

Рабочая программа дисциплины

Разработка серверных и клиентских приложений

<i>Направление подготовки</i>	Информационные системы и технологии
<i>Код</i>	09.03.02
<i>Направленность (профиль)</i>	Проектирование, разработка и сопровождение информационных систем
<i>Квалификация выпускника</i>	(степень) бакалавр

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Профессиональные	-	ПК-3
Профессиональные	-	ПК-4

2. Компетенции и индикаторы их достижения

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ПК-3	ПК-3. Способен использовать математический аппарат, методологию программирования и современные компьютерные технологии для решения практических задач получения, хранения, обработки и передачи информации	ПК-3.1. Осуществление технологической поддержки при установке и настройке операционных систем и сетевых устройств в рамках типовых регламентов организации. ПК-3.2. Моделирование бизнес-процессов в типовой ИС. ПК-3.3. Согласование и утверждение требований к установке и настройке операционных систем и сетевых устройств. ПК-3.4. Проверка соответствия серверов требованиям ИС к оборудованию и программному обеспечению. ПК-3.5. Установка серверной части ИС; верификация правильности установки серверной части ИС. ПК-3.6. Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС.
ПК-4	ПК-4. Способен обеспечивать техническую поддержку процессов создания (модификации) и сопровождения ИС, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-4.1. Адаптация бизнес-процессов к возможностям типовой ИС. Разработка модели бизнес-процессов. Проектирование и дизайн ИС. ПК-4.2. Сбор данных о запросах и потребностях заказчика применительно к типовой ИС. Документирование собранных данных в соответствии с регламентами организации. ПК-4.3. Согласование и утверждение требований к типовой ИС. ПК-4.4. Разработка прототипов ИС на базе типовой ИС в соответствии с требованиями. ПК-4.5. Интеграция ИС с существующими ИС заказчика ПК-4.9. Проведение прямо-сдаточных испытаний (валидации) ИС в соответствии с установленными регламентами. ПК-4.10. Документальное оформление результата прямо-сдаточных испытаний в соответствии с установленными регламентами.

3. Описание планируемых результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки).

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ПК-3		
	- Основные концепции и принципы работы корпоративных информационных систем. - Методологии программирования и алгоритмы, используемые для обработки данных. - Основные математические модели и методы, применяемые в задачах обработки информации. - Современные технологии и инструменты для хранения и передачи данных. - Правила и стандарты, регулирующие безопасность и защиту информации в корпоративных системах.	- Применять математический аппарат для моделирования и анализа данных в корпоративных информационных системах. - Разрабатывать и оптимизировать алгоритмы для обработки и анализа больших объемов информации. - Использовать программные средства для проектирования и внедрения корпоративных информационных систем. - Осуществлять интеграцию различных информационных систем и баз данных. - Проводить тестирование и отладку программного обеспечения, обеспечивая его функциональность и безопасность.	- Навыками работы с современными языками программирования и инструментами разработки программного обеспечения. - Умением использовать специализированные программные пакеты для анализа данных и создания отчетов. - Способностью разрабатывать и внедрять эффективные решения для управления данными в корпоративной среде. - Компетенцией в области защиты информации и соблюдения норм безопасности при работе с корпоративными данными. - Умением работать в команде, используя современные средства коммуникации и управления проектами для достижения общих целей.

Код компетенции	ПК-4		
	- методы оценки объемов и сроков выполнения работ, технологии выполнения работ в организации; - архитектуру, устройство и функционирование вычислительных систем, коммуникационное оборудование, сетевые протоколы; - основы современных операционных систем, основы современных систем управления базами данных, устройство и функционирование современных ИС; теорию баз данных, системы хранения и анализа баз данных; - основы программирования, современные объектно-ориентированные языки программирования, современные структурные языки программирования, языки современных бизнес-приложений; - современные методики тестирования разрабатываемых ИС, современные стандарты информационного взаимодействия систем; - программные	- работать с современными системами программирования, конструировать программное обеспечение и базы данных, разрабатывать основные программные документы; - оценивать объемы и сроки выполнения работ; - разрабатывать руководство программиста к модифицированным элементам типовой ИС; - разрабатывать руководство администратора к модифицированным элементам типовой ИС; - разрабатывать руководство пользователя к модифицированным элементам типовой ИС; - разрабатывать ТЗ.	- навыками конструирования программного обеспечения и баз данных; - навыками разработки интерфейсов обмена данными, форматов обмена данными, технологий обмена данными между ИС и существующими системами в соответствии с техническим заданием.

	средства и платформы инфраструктуры информационных технологий организаций; - современные подходы и стандарты автоматизации организации (например, CRM, MRP, ERP, ITIL, ITSM); - основы теории систем и системного анализа; - методики описания и моделирования бизнес-процессов, средства моделирования бизнес-процессов; - системы классификации и кодирования информации, в том числе присвоение кодов документам и элементам справочников; - отраслевую нормативную техническую документацию.		
--	---	--	--

4. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Разработка серверных и клиентских приложений» относится к части, формируемой участниками образовательных отношений учебного плана ОПОП.

