

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
"Ивановский государственный политехнический университет"**

УТВЕРЖДАЮ

Директор института _____

_____ 2026 г.

Основы разработки сетевых приложений

рабочая программа дисциплины (модуля)

Закреплена за кафедрой	ИТиС (Информационных технологий и сервиса)		
Учебный план	09.03.02 Информационные системы и технологии		
Направленность (профиль)	Дизайн и продвижение цифрового продукта		
Квалификация	бакалавр		
Форма обучения	очная		
Общая трудоемкость	2 ЗЕТ		
Часов по учебному плану	64	Виды контроля в семестрах: зачет, 7 семестр	
в том числе:			
аудиторные занятия	26		
самостоятельная работа	38		

Распределение часов дисциплины по семестрам

Семестр	7		Итого	
Вид занятий	УП	РП	УП	РП
Лекции	14	14	14	14
Лабораторные	12	12	12	12
Итого ауд.	26	26	26	26
Контактная работа	26	26	26	26
Сам. работа	38	38	38	38
Часы на контроль				
Итого	64	64	64	64

Программу составил(и):

Ахмадулина Ю.С., ст.преподаватель, _____

Рецензент(ы):

Арбузова А.А., доцент, _____

Рабочая программа дисциплины

Основы разработки сетевых приложений

Разработана в соответствии с ФГОС ВО

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.02 Информационные системы и технологии, утвержден приказом Министерства образования и науки Российской Федерации № 926 от 19.09.2017

составлена на основании учебного плана

09.03.02 Информационные системы и технологии

утвержденного ученым советом вуза от 26.03.2026 протокол №6

Рабочая программа одобрена на заседании кафедры

ИТиС (Информационных технологий и сервиса)

Зав. кафедрой Арбузова А.А. _____

Согласовано

Зав. выпускающей кафедрой ИТиС Арбузова А.А. _____

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	
Цели:	Сформировать у студентов базовые теоретические знания и практические навыки разработки сетевых приложений (клиент-серверное взаимодействие, протоколы HTTP/HTTPS, REST API, веб-сокеты). Развить способность самостоятельно проектировать индивидуальный образовательный маршрут для профессионального роста в области сетевых технологий. Воспитать творческое отношение к решению профессиональных задач, умение предлагать нестандартные архитектурные решения для сетевых компонентов цифровых продуктов.
Задачи:	Изучить основные сетевые протоколы (TCP/IP, HTTP/HTTPS, WebSocket, DNS) и принципы клиент-серверной архитектуры. Научить студентов создавать простые сетевые приложения (HTTP-клиенты и серверы) с использованием языков программирования (Python, JavaScript) и соответствующих библиотек. Освоить работу с публичными API: отправка запросов, обработка ответов в формате JSON, обработка ошибок и таймаутов. Сформировать навыки отладки сетевых приложений с помощью инструментов (Postman, cURL, браузерные инструменты разработчика). Развить умение подбирать учебные ресурсы и самостоятельно осваивать новые технологии для решения конкретных задач (индивидуальный образовательный маршрут). Поощрять творческий подход при проектировании сетевой архитектуры (выбор между REST, GraphQL, gRPC, использование веб-сокеты и т.д.) для цифровых продуктов.

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ОП:	Б1.В.ДВ.06.01
2.1	Требования к предварительной подготовке обучающегося:
	Дисциплина «Основы разработки сетевых приложений» является важным разделом цикла учебных дисциплин по информационным технологиям, базирующимся на освоении ряда дисциплин: Облачные технологии в профессиональной деятельности, Системы автоматизированного проектирования «Основы разработки сетевых приложений» представляет собой интегрирующий практико-ориентированный курс.
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
	Дисциплина «Основы разработки сетевых приложений» является дисциплиной профессиональной части блока Б1.В.ДВ.06.01, позволяет студентам осознать принципы действия сквозных цифровых технологий и инструментов, применяющихся в цифровой экономике. Предшествует дисциплинам, формирующим применение сквозных технологий как профессиональную компетенцию: Проектирование информационных систем и технологий, Производственная практика (преддипломная), Выполнение и защита выпускной квалификационной работы.

3. ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ИНДИКАТОРЫ ИХ ДОСТИЖЕНИЯ	
ДПК-1:Способен решать профессиональные задачи с помощью знаний, приобретенных в процессе формирования индивидуальной образовательной траектории	
ДПК-1.1. Проектировать индивидуальные образовательные маршруты для построения профессиональной карьеры и определять стратегию профессионального развития	
ДПК-1.2. Формирование творческого отношения к решению профессиональных задач	
ДПК-1.3. Формирование практических навыков в области информационно-коммуникационных технологий	
В результате освоения дисциплины (модуля) обучающийся должен:	
Знать:	современные роли в области разработки сетевых приложений (бэкенд-разработчик, DevOps, сетевик, архитектор API); ключевые технологии, языки и фреймворки (Python + Django/Flask, JavaScript + Node.js, протоколы HTTP/HTTPS, WebSocket); ресурсы для самостоятельного изучения (онлайн-курсы, документация, сообщества); принципы проектирования удобных и интуитивных сетевых интерфейсов (API First, REST, HATEOAS); методы нестандартной отладки сетевых приложений (использование прокси, мокирование сервисов); примеры креативных решений в клиент-серверной архитектуре (веб-сокеты для живых чатов, long polling, Server-Sent Events); основы сетевых протоколов (HTTP/HTTPS, TCP/IP, WebSocket, DNS); принципы работы клиент-серверной архитектуры и REST API; базовые инструменты разработки и отладки сетевых приложений (Postman, cURL, браузерные инструменты разработчика, Wireshark); современные форматы обмена данными (JSON, XML, Protobuf).

Уметь:	оценивать свой текущий уровень владения сетевыми технологиями; подбирать учебные материалы и практические проекты для углубления знаний в конкретной области (например, REST API или веб-сокеты); составлять личный план развития на 6–12 месяцев с учётом карьерных целей в цифровом продукте; предлагать альтернативные способы организации сетевого взаимодействия (например, выбор между REST, GraphQL или gRPC в зависимости от контекста); создавать прототипы сетевых функций с использованием необычных источников данных (API сторонних сервисов, IoT); визуализировать архитектуру сетевого приложения нестандартными способами (например, интерактивные диаграммы в Mermaid); создавать простейшие сетевые приложения (например, клиент для отправки GET/POST запросов) на одном из языков (Python, JavaScript); использовать публичные API для получения данных (погода, курс валют, геолокация); настраивать базовый веб-сервер (локально) и тестировать его с помощью cURL или Postman; анализировать сетевой трафик веб-приложения с помощью инструментов разработчика.
Владеть:	навыками поиска, анализа и интеграции новых знаний в области сетевых приложений; способностью корректировать образовательный маршрут на основе обратной связи от профессиональной среды; приёмами документирования своих достижений (портфолио проектов, сертификаты); навыками генерации идей для улучшения сетевой части цифрового продукта (оптимизация запросов, кэширование, офлайн-режимы); приёмами дизайн-мышления применительно к сетевым интерфейсам; способностью защищать свои архитектурные предложения перед командой; навыками написания кода для HTTP-запросов (библиотеки requests в Python, fetch в JS); приёмами обработки ответов API (парсинг JSON, обработка ошибок, таймауты); навыками развёртывания тестового сетевого приложения на локальной или облачной платформе (например, Heroku, Render); умением документировать разработанное сетевое приложение (README, описание эндпоинтов).

