

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Порохня Андрей Алексеевич

Должность: И.о. директора института перспективной инженерии

Дата подписания: 19.06.2026 16:38:42

Уникальный программный ключ:

990bea3aeec848c23bd8699e48288f8d1960c600

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

И.о. директора института

А.А. Порохня

Ф.И.О.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Междисциплинарный проектный практикум-2

Направление подготовки
Направленность (профиль)

09.03.04 Программная инженерия
Инженерия сложных программных
систем

Год начала обучения

2026

Форма обучения

очная

Реализуется в семестре

3,4

Разработано

Заведующая межинститутской базовой
кефелры

(должность разработчика)

Новикова Елена Николаевна

Ф.И.О.

Ставрополь 2026 г.

1. Цель и задачи освоения дисциплины (модуля)

Цель дисциплины: Формирование у студентов комплексных практических навыков сквозной разработки веб-приложений в условиях командной проектной деятельности, интегрирующих знания и умения, полученные при изучении дисциплин «Базы данных», «Технологии разработки программного обеспечения», «UI/UX дизайн», «Разработка Web-приложений», а также развитие профессиональных компетенций программного инженера в области проектирования, реализации, тестирования, развертывания и сопровождения программных продуктов.

Задачи освоения дисциплины:

1. Организовать командную работу с использованием Git и инструментов управления задачами
2. Спроектировать архитектуру веб-приложения на основе паттерна MVC/MVT
3. Реализовать backend-часть на Django или ASP.NET Core с использованием ORM
4. Разработать frontend-часть на React с компонентным подходом
5. Написать модульные и интеграционные тесты для backend и frontend
6. Контейнеризовать приложение с помощью Docker и настроить CI/CD
7. Спроектировать базу данных и выполнить миграции
8. Настроить мониторинг и логирование приложения
9. Интегрировать AI-сервис (OpenAI API) для расширения функциональности
10. Оформить техническую документацию и представить проект на защите

Обеспечить безопасность приложения (аутентификация, защита от уязвимостей)

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Междисциплинарный проектный практикум-2» относится к части, формируемой участниками образовательных отношений.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесённых с планируемыми результатами освоения образовательной программы

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-1 Способен осуществлять управление программными проектами: планировать, контролировать сроки и ресурсы, применять системы контроля версий и инструменты управления задачами	ИД-1пк-1: Применяет методы декомпозиции работ (WBS), оценки трудоёмкости (PERT, Planning Poker) и календарного планирования (диаграмма Ганта, критический путь) для формирования плана проекта	Применяет методы декомпозиции работ (WBS) для разбиения проекта на задачи, использует Planning Poker для оценки трудоёмкости и строит диаграмму Ганта для календарного планирования этапов разработки.
	ИД-2пк-1: Использует системы контроля версий (Git, Git Flow) и платформы совместной разработки (GitLab,	Использует Git для ведения репозитория (ветки feature/develop/main), организует командную работу через pull requests и code review, применяет Git Flow для управления релизами.

	GitHub) для организации командной работы	
	ИД-3пк-1: Применяет инструменты управления задачами (Jira, Trello, YouTrack) для отслеживания прогресса (burndown chart), ведения бэклога и контроля исполнения	Применяет инструменты управления задачами (Jira / Trello) для создания бэклога, отслеживает прогресс выполнения с помощью burndown chart и контролирует исполнение задач по спринтам.
	ИД-4пк-1: Анализирует план-фактные отклонения по срокам, ресурсам и бюджету, управляет изменениями и рисками, разрабатывает корректирующие действия	Анализирует план-фактные отклонения по срокам выполнения лабораторных работ, управляет изменениями в техническом задании и разрабатывает корректирующие действия при возникновении рисков.
ПК-2 Способен выполнять проектирование и разработку архитектуры программных систем среднего и крупного масштаба с применением современных паттернов и методов проектирования	ИД-2пк-2 Документирует архитектуру с использованием нотаций C4 и UML, фиксирует архитектурные решения в формате ADR	Способен спроектировать архитектуру веб-приложения на основе паттерна MVC/MVT, построить ER-диаграмму и описать архитектуру в нотации C4 или UML
ПК-3 Владеет навыками разработки программного обеспечения: способен реализовывать алгоритмы, писать код на современных языках программирования, создавать веб- и desktop-приложения, работать с базами данных	ИД-2пк-3 Создает клиент-серверные, веб- (frontend + backend) и desktop-приложения с графическим интерфейсом, применяя современные фреймворки (React, Spring, Django, WPF) и протоколы (REST, gRPC, WebSocket)	Способен реализовать backend на Django/ASP.NET Core с ORM, frontend на React, интегрировать их через REST API и реализовать систему аутентификации пользователей
ПК-4 Способен проводить	ИД-1пк-4 Применяет методы тест-дизайна	Применяет методы тест-дизайна (классы эквивалентности, граничные значения,

