МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ
Декан факультета информационных технологий и компьютерной безопасности

_____ Бредихин А.В.

23.06.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Технологии разработки клиент-серверных приложений»

Направление подготовки 09.03.02 Информационные системы и технологии

Профиль Информационные системы и технологии цифровизации

Квалификация выпускника бакалавр

Нормативный период обучения 4 года

Форма обучения очная

Год начала подготовки 2023

Авторы программы: _____ Е.Н. Королев

Директор Воронежского филиала
АО «Неофлекс Консалтинг» _____ А.М. Нагайцев

Заведующий кафедрой
искусственного интеллекта
и цифровых технологий _____ П.Ю. Гусев

Руководитель ОПОП _____ Д.В. Иванов

Воронеж 2026

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

Освоить технологии создания клиент-серверных приложений на Java

1.2. Задачи освоения дисциплины

– изучить технологии написания клиентской части клиент-серверного приложения;
– изучить технологии написания серверной части клиент-серверного приложения;
– освоить разработку двухуровневых клиент-серверных приложений;
– освоить разработку трехуровневых клиент-серверных приложений.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Технологии разработки клиент-серверных приложений» относится к дисциплинам части, формируемой участниками образовательных отношений (дисциплина по выбору) блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Технологии разработки клиент-серверных приложений» направлен на формирование следующих компетенций:

ПК-2 - Способен проектировать архитектуру информационных систем в соответствии с современными требованиями информационного взаимодействия

ПК-4 - Способен проектировать и разрабатывать базы данных

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ПК-2	знать технологии проектирования архитектуры информационных систем с клиент-серверной архитектурой в соответствии с современными требованиями информационного взаимодействия
	уметь выбирать и применять технологии проектирования архитектуры информационных систем с клиент-серверной архитектурой в соответствии с современными требованиями информационного взаимодействия
	владеть инструментальными средствами проектирования архитектуры информационных систем с клиент-серверной архитектурой в соответствии с современными требованиями информационного взаимодействия
ПК-4	знать технологии проектирования и разработки баз данных при разработке клиент-серверных приложений

	Уметь выбирать и применять технологии проектирования и разработки баз данных при разработке клиент-серверных приложений
	владеть инструментальными средствами для проектирования и разработки баз данных при разработке клиент-серверных приложений

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Технологии разработки клиент-серверных приложений» составляет 8 з.е.

Распределение трудоемкости дисциплины по видам занятий

очная форма обучения

Виды учебной работы	Всего часов	Семестры	
		5	6
Аудиторные занятия (всего)	162	72	90
В том числе:			
Лекции	72	36	36
Лабораторные работы (ЛР) в том числе в форме практической подготовки	90 28	36 12	54 16
Самостоятельная работа	90	36	54
Курсовой проект	+		+
Часы на контроль	36		36
Виды промежуточной аттестации - экзамен	+	зачет	экз
Общая трудоемкость: академические часы зач.ед.	 288 8	 108 3	 180 5

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Обзор интегрированных IDE и облачных средств разработки. Основы написания клиентской части на Java	Java, коллекции, Java FX, JDBC, многопоточность	12	12	12	36
2	Сетевое программирование на Java	Пакет java.net	4	4	4	12
3	Технологии создания WEB ориентированной клиентской части приложения	HTML, CSS, JS	8	8	8	24
4	Технологии написания серверных компонент	Рефлексия и аннотации. Servlet, JSP	12	12	12	36

5	Технологии создания 3-х звенных клиент серверных приложений. Основные паттерны. Использование Docker.	Jakarta EE, UML, JPA	12	18	18	48
6	REST архитектура	HTTP запрос, структура. Сервлеты. REST. JSON. AJAX	4	6	6	16
7	SOA архитектура	XML, SOA	8	12	12	32
8	Spring FrameWork	Spring FrameWork. Spring MVC	6	9	9	24
9	Микросервисы	Написание микросервисов с использованием Spring Boot	6	9	9	24
		Итого	**72**	**90**	**90**	**252**

