



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«МИРЭА – Российский технологический университет»

**Общий факультет (Фрязино)**

УТВЕРЖДАЮ

Директор филиала РТУ МИРЭА в г.  
Фрязино

\_\_\_\_\_ Макарова Л.А.

«\_\_» \_\_\_\_\_ 2023 г.

**Рабочая программа дисциплины (модуля)  
Программирование бизнес-приложений**

Читающее подразделение	<b>кафедра общенаучных дисциплин</b>
Направление	<b>09.03.01 Информатика и вычислительная техника</b>
Направленность	<b>Цифровизация предприятий в области радиоэлектроники</b>
Квалификация	<b>бакалавр</b>
Форма обучения	<b>очная</b>
Общая трудоемкость	<b>4 з.е.</b>

**Распределение часов дисциплины и форм промежуточной аттестации по семестрам**

Семестр	Зачётные единицы	Распределение часов							Формы промежуточной аттестации
		Всего	Лекции	Лабораторные	Практические	Самостоятельная работа	Контактная работа в период практики и (или) аттестации	Контроль	
4	4	144	16	0	32	60	2,25	33,75	Зачет, Курсовая работа

Программу составил(и):

*старший преподаватель, Рязанская Юлия Эдуардовна* \_\_\_\_\_

Рабочая программа дисциплины

**Программирование бизнес-приложений**

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)

составлена на основании учебного плана:

направление: 09.03.01 Информатика и вычислительная техника

направленность: «Цифровизация предприятий в области радиоэлектроники»

Рабочая программа одобрена на заседании кафедры

**кафедра общенаучных дисциплин**

Протокол от 13.01.2023 № 6

Зав. кафедрой Щучкин Григорий Григорьевич \_\_\_\_\_



## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

Дисциплина «Программирование бизнес-приложений» имеет своей целью способствовать формированию у обучающихся компетенций, предусмотренных данной рабочей программой в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.01 Информатика и вычислительная техника с учетом специфики направленности подготовки – «Цифровизация предприятий в области радиоэлектроники».

## 2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Направление:	09.03.01 Информатика и вычислительная техника
Направленность:	Цифровизация предприятий в области радиоэлектроники
Блок:	Дисциплины (модули)
Часть:	Часть, формируемая участниками образовательных отношений
Общая трудоемкость:	4 з.е. (144 акад. час.).

## 3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

В результате освоения дисциплины обучающийся должен овладеть компетенциями:

**УК-1** - Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

**ПК-1** - Способен проводить концептуальное и логическое проектирование информационных систем

**ПК-2** - Способен выполнять работы по созданию (модификации) информационной системы и ее частей, автоматизирующей задачи организационного управления, согласно установленному проекту

### ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), ХАРАКТЕРИЗУЮЩИЕ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

**УК-1** : Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

**УК-1.1** : Анализирует задачу, выделяя её базовые составляющие и осуществляет поиск информации для её решения

**Знать:**

- основы работы с HTML5 и JavaScript

**УК-1.2** : Определяет, интерпретирует и ранжирует информацию, требуемую для решения поставленной задачи

**Знать:**

- основы работы с фреймворками ReactJs и VueJS

**Уметь:**

- применять фреймворки при разработке веб-приложения

**УК-1.3** : Применяет системный подход для решения поставленных задач

**Знать:**

- основные понятия при работе с вёрсткой

**Владеть:**

- навыками верстки с применением JavaScript

**ПК-2 : Способен выполнять работы по созданию (модификации) информационной системы и ее частей, автоматизирующей задачи организационного управления, согласно установленному проекту**

**ПК-2.2 : Выполняет работы по созданию информационной системы и ее компонентов в части настройки рабочего окружения, создания архитектуры информационной системы и её функционирующих модулей**

**Знать:**

- Возможности ИС
- Языки современных бизнес-приложений
- Основные принципы обучения
- Основы менеджмента, в том числе менеджмента качества
- Программные средства и платформы инфраструктуры информационных технологий организаций
- Системы хранения и анализа баз данных
- Современные объектно-ориентированные языки программирования
- Современные структурные языки программирования
- Современный отечественный и зарубежный опыт в профессиональной деятельности
- Технологии подготовки и проведения презентаций