тестирование, отладку и оценивать качество программного обеспечения, применяя инструменты автоматизированного тестирования и методы обеспечения надежности и безопасности	(классы эквивалентности, граничные значения, pairwise) и пишет unit-, интеграционные и E2E-тесты с использованием JUnit, pytest, Jest, Cypress, Selenium	pairwise) и пишет unit-, интеграционные и E2E-тесты с использованием JUnit, pytest, Jest, Cypress, Selenium Способен написать модульные и интеграционные тесты для backend и frontend, настроить автоматический запуск тестов в CI/CD пайплайне
ПК-5 Способен осуществлять сопровождение, настройку, развертывание и эксплуатацию программного обеспечения, в том числе в средах контейнеризации и облачных платформах	ИД-1пк-5 Контейнеризует приложения с помощью Docker (Dockerfile, multi-stage builds) и оркестрирует многоконтейнерные стеки через Docker Compose	Способен контейнеризовать приложение с помощью Docker, настроить оркестрацию контейнеров через Docker Compose и развернуть приложение на облачной платформе
ПК-6 Владеет методами работы с данными и информационными системами: способен проектировать базы данных, выполнять анализ и обработку данных, интегрировать ПО с внешними системами	ИД-1пк-6 Проектирует реляционные схемы (нормализация до 3НФ) и нереляционные структуры данных, управляет миграциями БД (Liquibase, Flyway)	Способен спроектировать нормализованную структуру базы данных (3НФ), выполнить миграции и реализовать сложные запросы с фильтрацией, агрегацией и поиском
ПК-7 Способен применять современные информационные технологии, инструменты и платформы (включая отечественные), использовать DevOps-практики, облачные сервисы и средства автоматизации	ИД-1пк-7 Использует современные IDE, линтеры, форматтеры и инструменты статического анализа (SonarQube) для обеспечения качества кода	Способен настроить CI/CD пайплайн (GitHub Actions/GitLab CI) для автоматического тестирования, линтинга и деплоя приложения
ПК-8 Способен адаптировать и применять методы искусственного	ИД-3пк-8 Интегрирует предобученные модели и API ИИ (Hugging Face,	Способен интегрировать готовый AI-сервис (OpenAI API или аналоги) для расширения функциональности приложения (например, рекомендации, классификация, генерация контента)

интеллекта и машинного обучения для решения профессиональных задач, включая интеграцию готовых моделей в программные продукты	OpenAI, YOLO) в программные продукты, разворачивает их как микросервисы (FastAPI, Flask, Streamlit, Docker) и оптимизирует инференс (кэширование, similarity cache, батчинг, асинхронная обработка через очереди)	
ПК-9 Умеет оформлять техническую документацию, готовить презентации и отчеты, вести деловую коммуникацию на русском и иностранном языках в профессиональной среде	ИД-1пк-9 Оформляет техническую документацию (ТЗ, пояснительная записка, руководство пользователя, API-спецификация) в форматах Markdown, OpenAPI, DocBook в соответствии с ГОСТ	Способен оформить техническую документацию проекта в формате Markdown (README, Wiki, API-документация) и подготовить презентацию для защиты проекта
ПК-10 Готов соблюдать правовые нормы, стандарты и требования информационной безопасности, участвовать в разработке нормативной и технической документации, учитывать законодательство в области ИТ	ИД-2пк-10 Реализует меры защиты информации: аутентификацию (JWT, OAuth2), авторизацию (RBAC), шифрование, защиту от типовых уязвимостей OWASP Top 10 (SQL-инъекции, XSS, CSRF, IDOR) с учётом требований регуляторов (ФСТЭК)	Способен обеспечить безопасное хранение паролей (хеширование с солью), защитить приложение от типовых уязвимостей (SQL-инъекции, XSS, CSRF) и настроить ролевую модель доступа (RBAC)

4. Объем учебной дисциплины и формы контроля *

Объем занятий: всего: 6 з.е. 216 ак.ч.	ОФО, в ак. часах
Контактная работа:	54
Лекции/из них практическая подготовка	
Лабораторных работ/из них практическая подготовка	54
Практических занятий/из них практическая	

