сочетании с внеаудиторной работой.

Информационные технологии, применяемые при изучении дисциплины: использование информационных ресурсов, доступных в информационно-телекоммуникационной сети Интернет.

Адаптивные образовательные технологии, применяемые при изучении дисциплины – для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

## 4. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем». Материалы для проведения текущего контроля успеваемости и промежуточной аттестации размещены в фонде оценочных средств по дисциплине Б1.В.ДВ.01.02 «Архитектура и проектирование программных систем».

### 4.1. Структура оценочных средств для текущего контроля и промежуточной аттестации по дисциплине (модулю)

| № п/п | Код и наименование индикаторов достижения результатов обучения по дисциплине | Результаты обучения | Наименование оценочного средства | |
|---|---|---|---|---|
| | | | Текущий контроль | Промежуточная аттестация |
| 1 | ПК-10.1. Знает архитектуру целевой аппаратной платформы, для которой разрабатывается программное обеспечение | **Знать:** основные архитектурные стили (монолит, микросервисы, клиент-сервер); принципы проектирования программных систем (SOLID, DRY, KISS); основные паттерны проектирования (порождающие, структурные, поведенческие); нотации для описания архитектуры (UML: диаграммы классов, компонентов, | Подготовка докладов/сообщений, вопросы для обсуждения по темам, задания открытого и закрытого типа | Вопросы на экзамене |
| 2 | ПК-10.2. Умеет применять технологию разработки компиляторов, загрузчики, сборщики | | Подготовка докладов/сообщений, вопросы для обсуждения по темам, задания открытого и закрытого типа | Вопросы на экзамене |

| | | | | |
|---|---|---|---|---|
| 3 | ПК-10.3. Владеет навыками сопровождения разработанных компиляторов, загрузчиков, сборщиков | последовательностей, развертывания). **Уметь:** анализировать требования и выбирать архитектурное решение для программной системы; разрабатывать UML-диаграммы для описания различных аспектов архитектуры системы; применять выбранные паттерны проектирования при разработке архитектуры; проектировать программные интерфейсы (API) и взаимодействие компонентов. **Владеть (демонстрировать в деятельности):** навыками выбора и обоснования архитектурного стиля для конкретной программной системы; навыками документирования архитектурных решений с использованием стандартных нотаций (Uml, C4 model); навыками рефакторинга и эволюционного изменения архитектуры; навыками представления и защиты архитектурных решений перед командой. | Подготовка докладов/сообщений, вопросы для обсуждения по темам, задания открытого и закрытого типа | Вопросы на экзамене |

## 4.2. Типовые задания для текущего контроля и вопросы (теоретические и практические) для промежуточной аттестации по дисциплине (модулю)

Задания для текущего контроля и вопросы (теоретические и практические) для промежуточной аттестации, необходимые для оценки образовательных достижений обучающихся.

### Текущий контроль успеваемости для обучающихся по очной форме

**Задание 1.** Соотнесите архитектурный стиль и его характеристику:

| Архитектурный стиль | Характеристика |
|---|---|
| 1. Монолит | А. Система разбита на множество небольших независимых сервисов |
| 2. Микросервисная | Б. Единое приложение, развертываемое как целое |

| Архитектурный стиль | Характеристика |
|---|---|
| 3. Событийно-ориентированная | В. Компоненты взаимодействуют через асинхронные события |
| 4. Клиент-сервер | Г. Разделение на поставщика ресурсов (сервер) и потребителя (клиент) |

**Задание 2.** Выберите один правильный вариант ответа.

Какой паттерн проектирования относится к **порождающим** паттернам?

1. Наблюдатель (Observer)
2. Фабричный метод (Factory Method)
3. Декоратор (Decorator)
4. Стратегия (Strategy)

**Задание 3.** Отметьте все верные утверждения для микросервисной архитектуры:

Все сервисы развертываются вместе в одном процессе
Сервисы могут быть развернуты и масштабированы независимо
Сервисы общаются друг с другом через легковесные протоколы (HTTP, gRPC)
Изменение в одном сервисе требует переразвертывания всей системы

**Задание 4.** Установите правильную последовательность этапов архитектурного проектирования:

1. Анализ нефункциональных требований
2. Выбор архитектурного стиля
3. Сбор и анализ функциональных требований
4. Документирование архитектурного решения
5. Оценка рисков и компромиссов

**Задание 5.** Какая диаграмма UML лучше всего подходит для отображения архитектуры системы на уровне взаимодействия компонентов?

1. Диаграмма классов
2. Диаграмма последовательности
3. Диаграмма компонентов
4. Диаграмма состояний

**Задание 6.** Расположите уровни C4 model от самого высокого (контекстный) к самому низкому (уровень кода):

| Уровень | Название |
|---|---|
| 1 | Код (Code) |

| Уровень | Название |
|---|---|
| 2 | Контейнеры (Containers) |
| 3 | Компоненты (Components) |
| 4 | Контекст системы (System Context) |

**Задание 7 (практическое).** На основе текстового описания системы (например, «Интернет-магазин с функцией оплаты через API банка и доставкой через курьерскую службу») разработайте диаграмму компонентов (Component Diagram), указав не менее 4 компонентов и связи между ними.

**Задание 8.** Прочитайте описание системы. Выберите наиболее подходящий архитектурный стиль и обоснуйте свой выбор (2-3 предложения).

**Описание:** Система должна обрабатывать до 100 000 запросов в секунду в пиковые часы. Разные модули системы обновляются независимо несколько раз в день. Команда разработки состоит из 5 независимых групп.

**Задание 9.** Какое из требований относится к нефункциональным**?**

1. «Система должна позволять пользователю создавать заказ»
2. «Система должна обрабатывать 1000 запросов в секунду»
3. «Система должна хранить информацию о товарах»
4. «Система должна отправлять уведомление о статусе заказа»

**Задание 10.** Приведите пример архитектурного риска для микросервисной системы и предложите способ его минимизации.

**Шкала оценивания результатов по заданиям для проведения текущего контроля успеваемости по дисциплине**

| % верных решений (ответов) | Шкала оценивания |
|---|---|
| 85-100 | 5 - отлично |
| 71-84 | 4 - хорошо |
| 50-70 | 3 - удовлетворительно |
| 0-49 | 2 - неудовлетворительно |

**Текущий контроль успеваемости для обучающихся по заочной форме**

Контрольная работа представляет собой систематическое, достаточно полное изложение авторского решения соответствующей проблемы и выполнение заданий в рамках дисциплины, которая является одним из видов текущего контроля успеваемости обучающихся заочной формы обучения.

Цели контрольной работы:

- проверка и оценка знаний обучающихся;

- закрепление практических навыков применения теоретических подходов и методов анализа на учебных примерах и задачах;

- получение информации об уровне самостоятельности и активности обучающегося, об эффективности форм и методов учебной работы.

Контрольные работы выполняются обучающимися в сроки, предусмотренные учебным планом и календарным учебным графиком.

Контрольная работа выполняется в рукописном или в печатном (компьютерном) варианте на листах формата А4 в 1 экземпляре с соблюдением установленного формата. Текст набирается шрифтом Times New Roman 12, через 1 интервал, абзацный отступ - 1,25 см, выравнивание по ширине страницы. Страница должна иметь следующие поля: левое - 25 мм, правое - 10 мм, верхнее и нижнее - 20 мм. Титульный лист содержит информацию об обучающемся выполнившем контрольную работу (ФИО обучающегося, направление подготовки, группа); наименование дисциплины; ФИО преподавателя, проверяющего работу.

