

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Ордена Трудового Красного Знамени федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский технический университет связи и информатики» (МТУСИ)

Рабочая программа дисциплины

FULL-STACK РАЗРАБОТКА

Направление подготовки
09.03.04 «Программная инженерия»

Направленность (профиль) программы
«ТОП-ИТ: Разработка и сопровождение программного обеспечения»

Квалификация выпускника
Бакалавр

Форма обучения
Очная

Москва, 2025 г.

Рабочая программа составлена с учетом требований федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.04 «Программная инженерия», утвержденного Приказом Министерства образования и науки РФ от 19.09.2017 № 920, и на основании учебного плана, утвержденного Ученым советом вуза 02.10.2025, протокол №2.

Разработчик(и) программы:

И.о. заведующего кафедрой ПИ, к.т.н.



М.С. Мосева

Рабочая программа утверждена на заседании кафедры ПИ.

И.о. заведующего кафедрой ПИ, к.т.н.



М.С. Мосева

Рабочая программа актуализируется (обновляется) ежегодно, в том числе в части программного обеспечения, материально-технического обеспечения, литературы.

Рабочая программа хранится на кафедре ПИ (Программная инженерия) и в деканате факультета ИТ (Информационные технологии).

1. Цели и задачи дисциплины (модуля)

Целями освоения дисциплины «Full-stack разработка» являются формирование у обучающихся целостного понимания архитектуры веб-приложений и освоение современных технологий комплексной full-stack разработки промышленного уровня, охватывающих клиентскую и серверную стороны, а также формирование базовых навыков работы с базами данных, инструментами контейнеризации и оркестрации.

Изучение дисциплины обеспечивает у обучающихся способность самостоятельно проектировать и реализовывать полнофункциональные веб-сервисы, проводить тестирование и оптимизацию производительности, применять методы защиты информации, организовывать процесс непрерывной интеграции и доставки, а также развивает навыки командной работы, эффективной коммуникации с заказчиком и управления жизненным циклом программного продукта.

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Full-stack разработка» включена в часть, формируемую участниками образовательных отношений блока дисциплин учебного плана (Б1.В.07). Дисциплина «Full-stack разработка» реализуется в соответствии с требованиями ФГОС ВО, ОПОП ВО и Учебного плана по направлению подготовки 09.03.04 «Программная инженерия», профиль «ТОП-ИТ: Разработка и сопровождение программного обеспечения».

Знания и умения, необходимые для успешного освоения дисциплины, формируются у обучающихся в результате изучения дисциплины «Введение в информационные технологии».

Материалы дисциплины используются при изучении курсов «Тестирование и отладка программного обеспечения», «Высоконагруженные приложения», а также при курсовом проектировании и выполнении выпускной квалификационной работы.

Рабочая программа дисциплины «Full-stack разработка» для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

3. Перечень планируемых результатов освоения дисциплины, соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной дисциплины направлено на формирование у обучающихся компетенций, представленных в таблице 1.

4. Структура и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часов). Процесс изучения дисциплины реализуется при очной форме обучения в 5, 6 семестрах. Промежуточная аттестация предусматривает зачет в 5 семестре, экзамен в 6 семестре.

4.1. Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 8 зач.ед. (288 часов), их распределение по видам работ по семестрам представлено в таблице 2.

Требования к результатам освоения дисциплины (модуля)

Таблица 1

№ п/п	Код компетенции	Содержание компетенции (или её части)	Индекс индикатора достижения компетенции	Содержание индикатора достижения компетенции	Результаты освоения индикатора достижения компетенции
1.	ПК-2	Способен выполнять интеграцию программных модулей и компонентов и проверять работоспособность выпусков программного продукта	ПК-2.1	Разрабатывает процедуры интеграции программных модулей	<p><i>Знает:</i> принципы API-first разработки и паттерны интеграции</p> <p><i>Умеет:</i> проектировать контракты данных между frontend и backend компонентами</p> <p><i>Владеет:</i> инструментами описания API и методологией непрерывной интеграции</p>
			ПК-2.2	Выполняет интеграцию программных модулей и компонентов и проверку работоспособности выпусков программного продукта	<p><i>Знает:</i> методы end-to-end тестирования и стратегии версионирования API</p> <p><i>Умеет:</i> настраивать пайплайны сборки и тестирования full-stack приложений</p> <p><i>Владеет:</i> инструментами e2e-тестирования и системами CI/CD</p>
2.	ПК-6	Способен участвовать в промышленной разработке программного обеспечения	ПК-6.1	Работает в соответствии с промышленными методологиями разработки	<p><i>Знает:</i> принципы промышленных методологий разработки и модели жизненного цикла ПО</p> <p><i>Умеет:</i> применять гибкие методологии при разработке fullstack-приложений на React и Python</p> <p><i>Владеет:</i> навыками оценки задач</p>