подготовка	
Самостоятельная работа	162
Формы контроля	
Экзамен	
Зачет	
Зачет с оценкой	±
Расчетно-графические работы	
Курсовые работа	
Контрольные работы	

* Дисциплина (модуль) предусматривает применение электронного обучения, дистанционных образовательных технологий

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием количества часов и видов занятий

№	Раздел (тема) дисциплины и краткое содержание	Формируемые компетенции, индикаторы	очная форма			
			Контактная работа обучающихся с преподавателем /из них в форме практической подготовки, часов			Самостоятельная работа, часов
			Лекции	Практические занятия	Лабораторные работы	
1	Тема №1. Анализ требований и проектирование архитектуры: Анализ пользовательских ролей (Аноним, Пользователь, Админ) и сценариев использования. Проектирование моделей данных (User, Project, Tag, Task, TaskTag). Построение ER-диаграммы и описание реализации паттерна MVC/MVT в проекте. Создание структуры проекта в выбранной технологии (Django + React или ASP.NET + React). Оформление технического задания и архитектурной документации.				2	10
2	Тема №2. Модели данных, миграции и начальное наполнение: Реализация классов моделей с правильными типами данных и атрибутами валидации. Настройка связей между моделями (One-to-Many, Many-to-Many). Создание и применение миграций для создания схемы базы данных. Реализация автоматического наполнения базы тестовыми данными (Seed Data) для отладки и демонстрации.				2	10
3	Тема №3. Реализация бизнес-логики. Контроллеры и сервисы: Реализация CRUD-операций для моделей Task, Project и Tag. Создание контроллеров/ViewSets для обработки HTTP-запросов. Внедрение сервисного слоя для отделения бизнес-логики от контроллеров. Реализация фильтрации, поиска и пагинации списков. Настройка валидации входных данных на стороне сервера.				2	10

4	Тема №4. Аутентификация и авторизация пользователей: Реализация системы регистрации новых пользователей и аутентификации (JWT-токены). Настройка защиты API-маршрутов (только для авторизованных пользователей). Реализация привязки данных к пользователю (каждый видит только свои задачи). Добавление ролевой модели (пользователь, администратор). Интеграция аутентификации с React-фронтом.				2	10
5	Тема №5. Документирование API и тестирование бэкенда: Настройка автоматической генерации документации API с использованием Swagger/OpenAPI. Написание модульных тестов для сервисного слоя (pytest / xUnit). Написание интеграционных тестов для API эндпоинтов. Настройка покрытия кода тестами (минимальный порог 70%). Реализация валидации входных данных и обработки ошибок.				2	10
6	Тема №6. Продвинутая бизнес-логика. Статистика и отчеты: Реализация подсчета статистики по задачам (количество по статусам, просроченные задачи, продуктивность по дням). Реализация экспорта данных в форматы CSV, JSON и PDF. Реализация полнотекстового поиска задач. Добавление иерархии задач (подзадачи). Реализация системы напоминаний о дедлайнах.				2	10
7	Тема №7. Введение в React. Компонентный подход: Создание React-проекта с использованием Vite. Настройка маршрутизации с React Router. Создание основных компонентов страниц (Dashboard, TaskList, TaskForm, LoginPage, RegisterPage). Реализация переиспользуемых компонентов (TaskCard, TaskFilter, Navbar). Настройка HTTP-клиента (axios) для связи с бэкендом				2	10

8	Тема №8. Управление состоянием и интеграция с API: Настройка Redux Toolkit для управления глобальным состоянием приложения. Реализация хранилища (store) и слайсов (slices) для задач, проектов, тегов и аутентификации. Реализация CRUD-операций на фронтенде с вызовом API через createAsyncThunk. Реализация фильтрации и поиска задач. Добавление отображения уведомлений (toast notifications).				2	10
9	Тема №9. Презентация и защита проекта (1 семестр): Подготовка презентации проекта (10-15 слайдов). Подготовка технической документации (README, Swagger, архитектурные диаграммы). Демонстрация работающего приложения (бэкенд + базовый фронтенд). Защита принятых архитектурных решений. Планирование работ на второй семестр с использованием инструментов управления задачами.				2	10
	Итого за 3 семестр				18	90
10	Тема №10. Продвинутая работа с формами и валидация: Реализация формы создания/редактирования задачи с комплексной валидацией с использованием Formik и Yup. Реализация динамического выбора тегов (мультиселект). Реализация возможности создания новых тегов «на лету» непосредственно из формы задачи. Добавление выбора проекта из выпадающего списка.				2	4
11	Тема №11. Календарь и временная шкала: Реализация компонента календаря (FullCalendar / react-big-calendar) для отображения задач с дедлайнами. Реализация возможности drag-and-drop для изменения дат задач. Реализация представления «Kanban board» (доска задач по статусам) с перетаскиванием карточек между колонками. Добавление представления «Timeline» (временная шкала).				2	4
12	Тема №12. Реальное время. Django Channels и WebSockets: Настройка WebSocket соединения с использованием Django Channels. Реализация real-time уведомлений о приближающихся дедлайнах. Реализация обновления списка задач в реальном времени при изменениях. Добавление браузерных push-уведомлений (Notification API). Реализация отображения онлайн-статуса пользователей.				2	4