Задания для контрольных работ разрабатываются преподавателем дисциплины по вариантам, которые содержат:

1) Задание в форме ответа на теоретический вопрос по теме (разделу) – объем не более 2-3 страниц;
2) Задания, составленные в форме тестов (2 задания открытого и закрытого типа, разработанные в фонде оценочных средств) или ответ на практическое задание (решить ситуацию, математическую задачу, расчетно-графическое задание и др.).

Готовая контрольная работа в электронном виде прикрепляется в электронную образовательную среду Moodle в профиль обучающегося выполнившего работу до начала сессии. Если работа в рукописном варианте, то она должна быть отсканирована и прикреплена.

**Шкала и критерии оценивания контрольной работы**

| № п/п | Критерии | Зачтено |
|---|---|---|
| **Теоретический вопрос** | | |
| 1 | Глубина проработки материала | Основные теоретические положения по вопросу раскрыты. Имеются элементы обоснования выводов |
| 2 | Представление | Имеются элементы систематизации информации, факты применения профессиональной терминологии |
| 3 | Использование рекомендованной литературы | Основные источники рекомендованной литературы использованы |
| 4 | Грамотность изложения и качество оформления | Продемонстрирована культура речи. Соблюдены основные требования к оформлению |
| **Выполнение тестовых заданий** | | |

Если работа не отвечает названным критериям, выставляется оценка «не зачтено».

## Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен)

### Теоретические вопросы к экзамену

1. Понятие архитектуры программной системы. Ее роль в жизненном цикле разработки ПО.
2. Архитектурные стили: понятие, классификация, критерии выбора.
3. Монолитная архитектура: преимущества, недостатки, варианты реализации.
4. Микросервисная архитектура: принципы, преимущества и вызовы.

5. Событийно-ориентированная архитектура (EDA): компоненты, сценарии применения.
6. Клиент-серверная архитектура: двухзвенная и многозвенная (N-tier) реализация.
7. Сервис-ориентированная архитектура (SOA): основные принципы и отличия от микросервисов.
8. Порождающие паттерны проектирования: назначение, примеры.
9. Структурные паттерны проектирования: назначение, примеры.
10. Поведенческие паттерны проектирования: назначение, примеры.
11. Паттерны распределенных систем: Circuit Breaker, SAGA, API Gateway (назначение и сценарии применения).
12. Функциональные и нефункциональные требования. Влияние нефункциональных требований на архитектуру.
13. Ключевые качественные атрибуты ПО: производительность, масштабируемость, доступность, надежность.
14. Архитектурные риски: понятие, примеры, методы управления.
15. Архитектурные компромиссы: принцип «не существует идеальной архитектуры».
16. Роль документации в архитектуре ПО. Подходы к документированию.
17. Нотация UML: основные типы диаграмм для описания архитектуры.
18. C4 model: уровни и назначение каждого уровня.
19. Архитектурные решения и их документирование (ADR — Architectural Decision Records).
20. Этапы архитектурного проектирования.
21. Принципы проектирования SOLID: назначение и применение.
22. Принципы DRY, KISS, YAGNI в контексте архитектуры ПО.
23. Методы оценки качества архитектуры: ATAM, Architecture Tradeoff Analysis Method.
24. Архитектура «чистых» слоев (Clean Architecture, Hexagonal Architecture).
25. Паттерн CQRS (Command Query Responsibility Segregation): суть, преимущества, недостатки.
26. Архитектура на основе очередей сообщений (Message Queue Architecture).
27. Шаблоны интеграции систем: точка-точка, через шину данных (ESB), через API Gateway.

**Практические задания к экзамену**

**1. Разработка функциональных требований для системы управления проектами**

**Задание**: Создайте документ с функциональными требованиями для системы управления проектами. Система должна включать следующие функции:

- Управление задачами (создание, редактирование, удаление задач).
- Назначение задач пользователям.

Отслеживание статуса выполнения задач.
Обмен сообщениями между членами команды.

