

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агафонов Александр Викторович

Должность: директор филиала

Дата подписания: 19.06.2025 15:33:27

Уникальный программный ключ:

23E608A9D511E5187040B00000000000

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
ЧЕБОКСАРСКИЙ ИНСТИТУТ ФИЛИАЛ) МОСКОВСКОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА

Кафедра Информационных технологий
и систем управления



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Интернет-программирование»

(наименование дисциплины)

Направление подготовки	09.03.01 «Информатика и вычислительная техника» (код и наименование направления подготовки)
Направленность (профиль) подготовки	«Программное обеспечение вычислительной техники и автоматизированных систем» (наименование профиля подготовки)
Квалификация выпускника	бакалавр
Форма обучения	очная, заочная
Год начала обучения	2025

Чебоксары, 2025

чая программа дисциплины разработана в соответствии с:

- Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 – Информатика и вычислительная техника, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 929 от 19 сентября 2017 г. зарегистрированный в Минюсте 10 октября 2017 года, рег. номер 48489 (далее – ФГОС ВО);

- Приказ Министерства образования и науки РФ от 14 октября 2015 г. № 1147 «Об утверждении Порядка приема на обучение по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры»;

- Учебным планом (очной, заочной форм обучения) по направлению подготовки 09.03.01 «Информатика и вычислительная техника».

Рабочая программа дисциплины включает в себя оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине (п.8 Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины)

Автор Пикина Наталия Евгеньевна, кандидат педагогических наук,
доцент кафедры информационных технологий и систем управления

(указать ФИО, ученую степень, ученое звание или должность)

Программа одобрена на заседании кафедры Информационных технологий и систем управления (протокол № 8 от 12.04.2025г).

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы (Цели освоения дисциплины)

1.1. Целями освоения дисциплины «Интернет-программирование» являются:

получение студентами теоретических знаний и необходимых практических навыков по коммуникационным технологиям, используемым в глобальной сети Интернет. Отличительной особенностью курса является посвящение обучаемых в технические детали работы Интернет, подробного изучения работы транспортных и прикладных протоколов TCP/IP. Изучение низкоуровневого программирования клиентских и серверных приложений с использованием сокетов. Front и back end разработка Web-приложений.

1.2. Области профессиональной деятельности и (или) сферы профессиональной деятельности, в которых выпускники, освоившие программу, могут осуществлять профессиональную деятельность:

Об Связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации средств вычислительной техники и информационных систем, управления их жизненным циклом).

1.3. К основным задачам изучения дисциплины относится подготовка обучающихся к выполнению трудовых функций в соответствии с профессиональными стандартами:

Код и наименование профессионального стандарта	Обобщенные трудовые функции			Трудовые функции		
	код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
06.001 Программист Профессиональный стандарт «Программист», утвержденный приказом Министерством труда и социальной защиты Российской Федерации от 18.11.2013 № 679н (зарегистрирован в Министерстве труда и социальной защиты Российской Федерации от 18 декабря 2013 г. №30635)	D	Разработка требований и проектирование программного обеспечения	6	Анализ требований к программному обеспечению	D/01.6	6
			6	Разработка технических спецификаций на программные компоненты и их взаимодействие	D/02.6	
			6	Проектирование программного обеспечения	D/03.6	
06.028 Системный программист	A	Разработка компонентов системных	6	Разработка драйверов устройств	A/01.6	6

Код и наименование профессионального стандарта	Обобщенные трудовые функции			Трудовые функции		
	код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
Профессиональный стандарт «Системный программист», утв. Приказом Министерством труда и социальной защиты РФ от 29 сентября 2020 года N 678н		программных продуктов	6	Разработка компиляторов, загрузчиков, сборщиков	A/02.6	6
				Разработка системных утилит	A/03.6	6
				Создание инструментальных средств программирования	A/04.6	6
06.015 Специалист по информационным системам Профессиональный стандарт «Специалист по информационным системам», утв. приказом Министерством труда и социальной защиты РФ 18 ноября 2014 г. №896н	С	Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	6	Разработка модели бизнес-процессов заказчика	C/08.6	6
				Выявление требований к ИС	C/11.6	6
				Анализ требований	C/12.6	6
				Разработка архитектуры ИС	C/14.6	6
				Проектирование и дизайн ИС	C/16.6	6
				Разработка баз данных ИС	C/17.6	6
				Организационное и технологическое обеспечение кодирования на языках программирования	C/18.6	6
				Организационное и технологическое обеспечение модульного тестирования ИС (верификации)	C/19.6	6
				Создание пользовательской документации к ИС	C/22.6	6
Организация репозитория хранения данных о создании (модификации) и вводе ИС в эксплуатацию	C/40.6	6				

1.4. Компетенции обучающегося, формируемые в результате освоения дисциплины

Наименование категории (группы) компетенций	Код и наименование компетенций	Код и наименование индикатора достижения компетенции	Перечень планируемых результатов обучения
<p>Разработка, отладка, модификация и поддержка системного программного обеспечения</p>	<p>ПК-3. Способен разрабатывать компоненты системных программных продуктов</p>	<p>ПК-3.1 Разрабатывает драйверы устройств</p>	<p><i>на уровне знаний:</i> знать базовые понятия компьютерных сетей, сетевых и веб-сервисов; <i>на уровне умений:</i> уметь программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; <i>на уровне навыков:</i> владеть базовыми навыками веб-программирования.</p>
		<p>ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики</p>	<p><i>на уровне знаний:</i> знать принципы реализации веб-приложений; <i>на уровне умений:</i> уметь проектировать и реализовывать веб-приложения; <i>на уровне навыков:</i> владеть навыками веб-программирования, включая работу с HTML, CSS, JavaScript и основными фреймворками.</p>
		<p>ПК-3.3 Разрабатывает системные утилиты</p>	<p><i>на уровне знаний:</i> знать понятия о стандартах в Интернет сетях и Web-проектах. <i>на уровне умений:</i> уметь разворачивать простейшие серверные программные сборки. <i>на уровне навыков:</i> владеть принципами разработки и поддержки</p>

			веб-сайтов и веб-приложений.
		ПК-3.4. Создает инструментальные средства программирования	<p><i>на уровне знаний:</i> знать методы и средства обеспечения информационной безопасности объектов профессиональной деятельности.</p> <p><i>на уровне умений:</i> базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.</p> <p><i>на уровне навыков:</i> реализовывать простые проекты, устранять базовые ошибки и оптимизировать пользовательский интерфейс,</p>

2. Место дисциплины в структуре ОПОП

Дисциплина Б1.Д(М).В.ДВ.3.2 «Интернет-программирование» реализуется в рамках вариативной части Блока 1 «Элективные дисциплины (модули)» программы бакалавриата.

Дисциплина «Интернет-программирование» преподается обучающимся по очной форме обучения – в 5-м семестре, по заочной форме – в 6-м семестре.

Дисциплина «Интернет-программирование» является промежуточным этапом формирования компетенций ПК-3 процессе освоения ОПОП.

Дисциплина «Интернет-программирование» основывается на знаниях, умениях и навыках, приобретенных при изучении дисциплин Перспективы развития информатики и вычислительной техники, Введение в информатику, Учебная практика: ознакомительная практика, Объектно-ориентированное программирование, Программирование для мобильных устройств, Структуры и алгоритмы обработки данных, Криптографические методы защиты информации, Защита информации, Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы)), Учебная практика: технологическая (проектно-технологическая) практика, Микропроцессорные устройства систем управления и является предшествующей для изучения дисциплин Функциональное и логическое программирование, Системное программирование, Государственная итоговая

аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

Формой промежуточной аттестации знаний обучаемых по очной форме обучения является зачет в 5-м семестре, по заочной форме – в 6-м семестре.