Практическая подготовка при освоении дисциплины (модуля) проводится путем непосредственного выполнения обучающимися отдельных элементов работ, связанных с будущей профессиональной деятельностью, способствующих формированию, закреплению и развитию практических навыков и компетенций по профилю соответствующей образовательной программы на практических занятиях и (или) лабораторных работах:

№ п/п	Перечень выполняемых обучающимися отдельных элементов работ, связанных с будущей профессиональной деятельностью	Формируемые профессиональные компетенции
1	Разработка компонент web приложений с использованием html+css+js	ПК-2, ПК-4
2	Разработка системы авторизации с использованием Servlet.	ПК-2, ПК-4
3	Разработка системы с использованием REST архитектуры.	ПК-2, ПК-4
4	Разработка системы авторизации с использованием WebSocket и SOA архитектуры	ПК-2, ПК-4
5	Разработка компонент системы с использованием Spring MVC	ПК-2, ПК-4
6	Разработка микросервисов с использованием Spring Boot	ПК-2, ПК-4

5.2 Перечень лабораторных работ

1. Лабораторная работы №1. ООП на Java. Работа с коллекциями.
2. Лабораторная работы №2. Разработка многопоточного приложения на Java.
3. Лабораторная работы №3. Работа с базой данных с использованием JDBC.
4. Лабораторная работы №4. Разработка сетевого приложения с использованием пакета java.net.

5. Лабораторная работы №5. Разработка компонент web приложений с использованием html.

6. Лабораторная работы №6. Разработка компонент web приложений с использованием html+css.

7. Лабораторная работы №7. Разработка компонент web приложений с использованием html+css+js.

8. Лабораторная работы №8. Использование рефлексии и аннотаций при написании серверных компонент.

9. Лабораторная работы №9. Разработка системы авторизации с использованием Servlet.

10. Лабораторная работы №10. Разработка системы авторизации с использованием JSP.

11. Лабораторная работы №11. Разработка компонент корзины покупателя с использованием EJB.

12. Лабораторная работы №12. Разработка компонент корзины покупателя с использованием EJB+JPA.

13. Лабораторная работы №13. Разработка системы с использованием REST архитектуры.

14. Лабораторная работы №14. Разработка системы авторизации с использованием WebSocket и SOA архитектуры.

15. Лабораторная работы №15. Разработка компонент системы с использованием Spring MVC.

16. Лабораторная работы №16. Разработка микросервисов с использованием Spring Boot.

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 6 семестре для очной формы обучения.

Примерная тематика курсового проекта: «Разработка клиент-серверного приложения с использованием Java технологий»

Задачи, решаемые при выполнении курсового проекта:

• выбрать технологии Java EE (Jakarta EE) для решения поставленной задачи;

• использовать REST архитектуру, использовать Spring, использовать Docker.

• спланировать перечень работ по проекту;

• разработать соответствующие компоненты клиент-серверного приложения.

Курсовой проект включат в себя графическую часть и расчетно-пояснительную записку.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

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

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компе-тенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ПК-2	знать технологии проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Знание UML, IDEF1x	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь выбирать и применять технологии проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Знание основных диаграмм UML, REST, SOA, JPA, Spring	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть инструментальными средствами проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Знание PluntUML	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ПК-4	знать технологии проектирования и разработки баз данных при разработке клиент-серверных приложений	Знание Jakarta EE, JS, HTML, JDBC.	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь выбирать и применять технологии проектирования и разработки баз данных при разработке клиент-серверных приложений	Знание FX, Spring, JPA	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть инструментальными средствами для проектирования и разработки баз данных при разработке клиент-серверных приложений	Владение NetBeans	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 5 и 6 семестре для очной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компе-тенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ПК-2	знать технологии проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70- 80%	В тесте менее 70% правильных ответов
	уметь выбирать и применять технологии проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть инструментальными средствами проектирования архитектуры информационных систем в соответствии с современными требованиями информационного взаимодействия	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ПК-4	знать технологии проектирования и разработки баз данных при разработке клиент-серверных приложений	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70- 80%	В тесте менее 70% правильных ответов
	Уметь выбирать и применять технологии проектирования и разработки баз данных при разработке клиент-серверных приложений	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

	владеть инструментальными средствами для проектирования и разработки баз данных при разработке клиент-серверных приложений	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Что такое сервлет?