Данная дисциплина взаимосвязана с другими дисциплинами, такими как: «Объектно-ориентированное программирование», «Введение в инженерную деятельность», «Операционные системы и среды», «Базы данных», «Основы использования и конфигурирования 1С: Предприятие».

В рамках освоения программы бакалавриата выпускники готовятся к решению задач профессиональной деятельности следующих типов:

Профиль (направленность) программы установлена путем ее ориентации на сферу профессиональной деятельности выпускников: Проектирование, разработка и сопровождение информационных систем

5. Объем дисциплины

<i>Виды учебной работы</i>	<i>Формы обучения</i>
	<i>Очная</i>
Общая трудоемкость: зачетные единицы/часы	4/144
Контактная работа:	
Занятия лекционного типа	18
Занятия семинарского типа	36
Промежуточная аттестация: экзамен	9
Самостоятельная работа (СРС)	81

6. Содержание дисциплины (модуля), структурированное по темам / разделам с указанием отведенного на них количества академических часов и видов учебных занятий

6.1. Распределение часов по разделам/темам и видам работы

6.1.1. Очная форма обучения

№ п/п	Раздел/тема	Виды учебной работы (в часах)						
		Контактная работа						Самосто ятельная работа
		Занятия лекционного типа		Занятия семинарского типа				
		Лекции	Иные учебные занятия	Практи ческие занятия	Семи нары	Лабора торные работы	Иные	
1.	Основы веб-разработки и архитектуры клиент-сервер	1			3			8
2.	Клиентская разработка: HTML, CSS, JavaScript	2			3			9
3.	Фреймворки и библиотеки для клиентской разработки	2			4			8
4.	Серверная разработка: языки и платформы	2			4			9
5.	Работа с базами данных	2			4			9
6.	API и взаимодействие клиент-сервер	2			5			10
7.	Безопасность веб-приложений	2			4			9
8.	Развёртывание и эксплуатация приложений	2			4			9
9.	Оптимизация и	3			5			10

	тестирование приложений							
	Промежуточная аттестация	9						
	Итого	18			36			81

6.2 Программа дисциплины, структурированная по темам / разделам

6.2.1 Содержание лекционного курса

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1.	Основы веб-разработки и архитектуры клиент-сервер	Принципы работы клиент-серверной архитектуры. Разделение ответственности: клиентская и серверная части. Протоколы взаимодействия (HTTP/HTTPS, WebSocket). REST и SOAP — сравнение подходов. Одностраничные приложения (SPA) и многостраничные приложения (MPA). Микросервисная и монолитная архитектуры.
2.	Клиентская разработка: HTML, CSS, JavaScript	Основы HTML5: семантическая разметка, формы, мультимедиа. CSS3: стилизация, Flexbox, Grid, адаптивный дизайн. JavaScript: синтаксис, переменные, функции, объекты, массивы. DOM-манипуляции и обработка событий. ES6+: стрелочные функции, модули, промисы, async/await. БЭМ-методология для структурирования CSS.
3.	Фреймворки и библиотеки для клиентской разработки	React: компоненты, состояние, props, хуки. Vue.js: реактивность, компоненты, маршрутизация. Angular: модули, сервисы, директивы. Управление состоянием: Redux, Vuex. Маршрутизация: React Router, Vue Router. Сборщики проектов: Webpack, Vite.
4.	Серверная разработка: языки и платформы	Обзор языков: Node.js, Python (Django/Flask), Java (Spring), PHP (Laravel), C# (ASP.NET). Выбор технологии под задачу. Структура серверного приложения. Middleware и обработка запросов. Работа с HTTP-серверами (Express.js, Koa).
5.	Работа с базами данных	Реляционные БД: PostgreSQL, MySQL. Нереляционные БД: MongoDB, Redis. ORM и ODM: Sequelize, Mongoose, SQLAlchemy. Проектирование схемы данных. Индексы, транзакции, миграции. Кэширование данных (Redis).
6.	API и взаимодействие	Проектирование RESTful API.

