

📍 **Киев**
ул. Жиланская, 128/28
(044) 537 22 60

• **Одесса**
ул. Садовая, 3
(048) 728 6660

• **Винница**
ул. Соборная, 20
(0432) 52 0820

• **Днепр**
пр-т Яворницкого, 101
(056) 766 0788

• **Житомир**
ул. Скорульского, 3
(067) 303 0533

• **Запорожье**
ул. Рекордная, 20
(067) 547 7455

• **Кривой Рог**
пр-т Мира, 44А
(067) 828 8586

• **Каменское**
ул. Медицинская, 2-Е
(050) 380 1838

• **Луцк**
ул. Евгения Сверстюка, 1
(097) 114 5250

• **Львов**
ул. Замарстыновская, 83а
(032) 240 3851

• **Мариуполь**
ул. Архитектора
Нильсена, 26/21
(0629) 41 2724

• **Николаев**
ул. Лягина, 4
(0512) 67 0053

• **Полтава**
ул. Пилипа Орлика, 36а
(0322) 7 3544

• **Ровно**
ул. Соборная, 16
(050) 445 2111

• **Тернополь**
ул. Крушельницкой, 18
(067) 968 9985

• **Ивано-Франковск**
ул. Новгородская, 49
(050) 488 9951

• **Харьков**
ул. Маломясницкая, 9/11
(057) 752 0501

• **Хмельницкий**
ул. Подольская, 93
(068) 300 3009

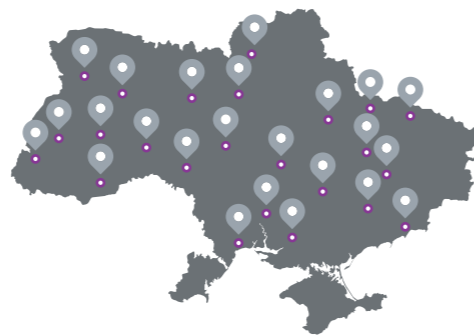
• **Херсон**
ул. Перекопская, 20
(067) 652 2762

• **Чернигов**
пр-т Мира, 44А
(0462) 91 9120

• **Черновцы**
ул. Кордуби, 25
(050) 366 9905

• **Ужгород**
вул. Льва Толстого, 33
(097) 920 08 05

Компьютерная Академия ШАГ
Филиалы
Украина



Филиалы
Международные



Программа обучения — как надо!*

*Ведущим IT-компаниям мира.

BACK-END

ПО ОКОНЧАНИИ КУРСА СЛУШАТЕЛЬ БУДЕТ:

- Разбираться в тонкостях создания веб-приложений с использованием PHP.
- Понимать особенности реализации механизмов ООП в PHP. Обрабатывать и анализировать данные форм.
- Использовать стандартные функции PHP. Работать с механизмом сессий.
- Взаимодействовать с источниками данных. Проектировать базы данных.
- Создавать запросы различной степени сложности. Внедрять AJAX в веб-приложения.
- Пользоваться системой контроля версий. Понимать основы командного взаимодействия.
- Применять паттерны проектирования. Использовать юнит-тестирование.
- Понимать и разбираться в тонкостях паттерна MVC. Создавать веб-проекты с использованием PHP и паттерна MVC.
- Использовать фреймворки Codeigniter и Laravel для создания веб-проектов.
- Применять различные конструкции HTML и использовать конструкции CSS.
- Анализировать и проводить отладку кода web-документов.
- Формировать содержимое web-документов для различных экранов — от стандартных браузеров до мобильных устройств.
- Быстро и качественно форматировать сложные web-документы.
- Владеть базовыми конструкциями языка JavaScript: переменными, условиями, циклами, строками, массивами, функциями и т.д.
- Обрабатывать возникающие ошибки.
- Разбираться в понятиях «событие», «обработчик события». Создавать функции-обработчики различных событий.
- Взаимодействовать с объектами BOM и DOM.
- Разбираться в тонкостях реализации клиентских сценариев для разных браузеров.
- Создавать формы и анализировать данные пользователя с использованием регулярных выражений.
- Сохранять пользовательские данные с помощью механизма cookies.
- Понимать особенности применения HTML5 по отношению к JavaScript.
- Сериализовать и парсить данные, используя JSON.
- Применять различные принципы создания асинхронных запросов с помощью AJAX.