1. **Сервлет является интерфейсом Java, реализация которого расширяет функциональные возможности сервера**.
2. Сервлет является интерфейсом Java, реализация которого расширяет функциональные возможности клиента.

2. Что такое контейнер сервлетов?

1. **Контейнер сервлетов — программа, представляющая собой сервер, который занимается системной поддержкой сервлетов и обеспечивает их жизненный цикл в соответствии с правилами, определёнными в спецификациях.**
2. Контейнер сервлетов — программа, представляющая собой клиентскую часть, которая занимается системной поддержкой сервлетов и обеспечивает их жизненный цикл в соответствии с правилами, определёнными в спецификациях.

3. Что такое дескриптор развертывания?

1. **Дескриптор развертывания — это конфигурационный файл артефакта, который будет развернут в контейнере сервлетов.**
2. Дескриптор развертывания — это конфигурационный файл апплета, который будет развернут в контейнере апплетов.

4. Что представляет собой объект ServletConfig?

1. **Интерфейс javax.servlet.ServletConfig используется для передачи конфигурационной информации сервлету.**
2. Интерфейс javax.servlet.ServletConfig используется для управления жизненным циклом сервлета.

5. Что представляет собой объект ServletContext?

1. **Интерфейс javax.servlet.ServletContext предоставляет доступ к параметрам веб приложения сервлету.**

2. Интерфейс javax.servlet.ServletContext предоставляет доступ к параметрам сервлета.

6. В чем отличия ServletContext и ServletConfig?

1. **ServletConfig является уникальным объектом для каждого сервлета, в то время как ServletContext уникальный для всего приложения.**
2. ServletContext является уникальным объектом для каждого сервлета, в то время как ServletConfig уникальный для всего приложения.

7. Что такое Request Dispatcher?

1. **Интерфейс RequestDispatcher используется для передачи запроса другому ресурсу**
2. Интерфейс RequestDispatcher используется для завершения работы сервлета.

8. Как получить адрес сервлета на сервере?

1. **Для получения актуального пути сервлета на сервере можно использовать эту конструкцию: getServletContext().getRealPath(request.getServletPath())**
2. Для получения актуального пути сервлета на сервере можно использовать эту конструкцию: getServletContext().getAddress(request.getAddress())

9. В каком случае вы будете переопределять метод service()?

1. **Метод service() переопределяется, когда мы хотим, чтобы сервлет обрабатывал как GET так и POST запросы в одном методе.**
2. Метод service() переопределяется, когда мы хотим, чтобы сервлет обрабатывал как только GET запросы в одном методе.
3. Метод service() переопределяется, когда мы хотим, чтобы сервлет обрабатывал как только POST запросы в одном методе.

10. Что вы знаете и в чем отличия методов forward() и sendRedirect()?

1. **RequestDispatcher forward() используется для проброски того же самого запроса к другому ресурсу, в то время как ServletResponse sendRedirect() это двухшаговый** метод.
2. ServletResponse sendRedirect() используется для проброски того же самого запроса к другому ресурсу, в то время как RequestDispatcher forward() это двухшаговый метод.

7.2.2 Примерный перечень заданий для решения стандартных задач

1. Что такое JPA?

1. JPA (Java Persistence API) это спецификация Java EE и Java SE, описывающая систему управления сохранением java объектов в таблицы реляционных баз данных в удобном виде.

2. JPA (Java Persistence API) это спецификация Java EE и Java SE, описывающая систему управления сохранением java объектов в XML файлы в удобном виде.

2. <u>Может ли Entity класс наследоваться от не Entity классов (non-entity classes)?</u>

1. Может.

2. Не может.

3. <u>Может ли Entity класс наследоваться от других Entity классов?</u>

1. **Может.**

2. Не может

4. <u>Может ли не Entity класс наследоваться от Entity класса?</u>

1. Может.

2. Не может

5. <u>Может ли Entity быть абстрактным классом?</u>

1. Может, при этом он сохраняет все свойства Entity, за исключением того, что его нельзя непосредственно инициализировать.