13	Тема №13. Совместная работа. Шаринг задач и проектов: Реализация системы приглашений пользователей в проекты (шаринг). Добавление ролей в проекте (владелец, редактор, читатель). Реализация комментариев к задачам с возможностью вложенных ответов. Добавление системы логов действий (история изменений задачи). Реализация уведомлений о назначении задачи.				2	4
14	Тема №14. Экспорт, импорт и отчеты в PDF: Реализация экспорта задач в CSV и JSON форматы. Реализация генерации PDF-отчетов с использованием reportlab. Реализация импорта задач из CSV/JSON файлов. Добавление возможности выбора полей для экспорта. Реализация генерации и отправки еженедельного отчета по email.				2	4
15	Тема №15. Интеграция с внешними API. Погода и календари: Интеграция API погоды (OpenWeatherMap) для отображения погоды на главном экране. Интеграция Google Calendar API для импорта событий. Реализация синхронизации задач с внешним календарем (экспорт/импорт). Добавление отображения мотивационных цитат (API Quotable). Реализация кэширования запросов к внешним API.				2	4
16	Тема №16. Контейнеризация с Docker: Создание Dockerfile для backend-приложения (Django). Создание Dockerfile для frontend-приложения (React) с многостадийной сборкой. Настройка docker-compose для всех сервисов (backend, frontend, PostgreSQL, Redis, Nginx). Настройка volumes для персистентности данных базы данных. Настройка переменных окружения через .env файл.				2	4
17	Тема №17. CI/CD пайплайн (GitHub Actions / GitLab CI): Настройка CI/CD пайплайна для автоматического тестирования при пуше в репозиторий. Добавление этапов линтинга и статического анализа кода. Настройка автоматической сборки Docker-образов. Настройка автоматического деплоя на тестовый сервер (staging). Настройка уведомлений о статусе сборки (Telegram).				2	4

18	Тема №18. Мониторинг и логирование: Настройка сбора метрик приложения с использованием Prometheus. Настройка визуализации метрик в Grafana (дашборды для RPS, latency, ошибок). Настройка централизованного сбора логов с использованием Loki и Promtail. Настройка алертов в Telegram/Slack при отклонении метрик. Создание дашбордов для мониторинга ключевых метрик приложения.				2	4
19	Тема №19. PWA (Progressive Web App): Создание манифеста (manifest.json) для PWA. Настройка Service Worker для кэширования статических ресурсов. Реализация офлайн-режима с кэшированием данных в IndexedDB. Добавление возможности «установки» приложения на устройство. Настройка push-уведомлений через сервис-воркер.				2	4
20	Тема №20. Интеграция AI/ML (Искусственный интеллект): Интеграция OpenAI API для генерации описания задачи и рекомендаций. Реализация автоматической рекомендации приоритета задачи на основе текста. Реализация AI-помощника для планирования дня. Добавление функции генерации мотивационных цитат. Реализация кэширования результатов AI-запросов с использованием Redis.				2	4
21	Тема №21. Оптимизация производительности: Проведение аудита производительности с использованием Lighthouse и Django Debug Toolbar. Оптимизация запросов к БД (устранение N+1, добавление индексов). Оптимизация фронтенда (code splitting, lazy loading, мемоизация компонентов). Внедрение кэширования на бэкенде (Redis). Оптимизация загрузки бандла (анализ и уменьшение размера).				2	4
22	Тема №22. Административная панель и дашборды: Создание кастомной административной панели с использованием Django Admin / React. Реализация управления пользователями (блокировка, изменение ролей). Добавление просмотра системных логов активности. Реализация дашборда с бизнес-метриками для администратора. Добавление управления системными настройками.				2	4

