

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
"Казанский (Приволжский) федеральный университет"
Институт информационных технологий и интеллектуальных систем



подписано электронно-цифровой подписью

Программа дисциплины

Фронтенд-разработка

Направление подготовки: 09.03.04 - Программная инженерия

Профиль подготовки: Разработка цифровых продуктов (с применением электронного обучения и дистанционных образовательных технологий) (Digital Product Development (with the use of e-learning and distance education technologies))

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО
2. Место дисциплины (модуля) в структуре ОПОП ВО
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
 - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
 - 4.2. Содержание дисциплины (модуля)
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
7. Перечень литературы, необходимой для освоения дисциплины (модуля)
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины (модуля) к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
13. Приложение №1. Фонд оценочных средств
14. Приложение №2. Перечень литературы, необходимой для освоения дисциплины (модуля)
15. Приложение №3. Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Программу дисциплины разработал(а)(и): ведущий специалист Якушенкова А.Д. (Институт информационных технологий и интеллектуальных систем, КФУ), ArDYakushenkova@kpfu.ru

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

Шифр компетенции	Расшифровка приобретаемой компетенции
ПК-4	Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения, способность формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

- Структуру и семантику HTML, основные и продвинутое CSS-свойства, принципы каскадности и адаптивной верстки.
- Основы и продвинутое возможности JavaScript: типы данных, функции, объекты, замыкания, асинхронность, работа с DOM.
- Концепции и синтаксис TypeScript, включая типизацию, интерфейсы, классы и основные принципы ООП.
- Основы React: JSX, компоненты, состояние и свойства, жизненный цикл компонентов.
- Принципы роутинга и авторизации в одностраничных приложениях на React.
- Методы и инструменты тестирования React-приложений, включая юнит- и интеграционные тесты.
- Использование системы контроля версий GIT

Должен уметь:

- Верстать адаптивные и кроссбраузерные страницы с использованием HTML и CSS, работать с макетами.
- Писать чистый, структурированный и эффективный JavaScript-код, использовать современные возможности языка.
- Использовать TypeScript для типизации кода и реализации ООП-подходов в проекте.
- Создавать и управлять React-компонентами, реализовывать обработку событий и управление состоянием.
- Настраивать маршрутизацию (роутинг) и реализовывать механизмы авторизации в React-приложениях.
- Писать и запускать тесты для компонентов и логики React-приложений, использовать инструменты тестирования.
- Создать репозиторий GIT и использовать его для ведения истории разработки

Должен владеть:

- Навыками создания полноценных фронтенд-приложений с использованием современных технологий и лучших практик.
- Способностью проектировать архитектуру приложений с использованием ООП и типизации в TypeScript.
- Умением интегрировать React с роутингом и системой авторизации для создания безопасных и удобных SPA.
- Навыками обеспечения качества кода через написание и поддержку тестов.
- Способностью самостоятельно решать технические задачи и применять новые знания на практике.

Должен демонстрировать способность и готовность:

- Самостоятельно разрабатывать адаптивные, функциональные и поддерживаемые веб-интерфейсы.
- Анализировать и исправлять ошибки в коде, оптимизировать производительность приложений.
- Работать с командой, используя современные инструменты разработки и контроля версий.
- Быстро осваивать новые технологии и применять их в рамках фронтенд-разработки.
- Принимать участие в проектировании архитектуры и выборе технических решений для веб-приложений.
- Использовать систему контроля версий GIT

2. Место дисциплины (модуля) в структуре ОПОП ВО

Данная дисциплина (модуль) включена в раздел "Б1.В.ДВ.02.03 Дисциплины (модули)" основной профессиональной образовательной программы 09.03.04 "Программная инженерия (Разработка цифровых продуктов (с применением электронного обучения и дистанционных образовательных технологий) (Digital Product Development (with the use of e-learning and distance education technologies)))" и относится к дисциплинам по выбору части ОПОП ВО, формируемой участниками образовательных отношений.

Осваивается на 2 курсе в 4 семестре.

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 3 зачетных(ые) единиц(ы) на 108 часа(ов).

Контактная работа - 72 часа(ов), в том числе лекции - 0 часа(ов), практические занятия - 0 часа(ов), лабораторные работы - 72 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 36 часа(ов).