**Требования**:

Определите основные функции системы и опишите каждую из них в формате: "Как пользователь, я хочу [функция], чтобы [цель]".
Определите минимальные и максимальные требования к системе.
Опишите сценарии использования, включающие типичные действия пользователя (например, как создаётся и назначается задача).

**2. Разработка технического задания для информационной системы интернет-магазина**

**Задание**: Разработайте структуру и основные разделы технического задания для информационной системы интернет-магазина. Техническое задание должно содержать:

Описание целей проекта и границ системы.

Функциональные требования.

Нефункциональные требования (производительность, безопасность, интерфейс).

Архитектурные ограничения.

**Требования**:

Опишите цели проекта, целевую аудиторию и основные бизнес-процессы.

Сформулируйте функциональные и нефункциональные требования.

Определите, какие интеграции с другими системами будут необходимы (например, с платёжными шлюзами или курьерскими службами).

### 3. Анализ бизнес-процессов и формализация требований для системы управления складом

**Задание**: Изучите бизнес-процессы, связанные с управлением складом, и составьте список функциональных требований для будущей системы управления складом. Бизнес-процессы могут включать:

Приём товара на склад.

Учёт товарных остатков.

Отправка товара.

**Требования**:

Опишите шаги бизнес-процессов в текстовой или графической форме (например, с помощью диаграммы активности).

Разработайте список функциональных требований, которые охватывают все бизнес-процессы.

Опишите условия и правила для выполнения операций (например, контроль остатков, автоматическая инвентаризация).

### 4. Разработка требований к пользовательскому интерфейсу системы бронирования

**Задание**: Сформулируйте требования к пользовательскому интерфейсу для системы бронирования номеров в гостинице. Система должна включать следующие функции:

Поиск доступных номеров.

Просмотр информации о номере (фотографии, цена, удобства).

Бронирование номера.

**Требования**:

Описать структуру и элементы экранов интерфейса, включая основные формы и кнопки.

Сформулируйте требования к отображению данных (например, сортировка номеров по цене или дате).

Опишите, каким образом будет организована навигация пользователя по системе.

### 5. Тестирование на основе требований

**Задание**: Разработайте спецификацию требований для интеграции информационной системы интернет-магазина с платёжной системой и курьерской службой. Спецификация должна включать:

Описание протоколов и форматов обмена данными.

Входные и выходные данные для интеграции с платёжной системой.

Способы обмена данными с курьерской службой.

**Требования**:

Составить описание функциональных и нефункциональных требований к простой системе (например, система регистрации пользователей).

На основе требований составить 3-4 тест-кейсов, включая позитивные и негативные сценарии.

Описать шаги выполнения тестов и ожидаемые результаты. **6. Описание нефункциональных требований для системы дистанционного обучения**

**Задание**: Сформулируйте нефункциональные требования для системы дистанционного обучения, предназначенной для учебного заведения. Нефункциональные требования должны включать:

Производительность системы (количество одновременно активных пользователей).

Безопасность (доступ к личным данным и защита учебных материалов).

Требования к интерфейсу (удобство и доступность).

**Требования**:

Опишите, какие показатели производительности должны быть обеспечены (например, время отклика сервера не более 2 секунд при доступе к учебным материалам).

Сформулируйте требования по защите данных пользователей (например, шифрование данных, авторизация с использованием двухфакторной аутентификации).

Укажите критерии удобства интерфейса (например, наличие версий для мобильных устройств, доступность для людей с ограниченными возможностями).

**7. Диаграмма прецедентов (Use Case Diagram)**

**Задание**: Разработайте диаграмму прецедентов для системы управления библиотекой. Включите основные прецеденты, такие как:

Регистрация читателя.

Поиск книги в каталоге.

Взятие книги на абонемент.

Возврат книги.

Управление каталогом книг библиотекарем.

**Требования**:

Определите всех участников системы (например, Читатель, Библиотекарь).

Включите прецеденты для каждого типа пользователя.