3. Объем дисциплины

очная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 5 в часах
Общая трудоёмкость дисциплины	3 з.е. - 108 ак.час	108 ак.час
Контактная работа - Аудиторные занятия	32	32
<i>Лекции</i>	16	16
<i>Лабораторные занятия</i>	16	16
<i>Семинары, практические занятия</i>	-	-
<i>Консультация</i>	-	-
Самостоятельная работа	76	76
Курсовая работа (курсовой проект)	-	-
Вид промежуточной аттестации	Зачет	Зачет

заочная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 6 в часах
Общая трудоёмкость дисциплины	3 з.е. - 108 ак.час	108 ак.час
Контактная работа - Аудиторные занятия	10	10
<i>Лекции</i>	4	4
<i>Лабораторные занятия</i>	6	6
<i>Семинары, практические занятия</i>	-	-
<i>Консультация</i>	-	-
Самостоятельная работа	94	94
Курсовая работа (курсовой проект)	-	-
Вид промежуточной аттестации	Зачет – 4 часа	Зачет – 4 часа

4. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

4.1 Учебно-тематический план

Очная форма обучения

Тема (раздел)	Количество часов				Код индикатора достижений компетенции
	контактная работа			самостоятельная работа	
	лекции	лабораторные занятия	семинары и практические занятия		
Тема 1. Основы организации Интернет сетей. Понятия	2	2	-	14	ПК-3.1., ПК-3.2.,

многоуровневой организации сети. Понятия интерфейса, протокола, адресации.					ПК-3.3., ПК-3.4.
Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	2	2	-	14	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.	4	4	-	16	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствам браузера. JavaScript. Формы	4	4	-	16	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	4	4	-	16	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Консультации		-		-	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Контроль (зачет)			-		ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
ИТОГО		32		76	

Заочная форма обучения

Тема (раздел)	Количество часов				Код индикатора достижений компетенции
	контактная работа			самостоятельная работа	
	лекции	лабораторные занятия	семинары и практические занятия		
Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.	2	-	-	18	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.

Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	-	2	-	18	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.	2	-	-	18	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствам браузера. JavaScript. Формы	-	2	-	20	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	-	2	-	20	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Консультации	-			-	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Контроль (зачет)	4				ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
ИТОГО		10		94	

4.2. Содержание дисциплины

Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.

Введение в Интернет-программирование: как сети влияют на разработку приложений.

Многоуровневая организация сети: модели OSI и TCP/IP и их применение в программировании.

Понятие интерфейса: взаимодействие между компонентами сети и приложениями.

Протоколы: определение и роль в обмене данными между клиентами и серверами.

Адресация: типы адресов (IP, MAC) и их значение для разработки сетевых приложений.

Примеры применения протоколов и адресации в реальных приложениях.
Задачи и вызовы, связанные с сетевым программированием.

Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений

Основы сокетов: что такое сокет и как он используется в сетевом программировании.

Создание клиентских приложений: реализация клиент-серверной архитектуры с использованием сокетов.

Создание серверных приложений: обработка запросов и управление соединениями.

Работа с потоками и асинхронное программирование в сетевых приложениях.

Обработка ошибок и исключений в сетевых приложениях.

Примеры создания простых клиентских и серверных приложений на разных языках программирования.

Тестирование и отладка сетевых приложений: инструменты и методы.

Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.

Протоколы верхнего уровня: их значение и применение в Интернет программировании.

Протокол FTP: создание клиента и сервера для передачи файлов, обработка команд.

Протоколы SMTP и POP3: работа с электронной почтой и реализация почтовых клиентов.

Протокол HTTP: работа с запросами и ответами, методы HTTP (GET, POST и др.).

Архитектура Web-приложений: взаимодействие клиентской и серверной части, MVC-паттерн.

Обработка HTTP-запросов: создание RESTful API и работа с JSON.

Примеры успешного использования протоколов в реальных веб-приложениях.

Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы

Основы HTML и CSS: создание структуры и стилей для веб-страниц.

Архитектура браузера: как браузеры обрабатывают HTML, CSS и JavaScript.

Использование шаблонизаторов для генерации динамического HTML-кода на стороне сервера.

Построение динамических страниц: взаимодействие с сервером и генерация контента.

AJAX и Fetch API: организация асинхронных запросов к серверу с использованием JavaScript.

Основы JavaScript: работа с DOM, обработка событий и создание интерактивных элементов.

Создание и обработка форм: валидация, отправка данных на сервер и обработка ответов.

Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.

Основы хранения данных в веб-приложениях: подходы и технологии.

Управление данными сессий: хранение и использование сессионной информации в приложениях.

Взаимодействие с базами данных: работа с реляционными и NoSQL базами данных.

Форматы данных: использование JSON и XML для обмена данными между клиентом и сервером.

Сравнение JSON и XML: преимущества и недостатки в контексте веб-программирования.

Использование SVG для векторной графики: создание графических элементов на веб-страницах.

Примеры работы с данными: сериализация и десериализация объектов, хранение и извлечение данных из баз данных.

5. Учебно-методическое обеспечение самостоятельной работы студентов

Самостоятельная работа проводится с целью: систематизации и закрепления полученных теоретических знаний и практических умений обучающихся; углубления и расширения теоретических знаний студентов; формирования умений использовать нормативную, правовую, справочную документацию, учебную и специальную литературу; развития познавательных способностей и активности обучающихся: творческой инициативы, самостоятельности, ответственности, организованности; формирование самостоятельности мышления, способностей к саморазвитию, совершенствованию и самоорганизации; формирования профессиональных компетенций; развитию исследовательских умений студентов.

Формы и виды самостоятельной работы студентов: чтение основной и дополнительной литературы – самостоятельное изучение материала по рекомендуемым литературным источникам; работа с библиотечным каталогом, самостоятельный подбор необходимой литературы; работа со словарем, справочником; поиск необходимой информации в сети Интернет; конспектирование источников; реферирование источников; составление аннотаций к прочитанным литературным источникам; составление рецензий и отзывов на прочитанный материал; составление обзора публикаций по теме; составление и разработка терминологического словаря; составление хронологической таблицы; составление библиографии (библиографической картотеки); подготовка к различным формам текущей и промежуточной аттестации (к тестированию, контрольной работе, зачету); выполнение домашних контрольных работ; самостоятельное выполнение практических заданий репродуктивного типа (ответы на вопросы, задачи, тесты; выполнение творческих заданий).

Технология организации самостоятельной работы обучающихся включает использование информационных и материально-технических ресурсов образовательного учреждения: библиотеку с читальным залом, компьютерные классы с возможностью работы в Интернет; аудитории (классы) для консультационной деятельности.

Перед выполнением обучающимися внеаудиторной самостоятельной работы преподаватель проводит консультирование по выполнению задания, который включает цель задания, его содержания, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. Во время выполнения обучающимися внеаудиторной самостоятельной работы и при необходимости преподаватель может проводить индивидуальные и групповые консультации.

Контроль самостоятельной работы студентов предусматривает: соотнесение содержания контроля с целями обучения; объективность контроля; валидность контроля (соответствие предъявляемых заданий тому, что предполагается проверить); дифференциацию контрольно-измерительных материалов.

Формы контроля самостоятельной работы: просмотр и проверка выполнения самостоятельной работы преподавателем; организация самопроверки, взаимопроверки выполненного задания в группе; обсуждение результатов выполненной работы на занятии; проведение письменного опроса; проведение устного опроса; организация и проведение индивидуального собеседования; организация и проведение собеседования с группой.

Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.	<ol style="list-style-type: none"> 1. Определение Интернет сети и ее архитектура. 2. Многоуровневая организация сети. Сравнить модели OSI и TCP/IP. 3. Раскрыть понятия интерфейса, протокола и адресации в контексте сетевых технологий. 4. Роль сетевых протоколов в процессе передачи данных. 5. Характеристика методов адресации: IPv4, IPv6 и MAC-адресация. 6. Описать взаимодействие между уровнями в модели OSI. Какие задачи решает каждый уровень? 7. Привести примеры использования сетевых протоколов в реальных сценариях (например, работа браузера, электронная почта). 	Изучение архитектуры клиент-серверных приложений. Подготовка сравнительной таблицы моделей OSI и TCP/IP. Анализ документации популярных сетевых протоколов.
Тема 2. Низкоуровневое программирование	<ol style="list-style-type: none"> 1. Сокеты и их роль в сетевом программировании. 2. Разница между потоковыми и датаграммными сокетами. 	Разработка простого клиента и сервера (например, на Python или

в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	3. Реализация клиент-серверной модели при помощи сокетов. 4. Подходы для обработки подключений на стороне сервера. 5. Обработка ошибок в сетевых приложениях. 6. Примеры программ на языке программирования (например, Python или C#), реализующих клиент и сервер. 7. Инструменты для тестирования и отладки сетевых приложений.	Node.js). Анализ работы асинхронных библиотек (asyncio, aiohttp, Node.js).
Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.	1. Задачи решают протоколы верхнего уровня. 2. Описать структуру и команды протокола FTP. Привести примеры использования. 3. Протоколы SMTP и POP3 при передаче электронной почты. 4. Принципы работы HTTP-протокола: структура запросов и ответов, методы. 5. Архитектура Web-приложения. Взаимодействие между клиентом и сервером. 6. Безопасность передачи данных (HTTPS). 7. Примеры HTTP-запросов с разными методами и объясните их назначение.	Разработка FTP-клиента для передачи файлов. Подготовка схемы архитектуры Web-приложения.
Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы	1. Структура HTML-документа. Элементы являются основными. 2. CSS и стилизация HTML-страницы. 3. Компоненты архитектуры браузера. 4. Роль шаблонизаторов на стороне сервера. 5. AJAX-запросы. 6. Взаимодействие JavaScript и DOM. Привести примеры манипуляций.	Создание шаблона веб-страницы с HTML и CSS. Разработка динамической формы с валидацией на JavaScript. Исследование DevTools в браузерах: отладка DOM, сетевых запросов и JS.
Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	1. Технологии для хранения данных в веб-приложениях. 2. Работа с сессиями. 3. Принципы работы с реляционными базами данных. 4. Отличие форматов JSON и XML. 5. Сериализация и десериализация данных. 6. SVG-графика в веб-разработке. 7. Привести примеры запросов к базе данных (SQL) и их назначение.	Реализация хранения пользовательских данных в сессии. Сравнение форматов JSON и XML на практике: создание и парсинг файлов. Разработка страницы с векторной графикой на SVG.

Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер

«Хорошо»	Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера
«Удовлетворительно»	Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой самостоятельной работы

6. Оценочные материалы (фонд оценочных средств) для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

6.1. Паспорт фонда оценочных средств

№	Контролируемые разделы (темы) дисциплины	Код и наименование компетенции	Индикатор достижения компетенции	Наименование оценочного средства
1.	Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4 Создает инструментальные средства программирования	Опрос, доклад, реферат, тест, зачет
2.	Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4 Создает инструментальные средства программирования	Опрос, доклад, реферат, тест, зачет
3.	Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура	ПК-3. Способен разрабатывать компоненты системных	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы,	Опрос, доклад, реферат, тест, зачет

	Web-приложения. Запросы.	программных продуктов	загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4 Создает инструментальные средства программирования	
4.	Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствам браузера. JavaScript. Формы	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4 Создает инструментальные средства программирования	Опрос, доклад, реферат, тест, зачет
5.	Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4 Создает инструментальные средства программирования	Опрос, доклад, реферат, тест, зачет

Этапы формирования компетенций в процессе освоения ОПОП прямо связаны с местом дисциплин в образовательной программе. Каждый этап формирования компетенции, характеризуется определенными знаниями, умениями и навыками и (или) опытом профессиональной деятельности, которые оцениваются в процессе текущего контроля успеваемости, промежуточной аттестации по дисциплине (практике) и в процессе итоговой аттестации.

Дисциплина «Интернет-программирование» является промежуточным этапом комплекса дисциплин, в ходе изучения которых у студентов формируются компетенции ПК-3.

Формирование компетенции ПК-3 начинается с изучения дисциплин «Перспективы развития информатики и вычислительной техники», «Введение в информатику», Учебная практика: ознакомительная практика, «Объектно-ориентированное программирование», «Программирование для мобильных устройств», «Структуры и алгоритмы обработки данных», «Криптографические методы защиты информации», «Защита информации», Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-

технологическая) практика, «Микропроцессорные устройства систем управления».

Завершается работа по формированию у студентов указанных компетенций в ходе «Функциональное и логическое программирование», «Системное программирование».

Итоговая оценка сформированности компетенций ПК-3 определяется в период Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

В процессе изучения дисциплины, компетенции также формируются поэтапно.

Основными этапами формирования ПК-3 при изучении дисциплины Б1.Д(М).В.ДВ.3.2 «Интернет-программирование» является последовательное изучение содержательно связанных между собой тем учебных занятий. Изучение каждой темы предполагает овладение студентами необходимыми дескрипторами (составляющими) компетенций. Для оценки уровня сформированности компетенций в процессе изучения дисциплины предусмотрено проведение текущего контроля успеваемости по темам (разделам) дисциплины и промежуточной аттестации по дисциплине – зачет.

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

6.2.1. Контрольные вопросы по темам (разделам) для опроса на занятиях

Тема (раздел)	Вопросы
Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.	ПК-3 1. Влияние сетевых технологий на разработку распределённых приложений. 2. Многоуровневая модель сети и её значение в интернет-программировании. 3. Принцип действия модели OSI и её сопоставление с TCP/IP. 4. Роль интерфейсов в сетевом взаимодействии приложений. 5. Протокол как способ стандартизации взаимодействия между клиентом и сервером. 6. Адресация в сетях: IP, MAC и доменные имена. 7. Механизмы разрешения доменных имён с использованием DNS. 8. Использование протоколов TCP и UDP в прикладных приложениях. 9. Задачи и проблемы, возникающие при разработке сетевых программ.

	<p>10. Примеры реализации клиент-серверного взаимодействия с учётом протоколов и адресации.</p>
<p>Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений</p>	<p>ПК-3</p> <ol style="list-style-type: none"> 1. Сокеты как основа сетевого взаимодействия на низком уровне. 2. Создание клиентского приложения с использованием TCP-сокетов. 3. Реализация серверного приложения, обрабатывающего множественные подключения. 4. Различия между протоколами TCP и UDP в реализации сокетов. 5. Использование неблокирующего ввода-вывода в асинхронном программировании. 6. Применение многопоточности для одновременной обработки нескольких клиентов. 7. Примеры кроссплатформенных реализаций сокет-программ на разных языках. 8. Обработка ошибок при установке и завершении соединения. 9. Принципы масштабируемости и устойчивости серверных решений. 10. Методы тестирования и отладки низкоуровневых сетевых программ.
<p>Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.</p>	<p>ПК-3</p> <ol style="list-style-type: none"> 1. Роль протоколов верхнего уровня в организации интернет-взаимодействия. 2. Протокол FTP: управление соединением, передача команд и файлов. 3. Создание почтовых клиентов с использованием SMTP и POP3. 4. Принцип работы HTTP-протокола и структура HTTP-запросов и ответов. 5. Методы HTTP (GET, POST, PUT, DELETE) и их назначение. 6. Архитектура Web-приложения и модель клиент-сервер. 7. Реализация REST API и взаимодействие с фронтендом. 8. Обработка данных в формате JSON в HTTP-запросах и ответах. 9. Взаимодействие между браузером и сервером на уровне HTTP. 10. Примеры использования HTTP и SMTP в разработке корпоративных приложений.
<p>Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы</p>	<p>ПК-3</p> <ol style="list-style-type: none"> 1. Использование HTML для создания структуры веб-документов. 2. Применение CSS для стилизации и адаптации интерфейсов. 3. Архитектура браузера и этапы рендеринга страницы. 4. Генерация HTML на сервере с помощью шаблонизаторов (например, Razor, Jinja). 5. Динамическое обновление содержимого страницы с помощью JavaScript. 6. Работа с DOM и событиями для создания интерактивных интерфейсов.

