

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор высшей школы
программной инженерии
А.В. Малеев**

	Рабочая программа дисциплины (модуля)
по дисциплине:	Фронтенд-разработка
по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии
курс:	1
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 2 (весенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 48 час.

Всего часов: 108, всего зач. ед.: 3

Программу составил: А.В. Селиванов

Программа обсуждена на заседании высшей школы программной инженерии 14.03.2025

Аннотация

Знакомство с Frontend-разработкой, ее принципами, целями и инструментами. В процессе прохождения курса студенты изучат основные правила и подходы к разработке современных web-приложений, научатся обрабатывать отправку формы, узнают все возможные способы её сохранения, поработают с авторизацией в web-приложениях.

1. Цели и задачи

Цель дисциплины

- познакомить студентов с таким направлением IT-разработки, как Frontend, дать студентам базовые и продвинутые навыки Frontend - разработки, а также познакомить их со всеми необходимыми принципами, правилами и подходами к разработке современных web-приложений.

Задачи дисциплины

- научить слушателей обработке отправки формы, показать способы её сохранения. Научить работе с авторизацией в web-приложениях. Научить разрабатывать собственное web-приложение.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
	ОПК-7.3 Умеет выявлять узкие места в процессе разработки и предлагать методы и инструменты для его оптимизации
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.1 Различает синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования
	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

1. Интерфейсы «ввода». File, geolocation, payment, battery status.
2. React devtools, create-react-app.
3. SPA. Spa, history api, solid state.
4. Виды и особенности применения web-воркеров, концепция PWA.
5. Архитектура современных web-приложений.

уметь:

1. Обработка отправки формы.
2. Сохранение в storage.
3. Авторизация в web-приложениях. Csrf, http-only cookies, cors, csp, js-api, oauth-провайдеры.
4. Разрабатывать собственное web-приложение.

владеть:

1. Взаимодействие с сервером.
2. Flow загрузки страницы, синхронные и асинхронные запросы, отладка сетевых запросов.
3. Построение сложного интерфейса пользователя. Best Practice использования React, паттерн Flux и Redux как реализация.
4. Доставка REAL-TIME сообщений. Websocket.
5. Оптимизация. Css-препроцессоры, css-modules, минификация и обфускация, кеширование и инвалидация кеша, шрифты и svg, sprite sheet (генерирование).

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

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Оптимизация	2	2		4
2	Web Workers	2	2		4
3	Push-уведомления	2	2		4
4	css	2	2		4
5	Современные возможности JS API	2	2		4
6	Новые возможности React	4	4		4
7	Новые возможности CRA	4	4		6
8	Мониторинг, профилирование, отладка, оптимизация	4	4		6
9	Основы безопасности веб приложений, деплой, инструменты командной разработки	4	4		6
10	Мета-лекция про фронтенд	4	4		6
Итого часов		30	30		48
Подготовка к экзамену		0 час.			
Общая трудоёмкость		108 час., 3 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 2 (Весенний)

1. Оптимизация

Обзор современного фронтенда. Подготовка структуры проекта. Npm. Базовая сборка. Создание Hello World приложения.

2. Web Workers

Виды и особенности применения web-воркеров. Концепция PWA. Создание манифеста приложения. Синхронизация вкладок. Фоновая отправка сообщений.

3. Push-уведомления

Разбор и сравнение основных платформ. Обзор готовых решений. Реализация простого клиента. Интеграция firebase cloud messaging.

4. css

File, geolocation, payment, battery status. Получение и кеширование геолокации. Загрузка файла и отрисовка preview.

5. Современные возможности JS API

Devtools. Html (форма и список сообщений). Css (сначала берем bootstrap). Es6, es2018. Dom, web components. Local/session storage. Обработка отправки формы, сохранение в storage.

6. Новые возможности React

Обзор фреймворков. Введение в react. React devtools, create-react-app (публичный ui-kit). Перенос кода в jsx, разделение на компоненты. Дополнение конфигурации сборки. Отправка (получение) данных на сервер.

7. Новые возможности CRA

Spa, history api. Solid state. Экран профиля. Настройка роутинга. Long-polling. Интеграция с Server API.

8. Мониторинг, профилирование, отладка, оптимизация

Best Practice использования React. Паттерн Flux и Redux как реализация. Определение моделей и хранилищ проекта. Реализация основных сценариев использования. Обработка ошибок сервера.

9. Основы безопасности веб приложений, деплой, инструменты командной разработки

Csrf, http-only cookies. Cors, csp. Is-api oauth-провайдеров. Прикрутка OAuth от основных соцсетей. Прикрутка ombed.