Покажите зависимости и отношения между прецедентами (например, расширение или включение).

**8. Диаграмма классов (Class Diagram)**

**Задание**: Разработайте диаграмму классов для интернет-магазина. Основные классы должны включать:

Пользователь.

Товар.

Заказ.

Корзина.

**Требования**:

Определите атрибуты и методы для каждого класса (например, для класса "Товар" укажите атрибуты: название, цена, количество).

Определите отношения между классами, такие как агрегация, композиция, наследование.

Покажите видимость атрибутов и методов (public, private, protected).

**9. Диаграмма последовательностей (Sequence Diagram)**

**Задание**: Создайте диаграмму последовательностей для процесса оформления заказа в интернет-магазине. Процесс должен включать:

Добавление товаров в корзину.

Оформление заказа.

Подтверждение оплаты.

**Требования**:

Укажите объекты, участвующие в процессе (например, Клиент, Корзина, Система оплаты).

Отобразите сообщения, передаваемые между объектами (например, "добавить товар в корзину", "проверить наличие товара").

Покажите порядок вызовов, используя стрелки и нумерацию.

**10. Диаграмма состояний (State Diagram)**

**Задание**: Разработайте диаграмму состояний для объекта "Заказ" в интернет-магазине. Состояния должны включать:

Создан.

Подтверждён.

Отправлен.

Доставлен.

Отменён.

**Требования**:

Определите все возможные состояния заказа и переходы между ними.

Укажите события, приводящие к изменениям состояния (например, "оплата подтверждена", "заказ отменён пользователем").

Учтите возможные альтернативные сценарии, такие как отмена заказа до или после подтверждения.

**11. Диаграмма деятельности (Activity Diagram)**

**Задание**: Создайте диаграмму деятельности для процесса регистрации нового пользователя на сайте интернет-магазина. Процесс должен включать:

Ввод информации пользователем.

Проверка введённых данных.

Подтверждение регистрации по электронной почте.

**Требования**:

Определите последовательность действий, от начала до завершения процесса.

Укажите ветвления (например, если данные введены неправильно, система запрашивает повторный ввод).

Используйте символы для начала и завершения процесса, а также для решений и действий.

**12. Диаграмма компонентов (Component Diagram)**

**Задание**: Разработайте диаграмму компонентов для системы управления гостиницей. Компоненты должны включать:

Система бронирования.

База данных гостей.

Система оплаты.

Интерфейс администратора.

**Требования**:

Определите основные компоненты системы и связи между ними.

Отобразите интерфейсы, которые компоненты предоставляют друг другу.

Покажите зависимости между компонентами (например, система бронирования зависит от базы данных гостей).

**13. Диаграмма развертывания (Deployment Diagram)**

**Задание**: Создайте диаграмму развертывания для онлайн-приложения для обмена сообщениями. Приложение должно включать:

Сервер приложений.

Базу данных.

Клиентские устройства (например, мобильные телефоны и компьютеры).

**Требования**:

Определите узлы (серверы, клиентские устройства) и покажите их взаимодействие.

Отобразите, какие программные компоненты развёрнуты на каждом узле.

Показать связи между узлами (например, взаимодействие клиента с сервером через Интернет).

**14. Диаграмма объектов (Object Diagram)**

**Задание**: Разработайте диаграмму объектов для сценария работы с интернет-магазином, в котором:

Существует один объект класса "Пользователь" (например, Иван).

Пользователь добавляет два объекта класса "Товар" в объект класса "Корзина".

**Требования**:

Покажите объекты, участвующие в сценарии (например, "Пользователь: Иван", "Товар: Ноутбук").

Отобразите конкретные атрибуты объектов с их значениями (например, название товара и цена).

Показать связи между объектами (например, связь между пользователем и корзиной).

**15. Нагрузочное тестирование веб-приложения**

**Задание**: Провести нагрузочное тестирование небольшого веб-приложения с использованием инструментов для тестирования производительности (например, JMeter или Gatling).

Оценить производительность веб-приложения при различных уровнях нагрузки.

**Требования**:

Настроить нагрузочный тест на веб-приложение, которое будет имитировать реальных пользователей (например, 100, 500 и 1000 пользователей одновременно).

Определить ключевые метрики, такие как время отклика, количество успешных/неуспешных запросов и пиковая нагрузка, при которой приложение перестает корректно работать.

**Критерии оценивания промежуточной аттестации: экзамен**

| Оценка | Критерии оценивания по экзамену |
|---|---|
| Высокий уровень «5» (отлично) | оценку «отлично» заслуживает обучающийся, освоивший знания, умения, компетенции и теоретический материал без пробелов; выполнивший все задания, предусмотренные учебным планом на высоком качественном уровне; практические навыки профессионального применения освоенных знаний сформированы. |
| Средний уровень | оценку «хорошо» заслуживает обучающийся, практически полностью |

| | |
|---|---|
| «4» (хорошо) | освоивший знания, умения, компетенции и теоретический материал, учебные задания не оценены максимальным числом баллов, в основном сформировал практические навыки. |
| Пороговый уровень «3» (удовлетворительно) | оценку «удовлетворительно» заслуживает обучающийся, частично с пробелами освоивший знания, умения, компетенции и теоретический материал, многие учебные задания либо не выполнил, либо они оценены числом баллов близким к минимальному, некоторые практические навыки не сформированы. |
| Минимальный уровень «2» (неудовлетворительно) | оценку «неудовлетворительно» заслуживает обучающийся, не освоивший знания, умения, компетенции и теоретический материал, учебные задания не выполнил, практические навыки не сформированы. |

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.

## 5. Методические указания для обучающихся по освоению дисциплины (модуля)

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

При проведении учебных занятий по дисциплине обеспечивается развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплины в форме курса, составленного на основе результатов научных исследований, проводимых институтом, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

**Результат обучения считается сформированным**, если теоретическое

содержание курса освоено полностью; при устных собеседованиях обучающийся исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, обучающийся способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий.

**Результат обучения считается несформированным**, если обучающийся при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям.

**Методические указания для обучающихся по освоению дисциплины на занятиях лекционного типа**

Лекционный курс предполагает систематизированное изложение основных вопросов тематического плана. В ходе лекционных занятий раскрываются базовые вопросы в рамках каждой темы дисциплины. Обозначаются ключевые аспекты тем, а также делаются акценты на наиболее сложные и важные положения изучаемого материала. Материалы лекций являются опорной основой для подготовки обучающихся к практическим занятиям и выполнения заданий самостоятельной работы, а также к мероприятиям текущего контроля успеваемости и промежуточной аттестации по дисциплине.

В ходе лекционных занятий необходимо вести конспектирование учебного материала. Возможно ведение конспекта лекций в виде интеллект-карт.

**Методические указания для обучающихся по освоению дисциплины на занятиях практического типа**

Практические (семинарские) занятия представляют собой детализацию лекционного теоретического материала, проводятся в целях закрепления курса и охватывают все основные разделы. Основной формой проведения семинаров и практических занятий является обсуждение наиболее проблемных и сложных вопросов по отдельным темам, а также решение задач и разбор примеров и ситуаций в аудиторных условиях.

Практические (семинарские) занятия обучающихся обеспечивают:

- проверку и уточнение знаний, полученных на лекциях;
- получение умений и навыков составления докладов и сообщений, обсуждения вопросов по учебному материалу дисциплины;
- подведение итогов занятий по рейтинговой системе, согласно технологической карте дисциплины.

**Методические указания по самостоятельной работе обучающихся**

Самостоятельная работа обеспечивает подготовку обучающегося к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающегося на занятиях и в качестве выполненных практических заданий и других форм текущего контроля.

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы.

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.