	<p>7. Использование Fetch API и AJAX для организации клиент-серверного обмена.</p> <p>8. Создание и валидация форм средствами HTML и JavaScript.</p> <p>9. Поддержка одностраничных приложений (SPA) с использованием JavaScript-фреймворков.</p> <p>10. Примеры интеграции шаблонов, стилей и скриптов в веб-приложениях.</p>
<p>Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.</p>	<p>ПК-3</p> <p>1. Использование сессий и куки для хранения пользовательских данных.</p> <p>2. Работа с реляционными базами данных через SQL-запросы.</p> <p>3. Взаимодействие с NoSQL-хранилищами в веб-приложениях.</p> <p>4. Принципы сериализации данных в формат JSON для передачи по сети.</p> <p>5. XML как универсальный формат хранения и обмена структурированными данными.</p> <p>6. Сравнительный анализ XML и JSON в задачах обмена данными.</p> <p>7. Создание SVG-графики и внедрение в интерфейс веб-приложения.</p> <p>8. Методы передачи данных между клиентом и сервером с помощью REST API.</p> <p>9. Хранение и извлечение данных в веб-приложении с использованием ORM.</p> <p>10. Примеры комплексной работы с формами, сессиями и базами данных.</p>

Шкала оценивания ответов на вопросы

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает ответ на каждый теоретический вопрос, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает теоретические вопросы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает теоретические вопросы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не знает ответов на поставленные теоретические вопросы.

6.2.2. Темы для докладов

Тема (раздел)	Вопросы
Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса,	<p>ПК-3</p> <p>1. Особенности разработки распределённых приложений</p> <p>2. Выбор сетевого протокола при разработке веб-приложения</p> <p>3. Работа с IP- и доменной адресацией в программном коде</p> <p>4. Реализация уровневой архитектуры взаимодействия клиента и сервера</p>

протокола, адресации.	5. Интерфейсы прикладного программирования для сетевых операций
Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	ПК-3 1. Реализация TCP-сервера с поддержкой нескольких клиентов 2. Построение клиентского приложения для отправки и получения данных 3. Программирование UDP-протокола: задачи реального времени 4. Применение многопоточности в сетевых приложениях 5. Примеры взаимодействия Python/Java/C# с сокетами
Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.	ПК-3 1. Создание REST API с использованием HTTP-протокола 2. Передача файлов через FTP с программной реализацией клиента 3. Работа с электронной почтой через SMTP и POP3 в коде 4. Разработка структуры Web-приложения: Backend + Frontend 5. Использование HTTP-заголовков для обработки сессий и безопасности
Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы	ПК-3 1. Разработка HTML-форм и их серверная обработка 2. Применение CSS-фреймворков в адаптивной вёрстке 3. Интерактивные элементы на JavaScript: события и DOM 4. Генерация страниц с шаблонизаторами (Jinja, Razor и др.) 5. Работа с данными форм через Fetch API и асинхронные запросы
Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	ПК-3 1. Интеграция с реляционными базами данных через ORM 2. Передача и обработка JSON-объектов в веб-приложениях 3. Серверная обработка пользовательских сессий и авторизации 4. Применение XML в API и настройке конфигураций 5. Использование SVG в динамических интерфейсах и графиках

Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему доклада, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает тему доклада, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает тему доклада и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой

6.2.3. Оценочные средства остаточных знаний (тест)

ПК-3.

1. Какой уровень модели OSI отвечает за установление, управление и завершение сеансов связи?

- 1) Сетевой
- 2) Транспортный
- 3) Сеансовый
- 4) Прикладной

2. Какой протокол используется для разрешения доменных имён в IP-адреса?

- 1) FTP
- 2) DNS
- 3) HTTP
- 4) SMTP

3. Что такое MAC-адрес?

- 1) Уникальный идентификатор сетевого интерфейса
- 2) IP-адрес устройства
- 3) Протокол маршрутизации
- 4) Сетевой

шлюз

4. Какой уровень модели TCP/IP обеспечивает маршрутизацию данных?

- 1) Прикладной
- 2) Сетевой
- 3) Транспортный
- 4) Канальный

5. Какой протокол используется для передачи электронной почты от клиента к серверу?

- 1) POP3
- 2) SMTP
- 3) IMAP
- 4) HTTP

6. Что означает термин "протокол" в контексте сетей?

- 1) Аппаратное устройство
- 2) Программное обеспечение
- 3) Набор правил для обмена данными
- 4) Сетевой

кабель

7. Какой уровень модели OSI отвечает за представление данных?

- 1) Сеансовый
- 2) Представительский

- 3) Транспортный
- 4) Прикладной

8. Какой протокол используется для безопасной передачи данных в Интернете?

- 1) HTTP
- 2) FTP
- 3) HTTPS
- 4) SMTP

9. Что такое IP-адрес?

- 1) Уникальный идентификатор сетевого интерфейса
- 2) Адрес электронной почты
- 3) Уникальный адрес устройства в сети
- 4) Протокол передачи данных

10. Какой протокол используется для передачи файлов между компьютерами в сети?

- 1) HTTP
- 2) FTP
- 3) SMTP
- 4) DNS

11. Что такое сокет в сетевом программировании?

- 1) Аппаратное устройство
- 2) Протокол передачи данных
- 3) Интерфейс для обмена данными между приложениями по сети
- 4) Тип кабеля

12. Какой метод используется для установления соединения клиентом в TCP-сокетах?

- 1) bind()
- 2) listen()
- 3) connect()
- 4) accept()

13. Какой метод используется сервером для ожидания входящих соединений в TCP-сокетах?

- 1) bind()
- 2) listen()
- 3) connect()
- 4) accept()

14. Какой метод используется для привязки сокета к определённому адресу и порту?

- 1) bind()
- 2) listen()
- 3) connect()
- 4) accept()

15. Какой метод используется для перевода сокета в режим прослушивания входящих соединений?

- 1) bind()
- 2) listen()
- 3) connect()
- 4) accept()

16. Какой протокол обеспечивает надёжную доставку данных?

- 1) UDP
- 2) TCP
- 3) ICMP
- 4) ARP

17. Какой протокол обеспечивает быструю, но ненадёжную доставку данных?

- 1) UDP
- 2) TCP
- 3) ICMP
- 4) ARP

18. Какой метод используется для отправки данных через сокет?

- 1) send()
- 2) receive()
- 3) connect()
- 4) bind()

19. Какой метод используется для получения данных через сокет?

- 1) send()
- 2) receive()
- 3) connect()
- 4) bind()

20. Какой метод используется для закрытия сокета?

- 1) close()
- 2) shutdown()
- 3) terminate()
- 4) stop()

21. Какой метод HTTP используется для получения данных с сервера?

- 1) POST
- 2) GET
- 3) PUT
- 4) DELETE

22. Какой метод HTTP используется для отправки данных на сервер?

- 1) GET
- 2) POST
- 3) PUT
- 4) DELETE

23. Какой код состояния HTTP означает успешный запрос?

- 1) 200
- 2) 404
- 3) 500
- 4) 301

24. Какой код состояния HTTP означает, что страница не найдена?

- 1) 200
- 2) 404
- 3) 500
- 4) 301

25. Какой код состояния HTTP означает внутреннюю ошибку сервера?

- 1) 200
- 2) 404
- 3) 500
- 4) 301

26. Какой код состояния HTTP означает перенаправление?

- 1) 200
- 2) 404
- 3) 500
- 4) 301

27. Какой протокол используется для получения электронной почты?

- 1) SMTP
- 2) POP3
- 3) FTP
- 4) HTTP

28. Какой протокол используется для отправки электронной почты?

- 1) SMTP
- 2) POP3
- 3) FTP
- 4) HTTP

29. Какой протокол используется для передачи файлов?

- 1) SMTP
- 2) POP3
- 3) FTP
- 4) HTTP

30. Что означает аббревиатура REST в контексте веб-сервисов?

- 1) Remote Execution Service Transfer
- 2) Representational State Transfer
- 3) Reliable Secure Transfer
- 4) Random Execution Service Transfer

31. Какой тег используется для создания гиперссылки в HTML?

- 1) <link>
- 2) <a>
- 3) <href>
- 4) <hyperlink>

32. Какой тег используется для вставки изображения в HTML?

- 1) <image>
- 2)
- 3) <src>
- 4) <picture>

33. Какой атрибут тега указывает путь к изображению?

- 1) href
- 2) src
- 3) link
- 4) path

34. Какой тег используется для создания таблицы в HTML?

- 1) <table>
- 2) <tab>
- 3) <tr>
- 4) <td>

35. Какой тег используется для создания строки в таблице?

- 1) <table>

- 2) <tab>
- 3) <tr>
- 4) <td>

36. Какой тег используется для создания ячейки в таблице?