			ПК-6.2	Использует инструменты промышленной разработки	<p><i>Знает:</i> современные инструменты промышленной разработки: системы контроля версий (Git), контейнеризацию (Docker), фреймворки тестирования</p> <p><i>Умеет:</i> интегрировать инструменты в CI/CD пайплайн для React-фронтенда и Python-бэкенда</p> <p><i>Владеет:</i> практиками работы с GitFlow и отладки в IDE (VSCode/PyCharm)</p>
			ПК-6.3	Разрабатывает масштабируемый и поддерживаемый код	<p><i>Знает:</i> принципы SOLID, DRY, KISS и паттерны проектирования для React-компонентов и Python-бэкенда</p> <p><i>Умеет:</i> разрабатывать масштабируемые компоненты на React и модули на Python</p> <p><i>Владеет:</i> методами документирования кода через и написания модульных тестов</p>
3.	ПК-8	Способен проектировать, внедрять и оптимизировать потоки работ CI/CD	ПК-8.1	Проектирует архитектуру потоков работ CI/CD с учетом жизненного цикла приложения и требований команды	<p><i>Знает:</i> этапы CI/CD пайплайна и стратегии деплоя</p> <p><i>Умеет:</i> проектировать пайплайны с учетом особенностей React-сборки и Python-депоя</p> <p><i>Владеет:</i> навыками интеграции инструментов мониторинга в пайплайн</p>
			ПК-8.2	Реализует и настраивает этапы потока работ: сборку, тестирование, развертывание, мониторинг	<p><i>Знает:</i> инструменты сборки, фреймворки тестирования и системы мониторинга</p> <p><i>Умеет:</i> настраивать автоматическое тестирование React-компонентов и Python-API</p> <p><i>Владеет:</i> практиками развертывания приложений в контейнерах и настройки алертинга</p>
4.	ПК-11	Осуществляет управление архитектурой единой информационной среды	ПК-11.3	Осуществляет контроль проектирования и документирования программного	<p><i>Знает:</i> стандарты документирования архитектуры и принципы согласованного взаимодействия компонентов</p>

				обеспечения и его интеграции с точки зрения единой информационной среды	<i>Умеет:</i> проводить ревью архитектурных решений на соответствие требованиям интеграции <i>Владеет:</i> методами проектирования и документирования full-stack решений (диаграммы компонентов, sequence-диаграммы)
--	--	--	--	---	---

**Распределение трудоёмкости дисциплины (модуля) по видам работ
по семестрам**

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 2

Вид учебной работы	Трудоёмкость			
	Всего час.	В т.ч. по семестрам		Из них практическая подготовка
		5	6	
Общая трудоёмкость дисциплины по учебному плану	288	108	180	81
1. Контактная работа:	123	55	68	
Аудиторная работа всего, в том числе:	118	54	64	
<i>лекции (Л)</i>	34	18	16	13
<i>практические занятия (ПЗ)</i>	16		16	
<i>лабораторные работы (ЛР)</i>	68	36	32	68
Иная контактная работа в семестре (ИКР)	1	1		
Контактная работа в сессию (КРС)	4		4	
2. Самостоятельная работа (СР), контроль	165	53	112	
Вид промежуточного контроля		<i>Зачет</i>	<i>Экзамен</i>	

4.2. Содержание дисциплины

Тематический план дисциплины

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 3

Наименование разделов и тем дисциплины	Всего	Аудиторная работа (по видам учебных занятий)			Самостоятельная работа (СР), ИКР, КРС, Контроль	Форма текущего контроля успеваемости/форма промежуточной аттестации
		Л	ПЗ	ЛР		
5 семестр						
Раздел 1. Frontend-разработка	54	8		18	27	Лабораторные работы
Раздел 2. Backend-разработка	54	10		18	27	
Всего за 5 семестр	108	18		36	54	Зачет
6 семестр						
Раздел 3. Оптимизация приложений	62	6	6	12	38	Лабораторные работы, практические занятия
Раздел 4. Системная архитектура	118	10	10	20	78	
Всего за 6 семестр	180	16	16	32	116	
Объем дисциплины (в академических часах)	288					Зачет, экзамен
Объем дисциплины (в зачетных единицах)	8					

4.3. Лекции/лабораторные/практические занятия

Содержание лекций/лабораторного практикума/практических занятий

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 4

№ п/п	Название раздела, темы	№ и название лекций/ лабораторных/ практических занятий	Код(ы) формируемых индикаторов компетенций
1.	Раздел 1. Frontend-разработка		
	Тема 1. Создание и настройка клиентского приложения	Лекция № 1 Введение в UI/UX-дизайн интерфейсов. Основы протоколов HTTP, REST и GraphQL	ПК-2.1, ПК-2.2
		Лекция № 2 Основы разработки на React: маршрутизация и типизация	ПК-2.1, ПК-2.2
		Лабораторная работа № 1 Проектирование UI/UX и создание каркасного прототипа	ПК-2.1, ПК-2.2
		Лабораторная работа № 2 Верстка компонентов и настройка React-Router	ПК-2.1, ПК-2.2
	Тема 2. Разработка клиентского приложения	Лекция № 3 Управление состоянием в React-приложениях: основы и библиотеки	ПК-2.1, ПК-2.2
		Лекция № 4 Работа с данными на фронтенде: запросы и кэширование	ПК-2.1, ПК-2.2
		Лекция № 5 Инструменты сборки и деплоймент фронтенд-приложений	ПК-2.1, ПК-2.2
		Лабораторная работа № 3 Настройка сервера и базовая маршрутизация	ПК-2.1, ПК-2.2
		Лабораторная работа № 4 Интеграция фронтенда с бэкендом: получение и отображение данных	ПК-2.1, ПК-2.2
2.	Раздел 2. Backend-разработка		
	Тема 1. Разработка серверной части приложения	Лекция № 6 Основы разработки API: маршруты и валидация	ПК-6.1, ПК-6.2, ПК-6.3
		Лабораторная работа № 5 Настройка сервера и базовая маршрутизация	ПК-6.1, ПК-6.2, ПК-6.3
		Лекция № 7 Введение в базы данных, схемы данных и ORM	ПК-6.1, ПК-6.2, ПК-6.3
		Лабораторная работа № 6 Реализация CRUD-эндпоинтов	ПК-6.1, ПК-6.2, ПК-6.3
		Лекция № 8 Основы безопасности серверных приложений и JWT	ПК-6.1, ПК-6.2, ПК-6.3
		Лабораторная работа № 7 Защита API: JWT и middleware	ПК-6.1, ПК-6.2, ПК-6.3