**Уметь:**

- Устанавливать программное обеспечение
- Устанавливать и настраивать прикладное ПО
- Кодировать на языках программирования
- Тестировать результаты собственной работы
- Использовать современные методы контроля производительности инфокоммуникационных систем
- Использовать современные измерительные приборы и программное обеспечение

**Владеть:**

- Осуществление выходного тестирования пользователей ИС
- Установка прикладного ПО, необходимого для функционирования ИС
- Сбор замечаний и пожеланий пользователей для развития ИС
- Настройка ИС для оптимального решения задач заказчика

**ПК-1 : Способен проводить концептуальное и логическое проектирование информационных систем**

**ПК-1.1 : Проводит анализ проблемной ситуации и разрабатывает требования к информационной системе**

**Знать:**

- Инструменты и методы прототипирования пользовательского интерфейса
- Основы программирования
- Программные средства и платформы инфраструктуры информационных технологий организаций
- Современные объектно-ориентированные языки программирования
- Современные подходы и стандарты автоматизации организации (например, CRM, MRP, ERP..., ITIL, ITSM)
- Современные структурные языки программирования
- Языки программирования и работы с базами данных
- Языки современных бизнес-приложений

**Уметь:**

- Кодировать на языках программирования
- Алгоритмизировать деятельность

**Владеть:**

- Согласование пользовательского интерфейса с заказчиком
- Разработка структуры программного кода ИС

**ПК-1.2 : Составляет концепцию, техническое задание и начальную документацию проекта на основе требований заказчика****Знать:**

- Диаграмма Ганта, метод "набегающей волны", типы зависимостей между работами
- Основы программирования
- Основы современных операционных систем
- Отраслевая нормативная техническая документация
- Программные средства и платформы инфраструктуры информационных технологий организаций
- Современные стандарты информационного взаимодействия систем
- Современные структурные языки программирования
- Языки современных бизнес-приложений

**Уметь:**

- Кодировать на языках программирования
- Тестировать результаты собственной работы

**Владеть:**

- Устранение обнаруженных несоответствий
- Анализ зафиксированных в системе учета дефектов и несоответствий в коде ИС и документации к ИС
- Установление причин возникновения дефектов и несоответствий
- Устранение дефектов и несоответствий
- Установление причин возникновения дефектов и несоответствий
- Устранение дефектов и несоответствий
- Фиксирование в системе учета факта внесения исправлений в код и документацию к ИС
- Параметризация дополнительных программных продуктов для тарификации сетевых ресурсов

**В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ) ОБУЧАЮЩИЙСЯ ДОЛЖЕН****Знать:**

- Современные структурные языки программирования
- Современные стандарты информационного взаимодействия систем
- Языки современных бизнес-приложений
- Языки современных бизнес-приложений
- Основы современных операционных систем
- Основы программирования
- Программные средства и платформы инфраструктуры информационных технологий организаций
- Отраслевая нормативная техническая документация
- Возможности ИС
- Современные структурные языки программирования
- Современные объектно-ориентированные языки программирования
- Технологии подготовки и проведения презентаций
- Современный отечественный и зарубежный опыт в профессиональной деятельности
- Основы менеджмента, в том числе менеджмента качества
- Основные принципы обучения
- Системы хранения и анализа баз данных
- Программные средства и платформы инфраструктуры информационных технологий организаций

- Диаграмма Ганта, метод "набегающей волны", типы зависимостей между работами
- Основы программирования
- Инструменты и методы прототипирования пользовательского интерфейса
- Современные структурные языки программирования
- Современные подходы и стандарты автоматизации организации (например, CRM, MRP, ERP..., ITIL, ITSM)
- Современные объектно-ориентированные языки программирования
- Программные средства и платформы инфраструктуры информационных технологий организаций
- основы работы с фреймворками ReactJs и VueJS
- основы работы с HTML5 и JavaScript
- основные понятия при работе с вёрсткой
- Языки программирования и работы с базами данных
- Языки современных бизнес-приложений