- 1) <table>
- 2) <tab>
- 3) <tr>
- 4) <td>

37. Какой тег используется для создания формы в HTML?

- 1) <form>
- 2) <input>
- 3) <textarea>
- 4) <button>

38. Какой атрибут HTML-элемента <form> указывает адрес обработки формы?

- 1) name
- 2) id
- 3) action
- 4) method

39. Какой атрибут тега <form> указывает метод отправки данных на сервер?

- 1) type
- 2) enctype
- 3) action
- 4) method

40. Какой тип элемента <input> используется для ввода пароля?

- 1) text
- 2) password
- 3) hidden
- 4) checkbox

41. Какой тег используется для создания выпадающего списка в HTML?

- 1) <select>
- 2) <input type="list">
- 3) <dropdown>
- 4) <optionbox>

42. Какой язык используется для стилизации веб-страниц?

- 1) HTML

- 2) JavaScript
- 3) CSS
- 4) PHP

43. Какой язык выполняется на стороне клиента в браузере?

- 1) PHP
- 2) Python
- 3) JavaScript
- 4) Ruby

44. Что делает функция document.getElementById() в JavaScript?

- 1) Удаляет элемент со страницы
- 2) Получает элемент по его классу
- 3) Возвращает элемент по его идентификатору
- 4) Создает новый элемент

45. Какой оператор в JavaScript используется для строгого сравнения значений и типов?

- 1) ===
- 2) !=
- 3) ====
- 4) =

Ключ к тесту:

1.3	2.2	3.1	4.2	5.2	6.3	7.2	8.3	9.3
10.2	11.3	12.3	13.4	14.1	15.2	16.2	17.1	18.1
19.2	20.1	21.2	22.2	23.1	24.2	25.3	26.4	27.2
28.2	29.1	30.3	31.2	32.2	33.2	34.2	35.1	36.4
37.1	38.3	39.4	40.2	41.1	42.3	43.3	44.3	45.3

Шкала оценивания результатов тестирования

% верных решений (ответов)	Шкала оценивания
85 - 100	отлично
70 - 84	хорошо
50- 69	удовлетворительно
0 - 49	неудовлетворительно

6.2.4 Примеры задач при разборе конкретных ситуаций

Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.

ПК-3.

1. Написать скрипт на Python, который получает IP-адрес устройства и определяет его сетевой адрес
2. Реализовать утилиту на языке C# или JavaScript, которая определяет, принадлежит ли IP-адрес определённой подсети.
3. Написать функцию на Python, которая возвращает MAC-адрес текущего устройства
4. Создать веб-страницу, которая отображает IP-адрес клиента, используя API стороннего сервиса.

Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений

ПК-3.

1. Написать клиент-серверное приложение на Python с использованием TCP. Клиент отправляет строку, сервер возвращает её в верхнем регистре.
2. Создать чат-приложение на JavaScript (Node.js) по протоколу UDP, где участники могут обмениваться сообщениями.
3. Реализовать TCP-сканер портов на Python, сканирующий диапазон и выводящий открытые порты.
4. Написать UDP-клиент, который повторно отправляет запрос, если в течение 5 секунд не получен ответ от сервера.

Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.

ПК-3.

1. Реализуйте на Python программу, которая подключается к FTP-серверу, авторизуется и скачивает файл.
2. Создайте SMTP-клиент на PHP, отправляющий email через внешний SMTP-сервер.
3. Разработайте простой HTTP-сервер на Node.js, возвращающий JSON-ответы.
4. С помощью JavaScript отправьте XMLHttpRequest на сервер и обработайте JSON-ответ.

Тема 4. HTML. CSS. Архитектура браузера. Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы

ПК-3.

1. Используя JavaScript написать приложение для решения квадратного уравнения.
2. Используя Python3 написать приложение для решения квадратного уравнения используя браузер в качестве клиента.
3. Используя PHP написать приложение для решения квадратного уравнения.

Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.

ПК-3.

1. Разработайте веб-приложение на PHP, сохраняющее данные формы в базу данных MySQL.
2. Напишите на Python API, возвращающее данные в формате JSON по GET-запросу.
3. Сконвертируйте данные из базы данных в XML-формат и сохраните их в файл.
4. Создайте страницу, загружающую SVG-графику из файла и отображающую её в браузере.

Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	обучающийся ясно изложил условие задачи, решение обосновал
«Хорошо»	обучающийся ясно изложил условие задачи, но в обосновании решения имеются сомнения;
«Удовлетворительно»	обучающийся изложил решение задачи, но обосновал его формулировками обыденного мышления;
«Неудовлетворительно»	обучающийся не уяснил условие задачи, решение не обосновал либо не сдал работу на проверку (в случае проведения решения задач в письменной форме).

6.2.5. Темы для рефератов

Тема (раздел)	Вопросы
Тема 1. Основы организации Интернет сетей. Понятия многоуровневой организации сети. Понятия интерфейса, протокола, адресации.	ПК-3 1. Реализация IP-сканера: анализ сетевых интерфейсов средствами программирования 2. Протокол ARP и его реализация на программном уровне 3. Создание простого сетевого анализатора на Python 4. Получение сетевой информации в приложениях: локальный IP, маска, шлюз 5. Работа с API внешних сервисов для определения IP-геолокации
Тема 2. Низкоуровневое программирование в Интернет сетях. Сокеты. Программирование клиентских и серверных приложений	ПК-3 1. Разработка TCP-клиента и сервера на Python/Java 2. Передача данных по UDP: особенности и реализация 3. Чат-приложение на основе сокетов: архитектура и исходный код 4. Безопасная передача данных: реализация TLS-соединения 5. Отладка и тестирование сокет-программ: методы и инструменты
Тема 3. Протоколы верхнего уровня. FTP. SMTP. POP3. HTTP протокол. Архитектура Web-приложения. Запросы.	ПК-3 1. Создание веб-сервера на Node.js: обработка HTTP-запросов 2. Отправка электронной почты из приложения: SMTP в действии 3. Разработка REST API с поддержкой авторизации 4. Протокол WebSocket: применение для обмена данными в реальном времени 5. Создание FTP-клиента на Python или Java
Тема 4. HTML. CSS. Архитектура браузера.	ПК-3 1. Разработка адаптивного веб-интерфейса с использованием

Шаблонизаторы. Построение страницы на стороне сервера. Организация запросов средствами браузера. JavaScript. Формы	Flexbox и Grid 2. Работа JavaScript с DOM: манипуляции и события 3. Использование шаблонов на стороне клиента: Handlebars, EJS 4. Создание SPA с помощью Vue/React: архитектура и принципы 5. Применение AJAX и Fetch API для взаимодействия с сервером
Тема 5. Хранение данных. Данные сессии. Взаимодействие с базами данных. JSON. Формат XML. Использование SVG.	ПК-3 1. Создание REST API с возвратом данных в формате JSON 2. Обработка XML-документов на стороне сервера (Python/PHP/Java) 3. Локальное хранение данных в браузере: localStorage, sessionStorage, IndexedDB 4. Работа с базами данных и экспорт в JSON/XML 5. Разработка системы комментариев с хранением данных на сервере

Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой самостоятельной работы

6.2. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОГО КОНТРОЛЯ

Вопросы для подготовки к промежуточной аттестации по итогам освоения дисциплины Интернет-программирование:

ПК-3.