Компьютерная Академия ШАГ

18 65

Стран мира

Филиалов

45000

Студентов

Аудиторные занятия — 368 академ. часов

Самостоятельная работа — 368 академ. часов

Soft Skills — 18 академ. часов

English for IT — 18 академ. часов

Всего за курс — 772 академ. часа

Длительность курса — 1 год

itstep.org

Back-end

Разработка веб-страниц на языке разметки HTML5 с использованием каскадных таблиц стилей CSS3

Введение в web-технологии. Структура HTML. Форматирование текста с помощью HTML

- Язык разметки гипертекста HTML.
- Тег – основной элемент структуры HTML. Правила записи тегов и их атрибутов в стандарте HTML5. Синтаксические отличия HTML4, XHTML и HTML5.
- Основные ошибки в записи тегов.
- Структура HTML5-документа.
- Кодировка страницы и теги <meta>. Классификация тегов: линейные и блочные.
- Модель форматирования текста: заголовки и абзацы. Элементы <p>, <h1>, <h6>.
- Выравнивание текста в блочных элементах: атрибут align.
- Классификация тегов: логическое и физическое форматирования.
- Практика: создание простейшей web-страницы.

Форматирование при помощи CSS. Списки. CSS отступы и поля

- CSS – каскадные таблицы стилей.
- Теги без форматирования: <div> – блочный, – линейный.
- Аналогия HTML и CSS на примере линейных и блочных тегов.
- Дополнительные свойства CSS для форматирования текста: letter-spacing, line-height, text-indent, text-transform, white-space, word-spacing.
- Использование атрибутов class и id для задания стилей.
- Использование внешних CSS файлов стилей.
- Практика: форматирование текста с помощью CSS.
- Создание списков.
- Создание вложенных списков.
- Форматирование списков с помощью CSS. Списки определений: элементы <dl>, <dd>, <dt>.
- Управление отступами и полями.
- Практика: создание списков.

Графика в web-дизайне. Оптимизация графики. Гиперссылки. Принципы навигации web-сайта

- Форматы графических файлов в web.
- Тег и его атрибуты (src, alt, width, height, border).
- Фон страницы – свойство background.

- Общие сведения о гиперссылках.
- Абсолютная и относительная адресация. Свойства CSS3.
- Создание меню с помощью структуры списков (,), его форматирование. Свойство display. Преобразование ссылки в блочный элемент. Псевдоклассы.
- CSS свойство cursor.
- Практика: работа по разработке галереи. Работа с мультимедиа.

Таблицы

- Создание простейшей таблицы. Теги <table>, <tr> и <td>.
- Объединение ячеек: атрибуты colspan, rowspan.
- Теги логического структурирования таблиц: <thead>, <tbody>, <tfoot>. Теги логического группирования столбцов: <colgroup>, <col>.
- Управление рамками таблицы: атрибуты frame, rules.
- Практика: создание сложных таблиц.
- Основы и пример табличной верстки: ее минусы.

Позиционирование. Верстка web-страниц блоками

- Свойство position.
- Свойства visibility, overflow.
- Основы верстки блоками. Правила верстки. Рассмотрение простейших структур страниц.
- Резиновая структура. Блоки с отрицательными margin.

Формы. Фреймы

- Управляющие элементы форм.
- Создание форм с помощью HTML.
- Элементы форм из HTML5.
- Валидация форм с помощью HTML5. Форматирование элементов форм с помощью CSS.
- Фреймы и их структура (теоретические сведения).

Создание web-приложений, исполняемых на стороне сервера при помощи языка программирования PHP, СУБД MySQL и технологии AJAX

Введение в web-программирование на PHP

- Принципы работы web-приложений. Обзор и сравнительный анализ существующих серверных web-технологий.
- Описание и обзор инструментария для разработчика: web-серверы, СУБД.

- Установка и настройка web-сервера Apache для Windows.
- Синтаксис включения PHP-кода в страницу. Разделение выражений. Комментарии.
- Правила именования переменных. Константы. Типы данных.
- Операторы и операнды.
- Условные конструкции. Массивы.
- Циклические конструкции.

Работа с формами, функции

- Способы связывания XHTML-формы и PHP-скрипта. Функции.
- Создание библиотек.
- Функции для работы с массивами, временем и со строками.
- Математические функции.

Системы контроля версий

- Зачем нужен контроль версий?
- Обзор систем. Git.

ООП, регулярные выражения

- Объектно-ориентированное программирование в PHP.
- Поддержка регулярных выражений в PHP.

Работа с файлами, ошибки и исключения, cookies и сессии

- Принципы взаимодействия с файлами и директориями посредством PHP. Ошибки и исключения.
- Cookies. Сессии.

Использование PHPUnit

- Модульное тестирование.
- Цели и задачи, необходимость модульного тестирования.
- Обзор инструментов для модульного тестирования.
- Инструмент PHPUnit.

Паттерны проектирования

- Понятие и причины создания паттернов проектирования.
- Принципы применения и выбор паттернов проектирования.
- Принципы разделения паттернов на категории.
- Использование UML при анализе паттернов проектирования.
- Порождающие, структурные и паттерны поведения.

Паттерн MVC

- Цели и задачи паттерна Model-View-Controller.
- Примеры использования паттерна MVC.

Принципы проектирования классов SOLID

- Обзор проблем, встречающихся при проектировании и разработке классов.
- Принципы проектирования и примеры использования классов SOLID.

Введение в работу с базами данных

- Введение в теорию баз данных.

- Запросы SELECT, INSERT, UPDATE, DELETE.
- Многотабличные базы данных, запросы. Нормализация.
- Функции агрегирования.
- Понятие группировки. Ключевое слово GROUP BY.
- Ключевое слово HAVING. Сравнительный анализ HAVING и WHERE.
- Подзапросы. Операторы подзапросов и объединений.
- План выполнения запроса.
- Оптимизация запросов.
- Понятие транзакции. Использование транзакций. Представления.
- Хранимые процедуры. Триггеры.

Взаимодействие с СУБД MySQL: административные возможности и взаимодействие с PHP-кодом

- Обзор возможностей PHP по поддержке работы с разными СУБД.
- Административные возможности СУБД MySQL.
- Взаимодействие PHP и MySQL.
- Библиотека SQLite.
- Расширение PHP Data Objects.
- Работа с графикой.

Работа в команде, управление программными проектами

- Что такое управление программными проектами?
- Причины возникновения дисциплины «Управление программными проектами».
- Диаграммы Ганта.
- Важные вопросы по управлению программными проектами.
- Анализ терминов предметной области. Характеристики проекта.
- Расходы, связанные с проектом.
- Общий обзор моделей и методологий процесса разработки.
- Подробнее о Scrum.

AJAX и PHP, Использование frameworks, CMS и PHP

- Цели и задачи AJAX. Практические примеры использования AJAX.
- Объект XMLHttpRequest. Пример использования XMLHttpRequest.
- Библиотеки для работы с AJAX.
- PHP frameworks. Обзор существующих решений.
- Использование CodeIgniter, Laravel.
- Web-мастеринг на основе CMS и обзор готовых проектов.

Разработка клиентских сценариев с использованием JavaScript и библиотеки jQuery

Введение в JavaScript

- Сценарии, выполняемые на стороне клиента.
- История создания JavaScript.
- Различия между JavaScript и Java, JScript, ECMAScript.
- Версии JavaScript.
- Понятие «Document Object Model».
- Понятие «Browser Object Model».

- Внедрение в HTML документы. Редакторы кода JavaScript.
- Тег <noscript>. Основы синтаксиса.
- Переменные. Правила именования переменных. Типы данных.
- Операторы. Ввод/вывод данных. Диалоговые окна.
- Условия. Циклы. Функции.
- Объект Arguments.
- Область видимости переменной. Ключевое this. Рекурсия.

Объект. Массивы. Объект Array. Строки. Объект String. Объект Date. Объект Math. Введение в объектно-ориентированное программирование

- Задержки и интервалы. Периодический вызов функций.
- Объект Date. Обработка даты и времени. Объект Math. Свойства и методы. Случайные числа.
- ООП. Три фундаментальных принципа ООП.
- Понятие класса и объекта в терминах JavaScript. Понятие «prototype».
- Свойства. Методы. Конструктор.
- Свойства-аксессоры.
- Наследование.

Обработка событий

- Событие. Обработчик события.
- Обработка событий в сценариях.
- Управление стилями элементов web-страницы.
- Объект Event и его свойства.
- Обработчики событий по умолчанию (стандартные обработчики), запрет вызова стандартного обработчика.
- Объект Image. Управление рисунками и ролловерами.

Browser Object Model. Document Object Model

- Объекты Browser Object Model.
- Document Object Model.
- Отличия DOM и BOM.
- Представление HTML-документа в виде дерева.
- Объекты модели DOM. Иерархия узлов. Модель событий DOM.
- Свойства и методы модели DOM.
- Изменение дерева DOM.
- Знакомство с объектами Document и Link. Особенности DOM в HTML5.
- Управление выделением и текстовым диапазоном: объекты Selection и TextRange.

Формы

- Применение форм. Размещение элементов формы в HTML.
- Коллекция Forms. Создание и программирование элементов формы.

Проверка достоверности форм. Использование cookie

- Объект RegExp. Правила записи регулярных выражений.
- Методы объектов String и RegExp для работы с регулярными выражениями. Проверка достоверно-

- сти данных формы.
- Cookie. Преимущества и недостатки cookie. Создание, использование и удаление cookie.

Рисование с помощью canvas, поддержка медиа-возможностей

- Базовые возможности canvas.
- Поддержка медиа возможностей.

JSON, AJAX

- Цели и задачи, синтаксис, объект.
- Настройка пользовательской сериализации в JSON. Метод toJSON. Синхронные и асинхронные запросы.
- AJAX. Объект XMLHttpRequest.
- Методы и свойства XMLHttpRequest.
- Понятие HTTP-заголовка.
- Использование метода GET. URL-кодирование.

Использование метода POST. Введение в jQuery

- Цели, задачи, версии jQuery.
- История создания jQuery.
- Подключение jQuery.
- Доступ к элементам страницы при помощи функции \$.
- Понятие и типы селектора.
- Traversing. Методы обхода DOM. Метод filter, next, nextAll, prev, prevAll, siblings и др.

События и jQuery

- Создание обработчиков событий с использованием jQuery.
- Удаление обработчиков событий.
- Объект Event и jQuery.
- Воздействие на обработку события. Запуск обработки события.

Стили и анимация

- Метод css. Отображение и скрытие элементов. Методы show и hide.
- Создание эффектов. Анимация.

Взаимодействие с DOM

- Создание новых элементов DOM.
- Вставка, передвижение и копирование элементов DOM.
- Взаимодействие с атрибутами.

AJAX и jQuery

- JSON. Механизмы AJAX внутри библиотеки jQuery.
- Использование метода GET.
- Использование метода POST.
- События и AJAX в рамках jQuery.
- Обработка ошибок.

Использование плагинов jQuery

- Подключение плагина.
- Примеры плагинов.

ИТОГОВЫЙ ПРОЕКТ

📍 **Киев**
ул. Жиланская, 128/28
(044) 537 22 60

• **Одесса**
ул. Садовая, 3
(048) 728 6660

• **Винница**
ул. Соборная, 20
(0432) 52 0820

• **Днепр**
пр-т Яворницкого, 101
(056) 766 0788

• **Житомир**
ул. Скороульського, 3
(067) 303 0533

• **Запорожье**
ул. Рекордная, 20
(067) 547 7455

• **Кривой Рог**
пр-т Мира, 44А
(067) 828 8586

• **Каменское**
ул. Медицинская, 2-Е
(050) 380 1838

• **Луцк**
ул. Евгения Сверстюка, 1
(097) 114 5250

• **Львов**
ул. Замарстыновская, 83а
(032) 240 3851

• **Мариуполь**
ул. Архитектора
Нильсена, 26/21
(0629) 41 2724

• **Николаев**
ул. Лягина, 4
(0512) 67 0053

• **Полтава**
ул. Пилипа Орлика, 36а
(0322) 7 3544

• **Ровно**
ул. Соборная, 16
(050) 445 2111

• **Тернополь**
ул. Крушельницкой, 18
(067) 968 9985

• **Ивано-Франковск**
ул. Новгородская, 49
(050) 488 9951

• **Харьков**
ул. Маломясницкая, 9/11
(057) 752 0501

• **Хмельницкий**
ул. Подольская, 93
(068) 300 3009

• **Херсон**
ул. Перекопская, 20
(067) 652 2762

• **Чернигов**
пр-т Мира, 44А
(0462) 91 9120

• **Черновцы**
ул. Кордуби, 25
(050) 366 9905

• **Ужгород**
вул. Льва Толстого, 33
(097) 920 08 05

Компьютерная Академия ШАГ
Филиалы
Украина



Филиалы
Международные



Программа обучения — как надо!*

*Ведущим IT-компаниям мира.

FRONT-END

ПО ОКОНЧАНИИ КУРСА СЛУШАТЕЛЬ БУДЕТ:

- Владеть основами взаимодействия с NodeJS. Производить сборку проекта.
- Разбираться в тонкостях использования Webpack.
- Понимать структуру Angular приложения.
- Применять правильные конструкции Angular в зависимости от поставленной задачи.
- Производить связывание данных в Angular приложении. Применять Dependency Injection.
- Разбираться в тонкостях реализации React приложений.
- Использовать Flux.
- Взаимодействовать с формами. Применять основы HTML: теги, атрибуты и способы.
- Структурирование содержимого web-страниц для создания форматированных документов.
- Применять основы CSS: значения, списки, цвета, шрифты и другие метрики форматирования.
- Владеть навыками проверки и отладки кода web-документов.
- Формировать содержимое web-документов разной сложности.
- Быстро и качественно форматировать сложные web-документы.
- Работать с базовыми конструкциями языка JavaScript такими как переменные, условия, циклы, массивы функции, и т.д.
- Применять ООП и его основные понятия. Исправлять возникающие ошибки.
- Разбираться в понятиях события, обработчика события. Создавать функции-обработчики различных событий.
- Понимать отличия BOM и DOM. Понимать отличия и взаимодействовать с объектами из BOM и DOM.
- Разбираться в тонкостях реализации клиентских сценариев в разных браузерах.
- Владеть принципами создания форм и анализа данных пользователя с использованием регулярных выражений.
- Сохранять пользовательские данные с помощью механизма cookies.
- Понимать особенности применения HTML5 по отношению к JavaScript.
- Сериализовать и парсить данные используя JSON.
- Владеть принципами создания асинхронных запросов при помощи Ajax.
- Разбираться в паттернах проектирования. Использовать системы контроля версий.
- Владеть принципами командной работы. Выполнять модульное тестирование.

Компьютерная Академия ШАГ

18 **65**

Стран мира

Филиалов

45000

Студентов

Аудиторные занятия — 368 академ. часов
Самостоятельная работа — 368 академ. часов
Soft Skills — 18 академ. часов
English for IT — 18 академ. часов
Всего за курс — 772 академ. часа
Длительность курса — 1 год

itstep.org

Программа обучения

Front-end

Разработка веб-страниц на языке разметки HTML5 с использованием каскадных таблиц стилей CSS3

Введение в Web-технологии. Структура HTML. Форматирование текста при помощи HTML

- Введение в предмет и языки разметки. Язык разметки гипертекста HTML.
- Структура HTML5 документа.
- Теги – основной элемент структуры HTML. Правила записи тегов и их атрибутов в стандарте HTML5. Синтаксические отличия HTML4, XHTML, HTML5.
- Классификация тегов: линейные и блочные, логическое и физическое форматирования.
- Основные ошибки в записях тегов.
- Кодировки страницы и теги <meta>.
- Модель форматирования текста: заголовки и абзацы. Элементы <p>, <h1>..<h6>. Выравнивание текста в блочных элементах: атрибут align.
- Практика: создание простейшей веб-страницы.