#### **Уметь:**

- Устанавливать и настраивать прикладное ПО
- Устанавливать программное обеспечение
- применять фреймворки при разработке веб-приложения
- Кодировать на языках программирования
- Тестировать результаты собственной работы
- Алгоритмизировать деятельность
- Кодировать на языках программирования
- Кодировать на языках программирования
- Тестировать результаты собственной работы
- Использовать современные методы контроля производительности инфокоммуникационных систем
- Использовать современные измерительные приборы и программное обеспечение

#### **Владеть:**

- Установка прикладного ПО, необходимого для функционирования ИС
- Сбор замечаний и пожеланий пользователей для развития ИС
- Осуществление выходного тестирования пользователей ИС
- Настройка ИС для оптимального решения задач заказчика
- Устранение обнаруженных несоответствий
- Анализ зафиксированных в системе учета дефектов и несоответствий в коде ИС и документации к ИС
- Разработка структуры программного кода ИС
- навыками верстки с применением JavaScript
- Согласование пользовательского интерфейса с заказчиком
- Установление причин возникновения дефектов и несоответствий
- Фиксирование в системе учета факта внесения исправлений в код и документацию к ИС
- Параметризация дополнительных программных продуктов для тарификации сетевых ресурсов
- Устранение дефектов и несоответствий
- Устранение дефектов и несоответствий
- Установление причин возникновения дефектов и несоответствий

## **4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

При проведении учебных занятий организация обеспечивает развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений и лидерских качеств.

<b>Код занятия</b>	<b>Наименование разделов и тем /вид занятия/</b>	<b>Сем.</b>	<b>Часов</b>	<b>Компетенции</b>
--------------------	--	-------------	--------------	--------------------

<b>1. Основы разработки бизнес-приложения.</b>				
<b>1.1</b>	<b>Типы современных бизнес-приложений (Лек).</b> Системы типа MRP, MRP II, ERP, ERP II, CRM, SCM, CSRP, B2C, B2B. RP – Material Resource Planning (Планирование потребностей в материальных ресурсах. Основная концепция MRP. Главное достижение MRP-систем. MRP II - Manufacturing Resource Planing (Планирование производственных ресурсов). ERP - Enterprise Resource Planning (Планирование ресурсов предприятия). Различие между концепциями MRP II и ERP. Инструментарий OLAP, средства поддержки принятия решений. Основные функции ERP-системы. CRM и SCM. CRM (Customer Relations Management) и SCM (Supply Chain Management) – управление отношениями соответственно с заказчиками и с поставщиками. CSRP – Customer Synchronized Resource Planning (Планирование ресурсов, синхронизированное с покупателем). ERP II – Enterprise Resource and Relationship Processing (Управление внутренними ресурсами и внешними связями предприятия). Управление взаимоотношениями с клиентами (CRM). Электронная коммерция. B2C (Business to Customer) и B2B (Business to Business) – обозначения широких классов программных продуктов, обслуживающих взаимоотношения предприятий с покупателями (B2C) и между собой (B2B). Пример B2C-системы.	4	2	
<b>1.2</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему:	4	2	
<b>1.3</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему:	4	2	
<b>1.4</b>	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: "Знакомство с фреймворками VueJS, ReactJS. Знакомство с платформой Node JS."	4	2	
<b>1.5</b>	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	
<b>1.6</b>	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе. Описание типа разрабатываемого приложения.	4	2	
<b>1.7</b>	<b>Основы разработки веб-приложений (Лек).</b> Современные языки разработки приложений. JavaScript. PHP. Python. Ruby. C#. Perl. Java. SQL. Программная часть. Серверная часть БД (базы данных). Серверная платформа. Области применения языков программирования. Системы управления базами данных.	4	2	
<b>1.8</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: изучение синтаксиса языка javascript ч.1	4	2	



1.9	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: изучение синтаксиса языка javascript ч.2	4	2	
1.10	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: "Знакомство с фреймворками VueJS, ReactJS. Знакомство с платформой Node JS."	4	2	
1.11	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	4	
1.12	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе. Описание используемых инструментов разработки приложения.	4	4	
1.13	<b>Веб-приложение (Web Application) (Лек).</b> Программа с двухуровневой архитектурой "клиент-сервер". Пользовательский интерфейс. Использование протоколов HTTP, HTTPS. Кроссплатформенность веб-приложений. Технологии программирования JavaScript, HTML5, CSS, фреймворки. Языки создания веб-приложений. Backend (серверная часть). Frontend (клиентская часть). DevOps программирование (автоматизация). Виды WEB-приложений. Portal Web App (порталы повышенного уровня сложности). Индивидуально разработанные системы. Системы электронной коммерции (E-commerce, Маркетплейсы). Этапы и основы разработки Web-приложений. Создание дизайна UI/UX. Backend разработка. Frontend разработка. Тестирование ПО. Размещение и развертывание web-приложения.	4	2	
1.14	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: развертывание серверной части для веб-приложения ч.1	4	2	
1.15	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: развертывание серверной части для веб-приложения ч.2	4	2	
1.16	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	
1.17	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: Описание используемых инструментов разработки приложения.	4	6	

<b>2. Разработка приложения</b>				
<b>2.1</b>	<b>Backend разработка (Лек).</b> PHP, Go, ASP.NET, C/C++, Python, Ruby, Java, API (REST, SOAP). Принципы работы HTTP. Модель OSI. Язык гипертекстовой разметки HTML. CSS (Cascading Style Sheets). Написание кода на стороне сервера. Написание кода для взаимодействия с базой данных.	4	2	УК-1.1, ПК-2.2
<b>2.2</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: разработка backend для приложения ч.1	4	2	УК-1.2, ПК-2.2
<b>2.3</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: разработка backend для приложения ч.2	4	2	УК-1.2, ПК-2.2
<b>2.4</b>	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: разработка backend	4	2	УК-1.2, ПК-2.2
<b>2.5</b>	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	УК-1.2, ПК-2.2
<b>2.6</b>	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: Описание используемых инструментов разработки приложения. Разработка backend для приложения. Описание процесса разработки.	4	2	УК-1.2, ПК-2.2
<b>2.7</b>	<b>Работа над графическим интерфейсом приложения (Лек).</b> Разница между React JS и Vue JS. Особенности React JS. Особенности Vue JS. Сравнение React JS и Vue JS. Ключевые различия между React JS и Vue JS. React JavaScript-библиотека для создания пользовательских интерфейсов. Спецификация типов элементов React. Использование нотации через точку в JSX типе. Выбор типа во время выполнения. Свойства в JSX. JavaScript выражения. Строковые литералы.	4	2	УК-1.1, УК-1.2, УК-1.3, ПК-1.1
<b>2.8</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: изучение фреймворков ReactJs и VueJs ч.1	4	2	УК-1.2, УК-1.3, ПК-1.1
<b>2.9</b>	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: изучение фреймворков ReactJs и VueJs ч.2	4	2	УК-1.2, УК-1.3, ПК-1.1
<b>2.10</b>	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: работа с фреймворками.	4	2	УК-1.1, УК-1.2, УК-1.3, ПК-1.1
<b>2.11</b>	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	УК-1.1, УК-1.2, УК-1.3, ПК-1.1

2.12	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: описание используемых инструментов разработки приложения. Описание фреймворков, какие возможности фреймворка будут использованы при курсовом проектировании.	4	2	УК-1.1, УК-1.2, УК-1.3, ПК-1.1
2.13	<b>Frontend разработка. (Лек).</b> Адаптивный дизайн, подход к дизайну сайтов, учитывающий их адаптацию под различные устройства и условия. Анимация, плавное изменение визуальных параметров объекта. Клиентсайд. Дизайн, визуальный облик. header. footer. MVP (Минимальный жизнеспособный продукт). SEO, или «оптимизация под поисковые системы».	4	2	ПК-1.1
2.14	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: разработка frontend части приложения ч.1	4	2	ПК-1.1
2.15	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: разработка frontend части приложения ч.2	4	2	ПК-1.1
2.16	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: разработка интерфейса приложения.	4	2	ПК-1.1
2.17	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	4	ПК-1.1
2.18	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: описание используемых инструментов разработки приложения. Предоставить описание интерфейса приложения, какие инструменты были использованы для его разработки.	4	6	ПК-1.1
2.19	<b>Тестирование, размещение и развертывание web-приложения (Лек).</b> Деплоймент приложения как критическая точка проекта разработки. Настройка сервера. Настройка приложения. Настройка клиента. Запуск сервера. Запуск клиента. Forever в node.js. События, доступные при использовании экземпляра Forever в node.js. Инструмента CLI.	4	2	ПК-1.1
2.20	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: деплоймент приложения ч.1	4	2	ПК-1.1
2.21	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: деплоймент приложения ч.2	4	2	ПК-1.1
2.22	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	ПК-1.1
2.23	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: описание используемых инструментов разработки приложения. Предоставить описание процесса деплоймента приложения.	4	4	ПК-1.1

2.24	<b>Исследование рынка и трендов при запуске продукта (Лек).</b> Анализ рынка – это процесс оценки, моделирования и выявления типичных особенностей рыночных тенденций. Понимания потребностей целевой аудитории; проверки конкурентоспособности продукта; изучения проблемных мест конкурентов; оптимизации уже существующих продуктов или услуг; минимизации рисков; планирования времени и ресурсов для создания продукта или услуги; создания правильных маркетинговых кампаний; привлечения инвесторов. Общие показатели рынка. Анализ спроса, целевой аудитории и конкурентов. Мониторинг спроса с помощью готовых сервисов. Google Trends. Яндекс.Вордстат. Обработка информации, визуализация данных.	4	2	
2.25	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: исследование рынка в области разрабатываемого продукта ч.1	4	2	
2.26	<b>Выполнение практических заданий (Пр).</b> Выполнение практической работы на тему: исследование рынка в области разрабатываемого продукта ч.1	4	2	
2.27	<b>Выполнение домашнего задания (Ср).</b> Выполнение домашней работы на тему: проведение исследования рынка.	4	2	
2.28	<b>Подготовка к аудиторным занятиям (Ср).</b> Повтор пройденного материала.	4	2	
2.29	<b>Выполнение курсовой работы (проекта) (Ср).</b> Формирование отчета по курсовой работе: формирование конечного отчета по курсовой работе. Исследование рынка по отношению к выпускаемому продукту.	4	2	
<b>3. Промежуточная аттестация (курсовая работа)</b>				
3.1	<b>Подготовка к сдаче промежуточной аттестации (КР).</b>	4	16,875	УК-1.1, УК-1.2, УК-1.3, ПК-1.1, ПК-1.2, ПК-2.2
3.2	<b>Контактная работа с преподавателем в период промежуточной аттестации (КрПА).</b>	4	2	УК-1.1, УК-1.2, УК-1.3, ПК-1.1, ПК-1.2, ПК-2.2
<b>4. Промежуточная аттестация (зачёт)</b>				
4.1	<b>Подготовка к сдаче промежуточной аттестации (Зачёт).</b>	4	16,875	УК-1.1, УК-1.2, УК-1.3, ПК-1.1, ПК-1.2, ПК-2.2
4.2	<b>Контактная работа с преподавателем в период промежуточной аттестации (КрПА).</b>	4	0,25	УК-1.1, УК-1.2, УК-1.3, ПК-1.1, ПК-1.2, ПК-2.2

## 5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

### 5.1. Перечень компетенций

Перечень компетенций, на освоение которых направлено изучение дисциплины «Программирование бизнес-приложений», с указанием результатов их формирования в процессе освоения образовательной программы, представлен в п.3 настоящей рабочей программы