1. Сокеты
2. Особенности программирования серверных приложений
3. Особенности программирования клиентских приложений
4. Программирование на стороне клиента. JavaScript
5. Особенности взаимодействия по CGI. Способы взаимодействия с браузером на языках высокого уровня
6. Программирование на стороне сервера. Сценарные языки общего назначения в Web-разработке
7. Специализированные сценарные языки для программирования на стороне сервера

8. Особенности использования баз данных в программировании
9. CMS
10. Принципы клиент-серверного взаимодействия в Web-разработке
11. Обработка форм с помощью JavaScript и серверных языков
12. Использование XMLHttpRequest и Fetch API
13. Асинхронность в JavaScript. Обещания (Promises) и async/await
14. Использование событий и обработчиков событий в JavaScript
15. Работа с DOM-деревом на языке JavaScript
16. Создание и валидация форм в HTML и JavaScript
17. Основы REST и RESTful-сервисов
18. Разработка REST API с использованием Node.js или PHP
19. AJAX-технологии и их роль в динамическом обновлении контента
20. Хранение и передача данных в формате JSON
21. Хранение и передача данных в формате XML
22. Разработка Web-приложений на Python (Flask, Django)
23. Разработка Web-приложений на PHP
24. Разработка Web-приложений на JavaScript с использованием Node.js
25. Работа с шаблонизаторами (например, Handlebars, EJS, Twig)
26. Работа с cookie, localStorage и sessionStorage
27. Принципы безопасности при передаче данных (HTTPS, шифрование)
28. Аутентификация и авторизация пользователей (сессии, токены)
29. Использование OAuth и JWT в Web-приложениях
30. Роутинг в Web-приложениях
31. Пагинация и фильтрация данных на сервере и клиенте
32. Основы работы с API сторонних сервисов
33. Использование fetch и axios для запросов к API
34. Особенности SPA (Single Page Application)
35. Использование фреймворков: Vue.js, React, Angular (вводное сравнение)
36. Разработка с использованием React (основы компонентов)
37. Хуки и управление состоянием в React
38. Разработка с использованием Vue.js (основы компонентов)
39. Организация структуры проекта во фреймворках
40. Использование NPM и систем сборки (Webpack, Vite)
41. Тестирование Web-приложений (юнит-тесты, e2e)
42. Принципы адаптивной и кроссбраузерной верстки
43. Основы CSS: каскадность, селекторы, наследование
44. Flexbox и Grid в современной вёрстке
45. Использование препроцессоров CSS (SASS, LESS)
46. Анимации и переходы в CSS
47. Основы мобильной Web-разработки
48. Прогрессивные Web-приложения (PWA)
49. Кеширование данных на клиенте и сервере
50. Использование сервис-воркеров (service workers)

51. Оптимизация скорости загрузки Web-страниц
52. Основы DevOps в Web-разработке
53. Деплой Web-приложений на сервер (введение)
54. Git и системы контроля версий в разработке Web-приложений
55. CI/CD для Web-проектов
56. Docker и контейнеризация Web-приложений
57. Основы безопасности Web-приложений (SQL-инъекции, XSS, CSRF)
58. Инструменты для отладки JavaScript и сетевых запросов
59. Логирование и мониторинг Web-приложений
60. Работа с WebSocket для двусторонней связи в реальном времени
61. Создание чата с использованием WebSocket
62. Организация маршрутизации в SPA
63. Использование middleware в серверной разработке
64. Интеграция с платежными системами (Stripe, PayPal)
65. Создание и подключение карты сайта, SEO-оптимизация
66. Использование CAPTCHA и reCAPTCHA в формах
67. Подключение и использование аналитики (Google Analytics, Yandex.Metrica)
68. Работа с почтовыми сервисами из Web-приложения
69. Создание административной панели для Web-приложения
70. Создание и реализация ролей пользователей (admin, user и др.)
71. Построение графиков и визуализация данных на клиенте (Chart.js, D3.js)
72. Организация загрузки и хранения файлов на сервере
73. Поддержка мультиязычности Web-приложений
74. Работа с push-уведомлениями в Web-приложениях
75. Микросервисная архитектура в Web-разработке

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

Основной целью проведения промежуточной аттестации является определение степени достижения целей по учебной дисциплине или ее разделам. Осуществляется это проверкой и оценкой уровня теоретической знаний, полученных обучающимися, умения применять их в решении практических задач, степени овладения обучающимися практическими навыками и умениями в объеме требований рабочей программы по дисциплине, а также их умение самостоятельно работать с учебной литературой.

Организация проведения промежуточной аттестации регламентирована «Положением об организации образовательного процесса в федеральном государственном автономном образовательном учреждении «Московский политехнический университет»

6.4.1. Показатели оценивания компетенций на различных этапах их формирования, достижение обучающимися планируемых результатов обучения по дисциплине

Код и наименование компетенции ПК-3. Способен разрабатывать компоненты программных продуктов				
Этап (уровень)	Критерии оценивания			
	неудовлетворительно	удовлетворительно	хорошо	отлично
знать	<p>Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний:</p> <ul style="list-style-type: none"> - базовые понятия компьютерных сетей, сетевых и веб-сервисов; - базовые принципы низкоуровневого сетевого программирование с использованием сокетов; - принципы реализации веб-приложений; - понятия о стандартах в Интернет сетях и Web-проектах. 	<p>Обучающийся демонстрирует неполное соответствие следующих знаний:</p> <ul style="list-style-type: none"> - базовые понятия компьютерных сетей, сетевых и веб-сервисов; - базовые принципы низкоуровневого сетевого программирование с использованием сокетов; - принципы реализации веб-приложений; - понятия о стандартах в Интернет сетях и Web-проектах. 	<p>Обучающийся демонстрирует частичное соответствие следующих знаний:</p> <ul style="list-style-type: none"> - базовые понятия компьютерных сетей, сетевых и веб-сервисов; - базовые принципы низкоуровневого сетевого программирование с использованием сокетов; - принципы реализации веб-приложений; - понятия о стандартах в Интернет сетях и Web-проектах. 	<p>Обучающийся демонстрирует полное соответствие следующих знаний:</p> <ul style="list-style-type: none"> - базовые понятия компьютерных сетей, сетевых и веб-сервисов; - базовые принципы низкоуровневого сетевого программирование с использованием сокетов; - принципы реализации веб-приложений; - понятия о стандартах в Интернет сетях и Web-проектах.
уметь	<p>Обучающийся не умеет или в недостаточной степени умеет выполнять</p> <ul style="list-style-type: none"> - программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; - проектировать и реализовывать веб-приложения; - разворачивать простейшие серверные программные сборки. 	<p>Обучающийся демонстрирует неполное соответствие следующих умений:</p> <ul style="list-style-type: none"> - программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; - проектировать и реализовывать веб-приложения; - разворачивать простейшие серверные программные сборки. 	<p>Обучающийся демонстрирует частичное соответствие следующих умений:</p> <ul style="list-style-type: none"> - программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; - проектировать и реализовывать веб-приложения; - разворачивать простейшие серверные программные сборки. 	<p>Обучающийся демонстрирует полное соответствие следующих умений :</p> <ul style="list-style-type: none"> - программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; - проектировать и реализовывать веб-приложения; - разворачивать простейшие серверные программные сборки.

владеть	Обучающийся не владеет или в недостаточной степени владеет: - понятийным аппаратом компьютерных сетей; - базовыми навыками веб-программирования; - базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.	Обучающийся владеет в неполном и проявляет недостаточность владения: - понятийным аппаратом компьютерных сетей; - базовыми навыками веб-программирования; - базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.	Обучающимся допускаются незначительные ошибки, неточности, затруднения, частично владеет - понятийным аппаратом компьютерных сетей; - базовыми навыками веб-программирования; - базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.	Обучающийся свободно применяет полученные навыки, в полном объеме владеет: - понятийным аппаратом компьютерных сетей; - базовыми навыками веб-программирования; - базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.
----------------	--	---	---	---

6.4.2. Методика оценивания результатов промежуточной аттестации

Показателями оценивания компетенций на этапе промежуточной аттестации по дисциплине «Интернет-программирование» являются результаты обучения по дисциплине.

Оценочный лист результатов обучения по дисциплине

Код компетенции	Знания	Умения	Навыки	Уровень сформированности и компетенции на данном этапе / оценка
ПК-3. Способен разрабатывать компоненты системных программных продуктов	на уровне знаний: знать базовые понятия компьютерных сетей, сетевых и веб-сервисов; знать базовые принципы низкоуровневого сетевого программирования с использованием сокетов; знать принципы реализации веб-приложений; знать понятия о стандартах в Интернет сетях и Web-проектах.	на уровне умений: уметь программировать простейшие клиент-серверные приложения с помощью сокетов на языках общего назначения; уметь проектировать и реализовывать веб-приложения; уметь разворачивать простейшие серверные программные сборки.	на уровне навыков: понятийным аппаратом компьютерных сетей; базовыми навыками веб-программирования; базовыми навыками программирования на языках общего назначения и администрирования в Интернет-сетях.	