10. Мета-лекция про фронтенд

Wwebsocket. Простой сокет к dev-серверу.-Подключение к centrifugo со стороны клиента.

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

компьютер и мультимедийное оборудование (проектор).

6.Перечень рекомендуемой литературы

Основная литература

1. Технологии разработки программного обеспечения, Электрон. версия печ. публикации / . — Санкт-Петербург, Питер, 2012

Фонд библиотеки МФТИ:

2. Баланов, А. Н. Бэкенд-разработка веб-приложений: архитектура, проектирование и управление проектами : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 312 с. — ISBN 978-5-507-52472-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/451820>

Дополнительная литература

Фонд библиотеки МФТИ:

1. Янцев, В. В. Разработка web-страниц на HTML, CSS и JavaScript : учебное пособие для вузов / В. В. Янцев. — Санкт-Петербург : Лань, 2024. — 148 с. — ISBN 978-5-507-49640-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/422462>

Рекомендуемая литература для самостоятельного изучения:

Yang, Y. LabVIEW graphical programming cookbook. Packt Publishing Ltd, 2014, p. 272.

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

Не предусмотрено

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

Устройство браузера – <https://html5rocks.appspot.com/en/tutorials/internals/howbrowserswork/>

JS и DOM – <http://learn.javascript.ru/>

node и npm – <https://nodejs.org/en/docs/> <https://docs.npmjs.com/>

React – <https://reactjs.org/docs/getting-started.html>

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

Для успешного освоения данной дисциплины студенту следует:

- посещать семинары, желательно (но не является обязательным условием успешного освоения дисциплины) конспектировать материалы занятий;
- выполнять домашние и практические задания, своевременно представлять на проверку преподавателю (при необходимости).

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии
курс:	<u>1</u>
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 2 (весенний) - Дифференцированный зачет

Разработчик: А.В. Селиванов

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
	ОПК-7.3 Умеет выявлять узкие места в процессе разработки и предлагать методы и инструменты для его оптимизации
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.1 Различает синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования
	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных

2. Показатели оценивания компетенций

В результате изучения дисциплины «Фронтенд-разработка» обучающийся должен:

знать:

1. Интерфейсы «ввода». File, geolocation, payment, battery status.
2. React devtools, create-react-app.
3. SPA. Spa, history api, solid state.
4. Виды и особенности применения web-воркеров, концепция PWA.
5. Архитектура современных web-приложений.

уметь:

1. Обработка отправки формы.
2. Сохранение в storage.
3. Авторизация в web-приложениях. Csrf, http-only cookies, cors, csp, js-api, oauth-провайдеры.
4. Разрабатывать собственное web-приложение.

владеть:

1. Взаимодействие с сервером.
2. Flow загрузки страницы, синхронные и асинхронные запросы, отладка сетевых запросов.
3. Построение сложного интерфейса пользователя. Best Practice использования React, паттерн Flux и Redux как реализация.
4. Доставка REAL-TIME сообщений. Websocket.
5. Оптимизация. Css-препроцессоры, css-modules, минификация и обфускация, кеширование и инвалидация кеша, шрифты и svg, sprite sheet (генерирование).

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлого занятия.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Задание по html/css.

Реализовать верстку страницы списка сообщений чата по прилагаемому макету. Для выравнивания сообщений в контейнере использовать flexbox.

2. Задание по js.

Реализовать обработку отправки формы сообщения. Содержимое формы преобразовывать в объект сообщения, содержащий информацию, о пользователе, времени отправки и содержимом сообщения. Объект сообщения сохранять в localStorage, а также передавать в событие “new-message”, порождаемое на объекте формы.

3. Задание на работу с API.

Реализовать обработку выбора файла в стандартном input[type=file].

4. Вывести в форме информацию о выбранном файле, если выбрано изображение – отобразить его миниатюру.

5. Задание по применению инструментов разработки.

Доработать конфигурацию сборки одностраничного приложения, таким образом, чтобы появилась возможность использования source-maps и dev-сервера с “горячей” перезагрузкой страницы.

Критерии оценивания

- оценка «отлично (10)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (9)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (8)» выставляется студенту, показавшему всесторонние систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, и правильное обоснование принятых решений
- оценка «хорошо (7)» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (6)» выставляется студенту, если он знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (5)» выставляется студенту, если он знает материал, и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «удовлетворительно (4)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «удовлетворительно (3)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет фрагментарно основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

- оценка «неудовлетворительно (2)» выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач
- оценка «неудовлетворительно (1)» выставляется студенту, который не знает формулировок основных понятий дисциплины

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет проводится по итогам текущей успеваемости и сдачи заданий и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме.