**План обучения студентов JavaScript**

**I. Базовый курс JavaScript (18 занятий - 72 часа)**

**1. Начало работы с JavaScript**

* Введение в JavaScript
* Справочники и спецификации
* Инструментарий (VS Code, DevTools)

**2. Основные понятия. Функции. Массивы и хэши**

* Переменные
* Константы
* Арифметические операции
* Строки и операции с ними
* Операции сравнения и логические операции
* Типы
* Преобразование типов
* Оптимизация при расчёте логических выражений
* Оператор if
* Оператор switch
* Операторы ввода-вывода
* Оператор выбора
* Прочие операции
* Циклы
* Функции
* Область видимости переменных
* Рекурсия
* Массивы
* Хэши (ассоциативные массивы)
* Ссылки

**3. Замыкания. ООП. Описание классов**

* JS код во внешнем файле
* Переменные типа Function
* Замыкания
* Класс и объект
* ООП и связанные с ним понятия
* Создание и удаление объектов
* Описание собственных классов
* Откуда берётся this?
* Приватные свойства и методы
* Функциоальное наследование
* Прототипное наследование
* Предопределённые классы и объекты
* Класс Object
* Объект Math

**4. Встроенные классы. Отладка и тестирование кода**

* Класс String
* Класс Number
* Класс Date
* Класс Array
* Исключения
* Отладка и тестирование кода

**5. Работа с DOM**

* DOM - Document Object Model
* Поиск элементов в дереве
* Работа со стилевыми свойствами элемента
* Работа с положением и размером элементов
* Работа с HTML-атрибутами тегов и свойствами DOM-элементов
* Управление содержимым элемента
* Работа с формами и элементами форм
* Работа с деревом элементов
* Манипуляция элементами дерева

**6. Формы. События и их обработка**

* События. Виды событий
* Алгоритм распространения событий
* Установка обработчиков и перехватчиков событий

**7. Объект события. События мыши, клавиатуры, тачскрина. Таймеры**

* Объект события
* События мыши, клавиатуры, тачскрина
* Делегирование обработки событий
* Таймеры

**8. Render tree**

* Предопределённый объект document
* Render tree

**9. JSON. Хранение данных на клиенте**

* JSON
* Встроенные методы для работы с JSON
* Порядок взаимодействия браузера с сервером в случае установки куки сервером
* Порядок взаимодействия браузера с сервером в случае уже установленного куки
* localStorage, sessionStorage

**10. jQuery (обзорно)**

* полезные функции
* jQuery Validation
* jQuery: AJAX, события

**11. Уязвимости. OWASP (обзорно)**

**12. CSS для JavaScript-разработчика (обзорно)**

**13. AJAX. SPA.** **WebSocket**

**14. Модульность. AMD-модули. Кроссбраузерность. Регулярные выражения**

**15. Лабораторная работа**

**II. Работа с современными фреймворками. Kendo Framework (6 занятий - 24 часа)**

**1. MVC. Архитектура MVC**

**2. MVVM**

* MVVM Overview
* ObservableObject
* Bindings

**3. DataSource**

* Local/Remote Binding
* Widget Binding
* Offline Support

**4. Templates**

* Templates Overview
* Performance

**5. Widget Basic**

* Data Attributes
* Methods and Events

**6. Widgets**

* Editors. AutoComplete, ComboBox, DatePicker, DropDownList, Editor, MaskedTextBox, MultiSelect, NumericTextBox, Slider, Upload, Validator
* Data Management. Grid, ListView
* Layout. Window, Tooltip, Popup, Notification
* Navigation. Menu, TabStrip, ToolBar, TreeView

**7. Лабораторная работа**

**III. Работа с современными фреймворками. React. Redux (12 занятий - 48h)**

**1. Современный веб**

* ES6+, TS
* WebComponents (Polymer), Virtual DOM
* Современные фреймворки: Angular, React
* Node.js, npm, Babel, Webpack

**2. React**

* Настройка окружения (Create React App), Hello World
* Основы работы с JSX
* Отображение элементов, виртуальный DOM
* Компоненты в React
* Состояние компонентов
* Жизненный цикл компонента
* Обработка событий
* Динамика в приложении
* Взаимодествие с DOM (keys & refs)
* Переиспользование компонентов (Higher-Order Components)
* Грамотное разбиение приложения на компоненты
* React Developer Tools

**3. Redux**

* Знакомство с Flux
* Ключевые отличия Redux и Flux
* Ключевые концепции и принципы Redux
* Actions
* Reducers
* Store
* Data Flow
* Middleware в Redux
* Redux DevTools
* React-redux для связи компонентов с логикой

**4. React Router**

* Зачем нужен роутинг и как он устроен
* Различные модули History
* Использование react-router
* Редиректы и переходы на странице
* Интеграция роутинга с Redux

**5. Дополнительно**

* Как работает React (Reconciliation, Fiber)
* Оптимизация производительности в React
* Иммутабельные данные (ImmutableJS)
* Работа с ImmutableJS в React/Redux приложении
* Server-side рендеринг
* Конфигурация Webpack, ESLint

**6. Лабораторная работа**