### 5.2. Типовые контрольные вопросы и задания

1. Делегирование событий.
2. Как `this` работает в JavaScript.
3. Прототипное наследование.
4. IIFE: ``function foo(){ }();``. Что необходимо изменить, чтобы это стало IIFE.
5. Различие между переменными, значение которых: ``null``, ``undefined`` и не объявлено.
6. Что такое замыкание и как/для чего его используют.
7. Можете ли вы описать основное различие между циклом ``forEach`` и циклом ``map()``?
8. Ajax (асинхронный JavaScript и XML).
9. Какие преимущества и недостатки в использовании Ajax.
10. Объясните, как работает JSONP (и почему это не совсем AJAX).
11. Поднятие (hoisting).
12. Всплытие событий (event bubbling).
13. Vue Devtools.
14. Разница между синхронными и асинхронными функциями.
15. Цикл событий (event loop).
16. Стек вызовов (call stack) и очередь событий (task queue).
17. Объявление функции.
18. Методы HTTP-запросов
19. Версии HTTP-протокола
20. Коды ответа (состояния) HTTP
21. Cookie
22. Максимальный размер cookie
23. Директива `use strict`
24. Статическая и динамическая типизации
25. Как клиент взаимодействует с сервером
26. Что такое REST?
27. Объяснить понятие мутабельность/иммутабельность?
28. Какие типы являются мутабельными и наоборот?
29. Какие методы HTTP-запросов вы знаете?
30. Какие версии HTTP-протокола вам известны?
31. Какие знаете коды ответа (состояния) HTTP?
32. Что такое Cross-Origin Resource Sharing? Как устранить проблемы с CORS?
33. Что такое cookie?
34. Какой максимальный размер cookie?
35. Что означает директива `use strict`?
36. Чем JS отличается при работе на front-end и back-end?
37. Что такое статическая и динамическая типизации?
38. Как клиент взаимодействует с сервером?
39. Что такое REST?
40. Объяснить понятие мутабельность/иммутабельность? Какие типы являются мутабельными и наоборот?
41. Как искать ошибки в коде? Используете ли вы дебаггер?
42. Каких известных людей из мира JS знаете?
43. Опишите процесс всплытия (bubbling) событий в DOM.
44. Как получить высоту блока? Его положение относительно границ документа?
45. Что такое webpack?

46. Чем отличается dev-сборник от prod?
47. Что такое блочная модель CSS?
48. Какие способы центрирования блочного контента по горизонтали и вертикали знаете?
49. Какие подходы в верстке вам известны (float, flex, grid, etc.)?
50. Как сделать приложение responsive?
51. Какие есть принципы семантической верстки?
52. Зачем нужны префиксы для некоторых CSS-свойств (-webkit-, -moz- и т. д.)?
53. Как упростить написание кросс-браузерных стилей?
54. Практические задачи: прокомментировать и исправить пример плохого CSS или HTML.
55. Что такое CSS-препроцессоры? Что нового они приносят в стандартный CSS?
56. Для чего предназначены методы setTimeout и setInterval?
57. Сравните подходы работы с асинхронным кодом: callbacks vs promises vs async / await.
58. Можно ли записывать новые свойства / функции в прототипы стандартных классов (Array, Object и т. д.)? Почему нет? В каких случаях это делать можно? Как обезопасить себя, если нужно расширить прототип?
59. Назовите методы массивов, какие помните, и скажите, для чего они нужны.
60. Какие методы перебора массива знаете? В чем их отличие?
61. Как работают операторы присваивания / сравнения / строчные / арифметические / битовые и т. д.?
62. Опишите назначение и принципы работы с коллекциями Map и Set.
63. Что означает глубокая (deep) и поверхностная (shallow) копия объекта? Как сделать каждую из них?
64. Какая разница между декларацией функции (function declaration) и функциональным выражением (function expression)?
65. Что такое анонимная функция?
66. Расскажите о стрелочных функциях (arrow function). В чем заключаются отличия стрелочных функций от обычных?
67. Классовые компоненты. В чем их особенность?
68. Какие данные лучше хранить в состоянии компонента, а какие передавать через пропсы? Приведите пример.
69. Работа с хуками. В чем их преимущества?
70. Работа с фрагментами и порталами. Зачем они нужны?
71. Какие вы знаете методы жизненного цикла компонента?
72. В каком методе жизненного цикла компонента лучше делать запросы на сервер? Почему?
73. В каком методе жизненного цикла компонента лучше делать подписку и отписку от листенера? Почему? Зачем отписываться?
74. Работа с контекстом. Когда его стоит использовать?
75. В чем особенность PureComponent?
76. Работа с мемоизованными селекторами (memoized selectors)? Для чего их используют и какой принцип работы?
77. В чем видите преимущества библиотеки React?
78. Почему библиотека React быстрая? Что такое Virtual DOM и Shadow DOM?
79. Зачем в списках ключи? Можно ли делать ключами индексы элементов массива? Когда это оправдано?
80. В чем основная идея Redux?
81. Работа со стилями в React.
82. React - это библиотека или фреймворк? Какая разница между этими двумя понятиями.
83. Можно ли использовать jQuery вместе с React? Почему да / нет?
84. Что такое codemod?
85. Библиотеки, которые используются в связке с React.
86. Стандарт разработки
87. Стейкхолдер.
88. Ресурсы проекта
89. Качество проекта
90. Системная документация