Оценка по дисциплине зависит от уровня сформированности компетенций, закрепленных за дисциплиной и представляет собой среднее арифметическое от выставленных оценок по отдельным результатам обучения (знания, умения, навыки).

Оценка «зачтено» выставляется, если среднее арифметическое находится в интервале от 2,4 до 5,0. Оценка «не зачтено» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

Промежуточная аттестация обучающихся в форме зачета проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по дисциплине «Интернет-программирование», при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «зачтено», или «не зачтено».

Шкала оценивания	Описание
Зачтено	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Не зачтено	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков по этапам (уровням) сформированности компетенций, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

7. Электронная информационно-образовательная среда

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде Чебоксарского института (филиала) Московского политехнического университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), как на территории филиала, так и вне ее.

Электронная информационно-образовательная среда – совокупность информационных и телекоммуникационных технологий, соответствующих

технологических средств, обеспечивающих освоение обучающимися образовательных программ в полном объеме независимо от места нахождения обучающихся.

Электронная информационно-образовательная среда обеспечивает:

а) доступ к учебным планам, рабочим программам дисциплин (модулей), практик, электронным учебным изданиям и электронным образовательным ресурсам, указанным в рабочих программах дисциплин (модулей), практик;

б) формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы;

в) фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения программы бакалавриата;

г) проведение учебных занятий, процедур оценки результатов обучения, реализация которых предусмотрена с применением электронного обучения, дистанционных образовательных технологий;

д) взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействия посредством сети "Интернет".

Функционирование электронной информационно-образовательной среды обеспечивается соответствующими средствами информационно-коммуникационных технологий и квалификацией работников, ее использующих и поддерживающих.

Функционирование электронной информационно-образовательной среды соответствует законодательству Российской Федерации.

Основными составляющими ЭИОС филиала являются:

а) сайт института в сети Интернет, расположенный по адресу www.polytech21.ru, <https://chebpolytech.ru/> который обеспечивает:

- доступ обучающихся к учебным планам, рабочим программам дисциплин, практик, к изданиям электронных библиотечных систем, электронным информационным и образовательным ресурсам, указанных в рабочих программах (разделы сайта «Сведения об образовательной организации»);

- информирование обучающихся обо всех изменениях учебного процесса (новостная лента сайта, лента анонсов);

- взаимодействие между участниками образовательного процесса (подразделы сайта «Задать вопрос директору»);

б) официальные электронные адреса подразделений и сотрудников института с Яндекс-доменом @polytech21.ru (список контактных данных подразделений Филиала размещен на официальном сайте Филиала в разделе «Контакты», списки контактных официальных электронных данных преподавателей размещены в подразделах «Кафедры») обеспечивают взаимодействие между участниками образовательного процесса;

в) личный кабинет обучающегося (портфолио) (вход в личный кабинет размещен на официальном сайте Филиала в разделе «Студенту» подразделе «Электронная информационно-образовательная среда») включает в себя портфолио студента, электронные ведомости, рейтинг студентов и обеспечивает:

- фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися,
- формирование электронного портфолио обучающегося, в том числе с сохранение работ обучающегося, рецензий и оценок на эти работы,
- г) электронные библиотеки, включающие электронные каталоги, полнотекстовые документы и обеспечивающие доступ к учебно-методическим материалам, выпускным квалификационным работам и т.д.:
Чебоксарского института (филиала) - «ИРБИС»
- д) электронно-библиотечные системы (ЭБС), включающие электронный каталог и полнотекстовые документы:
 - ЭБС «ЛАНЬ» -<https://e.lanbook.com/>
 - Образовательная платформа Юрайт - <https://urait.ru>
 - IPR SMART -<https://www.iprbookshop.ru/>
- е) платформа цифрового образования Политеха -<https://lms.mospolytech.ru/>
- ж) система «Антиплагиат» -<https://www.antiplagiat.ru/>
- з) система электронного документооборота DIRECTUM Standard — обеспечивает документооборот между Филиалом и Университетом;
- и) система «1С Управление ВУЗом Электронный деканат» (Московский политехнический университет) обеспечивает фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися;
- к) система «POLYTECH systems» обеспечивает информационное, документальное автоматизированное сопровождение образовательного процесса;
- л) система «Абитуриент» обеспечивает документальное автоматизированное сопровождение работы приемной комиссии.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебник для вузов / В. В. Соколова. — Москва : Издательство Юрайт, 2025. — 160 с. — (Высшее образование). — ISBN 978-5-534-16302-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561336>.
2. Янцев, В. В. Разработка web-страниц на HTML, CSS и JavaScript : учебное пособие для вузов / В. В. Янцев. — Санкт-Петербург : Лань, 2024. — 148 с. — ISBN 978-5-507-49640-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/422462> (дата обращения: 18.05.2025). — Режим доступа: для авториз. пользователей.
3. Золкин, А. Л. Техническое и программное обеспечение вычислительных машин, систем и сетей : учебник для вузов / А. Л. Золкин. — Санкт-Петербург : Лань, 2025. — 164 с. — ISBN 978-5-507-51547-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

<https://e.lanbook.com/book/450872>. — Режим доступа: для авториз. пользователей.

Дополнительная литература:

1. Сысолетин, Е. Г. Разработка интернет-приложений : учебник для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2025. — 80 с. — (Высшее образование). — ISBN 978-5-534-17124-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562916>.

2. Гаврилов, М. В. Информатика и информационные технологии : учебник для вузов / М. В. Гаврилов, В. А. Климов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 355 с. — (Высшее образование). — ISBN 978-5-534-15819-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/535560>.

3. Коммутация и маршрутизация в компьютерных сетях : методические указания / составитель А. В. Попов. — Воронеж : ВГТУ, 2024. — 41 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/417497>. — Режим доступа: для авториз. пользователей.

Периодика:

Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Компьютерные технологии, управление, радиоэлектроника»: Научный рецензируемый журнал. <https://vestnik.susu.ru/ctcr> - Текст: электронный.

9. Профессиональные базы данных и информационно-справочные системы

Профессиональная база данных и информационно-справочные системы	Информация о праве собственности (реквизиты договора)
Университетская информационная система РОССИЯ https://uisrussia.msu.ru/	Тематическая электронная библиотека и база для прикладных исследований в области экономики, управления, социологии, лингвистики, философии, филологии, международных отношений, права. свободный доступ
научная электронная библиотека Elibrary http://elibrary.ru/	Научная электронная библиотека eLIBRARY.RU - это крупнейший российский информационно-аналитический портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты более 26 млн научных статей и публикаций, в том числе электронные версии более 5600 российских научно-технических журналов, из которых более 4800 журналов в открытом доступе свободный доступ
сайт Института научной информации по	Библиографические базы данных ИНИОН РАН по социальным и гуманитарным наукам ведутся с начала 1980-х годов. Общий объем массивов составляет более

<p>общественным наукам РАН. http://www.inion.ru</p>	<p>3 млн. 500 тыс. записей (данные на 1 января 2012 г.). Ежегодный прирост — около 100 тыс. записей. В базы данных включаются аннотированные описания книг и статей из журналов и сборников на 140 языках, поступивших в Фундаментальную библиотеку ИНИОН РАН. Описания статей и книг в базах данных снабжены шифром хранения и ссылками на полные тексты источников из Научной электронной библиотеки.</p>
<p>Федеральный портал «Российское образование» [Электронный ресурс] – http://www.edu.ru</p>	<p>Федеральный портал «Российское образование» – уникальный интернет-ресурс в сфере образования и науки. Ежедневно публикует самые актуальные новости, анонсы событий, информационные материалы для широкого круга читателей. Еженедельно на портале размещаются эксклюзивные материалы, интервью с ведущими специалистами – педагогами, психологами, учеными, репортажи и аналитические статьи. Читатели получают доступ к нормативно-правовой базе сферы образования, они могут пользоваться самыми различными полезными сервисами – такими, как онлайн-тестирование, опросы по актуальным темам и т.д.</p>
<p>Информационные технологии – периодическое научно-техническое издание в области информационных технологий, автоматизированных систем и использования информатики в различных приложениях novtex.ru</p>	<p>Издательство выпускает теоретические и прикладные научно-технические журналы, обеспечивающие научной, производственной, обзорно-аналитической и образовательной информацией руководящих работников и специалистов промышленных предприятий, научных академических и отраслевых организаций, а также учебных заведений в области приоритетных направлений развития науки и технологий.</p>
<p>Ассоциация инженерного образования России http://www.ac-raee.ru/</p>	<p>Совершенствование образования и инженерной деятельности во всех их проявлениях, относящихся к учебному, научному и технологическому направлениям, включая процессы преподавания, консультирования, исследования, разработки инженерных решений, оказания широкого спектра образовательных услуг, обеспечения связей с общественностью, производством, наукой и интеграции в международное научно-образовательное пространство. свободный доступ</p>