Форматирование с помощью CSS. Списки. CSS отступы и поля

- CSS – каскадные таблицы стилей.
- Сравнение HTML и CSS на примере линейных и блочных тегов.
- Теги без форматирования <div> – блочный, – линейный.
- Использование атрибутов class и id для задания стилей.
- Использование внешних CSS-файлов стилей.
- Дополнительные свойства CSS для форматирования текста: letter-spacing, line-height, text-indent, text-transform, white-space, word-spacing.
- Практика: форматирование текста при помощи CSS.
- Создание списков.
- Создание вложенных списков.
- Форматирование списков с помощью CSS.
- Списки определений: элементы <dl>, <dd>, <dt>.
- Управление отступами: практические примеры использования.

Materialize

- История создания Materialize.
- Цели и задачи Materialize.
- Подключение Materialize.
- Material design.
- Система сеток в Materialize.
- Понятие Helper. Виды Helpers.

- Навигация и Bootstrap.
- Использование изображений и видео.
- Скроллинг.
- Практические примеры использования.

Less

- История создания Less.
- Цели и задачи Less.
- Подключение и компиляция Less.
- Переменные.
- Миксины.
- Функции.
- Практические примеры использования.

Разработка клиентских сценариев с использованием JavaScript и библиотеки jQuery

Введение в JavaScript

- Сценарии, выполняемые на стороне клиента.
- JavaScript.
- История создания JavaScript.
- Различия между JavaScript и Java, JScript, ECMAScript.
- Версии JavaScript.
- Понятие Document Object Model.
- Понятие Browser Object Model.
- Внедрение в HTML документы. Редакторы кода JavaScript.
- Тег <noscript>.
- Основы синтаксиса.
- Переменные. Правила именования переменных.
- Типы данных.
- Операторы.
- Ввод/вывод данных. Диалоговые окна.
- Условия.
- Циклы.
- Функция.
- Объект arguments.
- Область видимости переменной. Ключевое this.
- Рекурсия.

Введение в объектно-ориентированное программирование

- Объекты. Массивы. Строки.
- Задержки и интервалы. Периодический вызов функций.
- Объект Date. Обработка даты и времени.
- Объект Math. Свойства и методы. Случайные числа.
- ООП. Три фундаментальных принципа.
- Понятие класса и объекта в терми-

нах JavaScript.

- Свойства.
- Методы.
- Свойства-аксессоры.
- Конструктор.
- Понятие prototype.
- Наследование.

Обработка событий

- Событие. Обработчик события.
- Обработка событий в сценариях.
- Управление стилями элементов веб-страницы.
- Объект event и его свойства.
- Обработчики событий по умолчанию (стандартные обработчики). Запрет вызова стандартного обработчика.
- Объект Image. Управление рисунками и ролловерами.

Browser Object Model. Document Object Model

- Отличия DOM от BOM.
- Объекты Browser Object Model.
- Document Object Model.
- Представление HTML - документа в виде дерева.
- Объекты модели DOM. Иерархия узлов.
- Модель событий DOM.
- Свойства и методы модели DOM.
- Изменение дерева DOM.
- Знакомство с объектами Document и Link.
- Управление выделением и текстовым диапазоном: объекты Selection и TextRange.
- Особенности DOM в HTML5.

Взаимодействие с DOM

- Создание новых элементов DOM.
- Вставка элементов DOM.
- Передвижение элементов DOM.
- Копирование элементов DOM.
- Взаимодействие с атрибутами.

Формы

- Применение форм. Размещение элементов формы в HTML.
- Коллекция Forms.
- Создание и программирование элементов формы.

Проверка достоверности форм. Использование Cookie

- Проверка достоверности данных формы.
- Объект RegExp. Правила записи регулярных выражений.
- Методы объектов String и RegExp для работы с регулярными выражениями.
- Понятие Cookie.
- Создание, использование и удаление cookie.
- Преимущества и недостатки cookie.

Рисование с помощью Canvas, поддержка медиа-возможностей

- Понятие Canvas и ее базовые возможности.
- Поддержка медиа-возможностей.

JSON, Ajax

- Цели и задачи. Синтаксис JSON.
- Объект JSON.
- Настройка пользовательской сериализации в JSON. Метод toJSON.
- Синхронные и асинхронные запросы.
- Ajax.
- Объект XMLHttpRequest.
- Методы и свойства XMLHttpRequest.
- Понятие HTTP заголовка.
- Использование метода GET.URL кодирование.
- Использование метода POST.