23	Тема №23. Развертывание в облако (Yandex Cloud / AWS): Выбор и настройка облачного провайдера. Настройка виртуальной машины для бэкенда. Настройка managed базы данных PostgreSQL. Настройка S3-совместимого хранилища для медиафайлов. Настройка домена и SSL-сертификата (HTTPS). Развертывание приложения в production.				2	4
24	Тема №24. Безопасность. Защита от уязвимостей: Проведение статического анализа безопасности (SAST) с использованием Bandit / SonarQube. Проведение динамического анализа безопасности (DAST) с OWASP ZAP. Устранение SQL-инъекций (использование ORM/параметризованных запросов). Устранение XSS-уязвимостей (санитизация вывода). Настройка CSRF-защиты, CORS политик и rate limiting.				2	4
25	Тема №25. Мобильная адаптация и UI/UX улучшения: Реализация адаптивного дизайна (mobile-first подход). Добавление мобильного меню (бургер-меню) для навигации. Реализация темной темы (dark mode) с сохранением выбора пользователя. Оптимизация touch-событий для мобильных устройств (свайпы, жесты). Проведение UX-тестирования и улучшение интерфейса.				2	4
26	Тема №26. Итоговое тестирование и приемка: Написание E2E-тестов для ключевых пользовательских сценариев (Cypress / Playwright). Проведение нагрузочного тестирования с использованием k6. Составление плана тестирования и отчета о тестировании. Проведение smoke-тестирования после деплоя. Проведение user acceptance testing (UAT) с заказчиком.				2	4
27	Тема №27. Финальная защита проекта: Подготовка финальной презентации проекта (20-25 слайдов). Демонстрация работающего приложения в облаке. Презентация архитектуры и технологических решений. Демонстрация CI/CD пайплайна и мониторинга. Ответы на вопросы комиссии. Подготовка итогового отчета по проекту.				2	4
	Экзамен					
	Итого за 4 семестр				36	72
	ИТОГО				54	162

6. Фонд оценочных средств по дисциплине (модулю)

Фонд оценочных средств (ФОС) по дисциплине (модулю) базируется на перечне осваиваемых компетенций с указанием индикаторов. ФОС обеспечивает объективный контроль достижения запланированных результатов обучения. ФОС включает в себя:

- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;

- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций (включаются в методические указания по тем видам работ, которые предусмотрены учебным планом и предусматривают оценку сформированности компетенций);

- типовые оценочные средства, необходимые для оценки знаний, умений и уровня сформированности компетенций.

ФОС является приложением к данной программе дисциплины.

7. Методические указания для обучающихся по освоению дисциплины

Приступая к работе, каждый студент должен принимать во внимание следующие положения.

Дисциплина построена по тематическому принципу, каждая тема представляет собой логически завершённый раздел.

Практические занятия проводятся с целью закрепления усвоенной информации, приобретения навыков ее применения при решении практических задач в соответствующей предметной области.

Самостоятельная работа студентов направлена на самостоятельное изучение дополнительного материала, подготовку к лабораторным занятиям, а также выполнения всех видов самостоятельной работы.

Для успешного освоения дисциплины, необходимо выполнить все виды самостоятельной работы, используя рекомендуемые источники информации.

8. Учебно-методическое и информационное обеспечение дисциплины

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

8.1.1. Перечень основной литературы:

1. Холоватюк, А. Django 4 : практика создания веб-сайтов / А. Холоватюк. — Санкт-Петербург : БХВ-Петербург, 2023. — 432 с. — ISBN 978-5-9775-6832-8.
2. Фримен, А. ASP.NET Core 8 для профессионалов / А. Фримен ; перевод с английского. — Москва : ДМК Пресс, 2024. — 848 с. — ISBN 978-5-93700-289-9.
3. Бэнкс, А. React в действии / А. Бэнкс, Е. Порселло ; перевод с английского. — Москва : ДМК Пресс, 2021. — 384 с. — ISBN 978-5-97060-912-5.
4. Мартин, Р. Чистая архитектура / Р. Мартин ; перевод с английского. — Санкт-Петербург : Питер, 2019. — 352 с. — ISBN 978-5-4461-1211-7.
5. Дейт, К. Дж. Введение в системы баз данных / К. Дж. Дейт ; перевод с английского. — 8-е изд. — Москва : Вильямс, 2016. — 1328 с. — ISBN 978-5-8459-2087-6.

