1. **Паспорт Образовательной программы**

«Технологии веб-дизайна и разработки»

|  |  |
| --- | --- |
| **Версия программы** | **1** |
| **Дата Версии** | 29.09.2020 |

1. **Сведения о Провайдере**

|  |  |  |
| --- | --- | --- |
| 1.1 | Провайдер | ГАПОУ ОНТ имени В.А. Сорокина |
| 1.2 | Логотип образовательной организации | 0RIEqzEHaG4 |
| 1.3 | Провайдер ИНН | ИНН 5615003044 |
| 1.4 | Ответственный за программу ФИО | Саладай Алексей Владимирович |
| 1.5 | Ответственный должность | Преподаватель специальных дисциплин ГАПО ОНТ им.В.А.Сорокина |
| 1.6 | Ответственный Телефон | 89619043928 |
| 1.7 | Ответственный Е-mail | saladay@yandex.ru |

1. **Основные Данные**

|  |  |  |
| --- | --- | --- |
| **№** | **Название** | **Описание** |
| 2.1 | Название программы | «Технологии веб-дизайна и разработки» |
| 2.2 | Ссылка на страницу программы | [http://portal.ontvkr.ru/course/view.php?id=133](https://vk.com/away.php?to=http%3A%2F%2Fportal.ontvkr.ru%2Fcourse%2Fview.php%3Fid%3D133&cc_key=) |
| 2.3 | Формат обучения | Онлайн |
|  | Подтверждение от ОО наличия возможности реализации образовательной программы с применением электронного обучения и (или) дистанционных образовательных технологий с возможностью передачи данных в форме элементов цифрового следа | hortal.ontvkr.ru  http://nt-orsk.ru/ |
| 2.4 | Уровень сложности | Базовый |
| 2.5 | Количество академических часов | **72**ч |
|  | Практикоориентированный характер образовательной программы: не менее 50 % трудоёмкости учебной деятельности отведено практическим занятиям и (или) выполнению практических заданий в режиме самостоятельной работы (кол-во академических часов) | 44ч |
| 2.6 | Стоимость обучения одного обучающегося по образовательной программе, а также предоставление ссылок на 3 (три) аналогичные образовательные программы иных организаций, осуществляющих обучение, для оценки объективности стоимости или обоснование уникальности представленной образовательной программы в случае отсутствия аналогичных образовательных программ на рынке образовательных услуг | 15000  <http://idpo.magtu.ru/index.php/component/jshopping/product/view/28/112?Itemid=0>  <http://bakalavr-magistr.ru/course/Prakticheskoe-veb-programmirovanie>  https://do.tusur.ru/web-development |
| 2.7 | Минимальное количество человек на курсе | 100 |
| 2.8 | Максимальное количество человек на курсе | 150 |
| 2.9 | Данные о количестве слушателей, ранее успешно прошедших обучение по образовательной программе | - |
| 2.10 | Формы аттестации | ДЭ |
|  | Указание на область реализации компетенций цифровой экономики, к которой в большей степени относится образовательная программа, в соответствии с Перечнем областей | Программирование и создание ИТ-продуктов |

1. **Аннотация программы**

Веб-разработчик является динамичной, постоянно меняющейся профессией, сферой деятельности которой является создание и функционирования веб-сайтов. Веб-разработчики используют для создания вебсайтов специальные программы и языки программирования и разметки, которые связывают ссылки на различные веб-страницы, другие веб-сайты, графические элементы, текст и фото в единый функциональный и удобный информационный продукт.

Компьютерные программы, заготовки и открытые электронные библиотеки используются в качестве технической базы. В своей работе разработчики сайтов обязаны обращать внимание на закон об авторском праве и этические вопросы.

В наши дни каждый может попробовать свои силы в веб-разработке, оказывая все большее влияние на разработчиков-профессионалов. Чтобы пробудить интерес у посетителей сайта, последние обязаны изучать новые техники и технологии производства сайтов и использовать их при решении оригинальных задач.

Веб-разработчик осведомлен как в области технологий, так и в графическом дизайне. На сайтах веб-технология используется в том числе для автоматизации функций и помощи в управлении контентом. Творческие способности нужны веб-разработчикам при подборе цветов, шрифтов и графики, для поддержки эффективной рабочей коммуникации с профессиональными дизайнерами, а также при разработке структуры сайта. Хорошо спланированный пользовательский интерфейс (ПИ) гарантирует интерес посетителя к открытой и другим страницам сайта и, как следствие, высокую его конверсию. Веб-разработчик так же обязан знать основы проектной работы, продукцию, которой посвящен контент сайта, знать технологии и методы программирования на стороне сервера и клиента, разбираться в основах программной архитектуры и базах данных для хранения информации и организации сложных веб-сервисов, и основы управления сайтом. Совместимость конечного продукта с современными версиями наиболее распространенных веб-браузеров, программам и устройств обязательна.

Курс повышения квалификации позволит слушателю получить знания, умения и навыки, необходимые для успешной деятельности по следующим направлениям:

- использование современных систем управления контентом;

- разработка графического дизайна веб-страниц;

- осуществление вёрстки веб-страниц;

- программирование на стороне клиента и на стороне сервера;

- работа с базами данных MySQL

Министерство образования Оренбургской области

Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»

Дополнительная профессиональная программа повышения квалификации

« Технологии веб-дизайна и разработки»

72 час.

**ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ**

**1.Цель программы**: дополнительная профессиональная программа повышения квалификации направлена на совершенствование и (или) получение новой компетенции, необходимой для профессиональной деятельности, и (или) повышение профессионального уровня в рамках имеющейся квалификации, по компетенции «Веб-дизайн и разработка».

**2.Планируемые результаты обучения:**

2.1.Знание (осведомленность в областях)

2.1.1. Спецификации стандарта компетенции «Веб-дизайн разработка»;

2.1.2. Требования охраны труда и техники безопасности;

2.1.3. Порядка работы с оргтехникой и правила технической безопасности;

2.1.4. Наиболее востребованные информационно-коммуникационные технологии;

2.1.5. Правил оформления информационно-презентационных материалов;

2.1.6. Разработки дизайна сайта; верстки сайта по макету;

2.1.7. Разработки клиентской части сайта;

2.1.8. Разработки серверной части сайта; разработка сайта с использованием CMS;

2.1.9. Порядка организации и проведения групповых и массовых мероприятий;

2.1.10 Понятия системы программирования;

2.1.11 Основных элементов процедурного языка программирования, структура программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;

2.1.12. Подпрограммы, составление библиотек программ;

2.1.13. Объектно-ориентированной модели программирования, понятие классов и объектов, их свойства и методы.

2.2. Умение (способность к деятельности)

2.2.1. Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;

2.2.2 Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;

2.2.3 Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;

2.2.4. Создавать веб-сайты полностью соответствующие текущим стандартам W3C (http://www.w3.org);

2.2.5. Создавать и модифицировать сайты с учетом Search Engine Optimization.

2.2.6. Разрабатывать анимацию для веб-сайта для повышения его доступности и визуальной привлекательности;

2.2.7. Создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;

2.2.8. Применять открытые библиотеки; создавать библиотеки и модули для выполнения повторяющихся задач;

2.2.9. Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;

2.2.10. Создавать SQL (Structured Query Language) запросы, используя корректный синтаксис (классический и PDO (PHP Data Object));

2.2.11. Обеспечивать безопасность (устойчивость веб-приложения к атакам и взлому);

2.2.12. Интегрировать существующий программный код с API (Application Programming Interfaces), библиотеками и фреймворками;

2.2.13. Разрабатывать объектно-ориентированный программный код;

2.2.14. Применять нормативные документы, определяющие требования к оформлению страниц ИР;

2.2.15. Определять возможности отображения веб-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов;

2.2.16. Применять специализированное программное обеспечение для верстки страниц ИР; использовать язык разметки страниц ИР;

2.2.17. Применять выбранные языки программирования для написания программного кода;

2.2.18. Использовать выбранную среду программирования и средства системы управления базами данных;

2.2.19. Использовать возможности имеющейся программной архитектуры ИР; мспользовать существующие типовые решения и шаблоны ИР;

2.2.20 применять методы и средства проектирования ИР, структур данных, баз данных, программных интерфейсов;

2.2.21 Применять методы и средства планирования и контроля (мониторинга) исполнения планов;

Размещать текстовую и графическую информацию на страницах сайта;

2.2.22. Редактировать гипертекстовую разметку сайта с использованием систем администрирования;

2.2.23. Применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов;

2.2.24. Осуществлять коммуникации с заинтересованными сторонами;

2.2.25. Разрабатывать программный код клиентской и серверной части веб-приложений;

2.2.26. Осуществлять оптимизацию веб-приложения с целью повышения его рейтинга в сети Интернет.

2.3.Навыки (использование конкретных инструментов)

2.3.1. создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;

2.3.2. обеспечения безопасность и защищенность созданных веб-приложений;

2.3.3. выбора цвета, типографики, композиции;

2.3.4 применения World Wide Web Consortium (W3C) стандартов HTML и CSS;

2.3.5 создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем;

2.3.6 разработки кода с использованием открытых библиотек и фреймворков, объектно-ориентированной парадигмы JavaScrip, Yii2, Laravel;

2.3.7 разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON;

2.3.8 разработки программного кода в соответствии с паттернами (например, MVC (Model View Controller);

2.3.9 поиска, выбора и подключения нужных плагинов/модулей;

**3.Категория слушателей** (возможно заполнение не всех полей)

* 1. Специалисты со среднем профессиональным образованием, бакалавры, специалисты с высшем профессиональным образованием
  2. -
  3. Базовый уровень-
  4. -