№ п/п	Название раздела, темы	№ и название лекций/ лабораторных/ практических занятий	Код(ы) формируемых индикаторов компетенций
		Лабораторная работа № 8 Интеграция фронтенда с бэкендом: получение и отображение данных	ПК-6.1, ПК-6.2, ПК-6.3
3.	Раздел 3. Оптимизация приложений		
	Тема 1. Оптимизация и кэширование приложений	Лекция № 1 Асинхронность и тестирование клиентских приложений	ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 1 Асинхронная обработка запросов и оптимизация	ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 2 Асинхронность и производительность серверных приложений на Python	ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 2 Оптимизация работы с БД: индексы и миграции	ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 3 Оптимизация производительности и доступности фронтенда	ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 3 Управление состоянием и кэширование данных	ПК-8.1, ПК-8.2, ПК-11.3
4.	Раздел 4. Системная архитектура		
	Тема 1. Архитектура и паттерны	Лекция № 4 Архитектура клиентских приложений: принципы и паттерны	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 5 Архитектура серверных приложений: монолит и микросервисы	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 4 Unit- и E2E-тестирование компонентов	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 6 Паттерны проектирования UI-компонентов и архитектурные подходы	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 7 Паттерны проектирования бэкенда: DI, фабрики и репозитории	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 5 Рефакторинг и применение паттернов	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 8 Оптимизация производительности и доступности фронтенда	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лабораторная работа № 6 Оптимизация производительности и доступности	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3
		Лекция № 9 DevOps и мониторинг серверных приложений: основы и безопасность	ПК-2.1, ПК-2.2, ПК-6.1, ПК-6.2, ПК-6.3, ПК-8.1, ПК-8.2, ПК-11.3

5. Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся направлена на углубленное изучение разделов и тем рабочей программы и предполагает изучение литературных источников, выполнение домашних заданий и проведение исследований разного характера. Работа основывается на анализе литературных источников и материалов, публикуемых в интернете, а также реальных речевых и языковых фактов, личных наблюдений. Также самостоятельная работа включает подготовку и анализ материалов по темам пропущенных занятий.

Самостоятельная работа по дисциплине включает следующие виды деятельности:

- работа с лекционным материалом, предусматривающая проработку конспекта лекций и учебной литературы;
- поиск (подбор) и обзор литературы, электронных источников информации по индивидуально заданной проблеме курса, написание доклада, исследовательской работы по заданной проблеме;
- выполнение задания по пропущенной или плохо усвоенной теме;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашней контрольной работы (решение заданий, выполнение упражнений);
- изучение материала, вынесенного на самостоятельную проработку (отдельные темы, параграфы);
- подготовка к тестированию;
- подготовка к лабораторным работам;
- подготовка к практическим работам;
- подготовка к зачету и экзамену.

Самостоятельная работа обучающихся над усвоением материала по дисциплине может выполняться в помещении для самостоятельной работы МТУСИ, посредством использования электронной библиотеки и ЭИОС.

5.1. Контрольные вопросы и задания (для самостоятельного изучения)

1. Какие способы хранения токенов аутентификации на клиенте вы знаете и в чем их плюсы и минусы?
2. Объясните разницу между аутентификацией и авторизацией на клиенте.
3. Опишите механизм работы JWT: как клиент получает токен, как он его использует и какие потенциальные риски существуют.
4. Как реализовать автоматическое обновление access-токена с помощью refresh-токена на клиенте?
5. Что такое CORS и почему он важен при организации клиентской аутентификации?
6. Приведите пример конфигурации React Router v6 для трёх страниц: Home, Profile (/:userId) и NotFound.
7. Как передать параметры маршрута в компонент React с помощью хуков?
8. В чем разница между BrowserRouter и HashRouter и когда один вариант предпочтительнее другого?
9. Как типизировать пропсы компонента в React с помощью TypeScript?
10. Как типизировать хук useState для хранения объекта с полями name: string и age: number?
11. Какие основные роли выполняют сборщики Webpack и Vite и в чем их ключевые отличия?
12. Что такое tree shaking и как его настроить в Webpack?
13. Зачем нужны ладеры и плагины в Webpack и приведите по одному примеру каждого?
14. Как организовать горячую перезагрузку модулей (HMR) в Vite или Webpack Dev Server?
15. Как разбить сборку на чанки (code splitting) в React-приложении с помощью динамического импорта?
16. В чем ключевые отличия между fetch API и библиотекой axios?