8.1.2. Перечень дополнительной литературы:

1. Фаулер, М. Шаблоны корпоративных приложений / М. Фаулер ; перевод с английского. — Москва : Вильямс, 2016. — 544 с. — ISBN 978-5-8459-1893-4.
2. Лутц, М. Программирование на Python для начинающих / М. Лутц ; перевод с английского. — 5-е изд. — Санкт-Петербург : БХВ-Петербург, 2020. — 1088 с. — ISBN 978-5-9775-5351-5.

3. Прайс, М. С# 12 и .NET 8 для профессионалов / М. Прайс ; перевод с английского. — Москва : ДМК Пресс, 2024. — 944 с. — ISBN 978-5-93700-290-5.
4. Рогов, Е. PostgreSQL : руководство администратора / Е. Рогов. — 3-е изд. — Санкт-Петербург : БХВ-Петербург, 2022. — 736 с. — ISBN 978-5-9775-6841-0.
5. Кейн, Ш. Docker в действии / Ш. Кейн, К. Мэттьюз ; перевод с английского. — Москва : ДМК Пресс, 2019. — 384 с. — ISBN 978-5-97060-757-2.
6. Тернбулл, Д. Prometheus : мониторинг и алертинг / Д. Тернбулл ; перевод с английского. — Москва : ДМК Пресс, 2020. — 544 с. — ISBN 978-5-97060-823-4.
7. Стойков, С. OWASP Top 10 : защита веб-приложений / С. Стойков. — Москва : ДМК Пресс, 2021. — 312 с. — ISBN 978-5-97060-904-0.
8. Чаконарджу, С. Git : практическое руководство / С. Чаконарджу ; перевод с английского. — Санкт-Петербург : Питер, 2022. — 320 с. — ISBN 978-5-4461-1772-3.
9. Хебль, М. TypeScript : полное руководство / М. Хебль ; перевод с английского. — Санкт-Петербург : Питер, 2022. — 576 с. — ISBN 978-5-4461-1724-2.
10. Рогов, Е. PostgreSQL изнутри / Е. Рогов. — 2-е изд. — Санкт-Петербург : БХВ-Петербург, 2024. — 560 с. — ISBN 978-5-9775-6934-9.

8.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине

8.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. djbook.ru - Документация Django на русском языке (перевод официальной документации). - URL: <https://djbook.ru/>
2. metanit.com - METANIT : руководство по ASP.NET Core. - URL: <https://metanit.com/sharp/aspnetcore/>
3. metanit.com - METANIT : руководство по Entity Framework Core. - URL: <https://metanit.com/sharp/efcore/>
4. ru.react.dev - Документация React на русском языке. - URL: <https://ru.react.dev/>
5. postgrespro.ru - PostgreSQL : документация на русском языке. - URL: <https://postgrespro.ru/docs/postgresql>
6. metanit.com - METANIT : руководство по JavaScript. - URL: <https://metanit.com/web/javascript/>
7. metanit.com - METANIT : руководство по TypeScript. - URL: <https://metanit.com/web/typescript/>
8. redux-toolkit.js.org - Redux Toolkit Documentation. - URL: <https://redux-toolkit.js.org/>
9. reactrouter.com - React Router Documentation. - URL: <https://reactrouter.com/>
10. docs.docker.com - Docker Documentation. - URL: <https://docs.docker.com/>
11. docs.github.com - GitHub Actions Documentation. - URL: <https://docs.github.com/en/actions>
12. prometheus.io - Prometheus Documentation. - URL: <https://prometheus.io/docs/>
13. grafana.com - Grafana Documentation. - URL: <https://grafana.com/docs/>
14. platform.openai.com - OpenAI API Documentation. - URL: <https://platform.openai.com/docs>

15. habr.com - Habr : профессиональное сообщество (статьи по веб-разработке).
- URL: <https://habr.com/> (дата обращения: 24.05.2025).
16. tproger.ru - Tproger : IT-портал (новости и полезные материалы). -
URL: <https://tproger.ru/>
17. stepik.org - Stepik : онлайн-курсы по Python, Django, React. -
URL: <https://stepik.org/>
18. developer.mozilla.org - MDN Web Docs (русскоязычная версия). -
URL: <https://developer.mozilla.org/ru/>
19. caniuse.com - Can I Use : совместимость веб-технологий. -
URL: <https://caniuse.com/>
20. docs.gitlab.com - GitLab Documentation. - URL: <https://docs.gitlab.com/>

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

При чтении лекций используется компьютерная техника, демонстрации презентационных мультимедийных материалов. На семинарских и практических занятиях студенты представляют презентации, подготовленные ими в часы самостоятельной работы.

