

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

**УТВЕРЖДЕНО**  
**Директор**

**А.В. Малеев**

	<b>Рабочая программа дисциплины (модуля)</b>
<b>по дисциплине:</b>	Веб-разработка
<b>по направлению:</b>	Программная инженерия
<b>профиль подготовки:</b>	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии МФТИ - Яндекс
<b>курс:</b>	2
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 3 (осенний) - 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. Созыкин, канд. техн. наук, доцент

Программа обсуждена на заседании высшей школы программной инженерии МФТИ - Яндекс 28.04.2023

## Аннотация

познакомить студентов 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-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Способен использовать информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности
	ОПК-2.2 Владеет навыками создания программного обеспечения для ЭВМ и систем различной архитектуры
ОПК-5 Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	ОПК-5.2 Способен оценивать концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роль людей, процессов, методов, инструментов и технологий обеспечения качества
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
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-3.2 Умеет выбирать языки программирования для написания программного кода с учетом технического задания
	ПК-3.3 Умеет излагать основные принципы построения и виды архитектуры программного обеспечения, методы и средства проектирования программного обеспечения, методологии разработки программного обеспечения и технологии программирования

	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных
ПК-5 Способен проектировать, разрабатывать, внедрять, сопровождать и снимать с эксплуатации информационные системы	ПК-5.3 Умеет разрабатывать прототип информационных систем в соответствии с требованиями и проводить его тестирование для проверки корректности архитектурных решений

### 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. Csrф, 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	4	4		4
2	Web Workers	3	3		4
3	Push-уведомления	3	3		4
4	css	3	3		4
5	Современные возможности JS API	2	2		4
6	Новые возможности React	5	5		7
7	Новые возможности CRA	2	2		7
8	Мониторинг, профилирование, отладка, оптимизация	2	2		6
9	Основы безопасности веб приложений, деплой, инструменты командной разработки	2	2		4
10	Мета-лекция про фронтенд	4	4		4
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: 3 (Осенний)

##### 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. Js-api oauth-провайдеров. Прикрутка OAuth от основных соцсетей. Прикрутка ombed.

##### 10. Мета-лекция про фронтенд

Websocket. 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. Паттерны проектирования, [учеб. пособие для вузов] / Эр. Фримен, Эл. Фримен . — Санкт-Петербург, Питер, 2017.— URL: <https://ibooks.ru/bookshelf/354827/reading> (дата обращения: 26.11.2020). - Полный текст (Режим доступа : из сети МФТИ / Удаленный доступ)

Литература, рекомендуемая для самостоятельного изучения:

- Дуглас Крокфорд, «Как устроен JavaScript»
- Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес "Паттерны объектно-ориентированного проектирования"

### Дополнительная литература

Литература, рекомендуемая для самостоятельного изучения:

- Yang, Y. LabVIEW graphical programming cookbook. Packt Publishing Ltd, 2014, p. 272.
- Роберт Мартин, «Чистая архитектура»

## **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)**

<b>по направлению:</b>	Программная инженерия
<b>профиль подготовки:</b>	Разработка программно-информационных систем высшая школа программной инженерии МФТИ - Яндекс высшая школа программной инженерии
<b>курс:</b>	2
<b>квалификация:</b>	бакалавр
Семестр, формы промежуточной аттестации: 3 (осенний) - Дифференцированный зачет	
<b>Разработчик:</b>	А.В. Созыкин, канд. техн. наук, доцент

## 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-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Способен использовать информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности
	ОПК-2.2 Владеет навыками создания программного обеспечения для ЭВМ и систем различной архитектуры
ОПК-5 Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	ОПК-5.2 Способен оценивать концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роль людей, процессов, методов, инструментов и технологий обеспечения качества
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
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-3.2 Умеет выбирать языки программирования для написания программного кода с учетом технического задания
	ПК-3.3 Умеет излагать основные принципы построения и виды архитектуры программного обеспечения, методы и средства проектирования программного обеспечения, методологии разработки программного обеспечения и технологии программирования
	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных
ПК-5 Способен проектировать, разрабатывать, внедрять, сопровождать и снимать с эксплуатации информационные системы	ПК-5.3 Умеет разрабатывать прототип информационных систем в соответствии с требованиями и проводить его тестирование для проверки корректности архитектурных решений

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Веб-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.