17. Как реализовать REST-запрос на получение списка объектов и отобразить их в React-компоненте?
18. Что такое GraphQL и чем он отличается от REST? Приведите пример запроса на получение данных пользователя по id?
19. Какие стратегии кэширования запросов на фронтенде вы знаете и как они работают?
20. Как React Query или SWR помогают в управлении запросами и кэшем?
21. В чем состоят основные отличия локального (useState/useReducer) и глобального (Redux) управления состоянием?
22. Опишите архитектуру Redux и как взаимодействуют actions, reducers и store.
23. Приведите пример action creator и reducer для добавления элемента в список задач.
24. Как использовать middleware в Redux для работы с асинхронными экшнами (redux-thunk или redux-saga)?
25. Что означает концепция «единого источника правды» (single source of truth) в контексте Redux и зачем она нужна?
26. Перечислите основные методы HTTP и их назначение.
27. Объясните принципы REST (назовите минимум 4).
28. В чем ключевые отличия REST от GraphQL?
29. Опишите структуру JWT и процесс верификации токена.
30. Что такое идиоматичность HTTP-методов? Приведите примеры.
31. Что такое SQL-инъекция? Приведите пример профилактики в ORM.
32. Для чего нужны CORS? Как реализовать их в веб-фреймворках?
33. Дайте определение ACID в контексте баз данных.
34. Что такое нормализация БД? Цель 1НФ, 2НФ, 3НФ?
35. Что такое транзакция? Как обеспечивается атомарность операций?
36. Зачем нужны миграции схемы БД? Инструменты для управления ими (назовите 1).
37. Опишите разницу между inner join и left outer join.
38. Опишите уровни изоляции транзакций.
39. Перечислите типы NoSQL БД и примеры использования (минимум 3).
40. Как JWT обеспечивает безопасность передачи данных?
41. Что такое сессия в контексте ORM? Как управлять её жизненным циклом?
42. Объясните принцип работы HTTPS. Зачем нужны SSL/TLS.
43. Что такое индексы в БД? Когда их использование нежелательно?
44. Как GraphQL обрабатывает ошибки? Сравните с REST.
45. Зачем в API используют коды состояния HTTP (например, 201, 400, 401)?
46. Что такое Data Transfer Object (DTO)? Зачем использовать отдельные схемы для запроса/ответа API?
47. В чем разница между реляционными и документными БД?
48. Приведите пример использования транзакции с ORM.
49. Как ORM реализует ленивую загрузку (lazy loading) связей?
50. Зачем в GraphQL используют фрагменты (fragments)?
51. Как устроены промисы в JavaScript и как они помогают управлять асинхронностью на клиенте?
52. В чем разница между колбэками, промисами и async/await?
53. Как обрабатывать ошибки в цепочке промисов и зачем нужен метод .catch()?
54. Что такое race-condition в асинхронном коде и как его избежать?
55. Как реализовать таймаут для асинхронной операции на клиенте?
56. Какие виды тестирования фронтенд-приложений вы знаете и в чем их отличия?
57. Как настроить Jest для тестирования React-компонентов и что такое snapshot-тестирование?
58. В чем преимущество React Testing Library перед Enzyme при тестировании UI?
59. Как написать unit-тест для хука useEffect, который делает HTTP-запрос?
60. Как организовать интеграционные тесты для проверки взаимодействия нескольких компонентов?