	клиент-сервер	Методы HTTP (GET, POST, PUT, DELETE, PATCH). Формат данных: JSON, XML. Аутентификация и авторизация: JWT, OAuth2, API-ключи. Документация API: Swagger/OpenAPI. GraphQL: запросы, мутации, резолверы.
7.	Безопасность веб-приложений	Основные угрозы: XSS, CSRF, SQL-инъекции, DDoS. Защита от атак: валидация ввода, санитанизация, CSP. HTTPS и SSL/TLS. Хранение паролей: хеширование (bcrypt). CORS и политика безопасности. Аудит безопасности и пентест.
8.	Развёртывание и эксплуатация приложений	Окружения: разработка, тестирование, продакшн. Контейнеризация: Docker, Docker Compose. Оркестрация: Kubernetes. CI/CD: GitLab CI, GitHub Actions, Jenkins. Облачные платформы: AWS, Azure, GCP, Heroku. Мониторинг и логирование (Prometheus, Grafana, ELK).
9.	Оптимизация и тестирование приложений	Виды тестирования: юнит, интеграционное, E2E. Инструменты: Jest, Mocha, Selenium, Cypress. Оптимизация производительности: минификация, бандлинг, ленивая загрузка. SEO-оптимизация клиентских приложений. Прогрессивные веб-приложения (PWA). Оценка производительности (Lighthouse, WebPageTest).

6.2.2 Содержание практических занятий

№ п/п	Наименование раздела дисциплины	Содержание практических занятий
1.	Основы веб-разработки и архитектуры клиент-сервер	Принципы работы клиент-серверной архитектуры. Разделение ответственности: клиентская и серверная части. Протоколы взаимодействия (HTTP/HTTPS, WebSocket). REST и SOAP — сравнение подходов. Одностраничные приложения (SPA) и многостраничные приложения (MPA). Микросервисная и монолитная архитектуры.
2.	Клиентская разработка: HTML, CSS, JavaScript	Основы HTML5: семантическая разметка, формы, мультимедиа. CSS3: стилизация, Flexbox, Grid, адаптивный дизайн. JavaScript: синтаксис, переменные, функции, объекты, массивы. DOM-манипуляции и обработка событий.

		ES6+: стрелочные функции, модули, промисы, async/await. БЭМ-методология для структурирования CSS.
3.	Фреймворки и библиотеки для клиентской разработки	React: компоненты, состояние, props, хуки. Vue.js: реактивность, компоненты, маршрутизация. Angular: модули, сервисы, директивы. Управление состоянием: Redux, Vuex. Маршрутизация: React Router, Vue Router. Сборщики проектов: Webpack, Vite.
4.	Серверная разработка: языки и платформы	Обзор языков: Node.js, Python (Django/Flask), Java (Spring), PHP (Laravel), C# (ASP.NET). Выбор технологии под задачу. Структура серверного приложения. Middleware и обработка запросов. Работа с HTTP-серверами (Express.js, Koa).
5.	Работа с базами данных	Реляционные БД: PostgreSQL, MySQL. Нереляционные БД: MongoDB, Redis. ORM и ODM: Sequelize, Mongoose, SQLAlchemy. Проектирование схемы данных. Индексы, транзакции, миграции. Кэширование данных (Redis).
6.	API и взаимодействие клиент-сервер	Проектирование RESTful API. Методы HTTP (GET, POST, PUT, DELETE, PATCH). Формат данных: JSON, XML. Аутентификация и авторизация: JWT, OAuth2, API-ключи. Документация API: Swagger/OpenAPI. GraphQL: запросы, мутации, резолверы.
7.	Безопасность веб-приложений	Основные угрозы: XSS, CSRF, SQL-инъекции, DDoS. Защита от атак: валидация ввода, санитанизация, CSP. HTTPS и SSL/TLS. Хранение паролей: хеширование (bcrypt). CORS и политика безопасности. Аудит безопасности и пентест.
8.	Развёртывание и эксплуатация приложений	Окружения: разработка, тестирование, продакшн. Контейнеризация: Docker, Docker Compose. Оркестрация: Kubernetes. CI/CD: GitLab CI, GitHub Actions, Jenkins. Облачные платформы: AWS, Azure, GCP, Heroku. Мониторинг и логирование (Prometheus, Grafana, ELK).
9.	Оптимизация и тестирование приложений	Виды тестирования: юнит, интеграционное, E2E. Инструменты: Jest, Mocha, Selenium, Cypress. Оптимизация производительности: минификация, бандлинг, ленивая загрузка. SEO-оптимизация клиентских приложений. Прогрессивные веб-приложения (PWA). Оценка производительности (Lighthouse, WebPageTest).