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)			
Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Примечание
Раздел 1. Тема 1. Введение в сетевые приложения. Клиент-серверная архитектура и протокол HTTP: понятие сетевого приложения, модели взаимодействия (клиент-сервер, P2P). Обзор протокола HTTP: структура запроса и ответа, методы (GET, POST, PUT, DELETE), заголовки, коды состояния. URL/URI. Анализ HTTP-трафика с помощью браузерных инструментов разработчика (вкладка Network). Отправка GET-запроса вручную через cURL. Создание простого HTML-клиента с формой.			
/Лек/	7	2	
/Лаб/	7	1	
/СР/	7	5	
Раздел 2. Тема 2. Разработка простого HTTP-клиента и сервера на Python: использование библиотеки requests для отправки HTTP-запросов из Python. Основы создания простого веб-сервера с помощью встроенного модуля http.server. Понятие эндпоинта и маршрутизации; написать скрипт, который отправляет GET-запрос к публичному API (например, погода, курсы валют) и выводит ответ в формате JSON. Создать простой локальный сервер, возвращающий разные данные в зависимости от URL.			
/Лек/	7	2	
/Лаб/	7	1	
/СР/	7	5	
Раздел 3. Тема 3. Работа с REST API. Форматы данных JSON и XML: принципы REST (ресурсы, stateless, методы). Форматы обмена данными: JSON (синтаксис, парсинг), XML (базовые теги). Сравнение. Инструменты для тестирования API: Postman, Insomnia. С помощью Postman отправить несколько запросов к открытому API (например, GitHub API). В Python выполнить POST-запрос с данными в формате JSON. Разобрать полученный ответ и вывести выборочные поля.			
/Лек/	7	2	
/Лаб/	7	2	
/СР/	7	5	
Раздел 4. Тема 4. Веб-сокеты (WebSocket). Реализация чата в реальном времени: ограничения HTTP (полудуплекс, pull-модель). Протокол WebSocket: рукопожатие, фреймы, поддержка в браузерах. Библиотеки websockets в Python, Socket.IO. Создать простой эхо-сервер на Python с использованием библиотеки websockets и клиент в браузере (JavaScript). Организовать отправку и приём сообщений. (Базовый чат между двумя вкладками).			
/Лек/	7	2	
/Лаб/	7	2	
/СР/	7	5	
Раздел 5. Тема 5. Асинхронность и производительность сетевых приложений: проблема блокирующих операций ввода-вывода. Асинхронные подходы: многопоточность, асинхронные фреймворки (asyncio, FastAPI, Node.js). Знакомство с aiohttp для асинхронных запросов. Написать асинхронный клиент, который одновременно отправляет запросы к нескольким API и обрабатывает ответы. Измерить время выполнения по сравнению с синхронным вариантом.			
/Лек/	7	2	
/Лаб/	7	2	
/СР/	7	5	
Раздел 6. Тема 6. Обработка ошибок, таймауты и повторные попытки. Безопасность сетевых запросов: типичные ошибки: сетевые таймауты, ошибки соединения, HTTP-коды ошибок. Стратегии повторных запросов (retry с экспоненциальной задержкой). Основы HTTPS, SSL/TLS, проверка сертификатов. Аутентификация (API-ключи, OAuth). Модифицировать клиент, чтобы он обрабатывал исключения (ConnectionError, Timeout) и автоматически повторял запрос при неудаче. Добавить API-ключ в заголовок и протестировать доступ к защищённому API (например, Weather API с ключом).			
/Лек/	7	2	
/Лаб/	7	2	
/СР/	7	5	

Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Примечание
Раздел 7. Тема 7. Развёртывание и тестирование сетевого приложения. Документирование API: развёртывание простого сетевого приложения на облачной платформе (Render, Heroku, Railway). Основы контейнеризации (Docker) для сетевых приложений. Автоматическое тестирование эндпоинтов (pytest, Postman Collections). Спецификация OpenAPI (Swagger) и документирование. Развернуть ранее написанный HTTP-сервер (из темы 2 или 3) на бесплатном облаке (например, Render или PythonAnywhere). Написать простой тест с помощью pytest и requests для проверки эндпоинтов. Сгенерировать базовую документацию (README.md с примерами запросов).			
/Лек/	7	2	
/Лаб/	7	2	
/СР/	7	8	
/За/	7	0	

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В рамках внутренней системы оценки качества подготовки для текущего контроля и промежуточной аттестации в университете применяется балльно-рейтинговая система оценки знаний обучающихся (очной формы обучения).

В рамках внутренней системы оценки качества подготовки для текущего контроля и промежуточной аттестации в университете применяется балльно-рейтинговая система оценки знаний обучающихся (очной формы обучения).