10. Программное обеспечение (лицензионное и свободно распространяемое), используемое при осуществлении образовательного процесса

<p>Аудитория</p>	<p>Программное обеспечение</p>	<p>Информация о праве собственности (реквизиты договора, номер лицензии и т.д.)</p>
------------------	--------------------------------	---

<p>№ 2196 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/магистратуры/бакалавриата/специалитета/ магистратуры,, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП»</p>	Windows 7 OLPNLAcdmc	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор № ППИ-126/2023 от 14.12.2023
	Google Chrome	Свободное распространяемое программное обеспечение (бессрочная лицензия)
	Zoom	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
<p>№2066 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория информационных технологий</p>	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор № ППИ-126/2023 от 14.12.2023
	Windows 7 OLPNLAcdmc Windows Server 2012	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Microsoft Office Standard 2019(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Eclipse IDE for Java EE Developers, NET Framework, JDK 8, Microsoft SQL Server Express Edition, Microsoft Visio Professional, Microsoft Visual Studio, MySQL Installer for Windows, NetBeans, SQL Server Management Studio, Microsoft SQL Server Java Connector,	свободно распространяемое программное обеспечение (бессрочная лицензия)

	Android Studio, IntelliJ IDEA.	
	КОМПАС-3D v20 и v21	Сублицензионный договор № Нп-22-00044 от 21.03.2022 (бессрочная лицензия)
	MathCADv.15	Сублиц.договор №39331/МОС2286 от 6.05.2013) номер лицензии-42661846 от 30.08.2007) (бессрочная лицензия)
	SimInTech	Отечественное программное обеспечение
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	AdobeFlashPlayer	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Visual Studio 2019	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Python 3.7	свободно распространяемое программное обеспечение (бессрочная лицензия)
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)
№ 1126 Помещение для самостоятельной работы обучающихся	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор № ППИ-126/2023 от 14.12.2023
	Windows 7 OLPNLAcdmc	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Гарант- справочно-правовая система	Договор №С-002-2025 от 09.01.2025
	Yandex браузер	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic(Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Zoom	

		свободно распространяемое программное обеспечение (бессрочная лицензия)
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)

11. Материально-техническое обеспечение дисциплины

Тип и номер помещения	Перечень основного оборудования и технических средств обучения
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры/бакалавриата/специалитета/ магистратуры,, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП» 2196 (Чебоксары, ул. К.Маркса, д.60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; стенды <u>Технические средства обучения:</u> мультимедийное оборудование (проектор, экран)
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория информационных технологий №2066 (Чебоксары, ул. К.Маркса, д.60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; автоматизированные рабочие места, автоматизированное рабочее место преподавателя, проектор и экран; маркерная доска; <u>Технические средства обучения:</u> компьютерная техника (процессор Core i3, оперативная память объемом не менее 8 Гб); сервер в лаборатории (8-ядерный процессор с частотой не менее 3 ГГц, оперативная память объемом не менее 16 Гб, жесткие диски общим объемом не менее 1 Тб)
Помещение для самостоятельной работы обучающихся № 1126 (г. Чебоксары, ул. К.Маркса. 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; <u>Технические средства обучения:</u> компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Филиала

12. Методические указания для обучающегося по освоению дисциплины

Методические указания для занятий лекционного типа

В ходе лекционных занятий обучающемуся необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации.

Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из основной и дополнительной литературы, рекомендованной преподавателем и предусмотренной учебной программой дисциплины.

Методические указания для занятий лабораторного типа.

Выполнению лабораторных работ предшествует проверка знаний студентов – их теоретической готовности к выполнению задания. Проверка знаний проводится в форме, которую определяет преподаватель дисциплины (тестирование, опрос).

При проведении лабораторных занятий выделяют следующие разделы:

- общие положения (перечень лабораторных или практических занятий);
- общие требования к выполнению работ;
- инструкция по каждой работе;
- справочные материалы и т. д.

Лабораторные занятия позволяют развивать у обучающегося творческое теоретическое мышление, умение самостоятельно изучать литературу, анализировать практику; учат четко формулировать мысль, вести дискуссию, то есть имеют исключительно важное значение в развитии самостоятельного мышления.

Готовясь к докладу или выступлению в рамках интерактивной формы, при необходимости, следует обратиться за помощью к преподавателю.

Методические указания к самостоятельной работе.

Самостоятельная работа обучающегося является основным средством овладения учебным материалом во время, свободное от обязательных учебных занятий. Самостоятельная работа обучающегося над усвоением учебного материала по учебной дисциплине может выполняться в библиотеке университета, учебных кабинетах, компьютерных классах, а также в домашних условиях. Содержание и количество самостоятельной работы обучающегося определяется учебной программой дисциплины, методическими материалами, практическими заданиями и указаниями преподавателя.

Самостоятельная работа в аудиторное время может включать:

- 1) конспектирование (составление тезисов) лекций;
- 2) выполнение контрольных работ;

- 3) решение задач;
- 4) работу со справочной и методической литературой;
- 5) работу с нормативными правовыми актами;
- 6) выступления с докладами, сообщениями на семинарских занятиях;
- 7) защиту выполненных работ;
- 8) участие в оперативном (текущем) опросе по отдельным темам изучаемой дисциплины;
- 9) участие в беседах, деловых (ролевых) играх, дискуссиях, круглых столах, конференциях;
- 10) участие в тестировании и др.

Самостоятельная работа во внеаудиторное время может состоять из:

- 1) повторения лекционного материала;
- 2) подготовки к практическим занятиям;
- 3) изучения учебной и научной литературы;
- 4) изучения нормативных правовых актов (в т.ч. в электронных базах данных);
- 5) решения задач, и иных практических заданий
- 6) подготовки к контрольным работам, тестированию и т.д.;
- 7) подготовки к практическим занятиям устных докладов (сообщений);
- 8) подготовки рефератов, эссе и иных индивидуальных письменных работ по заданию преподавателя;
- 9) выполнения курсовых работ, предусмотренных учебным планом;
- 10) выполнения выпускных квалификационных работ и др.
- 11) выделения наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями на консультациях.
- 12) проведения самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов, написания рефератов и эссе по отдельным вопросам изучаемой темы.

Текущий контроль осуществляется в форме устных, тестовых опросов, докладов, творческих заданий.

В случае пропусков занятий, наличия индивидуального графика обучения и для закрепления практических навыков студентам могут быть выданы типовые индивидуальные задания, которые должны быть сданы в установленный преподавателем срок.

13. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Обучение по дисциплине «Интернет-программирование» инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется преподавателем с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Для студентов с нарушениями опорно-двигательной функции и с ОВЗ по слуху предусматривается сопровождение лекций и практических занятий мультимедийными средствами, раздаточным материалом.

Для студентов с ОВЗ по зрению предусматривается применение технических средств усиления остаточного зрения, а также предусмотрена возможность разработки аудиоматериалов.

По дисциплине «Интернет-программирование» обучение инвалидов и лиц с ограниченными возможностями здоровья может осуществляться как в аудитории, так и с использованием электронной информационно-образовательной среды, образовательного портала и электронной почты.

ЛИСТ ДОПОЛНЕНИЙ И ИЗМЕНЕНИЙ

рабочей программы дисциплины

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения _____

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202__-202__ учебном году на заседании кафедры, протокол № от « » 202 г.

Внесены дополнения и изменения _____