## 6. Материально-техническое и учебно-методическое обеспечение по дисциплине (модулю)

**Основная литература:**

1. Новожилов, О. П. Архитектура ЭВМ и вычислительных систем : учебник для вузов / О. П. Новожилов. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 505 с. — (Высшее образование). — ISBN 978-5-534-20365-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/568920

2. Ткаченко, С. Н., Методы и средства проектирования информационных систем и технологий + еПриложение : учебник / С. Н. Ткаченко, Б. Р. Мищук. — Москва : КноРус, 2022. — 222 с. — ISBN 978-5-406-09467-9. — URL: https://book.ru/book/943815— Текст : электронный.

3. Зыков, С. В. Архитектура информационных систем. Основы проектирования : учебник для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2025. — 260 с. — (Высшее образование). — ISBN 978-5-534-21538-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/575500

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

### Электронно-библиотечные системы (ЭБС) и базы данных

Доступ к ЭБС предоставляется из любой точки, в которой имеется доступ к сети «Интернет», как на территории Института, так и вне ее (удаленный доступ).

1. Образовательная платформа «ЮРАЙТ» - URL: https://urait.ru/
2. Электронно-библиотечная система «BOOK.ru» - URL: https://www.book.ru
3. Научная электронная библиотека eLIBRARY.RU - URL: https://elibrary.ru (крупнейшая российская база научных публикаций, доступ к рефератам и полным текстам статей).
4. КиберЛенинка - URL: https://cyberleninka.ru (научная электронная библиотека открытого доступа).

**Информационные справочные системы**

1. Справочная правовая система «Консультант Плюс» - доступ по локальной сети с компьютеров библиотеки и компьютерных классов (актуальная база законодательства РФ, в т.ч. в сфере образования и социальной защиты).
2. Справочная правовая система «Гарант» - доступ по локальной сети с компьютеров библиотеки и компьютерных классов (актуальная база законодательства РФ, включая сферы образования и социальной защиты).

**Профессиональные сообщества и IT-порталы**

1. Habr - URL: https://habr.com/ (крупнейшее русскоязычное сообщество IT-специалистов, статьи, новости, обсуждения).

2. Stack Overflow - URL: https://stackoverflow.com/ (международный ресурс для программистов, вопросы и ответы).

3. GitHub - URL: https://github.com/ (платформа для хостинга кода, совместной разработки и открытых проектов).

4. CodeProject - URL: https://www.codeproject.com/ (статьи, примеры кода, обсуждения для разработчиков).

5. Microsoft Learn - URL: https://learn.microsoft.com/ (бесплатные учебные материалы и документация по продуктам Microsoft, .NET, Azure и др.).

6. MDN Web Docs - URL: https://developer.mozilla.org/ (ресурс для **веб-разработчиков с документацией по HTML, CSS, JavaScript и API).**

**Научные базы данных зарубежных издательств (открытый доступ)**

7. IEEE Xplore - URL: https://ieeexplore.ieee.org/ (доступ к книгам, статьям и материалам конференций по компьютерным наукам, электротехнике и информационным технологиям).

8. ACM Digital Library - URL: https://dl.acm.org/ (библиотека статей и материалов конференций Ассоциации вычислительной техники).

9. SpringerLink - URL: https://link.springer.com/ (книги и журналы издательства Springer, включая серии по информатике).

10. Wiley Online Library - URL: https://onlinelibrary.wiley.com/ (коллекция журналов и книг по направлению «Computer Science & Information Technology»).

**Образовательные платформы и онлайн-курсы**

11. Национальная платформа «Открытое образование» - URL: https://openedu.ru/ (курсы ведущих российских вузов).

12. Stepik - URL: https://stepik.org/ (российская образовательная платформа с курсами по программированию и информатике).

13. Intuit - URL: https://intuit.ru/ (национальный открытый университет, курсы по информационным технологиям).

**Официальные органы государственной власти и управления**

1. Министерство науки и высшего образования Российской Федерации - URL: https://www.minobrnauki.gov.ru/.