6.2.3 Содержание самостоятельной работы

№ п/п	Наименование раздела дисциплины	Содержание самостоятельной работы
1.	Основы веб-разработки и архитектуры клиент-сервер	Принципы работы клиент-серверной архитектуры. Разделение ответственности: клиентская и серверная части. Протоколы взаимодействия (HTTP/HTTPS, WebSocket). REST и SOAP — сравнение подходов. Одностраничные приложения (SPA) и многостраничные приложения (MPA). Микросервисная и монолитная архитектуры.
2.	Клиентская разработка: HTML, CSS, JavaScript	Основы HTML5: семантическая разметка, формы, мультимедиа. CSS3: стилизация, Flexbox, Grid, адаптивный дизайн. JavaScript: синтаксис, переменные, функции, объекты, массивы. DOM-манипуляции и обработка событий. ES6+: стрелочные функции, модули, промисы, async/await. БЭМ-методология для структурирования CSS.
3.	Фреймворки и библиотеки для клиентской разработки	React: компоненты, состояние, props, хуки. Vue.js: реактивность, компоненты, маршрутизация. Angular: модули, сервисы, директивы. Управление состоянием: Redux, Vuex. Маршрутизация: React Router, Vue Router. Сборщики проектов: Webpack, Vite.
4.	Серверная разработка: языки и платформы	Обзор языков: Node.js, Python (Django/Flask), Java (Spring), PHP (Laravel), C# (ASP.NET). Выбор технологии под задачу. Структура серверного приложения. Middleware и обработка запросов. Работа с HTTP-серверами (Express.js, Koa).
5.	Работа с базами данных	Реляционные БД: PostgreSQL, MySQL. Нереляционные БД: MongoDB, Redis. ORM и ODM: Sequelize, Mongoose, SQLAlchemy. Проектирование схемы данных. Индексы, транзакции, миграции. Кэширование данных (Redis).
6.	API и взаимодействие клиент-сервер	Проектирование RESTful API. Методы HTTP (GET, POST, PUT, DELETE, PATCH). Формат данных: JSON, XML. Аутентификация и авторизация: JWT, OAuth2, API-ключи. Документация API: Swagger/OpenAPI. GraphQL: запросы, мутации, резолверы.
7.	Безопасность	Основные угрозы: XSS, CSRF, SQL-инъекции, DDoS.

	веб-приложений	Защита от атак: валидация ввода, санитализация, CSP. HTTPS и SSL/TLS. Хранение паролей: хеширование (bcrypt). CORS и политика безопасности. Аудит безопасности и пентест.
8.	Развёртывание и эксплуатация приложений	Окружения: разработка, тестирование, продакшн. Контейнеризация: Docker, Docker Compose. Оркестрация: Kubernetes. CI/CD: GitLab CI, GitHub Actions, Jenkins. Облачные платформы: AWS, Azure, GCP, Heroku. Мониторинг и логирование (Prometheus, Grafana, ELK).
9.	Оптимизация и тестирование приложений	Виды тестирования: юнит, интеграционное, E2E. Инструменты: Jest, Mocha, Selenium, Cypress. Оптимизация производительности: минификация, бандлинг, ленивая загрузка. SEO-оптимизация клиентских приложений. Прогрессивные веб-приложения (PWA). Оценка производительности (Lighthouse, WebPageTest).

7. Текущий контроль по дисциплине (модулю) в рамках учебных занятий

В рамках текущего контроля преподаватель самостоятельно может проводить следующие мероприятия:

№ п/п	Контролируемые разделы (темы)	Наименование оценочного средства
1.	Основы веб-разработки и архитектуры клиент-сервер	Устный опрос, проблемно-аналитические задания, дискуссия, реферат
2.	Клиентская разработка: HTML, CSS, JavaScript	Устный опрос, проблемно-аналитические задания
3.	Фреймворки и библиотеки для клиентской разработки	Устный опрос, проблемно-аналитические задания, тестирование
4.	Серверная разработка: языки и платформы	Устный опрос, Проблемно-аналитические задания, презентация
5.	Работа с базами данных	Устный опрос, проблемно-аналитические задания, тестирование
6.	API и взаимодействие клиент-сервер	Устный опрос, Проблемно-аналитические задания
7.	Безопасность веб-приложений	Устный опрос, Проблемно-аналитические задания
8.	Развёртывание и эксплуатация приложений	Устный опрос, Проблемно-аналитические задания
9.	Оптимизация и тестирование приложений	Устный опрос, проблемно-аналитические задания, тестирование

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

8.1. Основная литература:

1. Позевалкин, В. В. Разработка веб-приложений на основе клиентских каркасов и библиотек : учебное пособие / В. В. Позевалкин, Н. Ф. Панова. — Оренбург : Оренбургский государственный университет, ЭБС АСВ, 2025. — 191 с. — ISBN 978-5-7410-3380-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/153228.html>

2. Щенёва, Ю. Б. Проектирование и разработка клиент-серверных приложений. Ч.1 : учебное пособие / Ю. Б. Щенёва. — Рязань : Рязанский государственный радиотехнический университет, 2024. — 124 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/150308.html>

8.2. Дополнительная литература:

1. Хабаров, А. Н. Разработка программных приложений : учебник / А. Н. Хабаров, А. Н. Ермакова. — Ставрополь : АГРУС, 2025. — 208 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/156614.html>

2. Сычев, А. В. Теория и практика разработки современных клиентских веб-приложений : учебное пособие / А. В. Сычев. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 482 с. — ISBN 978-5-4497-0943-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146402.html>