2. Не может.

6. <u>Может ли встраиваемый (Embeddable) класс содержать другой встраиваемый (Embeddable) класс?</u>

1. **Может.**
2. Не может.

7. <u>Может ли встраиваемый (Embeddable) класс содержать связи (relationship) с другими Entity или коллекциями Entity?</u>

1. **Может, но только в случае если такой класс не используется как первичный ключ или ключ map'ы.**
2. Не может.

8. <u>Что такое JAXB.</u>

1. Java Architecture for XML Binding (JAXB) позволяет Java разработчикам ставить в соответствие Java классы и XML представления.

2. Java Architecture for XML Binding (JAXB) позволяет Java разработчикам ставить в соответствие Java классы и таблицы базы данных.

9. <u>Какие HTTP методы поддерживаются в REST?</u>

1. **GET; POST; PUT; DELETE; OPTIONS; HEAD;**
2. GET; POST;

3. PUT; DELETE; OPTIONS; HEAD.

10. <u>Какая разница между GET и POST запросами?</u>

1. GET передает данные серверу используя URL, когда POST передает данные, используя тело HTTP запроса.

2. POST передает данные серверу используя URL, когда GET передает данные, используя тело HTTP запроса.

7.2.3 Примерный перечень заданий для решения прикладных задач

1. <u>Что означает аннотация @ManagedProperty?</u>

1. Аннотация @ManagedProperty позволяет внедрять один управляемый бин в другой.

2. Аннотация @ManagedProperty позволяет ручное управление бином.

2. <u>На что указывает аннотация @ApplicationScoped?</u>

1. **Аннотация @ApplicationScoped указывает на то, что управляемый бин будет доступен на протяжении времени жизни приложения.**

2. Аннотация @ApplicationScoped указывает на то, что управляемый бин будет доступен на протяжении времени жизни сессии.

3. <u>Объясните разницу между required и requiredMessage атрибутов в теге <h:inputText>.</u>

1. Эти два тега объявляют необходимость обязательного заполнения текстового поля, а так же сообщения, которое будет выполнено при нарушении этого правила.

2. Эти два тега объявляют необходимость обязательной отправки формы на обработку.

<u>4. Объясните назначение тега <h:form>.</u>

1. Тег <h:form> генерирует элемент формы HTML, который использует «POST» запрос для подтверждения формы и пересылки данных из формы.

2. Тег <h:form> генерирует элемент формы HTML, который использует «GET» запрос для подтверждения формы и пересылки данных из формы.

<u>5. Какое назначение тега facelets?</u>

1. JSF предоставляет специальный набор тегов, которая дает большую гибкость для управления общими тегами / частями в одном месте для нескольких приложений.

2. Набор тегов для прорисовки окон

6. Что такое конфигурационный файл Hibernate?

1. **Файл конфигурации Hibernate содержит в себе данные о базе данных и необходим для инициализации SessionFactory.**
2. Файл конфигурации JSF содержит в себе данные о базе данных и необходим для инициализации SessionFactory.

7. Что такое Hibernate mapping file?

1. **Файл отображения (mapping file) используется для связи entity бинов и колонок в таблице базы данных.**
2. Файл отображения (mapping file) используется для связи xml файлов и колонок в таблице базы данных.

8. Является ли Hibernate SessionFactory потокобезоспансым?

1. **Так как объект SessionFactory immutable (неизменяемый), то да, он потокобезопасный.**
2. Нет, так как объект SessionFactory mutable (изменяемый)

9. Какие типы коллекций в Hibernate вы знаете?

1. **Bag, Set, List, Array, Map**
2. Vector, List, Array, Map

10. Что такое Managed Bean?

1. **В технологии JSF управляемые бины применяются для сохранения состояния web-страниц.**
2. В технологии JSF управляемые бины применяются для сохранения состояния базы данных.

7.2.4 Примерный перечень вопросов для подготовки к зачету

1. HTML, структура. Теги head, body, b, d, u, hr, font. Описание, синтаксис, атрибуты.
2. HTML, структура. Теги a, ul,ol,input. Описание, синтаксис, атрибуты.
3. HTML, структура. Теги img, button, form, textarea, table. Описание, синтаксис, атрибуты.
4. HTML, структура. Теги head, body, b, d, u, hr, font. Описание, синтаксис, атрибуты.
5. Особенности HTML5. Проверка форм.
6. CSS, назначение. Способы применения.
7. Использование универсальных, теговых и псевдоклассов в CSS/
8. Основы JavaScript. Включения, команды, комментарии, переменные.
9. Операторы, выражения, конструкции JavaScript.
10. Функции, циклы, события JavaScript.
11. Объекты в JavaScript, свойства объектов, методы объектов.

12. Организация массивов в JavaScript. Создание массивов, доступ к переменным. Свойства объекта Array, методы Array. Работа с объектами в JavaScript.
13. JDBC, назначение, основные классы
14. Пакет java.net
15. Создание многопоточных приложений, способы.
16. Синхронизация многопоточных приложений, способы.
17. Рефлексия и назначение
18. Аннотации стандартные и собственные.
19. Сервлеты, назначение и методы жизненного цикла.
20. JSP

7.2.5 Примерный перечень заданий для подготовки к экзамену

1. Типы клиент-серверной архитектуры, технологии реализации.
2. Трехзвенная клиент-серверная архитектура, технологии реализации на Java.
3. Что такое Java EE? Какие модули входят в Java EE? Какая разница между .jar, .war и .ear файлами?
4. Что такое сервлет? Какова структура веб-проекта? Что такое контейнер сервлетов?
5. Что представляет собой объект ServletConfig? Что представляет собой объект ServletContext? В чем отличия ServletContext и ServletConfig?
6. Что такое Request Dispatcher? Как получить адрес сервлета на сервере? Как получить ip адрес клиента на сервере?
7. Каков жизненный цикл сервлета и когда какие методы вызываются? Какие методы необходимо определить при создании сервлетов?
8. Что вы знаете и в чем отличия методов forward() и sendRedirect()?Стоит ли волноваться о “многопоточной безопасности” работая с сервлетами? В чем разница между методами GET и POST?
9. Как применяются Cookies в сервлетах? Написать сервлет, реализующий загрузку файла на сервер.
10. Что такое JPA? В чем её отличие JPA от Hibernate?
11. Что такое Entity? Какие требования JPA к Entity классам вы можете перечислить (не менее шести требований)?
12. Что такое EntityManager и какие основные его функции вы можете перечислить? Какой аннотацией можно управлять кешированием JPA для данного Entity?
13. Что такое веб сервисы? В чем разница между SOA и web service?
14. Что такое SOAP? Что такое REST? В чем разница между REST и SOAP веб сервисами?
15. Расскажите о JAXB.
16. Что такое JSF? Что такое Managed Bean? Каковы три типа тегов для текстовых полей существуют в JSF?

17. Какие фазы жизненного цикла в JSF вы знаете?
18. Каковы различные типы событий в JSF?
19. Какие теги для валидации существуют в JSF?
20. Объясните архитектуру JSF.
21. Что такое jsp и зачем он нужен? Расскажите об этапах (фазах) жизненного цикла jsp.
22. Расскажите о методах жизненного цикла jsp. Какие методы жизненного цикла JSP могут быть переопределены?
23. Какие неявные, внутренние объекты и методы есть на jsp странице?
24. Что вы знаете о PageContext и какие преимущества его использования?
25. Что такое JSTL (Jsp Standard tag library)?
26. Что вы знаете о написании пользовательских jsp тегов?
27. Как можно обработать ошибки jsp страниц?
28. Что такое Hibernate Framework? Какие важные преимущества дает использование Hibernate Framework?
29. Каковы преимущества Hibernate над JDBC? Назовите некоторые важные интерфейсы Hibernate.
30. Enterprise Java Beans - EJB
31. Session bean, stateless
32. Session bean, stateful
33. Session bean, singleton
34. Основные аннотации EJB3
35. JPA
36. AJAX
37. REST архитектура
38. Spring FrameWork
39. Spring MVC
40. Микросервисы
41. Docker

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Экзамен проводится по билетам, каждый из которых содержит 2 *вопроса.*

1. Оценка «Неудовлетворительно» ставится в случае, если студент отвечает менее 50% необходимой информации.