61. Что такое мокинг (mocking) и стабы (stubs) при тестировании клиентского кода?
62. Как запускать end-to-end тесты с помощью Cypress или Playwright и какие задачи они решают?
63. Какие принципы лежат в основе архитектуры Flux и почему она важна для клиентских приложений?
64. В чем состоят отличия между MVC, MVP и MVVM на фронтенде?
65. Что такое модульная архитектура и как разделять приложение на самостоятельные модули?
66. Как спроектировать архитектуру микрофронтендов и какие есть риски такого подхода?
67. Что такое Dependency Injection в контексте JavaScript-приложений и как его реализовать?
68. В чем заключается суть паттерна «композиция над наследованием» при построении UI-компонентов?
69. Как работают Higher-Order Components в React и в каких сценариях их применять?
70. Что такое render props и как они помогают переиспользовать логику в компонентах?
71. Опишите паттерн «compound components» и приведите пример его использования.
72. Какие техники ленивой загрузки (lazy loading) компонентов и ресурсов вы знаете?
73. Как использовать React.memo и useMemo для оптимизации рендеринга?
74. Что такое виртуальный скролл и когда стоит применять такую оптимизацию?
75. Какие правила WCAG следует учитывать при разработке доступного интерфейса?
76. Как правильно использовать ARIA-атрибуты для управления ролями и состоянием элементов?
77. Как обеспечить клавиатурную навигацию по интерактивным компонентам?
78. Что такое семантический HTML и почему он важен для accessibility и SEO?
79. Как проверять контрастность текста и фона, чтобы соответствовать стандартам доступности?
80. Какие компоненты включает проектирование API (маршрутизация, валидация и т.д.)?
81. Объясните принцип внедрения зависимостей (DI) в серверных приложениях.
82. Как асинхронная обработка улучшает производительность I/O-bound приложений?
83. Опишите шаблон "Репозиторий". Как он абстрагирует доступ к данным?
84. Сравните архитектуры "Монолит" и "Микросервисы": преимущества и недостатки.
85. Какие паттерны используются для управления созданием объектов (фабрики, строители)?
86. Как кэширование ответов API улучшает производительность? Методы реализации.
87. Зачем фронтенду lazy loading? Примеры реализации для ресурсов.
88. Какие стратегии кэширования API вы знаете?
89. Что такое Domain-Driven Design (DDD)?
90. Как мониторить ошибки серверных приложений в продакшене? Инструменты (1 пример).
91. Зачем в микросервисах используют брокеры сообщений (RabbitMQ, Kafka)?
92. Что такое CDN? Как он ускоряет доставку контента?
93. Опишите стратегии деплоя: Canary vs Blue-Green.
94. Какие данные нельзя логировать из соображений безопасности?
95. Объясните паттерн "Circuit Breaker". Где он применяется?
96. Зачем нужен rate limiting в API? Способы реализации.
97. Какие шаги включает рефакторинг монолита в микросервисы?
98. Что такое health-check и как его использовать в оркестраторах (Kubernetes)?
99. Какие преимущества дает Pydantic/схемы для валидации данных?
100. Как Prometheus/Grafana помогают в мониторинге инфраструктуры?
101. Что такое "Инфраструктура как код" (IaC)? Инструменты (1 пример).
102. Как реализовать аутентификацию в API (OAuth2, JWT)?
103. Зачем используют Server-Side Rendering (SSR) для фронтенда?
104. Какие этапы включает CI/CD пайплайн для бэкенд-приложения?

6. Оценочные материалы для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Оценочные материалы (оценочные средства) для проведения текущего контроля и промежуточной аттестации по дисциплине «Full-stack разработка» прилагаются.

6.1. Перечень видов оценочных средств

1. Теоретические вопросы к промежуточному контролю.
2. Компетентностно-ориентированные тесты к промежуточному контролю.
3. Практические задания и задачи к промежуточному контролю.
4. Лабораторные работы.

7. Учебно-методическое и информационное обеспечение дисциплины (модуля)

7.1. Основная литература

1. Саблина, В. А. Основы программирования на JavaScript : учебное пособие / В. А. Саблина, Е. А. Трушина. — Рязань : Рязанский государственный радиотехнический университет, 2022. — 96 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/134868.html> (дата обращения: 15.03.2024). — Режим доступа: для авторизир. пользователей
2. Сакулин, С. А. Основы интернет-технологий: HTML, CSS, JavaScript, XML : учебное пособие / С. А. Сакулин. — Москва : Московский государственный технический университет имени Н.Э. Баумана, 2017. — 112 с. — ISBN 978-5-7038-4724-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/134965.html> (дата обращения: 13.12.2023). — Режим доступа: для авторизир. пользователей
3. Северанс, Ч. Р. Python для всех / Ч. Р. Северанс ; перевод А. В. Снастин. — Москва : ДМК Пресс, 2022. — 262 с. — ISBN 978-5-93700-104-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/126314.html> (дата обращения: 05.12.2022). — Режим доступа: для авторизир. пользователей

7.2. Дополнительная литература

1. П, Т. Многопоточный JavaScript / Хантер Т. П, Б. Инглиш ; перевод А. А. Слинкина. — Москва : ДМК Пресс, 2022. — 188 с. — ISBN 978-5-93700-129-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/126322.html> (дата обращения: 05.12.2022). — Режим доступа: для авторизир. пользователей
2. Уилкс, М. Профессиональная разработка на Python / М. Уилкс ; перевод А. А. Слинкин. — Москва : ДМК Пресс, 2021. — 502 с. — ISBN 978-5-97060-930-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125136.html> (дата обращения: 20.10.2022). — Режим доступа: для авторизир. пользователей

8. Требования к условиям реализации дисциплины (модуля)

8.1. Общесистемные требования

Электронная информационно-образовательная среда ФГБОУ ВО «МТУСИ»

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде МТУСИ из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории МТУСИ, так и вне ее:

<https://mtuci.ru/> - адрес официального сайта университета;

<https://mtuci.ru/education/eios/> - электронная информационно-образовательная среда МТУСИ;

<http://elib.mtuci.ru/catalogue/> - каталог электронной библиотеки МТУСИ.

Электронно-библиотечные системы (электронные библиотеки)

№ п/п	Ссылка на информационный ресурс	Наименование образовательного ресурса	Доступность
1	http://iprbookshop.ru/	ЭБС IPRSmart	Индивидуальный неограниченный доступ из любой точки, в которой имеется доступ к сети Интернет
2	https://e.lanbook.com/	ЭБС ЛАНЬ	
3	https://znanium.com/	ЭБС ZNANIUM	
4	http://book.ru/	ЭБС BOOK.RU	
5	https://urait.ru/	образовательная платформа Юрайт	
6	https://www.elibrary.ru/defaultx.asp	Научная электронная библиотека	

8.2. Материально-техническое и учебно-методическое обеспечение дисциплины (модуля)

МТУСИ располагает материально-техническим обеспечением образовательной деятельности (помещениями и оборудованием) для реализации программы дисциплины (модуля).