8.3. Периодические издания

1. Вестник Московского государственного технического университета имени Н.Э. Баумана. Серия Естественные науки. ISSN 1812-3368. <https://www.iprbookshop.ru/23124.html>

2. Открытые Системы. СУБД. ISSN 1028-7493. <https://www.iprbookshop.ru/76383.html>

3. Информационные технологии моделирования и управления. ISSN 1813-9744. <https://www.iprbookshop.ru/43350.html>

9. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины (модуля)

1. Федеральный портал «Российское образование» <http://www.edu.ru>
2. Электронно-библиотечная система «Научная электронная библиотека eLIBRARY.RU» <https://www.elibrary.ru>
3. Электронно-библиотечная система IPR BOOKS <https://www.iprbookshop.ru>
4. Электронная библиотечная система <https://biblioclub.ru>

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

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности – лекций, семинарских занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

1. Работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;

2. Внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
3. Выполнение самостоятельных практических работ;
4. Подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами окружающего нас мира. Важным составляющим в изучении данного курса является решение ситуационных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии и т.д.

Для лучшего запоминания материала целесообразно использовать индивидуальные особенности и разные виды памяти: зрительную, слуховую, ассоциативную. Успешному запоминанию также способствует приведение ярких свидетельств и наглядных примеров. Учебный материал должен постоянно повторяться и закрепляться.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

1. Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.
2. Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.
3. Время непосредственно перед экзаменом (зачетом) лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

1. Microsoft Windows Server;
 2. Семейство ОС Microsoft Windows;
 3. Libre Office свободно распространяемый офисный пакет с открытым исходным кодом;
 4. Информационно-справочная система: Система КонсультантПлюс (КонсультантПлюс);
 5. Информационно-правовое обеспечение Гарант: Электронный периодический справочник «Система ГАРАНТ» (Система ГАРАНТ);
- Перечень используемого программного обеспечения указан в п.12 данной рабочей программы дисциплины.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

12.1. Учебная аудитория для проведения учебных занятий, предусмотренных образовательной программой, оснащенная оборудованием и техническими средствами обучения.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя, проектор, экран, колонки

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Windows 10, КонсультантПлюс, Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Яндекс Браузер, LibreOffice, МТС Линк.

Подключение к сети «Интернет» и обеспечение доступа в электронную информационно-образовательную среду ММУ.

12.2. Помещение для самостоятельной работы обучающихся.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; комплект мебели для преподавателя; доска (маркерная).

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; колонки; проектор, экран.

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Windows Server 2016, Windows 10, Microsoft Office, КонсультантПлюс, Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Яндекс Браузер, LibreOffice, МТС Линк, Gimp, Paint.net, AnyLogic, Inkscape.

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ММУ.

13. Образовательные технологии, используемые при освоении дисциплины

Для освоения дисциплины используются как традиционные формы занятий – лекции (типы лекций – установочная, вводная, текущая, заключительная, обзорная; виды лекций – проблемная, визуальная, лекция конференция, лекция консультация); и семинарские (практические) занятия, так и активные и интерактивные формы занятий - деловые и ролевые игры, решение ситуационных задач и разбор конкретных ситуаций.

На учебных занятиях используются технические средства обучения мультимедийной аудитории: компьютер, монитор, колонки, настенный экран, проектор, микрофон, пакет программ Microsoft Office для демонстрации презентаций и медиафайлов, видеопроектор для демонстрации слайдов, видеосюжетов и др. Тестирование обучаемых может осуществляться с использованием компьютерного оборудования университета.

13.1. В освоении учебной дисциплины используются следующие традиционные образовательные технологии:

- чтение проблемно-информационных лекций с использованием доски и видеоматериалов;
- семинарские занятия для обсуждения, дискуссий и обмена мнениями;
- контрольные опросы;

- консультации;
- самостоятельная работа студентов с учебной литературой и первоисточниками;
- подготовка и обсуждение рефератов (проектов), презентаций (научно-исследовательская работа);
- тестирование по основным темам дисциплины.

13.2. Активные и интерактивные методы и формы обучения

Из перечня видов: («мозговой штурм», анализ НПА, анализ проблемных ситуаций, анализ конкретных ситуаций, инциденты, имитация коллективной профессиональной деятельности, разыгрывание ролей, творческая работа, связанная с освоением дисциплины, ролевая игра, круглый стол, диспут, беседа, дискуссия, мини-конференция и др.) используются следующие:

- диспут;
- анализ проблемных, творческих заданий, ситуационных задач;
- ролевая игра;
- круглый стол;
- мини-конференция;
- дискуссия;
- беседа.