91. Документация на программное обеспечение
92. Риски проекта
93. Обеспечение непрерывности проекта
94. BSC-инструменты
95. Стандарты поддержки ЖЦ ИС и ИТ
96. Ключевые стейкхолдеры и ценность ИТ
97. Условия осуществления проекта разработки и эксплуатации ИС и ИТ
98. Управление человеческими ресурсами на этапе внедрения ИС и ИТ
99. Решение проблем на стадии внедрения (управление коммуникациями).
100. Управление качеством ИС и ИТ
101. Стандарты управления качеством.
102. Эксплуатационная документация
103. Метрики в проектах по разработке ПО. Упреждающий анализ
104. Диагностические метрики.
105. Ретроспективные метрики.
106. Архитектура, трехуровневая модель.
107. Шаблоны архитектуры приложений.
108. Критерии хорошей ИТ-архитектуры.
109. Управление структурой БД.
110. Управление параллельной обработкой данных (блокировки).
111. Распределение прав и обязанностей по обработке данных (один из факторов обеспечения безопасности информации).
112. Обеспечение безопасности БД.
113. Осуществление процедур сохранения, восстановления и переноса БД.
114. Управление СУБД.
115. Поддержание репозитория данных в рабочем состоянии.
116. Функции администратора по управлению структурой БД
117. Требования к технологии проектирования
118. Технология проектирования
119. Средства спецификации. Классификация спецификаций
120. Классификация видов тестирования
121. Функциональное тестирование
122. Тестирование безопасности
123. Нефункциональные виды тестирования
124. Компонентное или Модульное тестирование
125. Баг Репорт
126. Разработка расписания: инструменты и методы
127. Метод критического пути
128. Методы исследования объекта автоматизации
129. Инструменты реализации проекта ИС и ИТ
130. Проектирование графического интерфейса программного продукта
131. Организация процесса разработки ПО и систем.

### 5.3. Фонд оценочных материалов

Полный перечень оценочных материалов представлен в приложении 1.

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 6.1. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Наименование помещения	Перечень основного оборудования
Учебная аудитория для проведения занятий лекционного и семинарского типа, групповых и индивидуальных	Мультимедийное оборудование, специализированная мебель, наборы демонстрационного оборудования и учебно-

консультаций, текущего контроля и промежуточной аттестации	наглядных пособий, обеспечивающие тематические иллюстрации.
Помещение для самостоятельной работы обучающихся	Компьютерная техника с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.
Компьютерный класс	Компьютерная техника с возможностью подключения к сети Интернет; Мультимедийное оборудование.

## 6.2. ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

1. P7-Офис.
2. jQuery. Свободное программное обеспечение (лицензия MIT)

## 6.3. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

### 6.3.1. Основная литература

1. Федотов А. В., Хомченко В. Г. Компьютерное управление в производственных системах [Электронный ресурс]: учебное пособие для впо. - Санкт-Петербург: Лань, 2020. - 620 с. – Режим доступа: <https://e.lanbook.com/book/140775>
2. Смирнов М. В., Исаев Р. А., Толмасов Р. С. Проектирование и разработка информационных систем и бизнес-приложений [Электронный ресурс]: Методические указания по выполнению курсового проекта. - М.: РТУ МИРЭА, 2020. - – Режим доступа: <https://library.mirea.ru/secret/04122020/2425.iso>
3. Смирнов М. В., Исаев Р. А., Толмасов Р. С. Проектирование и разработка информационных систем и бизнес-приложений: Методические указания [Электронный ресурс]:. - Москва: РТУ МИРЭА, 2020. - 44 с. – Режим доступа: <https://e.lanbook.com/book/163878>
4. Разработка инновационных механизмов повышения конкурентоспособности выпускников ИТ-специальностей вуза в условиях многопромышленного города [Электронный ресурс]: сборник статей. - Москва: ФЛИНТА, 2019. - 152 с. – Режим доступа: <https://e.lanbook.com/book/125425>
5. Андрианова Е. Г., Башлыкова А. А., Даева С. Г., и др. Корпоративные информационные системы [Электронный ресурс]: метод. рекомендации. - М.: РТУ МИРЭА, 2020. - – Режим доступа: <https://library.mirea.ru/secret/26082020/2375.iso>
6. Астапчук В. А., Терещенко П. В. Корпоративные информационные системы: требования при проектировании [Электронный ресурс]: Учебное пособие для вузов. - Москва: Юрайт, 2021. - 113 с – Режим доступа: <https://urait.ru/bcode/472111>