Для осуществления образовательного процесса по дисциплине используются учебные аудитории для проведения учебных занятий, оснащенные оборудованием и техническими средствами обучения:

1. Учебная аудитория для проведения лекционных занятий, оснащенная оборудованием и техническими средствами обучения (мультимедийным проектором, экраном, компьютерной техникой), укомплектованная учебной мебелью (парты, доска).

2. Учебные аудитории для проведения лабораторных работ, практических занятий, оснащенные оборудованием и техническими средствами обучения (мультимедийным проектором, экраном, компьютерной техникой), укомплектованные учебной мебелью (парты, доска).

3. Учебная аудитория для проведения консультаций, текущего контроля и промежуточной аттестации, оснащенная компьютерной техникой.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду МТУСИ.

8.3. Необходимый комплект лицензионного программного обеспечения

МТУСИ обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства:

№	Наименование	Лицензия	Вид лицензии
1	Среда разработки Visual Studio Code		

2	Фреймворк FastAPI		
3	Фреймворк React		
4	Платформа контейнеризации Docker		
6	Система контроля версий GitFlic		

8.4. Перечень профессиональных баз данных и информационных справочных систем

Современные профессиональные базы данных:

1. Федеральный портал «Российское образование»: [Электронный ресурс] – Режим доступа: <http://www.edu.ru/> (открытый доступ)
2. Федеральный центр информационно-образовательных ресурсов: [Электронный ресурс] – Режим доступа: <http://fcior.edu.ru/> (открытый доступ)

Информационные справочные системы:

1. Портал Федеральных государственных образовательных стандартов высшего образования: <https://fgosvo.ru>
2. Справочно-правовая система Консультант – Режим доступа: <https://www.consultant.ru/>
3. Справочно-правовая система Гарант – Режим доступа: <https://www.garant.ru/>
4. Портал Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации: <https://digital.gov.ru/ru/documents/>

9. Методические рекомендации для участников образовательного процесса, определяющие особенности освоения учебной дисциплины обучающимся с ограниченными возможностями здоровья (ОВЗ) в условиях инклюзивного образования

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе проведения учебных занятий преподавателю следует стремиться к созданию гибкой и вариативной организационно-методической системы обучения, адекватной образовательным потребностям данной категории обучающихся, которая позволит не только обеспечить преемственность систем общего (инклюзивного) и высшего образования, но и будет способствовать формированию у них компетенций, предусмотренных ФГОС ВО, ускорит темпы профессионального становления, а также будет способствовать их социальной адаптации.

В процессе преподавания учебной дисциплины необходимо способствовать созданию на каждом занятии толерантной социокультурной среды, необходимой для формирования у всех обучающихся гражданской, правовой и профессиональной позиции соучастия, готовности к полноценному общению, сотрудничеству, способности толерантно воспринимать социальные, личностные и культурные различия, в том числе и характерные для обучающихся с ОВЗ.

Посредством совместной, индивидуальной и групповой работы необходимо способствовать формированию у всех обучающихся активной жизненной позиции и развитию способности жить в мире разных людей и идей, а также обеспечить соблюдение обучающимися их прав и свобод и признание права другого человека, в том числе и обучающихся с ОВЗ на такие же права.

В процессе обучения студентов с ОВЗ в обязательном порядке необходимо учитывать рекомендации службы медико-социальной экспертизы или психолого-медико-педагогической комиссии, обусловленные различными стартовыми возможностями данной категории обучающихся (структурой, тяжестью, сложностью дефектов развития).

В процессе овладения обучающимися с ОВЗ компетенциями, предусмотренными рабочей программой дисциплины (РПД), преподавателю следует неукоснительно руководствоваться следующими принципами построения инклюзивного образовательного пространства:

– принцип индивидуального подхода, предполагающий выбор форм, технологий, методов и средств обучения и воспитания с учетом индивидуальных образовательных потребностей каждого

из обучающихся с ОВЗ, учитывающий различные стартовые возможности данной категории обучающихся (структуру, тяжесть, сложность дефектов развития);

- принцип вариативной развивающей среды, который предполагает наличие в процессе проведения учебных занятий и самостоятельной работы обучающихся необходимых развивающих и дидактических пособий, средств обучения, а также организацию безбарьерной среды, с учетом структуры нарушения в развитии (нарушения опорно-двигательного аппарата, зрения, слуха и др.);

- принцип вариативной методической базы, предполагающий возможность и способность использования преподавателем в процессе овладения обучающимися с ОВЗ данной учебной дисциплиной, технологий, методов и средств работы из смежных областей, применение методик и приемов тифло-, сурдо-, олигофренопедагогики, логопедии;

- принцип модульной организации основной образовательной программы, подразумевающий включение в основную образовательную программу модулей из специальных коррекционных программ, способствующих коррекции и реабилитации обучающихся с ОВЗ, а также необходимости учета преподавателем конкретной учебной дисциплины их роли в повышении качества профессиональной подготовки данной категории обучающихся;

- принцип самостоятельной активности обучающихся с ОВЗ, предполагающий обеспечение самостоятельной познавательной активности данной категории обучающихся посредством дополнения раздела РПД «Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)» заданиями, учитывающими различные стартовые возможности данной категории обучающихся (структуру, тяжесть, сложность дефектов развития).