13.3. Особенности обучения инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ)

При организации обучения по дисциплине учитываются особенности организации взаимодействия с инвалидами и лицами с ограниченными возможностями здоровья (далее – инвалиды и лица с ОВЗ) с целью обеспечения их прав. При обучении учитываются особенности их психофизического развития, индивидуальные возможности и при необходимости обеспечивается коррекция нарушений развития и социальная адаптация указанных лиц.

Выбор методов обучения определяется содержанием обучения, уровнем методического и материально-технического обеспечения, особенностями восприятия учебной информации студентами с инвалидностью и студентами с ограниченными возможностями здоровья и т.д. В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение и дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены печатными и электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Автономная некоммерческая организация высшего образования
«МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ»

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ**

Разработка серверных и клиентских приложений

<i>Направление подготовки</i>	Информационные системы и технологии
<i>Код</i>	09.03.02
<i>Направленность (профиль)</i>	Проектирование, разработка и сопровождение информационных систем
<i>Квалификация выпускника</i>	(степень) бакалавр

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Профессиональные	-	ПК-3
Профессиональные	-	ПК-4

2. Компетенции и индикаторы их достижения

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ПК-3	ПК-3. Способен использовать математический аппарат, методологию программирования и современные компьютерные технологии для решения практических задач получения, хранения, обработки и передачи информации	ПК-3.1. Осуществление технологической поддержки при установке и настройке операционных систем и сетевых устройств в рамках типовых регламентов организации. ПК-3.2. Моделирование бизнес-процессов в типовой ИС. ПК-3.3. Согласование и утверждение требований к установке и настройке операционных систем и сетевых устройств. ПК-3.4. Проверка соответствия серверов требованиям ИС к оборудованию и программному обеспечению. ПК-3.5. Установка серверной части ИС; верификация правильности установки серверной части ИС. ПК-3.6. Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС.
ПК-4	ПК-4. Способен обеспечивать техническую поддержку процессов создания (модификации) и сопровождения ИС, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-4.1. Адаптация бизнес-процессов к возможностям типовой ИС. Разработка модели бизнес-процессов. Проектирование и дизайн ИС. ПК-4.2. Сбор данных о запросах и потребностях заказчика применительно к типовой ИС. Документирование собранных данных в соответствии с регламентами организации. ПК-4.3. Согласование и утверждение требований к типовой ИС. ПК-4.4. Разработка прототипов ИС на базе типовой ИС в соответствии с требованиями. ПК-4.5. Интеграция ИС с существующими ИС заказчика ПК-4.9. Проведение приемо-сдаточных испытаний (валидации) ИС в соответствии с установленными регламентами. ПК-4.10. Документальное оформление результата приемо-сдаточных испытаний в соответствии с установленными регламентами.

3. Описание планируемых результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки).

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ПК-3		
	- Основные концепции и принципы работы корпоративных информационных систем. - Методологии программирования и алгоритмы, используемые для обработки данных. - Основные математические модели и методы, применяемые в задачах обработки информации. - Современные технологии и инструменты для хранения и передачи данных. - Правила и стандарты, регулирующие безопасность и защиту информации в корпоративных системах.	- Применять математический аппарат для моделирования и анализа данных в корпоративных информационных системах. - Разрабатывать и оптимизировать алгоритмы для обработки и анализа больших объемов информации. - Использовать программные средства для проектирования и внедрения корпоративных информационных систем. - Осуществлять интеграцию различных информационных систем и баз данных. - Проводить тестирование и отладку программного обеспечения, обеспечивая его функциональность и безопасность.	- Навыками работы с современными языками программирования и инструментами разработки программного обеспечения. - Умением использовать специализированные программные пакеты для анализа данных и создания отчетов. - Способностью разрабатывать и внедрять эффективные решения для управления данными в корпоративной среде. - Компетенцией в области защиты информации и соблюдения норм безопасности при работе с корпоративными данными. - Умением работать в команде, используя современные средства коммуникации и

			управления проектами для достижения общих целей.
Код компетенции	ПК-4		
	- методы оценки объемов и сроков выполнения работ, технологии выполнения работ в организации; - архитектуру, устройство и функционирование вычислительных систем, коммуникационное оборудование, сетевые протоколы; - основы современных операционных систем, основы современных систем управления базами данных, устройство и функционирование современных ИС; теорию баз данных, системы хранения и анализа баз данных; - основы программирования, современные объектно-ориентированные языки программирования, современные структурные языки программирования, языки современных бизнес-приложений; - современные методики тестирования разрабатываемых ИС, современные стандарты информационного взаимодействия систем; - программные средства	- работать с современными системами программирования, конструировать программное обеспечение и базы данных, разрабатывать основные программные документы; - оценивать объемы и сроки выполнения работ; - разрабатывать руководство программиста к модифицированным элементам типовой ИС; - разрабатывать руководство администратора к модифицированным элементам типовой ИС; - разрабатывать руководство пользователя к модифицированным элементам типовой ИС; - разрабатывать ТЗ.	- навыками конструирования программного обеспечения и баз данных; - навыками разработки интерфейсов обмена данными, форматов обмена данными, технологий обмена данными между ИС и существующими системами в соответствии с техническим заданием.