Контроль (зачёт / экзамен) - 0 часа(ов).

Форма промежуточного контроля дисциплины: зачет с оценкой в 4 семестре.

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

4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)

N	Разделы дисциплины / модуля	Се-местр	Виды и часы контактной работы, их трудоемкость (в часах)						Само-стоя-тельная ра-бота
			Лекции, всего	Лекции в эл. форме	Практи-ческие занятия, всего	Практи-ческие в эл. форме	Лабора-торные работы, всего	Лабора-торные в эл. форме	
1.	Тема 1. Продвинутый JavaScript	4	0	0	0	0	8	0	10
2.	Тема 2. TypeScript и ООП	4	0	0	0	0	8	0	11
3.	Тема 3. TypeScript и ООП. Продолжение	4	0	0	0	0	16	0	10
4.	Тема 4. ООП в интерфейсах	4	0	0	0	0	16	0	10
5.	Тема 5. Основы React	4	0	0	0	0	8	0	11
6.	Тема 6. Роутинг и авторизация	4	0	0	0	0	8	0	10
7.	Тема 7. Тестирование React-приложений	4	0	0	0	0	8	0	10
	Итого		0	0	0	0	72	0	72

4.2 Содержание дисциплины (модуля)

Тема 1. Продвинутый JavaScript

После прохождения тем "Продвинутый JavaScript. Асинхронность" и "Работа с API" студент должен хорошо понимать природу асинхронных операций в JavaScript и уметь эффективно работать с ними. Он должен знать, что асинхронность позволяет выполнять задачи, не блокируя основной поток выполнения, используя такие механизмы, как колбэки, таймеры и промисы. Студент должен понимать работу Event Loop - ключевого механизма обработки очереди событий и асинхронных вызовов в JavaScript. Он должен уметь создавать и использовать промисы для управления асинхронным кодом, а также применять конструкции `async/await` для упрощения работы с промисами. В рамках работы с API студент должен знать основные принципы протокола HTTP, уметь формировать и отправлять запросы к серверу, обрабатывать ответы и разбирать данные в формате JSON. Он должен владеть навыками использования инструментов разработчика, таких как вкладка Network, для отладки сетевых запросов. Кроме того, студент должен понимать особенности стандартной отправки форм и уметь интегрировать взаимодействие с внешними сервисами через API в свои приложения. Эти знания позволят создавать динамичные веб-приложения, которые эффективно взаимодействуют с сервером, обеспечивая современный пользовательский опыт.

Тема 2. TypeScript и ООП

После изучения тем "Введение в TypeScript", "Типизация кода" и "Введение в ООП" студент должен понимать ключевые отличия TypeScript от JavaScript, в частности важность статической типизации для повышения надежности и читаемости кода. Он должен знать, что TypeScript позволяет создавать самодокументируемый код, облегчая поддержку и масштабирование проектов, а также уметь настраивать и интегрировать TypeScript в существующие проекты с использованием соответствующего инструментария. Студент должен уметь работать с базовыми и сложными типами данных, типизировать функции, массивы, объекты и создавать собственные типы для более точного описания структуры данных. В области объектно-ориентированного программирования студент должен знать основные парадигмы и стили программирования, уметь создавать и работать с объектами, понимать контекст выполнения и особенности привязки this. Он должен владеть синтаксисом классов в JavaScript и TypeScript, уметь типизировать поля и методы, а также использовать интерфейсы для описания контрактов классов. Кроме того, студент должен понимать и применять основные принципы ООП - инкапсуляцию, наследование и полиморфизм - и осознавать, как они взаимодействуют для создания гибких и расширяемых программных систем. Эти знания формируют фундамент для разработки масштабируемых и поддерживаемых приложений с использованием современных подходов.

Тема 3. TypeScript и ООП. Продолжение

После изучения основ TypeScript и архитектуры приложений студент получает прочную базу для разработки современных фронтенд-приложений с использованием строгой типизации и продуманных архитектурных решений. Он научится правильно типизировать DOM-элементы и события, что значительно повысит надежность и удобство работы с пользовательским интерфейсом. Понимание дженериков и условных типов позволит создавать более универсальный и гибкий код, а также использовать и создавать собственные библиотеки, что улучшит масштабируемость проектов. Изучение принципов SOLID, DRY, KISS и других методик проектирования помогает писать чистый, поддерживаемый и легко читаемый код, а практика композиции классов и разработки без кода развивает навыки структурирования приложений. В результате студент сможет создавать сложные архитектуры, разделять ответственность между моделями, контроллерами и отображениями, а также эффективно управлять событиями через брокер событий. Дополнительно освоение расширенного TypeScript и ООП, включая статические методы, интерфейсы, абстрактные классы и методы, обеспечивает глубокое понимание объектно-ориентированного программирования и позволяет строить масштабируемые и устойчивые приложения. Такой комплексный подход дает уверенность в работе с современными фреймворками и библиотеками, а также закладывает фундамент для дальнейшего профессионального роста во фронтенде.

Тема 4. ООП в интерфейсах

После изучения темы паттернов проектирования студент освоит важные шаблоны, которые помогут создавать гибкие и масштабируемые фронтенд-приложения. Он познакомится с паттернами Singleton, Adapter, Facade, Proxy, Observer и Builder, что позволит эффективно управлять состоянием приложения, адаптировать интерфейсы, упрощать взаимодействие между компонентами и строить сложные объекты поэтапно. В разделе работы с окружением студент научится сохранять данные в браузере, работать с файлами и использовать стандартные обсерверы, что существенно расширит возможности взаимодействия с пользователем и системой. Изучение ООП в интерфейсах и проектирование UI-библиотеки дадут практические навыки создания повторно используемых компонентов - тултипов, дропдаунов, модальных окон, галерей, аккордеонов, табов и слайдеров. Особое внимание уделяется асинхронным операциям, например, дропдауну с поиском и созданию чата, что подготовит к реальным задачам фронтенд-разработки. Такой комплексный подход формирует у студента понимание, как проектировать и реализовывать сложные интерфейсы, используя принципы объектно-ориентированного программирования и паттерны проектирования, что значительно повысит качество и поддержку кода в реальных проектах.

Тема 5. Основы React

Изучая концепцию SPA (Single Page Application) и популярные фреймворки, студент получит глубокое понимание того, как строятся современные веб-приложения с динамическим обновлением контента без перезагрузки страницы. Он познакомится с принципами реактивного программирования и особенностями таких фреймворков, как React, Angular и Vue, что позволит выбрать оптимальный инструмент для решения конкретных задач. В разделе по настройке и сборке проекта студент освоит ключевые инструменты сборки, включая Webpack, CRA (Create React App) и Vite, научится оптимизировать процесс разработки и разделять настройки для продакшн и девелопмент-сред. Основы React охватывают устройство Virtual DOM, синтаксис JSX, создание и использование компонентов, работу с пропсами и дочерними элементами, обработку событий и стилизацию, что формирует базовые навыки разработки интерфейсов. Важным этапом станет изучение состояния и эффектов через хуки - useState, useEffect, useRef и кастомные хуки, которые позволяют управлять данными и жизненным циклом компонентов более эффективно. Студент также познакомится с жизненным циклом классовых и функциональных компонентов, научится создавать чистые компоненты и обрабатывать ошибки с помощью Error Boundaries. Завершает обучение работа с библиотеками компонентов и концепция изоляции с использованием Storybook и CSS-модулей, что помогает создавать и тестировать UI-компоненты в изоляции, улучшая качество и поддержку кода. Такой комплексный подход обеспечивает студенту уверенные навыки разработки современных SPA-приложений с использованием передовых инструментов и методик.

Тема 6. Роутинг и авторизация

Студент изучит продвинутые возможности React, включая работу с глобальным состоянием через Redux и другие библиотеки, такие как MobX и Effector. Особое внимание уделяется React Router для организации маршрутизации в приложениях, а также механизмам авторизации пользователей. В рамках проекта "Stellar Burger" студент применит полученные знания на практике, реализуя роутинг и защиту маршрутов. Также будут рассмотрены дополнительные технологии, включая Redux Saga, DnD и WebSockets.

Тема 7. Тестирование React-приложений

Студент изучит методы оптимизации производительности React-приложений, что позволит значительно ускорить работу интерфейса и улучшить пользовательский опыт. В рамках курса он освоит основы тестирования с использованием Jest для юнит-тестов и Cypress для энд-ту-энд тестирования, применяя эти инструменты на примере формы регистрации. Кроме того, студент научится готовить проект к продакшену, включая настройку сборки, минимизацию кода и другие важные шаги для стабильной и эффективной работы приложения в реальных условиях.

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства науки и высшего образования Российской Федерации от 6 апреля 2021 года №245)

Письмо Министерства образования Российской Федерации №14-55-99б/ин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

6. Фонд оценочных средств по дисциплине (модулю)

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

В фонде оценочных средств содержится следующая информация:

- соответствие компетенций планируемым результатам обучения по дисциплине (модулю);
- критерии оценивания сформированности компетенций;
- механизм формирования оценки по дисциплине (модулю);
- описание порядка применения и процедуры оценивания для каждого оценочного средства;
- критерии оценивания для каждого оценочного средства;
- содержание оценочных средств, включая требования, предъявляемые к действиям обучающихся, демонстрируемым результатам, задания различных типов.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе дисциплины (модулю).

7. Перечень литературы, необходимой для освоения дисциплины (модуля)

Освоение дисциплины (модуля) предполагает изучение основной и дополнительной учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями;
- в печатном виде - в Научной библиотеке им. Н.И. Лобачевского. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования Научной библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,5 экземпляра (для обучающихся по ФГОС 3++ - не менее 0,25 экземпляра) каждого из изданий основной литературы и не менее 0,25 экземпляра дополнительной литературы на каждого обучающегося из числа лиц, одновременно осваивающих данную дисциплину.

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов Научной библиотеки КФУ.

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

CSS-Tricks (документация CSS и адаптивная верстка) - <https://css-tricks.com>

HTML Living Standard (спецификация по HTML) - <https://html.spec.whatwg.org>

JavaScript-Info (документация по JavaScript) - <https://javascript.info>

MDN Web Docs - <https://developer.mozilla.org>

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

Вид работ	Методические рекомендации
лабораторные работы	Установить редактор кода VisualStudio Code и систему контроля версий Git. Для удобства использования Git рекомендуется установить GUI, например Git Extensions или GitHub Desktop в случае использования Mac OS. Каждая работа должна фиксироваться в персональном репозитории Git. Код лабораторной работы должен содержать осмысленные комментарии в виде пояснений функционала, реализуемого или используемого в работах.
самостоятельная работа	Студенты должны самостоятельно исследовать документацию (например, MDN Web Docs) для решения поставленных проблем и расширения функционала своих проектов. Для отработки навыков рекомендуется использовать интерактивные платформы, такие как freeCodeCamp или Codecademy. Ключевая цель - создание и постоянное пополнение личного портфолио на GitHub для демонстрации прогресса. Все задачи должны быть выполнены самостоятельно, а код - содержать комментарии, объясняющие ключевые решения.
зачет с оценкой	Для подготовки к зачету с оценкой студентам следует вернуться к ранее выполненным лабораторным и домашним заданиям, осмыслить написанный код и проверить, что все их задания выложены в персональный репозиторий. Проверить, что работы компилируются и реализуют требуемый функционал задачи, проверить уровни доступа к репозиторию, чтобы преподаватель мог проверить работы.

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

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

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

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Помещения для самостоятельной работы обучающихся, укомплектованные специализированной мебелью (столы и стулья) и оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду КФУ.

Учебные аудитории для контактной работы с преподавателем, укомплектованные специализированной мебелью (столы и стулья).

Компьютер и принтер для распечатки раздаточных материалов.

12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;
- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:
- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;
- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 09.03.04 "Программная инженерия" и профилю подготовки "Разработка цифровых продуктов (с применением электронного обучения и дистанционных образовательных технологий) (Digital Product Development (with the use of e-learning and distance education technologies))".