Информационные справочные системы:

Информационно-справочные и информационно-правовые системы, используемые при изучении дисциплины:

1. Электронная информационно-образовательная среда (ЭИОС) [вуза]. - [Место доступа] : внутренний портал вуза.
2. Электронно-библиотечная система (ЭБС) [наименование, например «Лань», «Юрайт», «Znaniium»]. - [Место доступа] : внутренний портал вуза.
3. Справочная система «ГОСТЭКСПЕРТ». - [Место доступа] : внутренний портал вуза.
4. КонсультантПлюс - справочная правовая система. - [Место доступа] : внутренний портал вуза.
5. Официальный портал ФСТЭК России. - URL: <https://fstec.ru/> (дата обращения: 24.05.2025).
6. MDN Web Docs (русскоязычная версия). -
URL: <https://developer.mozilla.org/ru/> (дата обращения: 24.05.2025).
7. Can I Use — совместимость веб-технологий. - URL: <https://caniuse.com/> (дата обращения: 24.05.2025).
8. OpenAPI Specification (русский перевод). -
URL: <https://swagger.io/specification/> (дата обращения: 24.05.2025).

Программное обеспечение:

1. Visual Studio Code - Редактор кода (универсальный). - Лицензия: MIT License (свободное распространение).
2. Visual Studio Community - IDE для разработки на C# / .NET. - Лицензия: Free (для образовательных учреждений).
3. PyCharm Community Edition - IDE для разработки на Python. - Лицензия: Apache License 2.0 (свободное распространение).
4. Git - Система контроля версий. - Лицензия: GNU General Public License v2 (свободное распространение).
5. Docker Desktop (Community Edition) - Контейнеризация приложений. - Лицензия:

- Free (для образования, индивидуального использования и открытых проектов).
6. Node.js - Среда выполнения JavaScript. - Лицензия: MIT License (свободное распространение).
 7. Python - Среда выполнения Python. - Лицензия: Python Software Foundation License (свободное распространение).
 8. .NET SDK - Среда выполнения .NET. - Лицензия: MIT License (свободное распространение).
 9. PostgreSQL - Система управления базами данных. - Лицензия: PostgreSQL License (свободное распространение).
 10. DBeaver / pgAdmin - Визуальный клиент для работы с базами данных. - Лицензия: Apache License 2.0 / PostgreSQL License (свободное распространение).
 11. Postman / Insomnia - Тестирование API. - Лицензия: Free / MIT License (свободное распространение).
 12. OWASP ZAP - Сканер безопасности веб-приложений. - Лицензия: Apache License 2.0 (свободное распространение).
 13. Prometheus - Система сбора метрик. - Лицензия: Apache License 2.0 (свободное распространение).
 14. Grafana - Визуализация метрик. - Лицензия: GNU Affero General Public License v3 (свободное распространение).
 15. Loki - Централизованное логирование. - Лицензия: GNU Affero General Public License v3 (свободное распространение).
 16. k6 - Нагрузочное тестирование. - Лицензия: GNU Affero General Public License v3 (свободное распространение).
 17. Cypress - E2E-тестирование. - Лицензия: MIT License (свободное распространение).
 18. Playwright - E2E-тестирование. - Лицензия: Apache License 2.0 (свободное распространение).
 19. GitLab / GitHub - Платформы совместной разработки. - Лицензия: Free (для образования) / MIT License.
 20. GitFlic - Российская платформа для совместной разработки (альтернатива GitHub). - Лицензия: Free (для образования).
 21. Jira / Trello / YouTrack - Инструменты управления задачами. - Лицензия: Free (для образования).
 22. VirtualBox / VMware - Виртуализация (при необходимости). - Лицензия: GNU General Public License v2 / Free for education.

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

Лекционные занятия	Учебная аудитория для проведения учебных занятий, оснащенная мультимедийным оборудованием и техническими средствами обучения.
Лабораторные работы	Учебная аудитория для проведения учебных занятий, оснащенная мультимедийным оборудованием и техническими средствами обучения.
Самостоятельная работа	Помещение для самостоятельной работы обучающихся оснащенное компьютерной техникой с возможностью подключения к сети "Интернет" и возможностью доступа к электронной информационно-образовательной среде университета

11. Особенности освоения дисциплины (модуля) лицами с ограниченными возможностями здоровья

Обучающимся с ограниченными возможностями здоровья предоставляются специальные учебники, учебные пособия и дидактические материалы, специальные технические средства обучения коллективного и индивидуального пользования, услуги ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

Освоение дисциплины (модуля) обучающимися с ограниченными возможностями здоровья может быть организовано совместно с другими обучающимися, а также в отдельных группах.

