



Автономная некоммерческая организация
дополнительного профессионального образования
(АНО ДПО «Инфосфера»)

Центр профессиональной подготовки
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ

**Рабочая программа дисциплины
«Основы Front-End разработки»**

Разработал:
преподаватель ИПС
АНО ДПО «Инфосфера»
А.А. Коптелин

Йошкар-Ола, 2017

Пояснительная записка

Целью курса "Front-end-программирование" является изучение теоретических основ и получение практических навыков разработки веб-сервисов. Студенты разрабатывают статические и динамические HTML страницы, изучают способы задания стилового оформления средствами CSS, осваивают принципы клиентского программирования на JavaScript, знакомятся с современными технологиями разработки HTML5 и CSS3.

Планируемые результаты обучения

Осуществляется самостоятельная или под руководством реализация алгоритмов с использованием языков программирования, написание программного кода для создания и актуализации баз данных, реализация запросов с использованием языков определения и манипулирования данными СУБД. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Осуществляется самостоятельное или под руководством оформление программного кода в соответствии с внутренними нормативными документами организации (регламентами, приказами, порядками) и, при необходимости, ГОСТами. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Осуществляется решение типовых задач. Полученные результаты представляются руководителю разработки программного обеспечения.

Проверка работоспособности программного обеспечения осуществляется программистом на основании функциональных требований и технических спецификаций на программное обеспечение самостоятельно путем разработки и исполнения сценариев проверки с применением методов и технологий тестирования и ревьюирования кода.

В ходе проверки работоспособности осуществляется анализ нарушений требований к программному обеспечению, принимаются решения и вносятся изменения в программный код. Программист несет ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Рефакторинг и оптимизация программного кода осуществляется на основании функциональных требований и технических спецификаций на программное обеспечение, в том числе с использованием специализированных программных средств. Программист несет ответственность за решение поставленных задач или результат деятельности группы

работников. Полученные результаты представляются руководителю разработки программного обеспечения.

В процессе интеграции программных модулей и компонент и верификации выпусков программного продукта осуществляется сборка модулей и компонент программного обеспечения, производится интеграция с внешней средой. Обеспечивается согласованное функционирование и требуемый уровень качества.

Проведение интеграции программных модулей и компонент и верификация выпусков программного продукта предполагают определение задач программной интеграции, распределение задач между подчиненными, обеспечение взаимодействия подчиненных сотрудников.

Программист несет ответственность за результат выполнения работ на уровне группы программистов. В процессе интеграции требуется взаимодействие с архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

Учебно-тематический план

№ урока	Тема и ее основное содержание	Кол-во часов
1	Введение в HTML <ul style="list-style-type: none">• Место верстки в процессе разработки• Требования к верстке• История HTML• Парные/одиночные теги	
2	Введение в HTML <ul style="list-style-type: none">• Doctype• Кодировка страницы• Обзор основных тэгов• Валидация страницы• Структура проекта• Соглашение о кодировании	
3	Работа с HTML-формами <ul style="list-style-type: none">• Формы• Методы отправки POST, GET Основные элементы внутри формы и их свойства	
4	Способы организации HTML документа <ul style="list-style-type: none">• Табличная и блочная верстка• Блочные и строчные элементы	
5	Основы CSS <ul style="list-style-type: none">• История и назначение• Способы подключения CSS к странице	

	<ul style="list-style-type: none"> • Приоритеты применения стилей • Виды селекторов (идентификатора, класса, тэга) • Вложенные, комбинированные селекторы 	
6	Основы CSS <ul style="list-style-type: none"> • Комментарии в CSS • Разбор свойств для шрифтового оформления • Настройка Adobe Photoshop для работы над шрифтами • Работа с фоном (css background) • Боксовая модель (css margin, padding, border) 	
7	Основы CSS <ul style="list-style-type: none"> • Горизонтальное выравнивание текста • Блочные и строчные элементы • Ширина и высота 	
8	Основы CSS <ul style="list-style-type: none"> • Позиционирование (position, display) • Поток страницы и свойство float • Очистка потока (clear) • Глубина (z-index) 	
9	Оптимизация Front-end, SEO-верстка	
10	CSS3	
11	Адаптивная верстка	
12	JavaScript. Синтаксис	
13	JavaScript работа со страницей HTML(DOM, селекторы, изменение свойств страницы)	
14	JavaScript объекты	
15	jQuery+AJAX, динамическая подгрузка данных, формат данных JSON	
16	JavaScript Canvas <ul style="list-style-type: none"> • Отрисовка изображений, подгрузка картинок, события мыши 	
17	JavaScript. Современные фреймворки	

Содержание курса

Введение

В результате прохождения данного курса студент приобретает практические навыки разработки веб-сервисов: разработка HTML страниц, клиентское программирование на JavaScript.

Тема 1. Создание HTML страниц

Введение в HTML. Основные теги

Рассматривается структура HTML документа. Исследуется вопрос задания правильного DOCTYPE для HTML страницы и его влияние на отображение страницы в наиболее распространенных браузерах FireFox, Internet Explorer, Chrome.

Исследуется вопрос рационального использования HTML тэгов для формирования страниц. Изучаются способы применения и атрибуты тэгов div, p, a, ul, table.

HTML Формы. Обзор методов запросов. Обработка форм. Метод POST

Изучается возможность создания форм в теле HTML страницы. Исследуются способы передачи данных веб-формы на сервер по протоколу HTTP – отличия методов GET и POST. Изучается способ кодирования данных при передаче по принципу URL Encode.

Тема 2. Задание оформления HTML страниц, каскадные таблицы стилей CSS

Рассказывается о возможности изменения внешнего вида элементов страницы при помощи каскадных стилей. Изучается структура внешнего файла CSS и способ его подключения к HTML странице. Привязывание стилей к заданному элементу страницы при помощи атрибута class.

Шрифты. Разбираются свойства для шрифтового оформления. Происходит знакомство со средой Adobe Photoshop. Рассматриваются средства для работы с фоном.

Позиционирование. Изучаются вопросы расположения элементов на странице, различия блочных и строчных элементов, методы задания ширины и высоты. Позиционирование средствами свойств position и display. Происходит знакомство со средствами управления потоком страницы.

Тема 3. Клиентское программирование на JavaScript

Синтаксис языка JavaScript. Применение JavaScript для задания динамического поведения HTML страницы.

Динамическое клиент-серверное программирование, AJAX

Рассматривается возможность передачи данных от клиента на сервер без перезагрузки страницы при помощи технологии AJAX. Исследуются особенности работы с технологией AJAX в браузерах Firefox, Chrome и Internet Explorer.

Библиотеки JavaScript. jQuery

Рассматриваются готовые решения задач клиентского программирования на основе библиотеки jQuery.

Тема 4. Интерактивные системы на основе HTML5. Анимация средствами Canvas

Возможности HTML5 для работы с аудио и видео. Canvas для рисования. Рассматривается requestAnimationFrame для создания анимаций.

Тема 5. Современные фреймворки для работы с JavaScript

Рассматривается одно или несколько решений на базе JavaScript для разработки клиентской части веб-сервисов.

Методические рекомендации

Курс формирует основу для дальнейшего изучения веб-технологий, дает практические навыки разработки веб-проектов достаточные для их применения при решении производственных задач.

Основными видами учебной работы являются лекции и практические занятия. Каждая тема подробно рассматривается на лекции. В ходе лекции возможны проведение опросов по уже пройденным темам, совместные со студентами рассуждения по новому материалу.

Практические занятия закрепляют изученный лекционный материал. Каждое практическое занятие подразумевает выполнение на компьютере ряда практических задач по наиболее важным темам.

Самостоятельная работа включает процедуры самообучения слушателей курса, инициируемые и управляемые со стороны преподавателя в режиме их подготовки к лекциям, семинарам, практическим работам, сдаче экзаменов и зачетов; процедуры самообучения в условиях свободы выбора заданий для самостоятельного освоения новых знаний, овладения умениями, используя весь арсенал современных источников информации (учебники, учебные пособия, ресурсы интернета, собственный опыт); консультирование преподавателя.

При организации учебных занятий используются активные методы обучения (работа в группах, взаимообучение, самоконтроль, индивидуальные задания дифференцированной сложности).

В процессе обучения возможно использование следующих тактических технологий: лекция классическая, лекция проблемная, лекция-визуализация, лекция-диалог, аудиторно-практическое занятие классическое, практикум-лабораторная работа, самообучение.

Контрольные задания

Контрольные вопросы для самостоятельной оценки качества освоения дисциплины

1. Создание HTML страниц.
2. HTML Формы. Обзор методов запросов. Обработка форм. Метод POST
3. Задание оформления HTML страниц, каскадные таблицы стилей CSS
4. Клиентское программирование на JavaScript
5. Динамическое клиент-серверное программирование, AJAX
6. Библиотеки JavaScript. jQuery
7. Интерактивные системы на основе HTML5. Анимация средствами Canvas
8. Современные фреймворки для работы с JavaScript