2. Министерство просвещения Российской Федерации - URL: https://edu.gov.ru/.

3. Министерство труда и социальной защиты Российской Федерации - URL: https://mintrud.gov.ru/.

4. Федеральная служба по надзору в сфере образования и науки (Рособрнадзор) - URL: http://obrnadzor.gov.ru/.

5. Государственная система правовой информации «Законодательство России» - URL: http://pravo.gov.ru/.

**Образовательные и справочные порталы**

Федеральный портал «Российское образование» - URL: http://www.edu.ru/.

Российское общество «Знание» - URL: https://znanierussia.ru/.

Справочно-информационный портал «Грамота.ру» - URL: http://gramota.ru/ (русский язык и культура речи).

Образовательный портал «Учеба» - URL: http://www.ucheba.com/.

Словари и энциклопедии на Академике - URL: https://dic.academic.ru/.

Проект Государственного института русского языка им. А.С. Пушкина «Образование на русском» - URL: https://pushkininstitute.ru/.

**Комплект лицензионного и свободно распространяемого программного обеспечения:**

1. LibreOffice - офисный пакет
2. PDFedit – программа для работы с pdf
3. Yandex Browser – браузер

4. Менеджер архивов
5. Libre Base – программа для работы с БД
6. Inkscape – ПО для компьютерной графики
7. DIA – ПО для блока схем и диаграмм
8. 1С: Предприятие – Учебная версия
9. GiMP - Программа обработки изображений

**Перечень материально-технического обеспечения включает:**

учебные аудитории для проведения занятий лекционного типа, занятий практического (семинарского) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы оснащенные компьютерной техникой с подключением к сети "Интернет" и доступом в электронную информационно-образовательную среду Института.

| Наименование помещения.<br>Перечень основного оборудования | Адрес |
|---|---|
| Учебная аудитория № 217<br>Учебная аудитория для проведения занятий лекционного и практического типа, выполнения курсовых работ, групповых и индивидуальных консультаций, текущего контроля, промежуточной и итоговой аттестации).<br>Оборудование:<br>рабочее место преподавателя (1); рабочие места обучающихся (36); ноутбук с лицензионным ПО (LibreOffice) и возможностью выхода в интернет (1); мультимедийное оборудование (1); доска учебная (1); книжный шкаф (1); сплит-система(1);<br>учебно-наглядные пособия;<br>доступ в электронную информационно-образовательную среду Института. | 350002, Краснодарский край, г. Краснодар, Центральный внутригородской округ, ул. им. Леваневского, д. 187/1 |
| Аудитория № 218<br>Помещение для самостоятельной работы обучающихся<br>Оборудование:<br>рабочие места обучающихся (17); персональный компьютер с лицензионным ПО и возможностью выхода в Интернет (17); книжный шкаф (1); сплит-система (1);<br>учебно-наглядные пособия;<br>доступ в электронную информационно-образовательную среду Института. | 350002, Краснодарский край, г. Краснодар, Центральный внутригородской округ, ул. им. Леваневского, д. 187/1 |
| Учебная аудитория № 303 (компьютерный класс)<br>Учебная аудитория для проведения занятий лекционного и практического типа, выполнения курсовых работ, групповых и индивидуальных консультаций, текущего контроля, промежуточной аттестации). | 350002, Краснодарский край, г. Краснодар, Центральный внутригородской округ, ул. им. Леваневского, д. 187/1 |

| | |
|---|---|
| Оборудование:<br>рабочее место преподавателя (1); рабочие места обучающихся (25); персональный компьютер с лицензионным ПО и возможностью выхода в интернет (26); мультимедийное оборудование (1); доска учебная (1); книжный шкаф (1); сплит-система(1);<br>учебно-наглядные пособия;<br>доступ в электронную информационно-образовательную среду Института. | |