Освоение дисциплины (модуля) обучающимися с ограниченными возможностями здоровья осуществляется с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья.

В целях доступности получения высшего образования по образовательной программе лицами с ограниченными возможностями здоровья при освоении дисциплины (модуля) обеспечивается:

1) для лиц с ограниченными возможностями здоровья по зрению:

- присутствие ассистента, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе, записывая под диктовку),
- письменные задания, а также инструкции о порядке их выполнения оформляются увеличенным шрифтом,
- специальные учебники, учебные пособия и дидактические материалы (имеющие крупный шрифт или аудиофайлы),
- индивидуальное равномерное освещение не менее 300 люкс,
- при необходимости студенту для выполнения задания предоставляется увеличивающее устройство;

2) для лиц с ограниченными возможностями здоровья по слуху:

- присутствие ассистента, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе, записывая под диктовку),
- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающемуся предоставляется звукоусиливающая аппаратура индивидуального пользования;
- обеспечивается надлежащими звуковыми средствами воспроизведения информации;

3) для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата (в том числе с тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;
- по желанию студента задания могут выполняться в устной форме.

12. Особенности реализации дисциплины с применением дистанционных образовательных технологий и электронного обучения

Согласно части 1 статьи 16 Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации» под *электронным обучением* понимается организация образовательной деятельности с применением содержащейся в базах данных и используемой при реализации образовательных программ информации и обеспечивающих ее обработку информационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу по линиям связи указанной информации, взаимодействие обучающихся и педагогических работников. Под *дистанционными образовательными технологиями* понимаются образовательные технологии, реализуемые в основном с применением информационно-

телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников.

Реализация дисциплины может быть осуществлена с применением дистанционных образовательных технологий и электронного обучения полностью или частично. Компоненты УМК дисциплины (рабочая программа дисциплины, оценочные и методические материалы, формы аттестации), реализуемой с применением дистанционных образовательных технологий и электронного обучения, содержат указание на их использование.

При организации образовательной деятельности с применением дистанционных образовательных технологий и электронного обучения могут предусматриваться асинхронный и синхронный способы осуществления взаимодействия участников образовательных отношений посредством информационно-телекоммуникационной сети «Интернет».

При применении дистанционных образовательных технологий и электронного обучения в расписании по дисциплине указываются: способы осуществления взаимодействия участников образовательных отношений посредством информационно-телекоммуникационной сети «Интернет» (ВКС-видеоконференцсвязь, ЭТ – электронное тестирование); ссылки на электронную информационно-образовательную среду СКФУ, на образовательные платформы и ресурсы иных организаций, к которым предоставляется открытый доступ через информационно-телекоммуникационную сеть «Интернет»; для синхронного обучения - время проведения онлайн-занятий и преподаватели; для асинхронного обучения - авторы онлайн-курсов.

При организации промежуточной аттестации с применением дистанционных образовательных технологий и электронного обучения используются Методические рекомендации по применению технических средств, обеспечивающих объективность результатов при проведении промежуточной и государственной итоговой аттестации по образовательным программам высшего образования - программам бакалавриата, программам специалитета и программам магистратуры с применением дистанционных образовательных технологий (Письмо Минобрнауки России от 07.12.2020 г. № МН-19/1573-АН "О направлении методических рекомендаций").

Реализация дисциплины с применением электронного обучения и дистанционных образовательных технологий осуществляется с использованием электронной информационно-образовательной среды СКФУ, к которой обеспечен доступ обучающихся через информационно-телекоммуникационную сеть «Интернет», или с использованием ресурсов иных организаций, в том числе платформ, предоставляющих сервисы для проведения видеоконференций, онлайн-встреч и дистанционного обучения (МТС-Линк), а также с использованием возможностей социальных сетей для осуществления коммуникации обучающихся и преподавателей.

Учебно-методическое обеспечение дисциплины, реализуемой с применением электронного обучения и дистанционных образовательных технологий, включает представленные в электронном виде рабочую программу, учебно-методические пособия или курс лекций, методические указания к выполнению различных видов учебной деятельности обучающихся, предусмотренных дисциплиной, и прочие учебно-методические материалы, размещенные в информационно-образовательной среде СКФУ.