## 6.4. РЕКОМЕНДУЕМЫЙ ПЕРЕЧЕНЬ СОВРЕМЕННЫХ ПРОФЕССИОНАЛЬНЫХ БАЗ ДАННЫХ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ

1. Russian Software Developer Network — сообщество русскоговорящих разработчиков программного обеспечения <https://www.rsdn.org>
2. Российский фонд фундаментальных исследований <https://www.rfbr.ru>
3. Информационный портал системы международного цитирования Scopus <https://www.scopus.com>
4. Информационный портал системы международного цитирования “Web of Science”  
<https://www.apps.webofknowledge.com>
5. Научная электронная библиотека <http://www.elibrary.ru>

## 6.5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Самостоятельная работа студента направлена на подготовку к учебным занятиям и на развитие знаний, умений и навыков, предусмотренных программой дисциплины.

В соответствии с учебным планом дисциплина может предусматривать лекции,



практические занятия и лабораторные работы, а также выполнение и защиту курсового проекта (работы). Успешное изучение дисциплины требует посещения всех видов занятий, выполнение заданий преподавателя и ознакомления с основной и дополнительной литературой. В зависимости от мероприятий, предусмотренных учебным планом и разделом 4, данной программы, студент выбирает методические указания для самостоятельной работы из приведенных ниже.

При подготовке к лекционным занятиям студентам необходимо: перед очередной лекцией необходимо просмотреть конспект материала предыдущей лекции. При затруднениях в восприятии материала следует обратиться к основным литературным источникам. Если разобраться в материале опять не удалось, то обратитесь к лектору (по графику его консультаций) или к преподавателю на практических занятиях.

Практические занятия завершают изучение наиболее важных тем учебной дисциплины. Они служат для закрепления изученного материала, развития умений и навыков подготовки докладов, сообщений, приобретения опыта устных публичных выступлений, ведения дискуссии, аргументации и защиты выдвигаемых положений, а также для контроля преподавателем степени подготовленности студентов по изучаемой дисциплине. При подготовке к практическому занятию студенты имеют возможность воспользоваться консультациями преподавателя.

При подготовке к практическим занятиям студентам необходимо: приносить с собой рекомендованную преподавателем литературу к конкретному занятию; до очередного практического занятия по рекомендованным литературным источникам проработать теоретический материал, соответствующей темы занятия; в начале занятий задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении при решении задач, заданных для самостоятельного решения; в ходе семинара давать конкретные, четкие ответы по существу вопросов; на занятии доводить каждую задачу до окончательного решения, демонстрировать понимание проведенных расчетов (анализов, ситуаций), в случае затруднений обращаться к преподавателю.

Студентам, пропустившим занятия (независимо от причин), не имеющим письменного решения задач или не подготовившихся к данному практическому занятию, рекомендуется не позже чем в 2-недельный срок явиться на консультацию к преподавателю и отчитаться по теме, изученную на занятии.

Методические указания, необходимые для изучения и прохождения дисциплины приведены в составе образовательной программы.

## **6.6. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОБУЧЕНИЮ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ**

Освоение дисциплины обучающимися с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Предполагаются специальные условия для получения образования обучающимися с ограниченными возможностями здоровья.

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. № АК-44/05вн) в курсе предполагается использовать 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. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Медиа материалы также следует использовать и адаптировать с учетом индивидуальных

особенностей обучения лиц с ОВЗ.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);

- в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);

- методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);

- выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);

- устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