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе проведения учебных занятий преподавателю необходимо осуществлять учет наиболее типичных проявлений психоэмоционального развития, поведенческих и характерологических особенностей, свойственных обучающимся с ОВЗ: повышенной утомляемости, лабильности или инертности эмоциональных реакций, нарушений психомоторной сферы, недостаточное развитие вербальных и невербальных форм коммуникации. В отдельных случаях следует учитывать их склонность к перепадам настроения, аффективность поведения, повышенный уровень тревожности, склонность к проявлениям агрессии, негативизма и т.д.

С целью коррекции и компенсации вышеперечисленных типичных проявлений психоэмоционального развития, поведенческих и характерологических особенностей, свойственных обучающимся с ОВЗ, преподавателю в ходе проведения учебных занятий следует использовать здоровьесберегающие технологии по отношению к данной категории обучающихся, в соответствии с рекомендациями службы медико-социальной экспертизы или психолого-медико-педагогической комиссии.

В группах, в состав которых входят обучающиеся с ОВЗ различной нозологии, при проведении учебных занятий преподавателю следует обратить особое внимание на следующее:

- при обучении студентов с дефектами слуха: на создание безбарьерной среды общения, которая определяется наличием у обучающихся данной категории индивидуальных слуховых аппаратов (или кохлеарных имплантатов), наличие технических средств, обеспечивающих передачу информации на зрительной основе (средств статической и динамической проекции, видеотехники, лазерных дисков, адаптированных компьютеров и т.д.);

- присутствие на занятиях тьютора, владеющего основами разговорной, дактильной и калькирующей жестовой речи;

- при обучении студентов с дефектами зрения: на наличие повышенной освещенности (не менее 1000 люкс) или локального освещения не менее 400-500 люкс, а также наличие оптических средств (лупы, специальных устройств для использования компьютера, телевизионных увеличителей, аудио оборудования для прослушивания «говорящих книг»), наличие комплекта письменных принадлежностей (бумага для письма рельефно-точечным шрифтом Брайля), учебных материалов с использованием шрифта Брайля, звукоусиливающей аппаратуры индивидуального пользования;

– при обучении студентов с нарушениями опорно-двигательной функции: предусматривается применение специальной компьютерной техники с соответствующим программным обеспечением, в том числе специальные возможности операционных систем, таких как экранная клавиатура, альтернативные устройства ввода информации, а также обеспечение безбарьерной архитектурной среды, обеспечивающей доступность маломобильным группам обучающихся с ОВЗ.

В группах, в состав которых входят обучающиеся с ОВЗ, с целью реализации индивидуального подхода, а также принципа индивидуализации и дифференциации, преподавателю следует использовать технологию нелинейной конструкции учебных занятий, предусматривающую одновременное сочетание фронтальных, групповых и индивидуальных форм работы с различными категориями обучающихся, в т.ч. и имеющими ОВЗ.

В процессе учебных занятий в группах, в состав которых входят обучающихся с ОВЗ, преподавателю желательно использовать технологии, направленные на решение дидактических, коммуникативных и компенсаторных задач, посредством использования информационно-коммуникативных технологий дистанционного и online обучения:

– стандартные технологии – например, компьютеры, имеющие встроенные функции настройки для лиц с ограниченными возможностями здоровья;

– доступные форматы данных, известные также как альтернативные форматы – например, доступный HTML, говорящие книги системы DAISY (Digital Accessible Information System – электронная доступная информационная система); а также «низко технологичные» форматы, такие, как система Брайля;

– вспомогательные технологии (ВТ) – это устройства, продукты, оборудование, программное обеспечение или услуги, направленные на усиление, поддержку или улучшение функциональных возможностей обучающихся с ОВЗ, к ним относятся аппараты, устройства для чтения с экрана, клавиатуры со специальными возможностями и т.д.;

– дистанционные образовательные технологии обучения студентов с ОВЗ предоставляют возможность индивидуализации траектории обучения данной категории обучающихся, что подразумевает индивидуализацию содержания, методов, темпа учебной деятельности обучающегося, возможность следить за конкретными действиями обучающегося с ОВЗ при решении конкретных задач, внесения, при необходимости, требуемых корректировок в деятельность обучающегося и преподавателя; данные технологии позволяют эффективно обеспечивать коммуникации обучающегося с ОВЗ не только с преподавателем, но и с другими обучающимися в процессе познавательной деятельности;

– наиболее эффективными формами и методами дистанционного обучения являются персональные сайты преподавателей, обеспечивающих онлайн поддержку профессионального образования обучающихся с ОВЗ, электронные УМК и РПД, учебники на электронных носителях, видеолекции и т.д.

В группах, в состав которых входят обучающиеся с ОВЗ, преподавателю желательно использовать в процессе учебных занятий технологии, направленные на активизацию учебной деятельности, такие как:

– система опережающих заданий, способствующих актуализации знаний и более эффективному восприятию обучающимися с ОВЗ данной учебной дисциплины;

– работа в диадах (парах) сменного состава, включающих обучающегося с ОВЗ и его однокурсников, не имеющих отклонений в психосоматическом развитии;

– опорные конспекты и схемы, позволяющие систематизировать и адаптировать изучаемый материал в соответствии с особенностями развития обучающихся с ОВЗ различной нозологии;

– бланковые методики, с использованием карточек, включающих индивидуальные многоуровневые задания, адаптированные с учетом особенностей развития и образовательных потребностей обучающихся с ОВЗ и их возможностей;

– методика ситуационного обучения (кейс-методы);

– методика совместного оставления проектов как способа достижения дидактической цели через детальную разработку актуальной проблемы, которая должна завершиться вполне реальным,

осязаемым практическим результатом, оформленным тем или иным образом временной инициативной группой разработчиков из числа обучающихся с ОВЗ и их однокурсников, не имеющих отклонений в психосоматическом развитии;

- методики совместного обучения, реализуемые в составе временных инициативных групп, которые создаются в процессе учебных занятий из числа обучающихся с ОВЗ и их однокурсников, не имеющих отклонений в психосоматическом развитии, с целью совместного написания докладов, рефератов, эссе, а также подготовки библиографических обзоров научной и методической литературы, проведения экспериментальных исследований, подготовки презентаций, оформления картотеки нормативно-правовых документов, регламентирующих профессиональную деятельность и т.п.

В процессе учебных занятий в группах, в состав которых входят обучающиеся с ОВЗ, преподавателю желательно использовать технологии, направленные на позитивное стимулирование их учебной деятельности:

- предоставлять реальную возможность для получения в процессе занятий индивидуальной консультативно-методической помощи;

- давать возможность для выбора привлекательного задания, после выполнения обязательного, предупреждать возникновение неконструктивных конфликтов между обучающимися с ОВЗ и их однокурсниками, исключая, таким образом, возможность возникновения у участников образовательного процесса стрессовых ситуаций и негативных реакций.

В группах, в состав которых входят обучающиеся с ОВЗ, в процессе учебных занятий преподавателю желательно использовать технологии, направленные на диагностику уровня и темпов профессионального становления обучающихся с ОВЗ, а также технологии мониторинга степени успешности формирования у них компетенций, предусмотренных ФГОС ВО при изучении данной учебной дисциплины, используя с этой целью специально адаптированные оценочные материалы и формы проведения промежуточной и итоговой аттестации, специальные технические средства, предоставляя обучающимся с ОВЗ дополнительное время для подготовки ответов, привлекая тьютеров.

По результатам текущего мониторинга степени успешности формирования у обучающихся с ОВЗ компетенций, предусмотренных ФГОС ВО в рамках изучения данной учебной дисциплины, при возникновении объективной необходимости, обусловленной оптимизацией темпов профессионального становления конкретного обучающегося с ОВЗ, преподавателю совместно с тьютером и службой психологической поддержки МТУСИ следует разработать адаптированный индивидуальный маршрут овладения данной учебной дисциплиной, адекватный его образовательным потребностям и возможностям.

10. Методические рекомендации обучающимся по освоению дисциплины

Приступая к изучению дисциплины, обучающимся необходимо внимательно ознакомиться с тематическим планом занятий, списком рекомендованной литературы. Следует уяснить последовательность выполнения индивидуальных учебных заданий. Самостоятельная работа обучающихся предполагает работу с научной и учебной литературой, умение создавать тексты. Уровень и глубина усвоения дисциплины зависят от активной и систематической работы на лекциях, изучения рекомендованной литературы, выполнения контрольных письменных заданий.

При изучении дисциплины обучающиеся выполняют следующие задания: изучают рекомендованную научно-практическую и учебную литературу; выполняют задания, предусмотренные для самостоятельной работы.

Основными видами аудиторной работы обучающихся являются лекции, лабораторные и практические занятия. В ходе лекций преподаватель излагает и разъясняет основные, наиболее сложные понятия темы, а также связанные с ней теоретические и практические проблемы, дает рекомендации на лабораторные и практические занятия и указания на самостоятельную работу.

Лабораторные 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 обучающихся по изучаемой дисциплине.

Качество учебной работы обучающихся преподаватель оценивает с использованием фонда оценочных средств дисциплины по организации самостоятельной работы по дисциплине.

УТВЕРЖДАЮ:
Декан факультета _____

« ____ » _____ 20 ____ г.

Лист актуализации рабочей программы дисциплины (модуля)

« _____ »

наименование

Направление: (код, название направления/специальности)

Направленность (профиль): _____

Форма обучения: _____

а) Рабочая программа действует без изменений.

б) В рабочую программу вносятся следующие изменения:

1);

2);

3)

Разработчик (и): _____
(ФИО, ученая степень, ученое звание)

« ____ » _____ 20 ____ г.

Рабочая программа пересмотрена и одобрена на заседании кафедры _____

_____ протокол № _____ от « ____ » _____ 20 ____ г.

Заведующий кафедрой _____