	и платформы инфраструктуры информационных технологий организаций; - современные подходы и стандарты автоматизации организации (например, CRM, MRP, ERP, ITIL, ITSM); - основы теории систем и системного анализа; - методики описания и моделирования бизнес-процессов, средства моделирования бизнес-процессов; - системы классификации и кодирования информации, в том числе присвоение кодов документам и элементам справочников; - отраслевую нормативную техническую документацию.		
--	---	--	--

3.2. Критерии оценки знаний студентов

Шкала оценивания	Индикаторы достижения	Показатели оценивания результатов обучения
ОТЛИЧНО/ЗАЧТЕНО	Знает:	- студент глубоко и всесторонне усвоил материал, уверенно, логично, последовательно и грамотно его излагает, опираясь на знания основной и дополнительной литературы, - на основе системных научных знаний делает квалифицированные выводы и обобщения, свободно оперирует категориями и понятиями.
	Умеет:	- студент умеет самостоятельно и правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, используя научные понятия, ссылаясь на нормативную базу.

	Владеет:	- студент владеет рациональными методами (с использованием рациональных методик) решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении продемонстрировал навыки - выделения главного, - связкой теоретических положений с требованиями руководящих документов, - изложения мыслей в логической последовательности, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
ХОРОШО/ЗАЧТЕНО	Знает:	- студент твердо усвоил материал, достаточно грамотно его излагает, опираясь на знания основной и дополнительной литературы, - затрудняется в формулировании квалифицированных выводов и обобщений, оперирует категориями и понятиями, но не всегда правильно их верифицирует.
	Умеет:	- студент умеет самостоятельно и в основном правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, не в полной мере используя научные понятия и ссылки на нормативную базу.
	Владеет:	- студент в целом владеет рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении смог продемонстрировать достаточность, но не глубинность навыков - выделения главного, - изложения мыслей в логической последовательности. - связки теоретических положений с требованиями руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
УДОВЛЕТВОРИТЕЛЬНО/ЗАЧТЕНО	Знает:	- студент ориентируется в материале, однако затрудняется в его изложении; - показывает недостаточность знаний основной и дополнительной литературы; - слабо аргументирует научные положения; - практически не способен сформулировать выводы и обобщения; - частично владеет системой понятий.
	Умеет:	- студент в основном умеет решить учебно-профессиональную задачу или задание, но допускает ошибки, слабо аргументирует свое решение, недостаточно использует научные понятия и руководящие документы.
	Владеет:	- студент владеет некоторыми рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении продемонстрировал недостаточность навыков - выделения главного, - изложения мыслей в логической последовательности. - связки теоретических положений с требованиями

		руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
Компетенция не достигнута		
НЕУДОВЛЕТВОРИТЕЛЬНО/ НЕ ЗАЧТЕНО	Знает:	- студент не усвоил значительной части материала; - не может аргументировать научные положения; - не формулирует квалифицированных выводов и обобщений; - не владеет системой понятий.
	Умеет:	студент не показал умение решать учебно-профессиональную задачу или задание.
	Владеет:	не выполнены требования, предъявляемые к навыкам, оцениваемым “удовлетворительно”.

При ответе на вопросы в рамках прохождения промежуточной аттестации (зачет/ зачет с оценкой/ экзамен) допускается вольная формулировка ответа, по смыслу раскрывающая содержание ответа, указанного в фонде оценочных средств, в качестве верного ответа.

4. Типовые контрольные задания и/или иные материалы для проведения промежуточной аттестации, необходимые для оценки достижения компетенции, соотнесенной с результатами обучения по дисциплине

5 СЕМЕСТР ПК-3

- Какой протокол используется для защищённой передачи данных в веб-приложениях?
 - FTP.
 - HTTP.
 - HTTPS.
 - SMTP.

Правильный ответ: в
- Какой метод HTTP используется для создания нового ресурса?
 - GET.
 - POST.
 - PUT.
 - DELETE.

Правильный ответ: б
- Какой язык программирования используется для клиентской разработки в браузере?
 - Python.

- б) Java.
- в) JavaScript.
- г) C++.

Правильный ответ: в

4. Что такое REST?
- а) Язык программирования.
 - б) Стил архитектуры ПО.
 - в) База данных.
 - г) Фреймворк.

Правильный ответ: б

5. Какой инструмент используется для контейнеризации приложений?
- а) Git.
 - б) Docker.
 - в) Webpack.
 - г) Jest.