При реализации образовательной программы в университете применяются электронное обучение и дистанционные образовательные технологии. Для проведения аудиторной и внеаудиторной контактной работы используются технологии видеоконференцсвязи: BBB в системе управления обучением LMS Moodle. Самостоятельная работа студентов осуществляется с помощью цифровых инструментов организации командной работы обучающихся: сервисы Яндекс, отечественные интерактивные доски. Для проведения всех видов занятий используется электронная информационно-образовательная среда вуза. При освоении разделов дисциплины используются следующие виды учебной работы: лекции, лабораторные работы, самостоятельная работа. При чтении лекций применяется объяснительно-иллюстративный метод изложения информации. При проведении практических занятий используются активные методы обучения. Лабораторные работы проводятся в диалоговой форме, используется демократический стиль взаимодействия преподавателя и обучающихся. Во время проведения лабораторных работ преподавателем постоянно задаются небольшие вопросы с просьбой ответить обучающихся на них как можно быстрее. Решение задач иллюстрируется на проекторе с использованием специализированного ПО (текстовые, табличные, графические редакторы, Jupiter notebook). Самостоятельная работа обучающихся заключается в работе с рекомендуемой литературой, рекомендованными образовательными платформами и ЭБС, рекомендованными онлайн-курсами, представленной печатными и электронными версиями учебников и методических указаний, в подготовке к практическим занятиям и к текущей аттестации. Также обучающиеся могут использовать материал, представленный преподавателем на своем курсе в электронной информационно-образовательной среде «Moodle», рекомендованные образовательные платформы, онлайн-курсы и ЭБС.

6. ОЦЕНОЧНЫЕ СРЕДСТВА

Фонд оценочных средств (ФОС) состоит из средств текущего контроля выполнения заданий, средств для промежуточных аттестаций. Оценочные средства содержат перечень:

- вопросов, ответы на которые дают возможность обучающемуся продемонстрировать, а преподавателю оценить степень усвоения теоретических и практических положений дисциплины;
- заданий, позволяющих оценить приобретенные обучающимися практические умения и навыки.

Текущий контроль знаний обучающихся проводится в виде аттестаций на 7, 13 неделях семестра. Результаты аттестации проставляются в ведомость в виде рейтинговой оценки. Максимальный балл в одной аттестации – 25. Для этого обучающемуся необходимо:

- своевременно и правильно выполнить полный объем работы,
- оформить и сдать в срок отчеты по лабораторным работам,
- уметь обобщить материал и сделать собственные выводы.

Промежуточный контроль в виде зачёта проводится в 7 семестре. Перевод рейтинговой оценки в итоговую оценку, проставляемую в зачетную книжку обучающегося, осуществляется в соответствии со следующей таблицей:

0-50 неудовлетворительно*

51-70 удовлетворительно

71-84 хорошо

85-100 отлично

* оценка «неудовлетворительно» в зачетную книжку не проставляется.

Вопросы к зачёту

Что такое клиент-серверная архитектура? Приведите пример.

Назовите основные методы HTTP-запросов (не менее 4). Какое их назначение?

Что означает код состояния HTTP 404? А 200? А 500?

Как отправить GET-запрос с помощью библиотеки requests в Python?

Что такое REST API? Назовите два основных принципа REST.

В каком формате чаще всего передают данные в современных API? Приведите пример одной строки в этом формате.

Как проанализировать сетевой трафик веб-страницы в браузере?

Для чего используется инструмент Postman?

Чем протокол WebSocket отличается от HTTP?

Как обработать ошибку соединения (ConnectionError) при сетевом запросе в Python?

Назовите два способа аутентификации при обращении к API.
 Как создать простейший HTTP-сервер на Python с помощью встроенного модуля?
 Что такое таймаут (timeout) сетевого запроса? Почему он важен?
 Как опубликовать (развернуть) простое сетевое приложение в облаке (например, на Render или PythonAnywhere)?
 Какая библиотека Python позволяет выполнять асинхронные HTTP-запросы? Приведите пример использования.