Перечень литературы, необходимой для освоения дисциплины (модуля)

Направление подготовки: 09.03.04 - Программная инженерия

Профиль подготовки: Разработка цифровых продуктов (с применением электронного обучения и дистанционных образовательных технологий) (Digital Product Development (with the use of e-learning and distance education technologies))

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

Основная литература:

1. Янцев, В. В. JavaScript. Креативное программирование : учебное пособие для вузов / В. В. Янцев. - 2-е изд., стер. - Санкт-Петербург : Лань, 2024. - 232 с. - ISBN 978-5-507-49267-1. - Текст : электронный // Лань : электронно-библиотечная система. - URL: <https://e.lanbook.com/book/383837> (дата обращения 10.12.2024). - Режим доступа: для авториз. пользователей.
2. Дронов В. А. HTML и CSS: 25 уроков для начинающих. / В.А. Дронов. - Санкт-Петербург : БХВ-Петербург, 2020. - 400 с. - ISBN 978-5-9775-4070-4. - URL: <https://ibooks.ru/bookshelf/385775> (дата обращения: 10.12.2024). - Текст: электронный.
3. Сидельников Грег. Наглядный CSS/ Пер. с английского С. Черников. - Санкт-Петербург : Питер, 2022. - 224 с. - ISBN 978-5-4461-1618-8. - URL: <https://ibooks.ru/bookshelf/377944> (дата обращения: 10.12.2024). - Текст: электронный.
4. Прохоренок Н. А. JavaScript и Node.js для веб-разработчиков. / Н.А. Прохоренок, В.А. Дронов. - Санкт-Петербург : БХВ-Петербург, 2022. - 768 с. - ISBN 978-5-9775-6847-0. - URL: <https://ibooks.ru/bookshelf/385754> (дата обращения: 10.12.2024). - Текст: электронный.

Дополнительная литература:

1. Никсон Робин. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 6-е изд. - Санкт-Петербург : Питер, 2023. - 832 с. - ISBN 978-5-4461-1970-7. - URL: <https://ibooks.ru/bookshelf/386792> (дата обращения: 10.12.2024). - Текст: электронный.
2. Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера. - 5-е изд., перераб. и доп. / Н.А. Прохоренок, В.А. Дронов. - Санкт-Петербург : БХВ-Петербург, 2019. - 912 с. - ISBN 978-5-9775-3986-9. - URL: <https://ibooks.ru/bookshelf/386508> (дата обращения: 10.12.2024). - Текст: электронный.
3. Дронов В. А. JavaScript. 32 урока для начинающих. - (Для начинающих) / В.А. Дронов. - Санкт-Петербург : БХВ-Петербург, 2024. - 576 с. - ISBN 978-5-9775-1942-7. - URL: <https://ibooks.ru/bookshelf/396454> (дата обращения: 10.12.2024). - Текст: электронный.
4. Свекис Лоренс Ларс. JavaScript с нуля до профи. - Санкт-Петербург : Питер, 2023. - 480 с. - ISBN 978-5-4461-2269-1. - URL: <https://ibooks.ru/bookshelf/390214> (дата обращения: 10.12.2024). - Текст: электронный.

*Приложение 3
к рабочей программе дисциплины (модуля)
Б1.В.ДВ.02.03 Фронтенд-разработка*

Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Направление подготовки: 09.03.04 - Программная инженерия

Профиль подготовки: Разработка цифровых продуктов (с применением электронного обучения и дистанционных образовательных технологий) (Digital Product Development (with the use of e-learning and distance education technologies))

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Операционная система Microsoft Windows 7 Профессиональная или Windows XP (Volume License)

Пакет офисного программного обеспечения Microsoft Office 365 или Microsoft Office Professional plus 2010

Браузер Mozilla Firefox

Браузер Google Chrome

Adobe Reader XI или Adobe Acrobat Reader DC

Kaspersky Endpoint Security для Windows

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен обучающимся. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, учебно-методические комплексы, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов высшего образования (ФГОС ВО) нового поколения.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен обучающимся. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "Консультант студента", доступ к которой предоставлен обучающимся. Многопрофильный образовательный ресурс "Консультант студента" является электронной библиотечной системой (ЭБС), предоставляющей доступ через сеть Интернет к учебной литературе и дополнительным материалам, приобретенным на основании прямых договоров с правообладателями. Полностью соответствует требованиям федеральных государственных образовательных стандартов высшего образования к комплектованию библиотек, в том числе электронных, в части формирования фондов основной и дополнительной литературы.