Правильный ответ: б

6. Какой фреймворк основан на компонентах и использует виртуальный DOM?
- а) Angular.
 - б) React.
 - в) Vue.js.
 - г) jQuery.

Правильный ответ: б

7. Какой тип базы данных хранит данные в виде документов JSON?
- а) Реляционная.
 - б) Графовая.
 - в) Документоориентированная.
 - г) Ключ-значение.

Правильный ответ: в

8. Что такое JWT?
- а) Протокол передачи данных.
 - б) Формат токена для аутентификации.
 - в) Язык разметки.
 - г) Инструмент тестирования.

Правильный ответ: б

9. Какой инструмент используется для сборки JavaScript-проектов?
- а) Docker.
 - б) Webpack.
 - в) PostgreSQL.
 - г) Redis.

Правильный ответ: б

10. Какая атака предполагает внедрение вредоносного кода в веб-страницу?
- а) SQL-инъекция.
 - б) XSS.
 - в) CSRF.
 - г) DDoS.

Правильный ответ: б

11. Какой инструмент используется для автоматизированного тестирования веб-интерфейсов?
- а) Jest.
 - б) Selenium.
 - в) Mocha.

г) Postman.

Правильный ответ: б

12. Что такое CI/CD?

- а) Метод шифрования данных.
- б) Процесс непрерывной интеграции и доставки.
- в) Тип базы данных.
- г) Протокол безопасности.

Правильный ответ: б

13. Какой инструмент используется для мониторинга приложений?

- а) Grafana.
- б) Docker.
- в) React.
- г) Webpack.

Правильный ответ: а

14. Какой язык используется в Node.js?

- а) Java.
- б) Python.
- в) JavaScript.
- г) Ruby.

Правильный ответ: в

15. Что такое ORM?

- а) Система управления базами данных.
- б) Технология отображения объектов в реляционные таблицы.
- в) Протокол передачи данных.
- г) Фреймворк для фронтенда.

Правильный ответ: б

5 СЕМЕСТР ПК-4

1. Какой формат данных чаще всего используется в REST API?

- а) XML.
- б) JSON.
- в) YAML.
- г) CSV.

Правильный ответ: б

2. Что такое SPA?

- а) Статический сайт.
- б) Одностраничное приложение.
- в) Серверная архитектура.
- г) Система аутентификации.

Правильный ответ: б

3. Какой инструмент используется для управления зависимостями в Node.js?

- а) npm.
- б) Git.
- в) Docker.
- г) Webpack.

Правильный ответ: а

4. Что означает CORS?

- а) Кросс-доменные запросы.
- б) Протокол шифрования.
- в) Метод аутентификации.

г) Тип базы данных.

Правильный ответ: а

5. Какой инструмент используется для документирования API?

а) Swagger.

б) Postman.

в) Jest.

г) Docker.

Правильный ответ: а

6. Что такое GraphQL?

а) База данных.

б) Язык запросов к API.

в) Фреймворк для бэкенда.

г) Протокол передачи файлов.

Правильный ответ: б

7. Какой инструмент используется для логирования и анализа логов?

а) ELK Stack.

б) Kubernetes.

в) React.

г) MySQL.

Правильный ответ: а

8. Что такое PWA?

а) Прогрессивное веб-приложение.

б) Протокол веб-аутентификации.

в) Платформа для разработки.

г) Метод оптимизации.

Правильный ответ: а

9. Какой метод HTTP используется для полного обновления ресурса?

а) GET.

б) POST.

в) PUT.

г) PATCH.

Правильный ответ: в

10. Какой инструмент используется для оркестрации контейнеров?

а) Docker.

б) Kubernetes.

в) Webpack.

г) npm.

Правильный ответ: б

11. Что такое «ленивая загрузка» (lazy loading) в веб-разработке?

а) Отложенная загрузка ресурсов (изображений, модулей) по мере необходимости.

б) Предварительная загрузка всех ресурсов при старте приложения.

в) Синхронная загрузка данных.

г) Удаление неиспользуемых ресурсов.

Правильный ответ: а

12. Какой фреймворк разработан компанией Google?

а) React.

б) Vue.js.

в) Angular.

г) Ember.js.

Правильный ответ: в

13. Что такое миграция базы данных?

- а) Перенос данных между серверами.
- б) Процесс изменения схемы базы данных с сохранением данных.
- в) Резервное копирование данных.
- г) Очистка устаревших данных.

Правильный ответ: б

14. Какой инструмент позволяет автоматизировать тестирование API?

- а) Postman.
- б) Selenium.
- в) Jest.
- г) Webpack.

Правильный ответ: а

15. Что означает термин «бандлинг» в контексте веб-разработки?

- а) Объединение нескольких файлов JavaScript/CSS в один для оптимизации загрузки.
- б) Разделение кода на модули.
- в) Шифрование данных.
- г) Создание резервных копий файлов.

Правильный ответ: а