ФОС по дисциплине представлен в ПРИЛОЖЕНИИ к РПД.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1 Основная литература

	Авторы, составители	Заглавие, назначение, ссылка	Издательство, год
Л1.1	В. В. Янцев.	Web-программирование на Python : учебное пособие для вузов . — 3-е изд., перераб. — 180 с. — Текст : электронный // ЭБС «Лань» , ,	Санкт-Петербург : Лань, 2024. ,
Л1.2	С. П. Хабаров, М. Л. Шилкина.	Построение распределенных систем на базе WebSocket : учебное пособие для вузов / 2-е изд., стер. . 216 с. Текст : электронный // ЭБС «Лань» , ,	Санкт-Петербург : Лань, 2022, ,
Л1.3	В. В. Кручинин	Разработка сетевых приложений : учебное пособие / . — Москва : ТУСУР ; — 120 с. Текст : электронный // ЭБС «Лань» , ,	Санкт-Петербург : Лань, 2023. ,

7.1.2 Дополнительная литература

	Авторы, составители	Заглавие, назначение, ссылка	Издательство, год
Л2.1	В. В. Боженко.	Клиент-серверные информационные системы : учебное пособие / 80 с. Текст : электронный // ЭБС «Лань» , ,	Санкт-Петербург : ГУАП, 2022, ,
Л2.2	Зарипов, А. А. Куликов, М. М. Благирев.	Разработка клиент-серверных приложений : практикум : в 2 ч. Ч. 2 / 80 с.Текст : электронный / ЭБС «Лань» , ,	Москва : РТУ МИРЭА, 2024, ,

7.1.3 Методические разработки

	Авторы, составители	Заглавие, назначение, ссылка	Издательство, год
Л3.1	П. Б. Акмаров.	Компьютерные сети. Лабораторный практикум : учебное пособие для вузов / 120 с. Текст : электронный // ЭБС «Лань» , ,	Санкт-Петербург : Лань, 2024. ,

7.2. Электронные учебные издания и электронные образовательные ресурсы

E-learning Портал электронного образования ИВГПУ: <https://moodle.ivgpu.ru/>

7.3. Информационное обеспечение дисциплины

7.3.1 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства

	Лицензионное программное обеспечение вуза: Microsoft Windows, Microsoft Office. Свободно распространяемое ПО вуза: программный пакет Moodle. Прикладное программное обеспечение: Яндекс, Microsoft Edge, Opera, Mozilla Firefox и др. Национальный открытый университет (ИНТУИТ). – Режим доступа: http://www.intuit.ru/ Единое окно доступа к образовательным ресурсам. – Режим доступа: http://window.edu.ru/ Официальный сайт ИВГПУ. – Режим доступа: http://ivgpu.ru/ .
--	---

7.3.2 Перечень профессиональных баз данных и информационных справочных систем

7.3.2.1	ЭБС издательства "Лань", https://e.lanbook.com/
7.3.2.2	Электронная библиотека ИВГПУ, https://lib.ivgpu.ru/
7.3.2.3	ЭБС "Университетская Библиотека Онлайн", https://biblioclub.ru

8. МТО (ОБОРУДОВАНИЕ И ТЕХНИЧЕСКИЕ СРЕДСТВА ОБУЧЕНИЯ)

При подготовке к лабораторным работам, самостоятельной работе, при выполнении домашних работ и РГР и для контроля знаний обучающихся могут использоваться информационные технологии. Для этой цели на кафедре ИТиС имеется компьютерный класс, в котором установлено 12 персональных компьютеров на базе процессоров Intel Celeron 1,33GHz. Класс входит в локальную сеть кафедры и университета. Имеется выход в глобальную сеть Интернет. На всех компьютерах установлены операционные системы семейства MS Windows и необходимое прикладное программное обеспечение. Для визуализации учебного материала используется мультимедийный проектор Benq и монитор с большой диагональю экрана (47"). В учебном процессе также могут использоваться компьютерные классы и другие технические средства, предоставляемые университетским департаментом информационных и вычислительных технологий. Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду Университета.

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

Выполнение самостоятельной работы требует от обучающихся внимательного и вдумчивого отношения к поставленным задачам.

Для успешного решения возникающих вопросов необходимо заранее составить план получения информации по изучаемым темам. Для этого можно проконсультироваться с преподавателем по вопросам использования литературы и данных компьютерной сети. При использовании компьютерной сети необходимо:

- выполнить поиск сайтов, располагающих интересующей Вас информацией;
- провести сравнение полноты материала и манеры изложения на различных сайтах;
- подобрать необходимый материал.

При подготовке к работам в учебных лабораториях и аудиториях необходимо тщательно подготовиться к выполнению лабораторных работ. Для этого нужно внимательно изучить рекомендованную литературу, составить собственный план выполнения лабораторной работы. Желательно до выполнения работы попытаться ответить на контрольные вопросы, выделив те из них, которые вызывают наибольшие затруднения для консультации у преподавателя. Фактически при правильной подготовке к выполнению лабораторной работы обучающиеся могут во время занятий получить квалифицированную консультацию по сложным вопросам. Самостоятельная работа обучающегося является наиболее продуктивной формой образовательной и познавательной деятельности в период обучения.

Целью самостоятельной работы обучающихся является усвоение учебной информации и превращение ее в знания, умения и владения.

Рекомендуемое распределение часов по темам приведено в п. 4. За отведенное время необходимо изучить все разделы дисциплины, выполнить лабораторные работы, подготовиться к аттестации. Самостоятельная работа является основной в работе обучающегося. Она требует активной мыслительной деятельности и может привести к желаемым результатам лишь при ее правильной организации. Неумение работать самостоятельно является одной из основных причин низкой успеваемости.

Самостоятельная работа состоит из следующих модулей:

- работа над темами для самостоятельного изучения;
- подготовка к лабораторным работам;
- выполнение промежуточных тестов по разделам дисциплины;
- подготовка к промежуточной аттестации.

Контроль самостоятельной работы обучающихся и качество освоения дисциплины осуществляется посредством:

- проверки выполнения лабораторных работ;
- отчета по лабораторным работам;
- результатов тестирования по разделам дисциплины;
- проведения промежуточной аттестации.

10. УСЛОВИЯ РЕАЛИЗАЦИИ ООП ВО ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

Преподавание дисциплины должно учитывать особенности познавательной деятельности и личностной особенности обучающихся из числа лиц с ограниченными возможностями здоровья (ОВЗ) и инвалидов (по возможности на лекциях, семинарах, лабораторных работах и т.д.). Учет особенностей обучающихся с инвалидностью в полной мере проявляется на предусмотренных для таких обучающихся индивидуальных консультациях по дисциплине.

Необходимо во время преподавания в общей группе учитывать реальные возможности лиц с ОВЗ и инвалидов. Коррекция методики преподавания для данной группы лиц возможна на индивидуальных занятиях. Преподаватель должен учитывать физиологические особенности обучаемого и разработать рекомендации и приемы выполнения поставленной задачи индивидуально. Возможен подбор индивидуального задания с коррекцией уровня сложности выполняемого задания.

Специальные условия для получения образования обучающимися с ограниченными возможностями здоровья условия обучения, воспитания и развития таких обучающихся, включающие в себя использование специальных образовательных программ и методов обучения и воспитания, специальных учебников, учебных пособий и дидактических материалов, специальных технических средств обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, проведение групповых и индивидуальных коррекционных занятий, обеспечение доступа в здания организаций, осуществляющих образовательную деятельность, и другие условия, без которых невозможно или затруднено освоение образовательных программ обучающимися с ограниченными возможностями здоровья.

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

Необходимо учитывать тот факт, что обучающиеся не всегда имеют физическую возможность посещать занятия. Поэтому целесообразно разрабатывать дистанционную информационную поддержку процесса обучения. Таким средством могут являться обучающие компьютерные фильмы по данной дисциплине, презентации, которые обучающийся может просматривать дома в комфортной среде и без лимита времени.

Обучающиеся из числа лиц с ОВЗ и инвалидов обеспечиваются печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

При проведении занятий в компьютерных лабораториях целесообразно использовать проектор для наиболее удобного размещения обучающегося с ОВЗ при изложении материала занятия.