ECMAScript 6

- История возникновения.
- Переменные.
- Тип данных Symbol.
- Функции-стрелки.
- Использование строк.
- Объекты и классы.
- Модули.
- Другие возможности ECMAScript 6.

Модульное тестирование

- Цели и задачи модульного тестирования.
- Необходимость модульного тестирования.
- Обзор инструментов для модульного тестирования.
- Практические примеры использования.

Паттерны проектирования

- Понятие паттерна проектирования.
- Причины возникновения паттернов проектирования.
- Принципы применения паттернов проектирования.
- Принципы выбора паттернов проектирования.
- Принципы разделения паттернов на категории.
- Введение в UML.
- Использование UML при анализе паттернов проектирования.
- Порождающие паттерны.
- Структурные паттерны.
- Паттерны поведения.

Паттерн MVC

- Цели и задачи паттерна Model-View-Controller.
- Model.
- View.
- Controller.
- Примеры использования паттерна MVC.

Принципы проектирования классов SOLID

- Принципы проектирования классов SOLID.
- Обзор проблем, встречающихся при проектировании и разработке классов.
- Примеры использования принципов SOLID.

Введение в jQuery

- Цели и задачи jQuery.
- История создания jQuery.
- Версии jQuery.

- Подключение jQuery.
- Доступ к элементам страницы при помощи функции \$.
- Понятие селектора.
- Типы селекторов.
- Traversing.
- Методы обхода DOM: filter, next, nextAll, prev, prevAll, siblings и др.

События и jQuery

- Создание обработчиков событий с использованием jQuery.
- Удаление обработчиков событий.
- Объект Event и jQuery.
- Воздействие на обработку события.
- Запуск обработки события.

Стили и анимация

- Метод CSS.
- Отображение и скрытие элементов с помощью методов show и hide.
- Создание эффектов.
- Анимация.

AJAX и jQuery

- Понятие JSON.
- Механизмы Ajax внутри библиотеки jQuery.
- Использование метода GET.
- Использование метода POST.
- События и Ajax в рамках jQuery.
- Обработка ошибок.

Использование jQuery плагинов

- Понятие плагина jQuery.
- Подключение плагина.
- Примеры плагинов.
- Работа в команде, управление программными проектами**
- Управление. Причины возникновения дисциплинарных проблем во время управления программными проектами.
- Диаграммы Ганта.
- Важные вопросы по управлению программными проектами.
- Анализ терминов предметной области.
- Характеристики проекта.
- Расходы, связанные с проектом.
- Общий обзор моделей и методологий процесса разработки.
- Подробнее о Scrum.

ЭКЗАМЕН

Использование фреймворков Angular, React

NodeJS

- Цели и задачи NodeJS.
- Настройка окружения.
- Инструменты для разработки и отладки.
- Структура приложения.
- Менеджер пакетов npm.
- Создание и регистрация собственных пакетов.
- Работа с событиями.
- Использование файловой системы.
- Модули по взаимодействию с HTTP и URL.

Сборка проектов с помощью Webpack 2 и других инструментов

- Сборка проектов.
- Проблемы, возникающие при сборке проекта.
- Инструменты для сборки проектов.
- Webpack 2.

Angular

- Цели и задачи Angular.
- Понятие одностраничного приложения (SPA).
- Анатомия Angular приложения.
- Настройка окружения.
- Понятие модуля.
- Компонент.
- Шаблоны.
- Интерполяция.
- Использование директив.
- Data binding в Angular.
- Pipes в Angular приложении.
- Сервисы.
- Dependency Injection.
- Использование HTTP запросов.
- Навигация и пересылка данных.
- Практические примеры использования.

React

- Цели и задачи React.
- Анатомия React приложения.
- Настройка окружения.
- Понятие Flux.
- Понятие JSX.
- Использование компонентов.
- Жизненный цикл приложения на React.
- Props и State.
- Controller views.
- Понятие Route.
- Использование форм.
- Использование Flux.
- Практические примеры использования.

ЭКЗАМЕН