2. Оценка «Удовлетворительно» ставится в случае, если студент отвечает более 50%, но менее 70% необходимой информации.

3. Оценка «Хорошо» ставится в случае, если студент отвечает более 70%, но менее 90% необходимой информации.

4. Оценка «Отлично» ставится, если студент отвечает более 90% необходимой информации.

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Java, коллекции, Java FX, JDBC, многопоточность	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
2	Пакет java.net	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
3	HTML, CSS, JS	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
4	Рефлексия и аннотации. Servlet, JSP	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
5	Jakarta EE, UML, JPA	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
6	HTTP запрос, структура. Сервлеты. REST. JSON. AJAX	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
7	XML, SOAP	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
8	Spring FrameWork. Spring MVC	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту
9	Написание микросервисов с использованием Spring Boot	ПК-2, ПК-4	Тест, защита лабораторных работ, требования к курсовому проекту

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется

проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Королев, Евгений Николаевич. Технологии сетевого программирования : учебное пособие / Е. Н. Королев ; Воронеж. гос. техн. ун-т .— Воронеж : Издательство ВГТУ, 2015 .— 115 с. ; 21 см .— Библиогр.: с. 114 .— 60.00 р., 250.
2. Королев, Е.Н. Программирование под Internet на языке Java : учеб. пособие / Е.Н.Королев .— Воронеж : ВГТУ, 2003 .— 209 с. — 57.00.
3. Королев, Е.Н. Проектирование и разработка приложений на языке Java : Учеб. пособие .— Воронеж : ГОУВПО "Воронежский государственный технический университет", 2008 .— 137 с. — 50-00.
4. Мухамедзянов Р.Р. JAVA. Серверные приложения [Электронный ресурс]/ Мухамедзянов Р.Р.— Электрон. текстовые данные.— М.: СОЛОН-ПРЕСС, 2010.— 336 с.— Режим доступа: http://www.iprbookshop.ru/65089.html.— ЭБС «IPRbooks»
5. Дубаков А.А. Сетевое программирование [Электронный ресурс]: учебное пособие/ Дубаков А.А.— Электрон. текстовые данные.— СПб.: Университет ИТМО, 2013.— 249 с.— Режим доступа: http://www.iprbookshop.ru/68118.html.— ЭБС «IPRbooks»

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Программное обеспечение
MySql
NetBeans 8.1
jdk 11

Ресурс информационно-телекоммуникационной сети «Интернет»
http://www.edu.ru/
Образовательный портал ВГТУ

Информационная справочная система
http://window.edu.ru
https://wiki.cchgeu.ru/
https://habr.com/ru/
http://javastudy.ru/interview/list-of-questions-javaee-interview/
https://netbeans.org/kb/docs/javaee/javaee-entapp-ejb_ru.html
http://www.knigafund.ru/ (ЭБС Книгафонд)
http://www.book.ru/ (ЭБС BOOK.ru)
http://ibooks.ru/ (ЭБС Ibooks (Айбукс))
http://e.laibrary.ru
http://habr.com

http://fstec.ru
http://wikipedia.org
http://intuit.ru

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Специализированная лекционная аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой
Дисплейный класс, оснащенный компьютерными программами для проведения лабораторного практикума

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Технологии разработки клиент-серверных приложений» читаются лекции, проводятся лабораторные работы.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомится с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоения учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;

	- выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Перечень вносимых изменений	Дата внесения изменений	Подпись заведующего кафедрой, ответственной за реализацию ОПОП
1	Актуализированы разделы в соответствии с изменениями в учебном плане	22.04.2025	
2	Актуализированы разделы 5, 7 и 8. Актуализация выполнена с привлечение специалистов из АО Неофлекс Консалтинг, в соответствии с соглашением № 2705/2026 от 27 мая 2026 г.	23.06.2026	