**4.Учебный план программы «**Технологии веб-дизайна и разработки **»**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№ п/п** | **Модуль** | **Всего, час** | **Виды учебных занятий** | | |
| **лекции** | **практические занятия** | **самостоятельная работа** |
| 1 | Модуль 1. Современные технологии в профессиональной сфере | 8 | 8 | - | 4 |
| 2 | Модуль 2. Графический дизайн веб-страниц | 8 | 2 | 4 | 2 |
| 3 | Модуль 3. Верстка страниц | 8 | 2 | 6 | - |
| 4 | Модуль 4. Основы CSS | 8 | 2 | 4 | 2 |
| 5 | Модуль 5. Программирование на стороне клиента и на стороне сервера | 8 | 2 | 6 |  |
| 6 | Модуль 6. Базы данных. Основы MySQL | 4 | 2 | 2 |  |
| 7 | Модуль 7. Системы управления контентом | 20 | 2 | 16 | 2 |
| **Итоговая аттестация** | |  | **Указывается вид** | | |
| 8 | | 8 | экзамен | | |

**5.Календарный план-график реализации образовательной** программы

(дата начала обучения – дата завершения обучения) в текущем календарном году, указания на периодичность набора групп (не менее 1 группы в месяц)

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Наименование учебных модулей** | **Трудоёмкость (час)** | **Сроки обучения** |
| 1 | Модуль 1. Современные технологии в профессиональной сфере | 8 | 01.11-02.11 |
| 2 | Модуль 2. Графический дизайн веб-страниц | 8 | 02.11-03.11 |
| 3 | Модуль 3. Верстка страниц | 8 | 03.11-04.11 |
| 4 | Модуль 4. Основы CSS | 8 | 05.11-08.11 |
| 5 | Модуль 5. Программирование на стороне клиента и на стороне сервера | 8 | 08.11-09.11 |
| 6 | Модуль 6. Базы данных. Основы MySQL | 4 | 09.11-10.11 |
| 7 | Модуль 7. Системы управления контентом | 20 | 11.11-14.11 |
|  | Итоговая аттестация (демонстрационный экзамен) | 8 | 15.11 |
| **Всего:** | | 72 | 01.11-15.11 |

**6.Учебно-тематический план программы «**  Технологии веб-дизайна и разработки **»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№ п/п** | **Модуль / Тема** | **Всего, час** | **Виды учебных занятий** | | | **Формы контроля** |
| **лекции** | **практические занятия** | **самостоятельная работа** |
| 1 | Модуль 1. Современные технологии в профессиональной сфере | 8 | 4 | - | 4 | - |
| 1.1 | Современные информационно-коммуникационные технологии в профессиональной деятельности | 4 | 2 | - | 2 | Самостоятельная работа |
| 1.2 | Перспективы развития веб-технологий. | 4 | 2 | - | 2 | Самостоятельная работа |
| 2 | Модуль 2. Графический дизайн веб-страниц | 8 | 2 | - | - | - |
| 2.1 | Создание дизайна по предоставляемым инструкциям и спецификациям | 2 | 1 | 1 | - | Практическая работа |
| 2.2 | Выбор цвета, типографики и композиции. Современные стили и тенденции дизайна | 2 | 1 | 1 | - | Практическая работа |
| 2.3 | Создание макета сайта согласно требованиям ТЗ | 4 | - | 2 | 2 | Практическая работа |
| 3 | Модуль 3. Верстка страниц | 8 | 2 | 6 | - | - |
| 3.1 | Структура HTML- Документа. Разметка текста. Ссылки и изображения. Знакомство с таблицами и формами | 4 | 2 | 2 | - | Практическая работа |
| 3.2 | Основы HTML5, Формы в HTML5 | 2 | - | 2 | - | Практическая работа |
| 3.3 | Форматирование документа. Списки.  Гиперссылки | 2 | - | 2 | - | Практическая работа |
| 4 | Модуль 4. Основы CSS | 8 | 2 | 4 | 2 | - |
| 4.1 | Работа со шрифтами в CSS | 2 | 1 | 1 | - | Практическая работа |
| 4.2 | Работа с текстом в CSS | 2 | 1 | 1 | 1 | Практическая работа |
| 4.3 | Работа с фоном с CSS | 1 | - | 1 | - | Практическая работа |
| 4.4 | Оформление списков в CSS | 1 | - | 1 | - | Практическая работа |
| 4.5 | Дополнительные элементы управления элементами в CSS | 1 | - | 1 | - | Практическая работа |
| 4.6 | Введение в анимацию | 1 | - | 1 | - | Практическая работа |
| 5 | Модуль 5. Программирование на стороне клиента и на стороне сервера | 8 | 2 | 6 | - | - |
| 5.1 | Основы программирования на JavaScript | 4 | 1 | 3 | - | Практическая работа |
| 5.2 | Основы программирования на PHP | 4 | 1 | 3 | - | Практическая работа |
| 6 | Модуль 6. Базы данных. Основы MySQL | 4 | 2 | 2 | - | - |
| 6.1 | Основные понятия MySQL | 2 | 1 | 1 | - | Практическая работа |
| 6.2 | Основы языка SQL | 2 | 1 | 1 | - | Практическая работа |
| 7 | Модуль 7. Системы управления контентом | 20 | 2 | 18 | - | - |
| 7.1 | Знакомство с CMS WordPress | 2 | 2 | - | - | Практическая работа |
| 7.2 | Выбор доменного имени и хостинга | 2 | - | 2 | - | Практическая работа |
| 7.3 | Установка и настройка WordPress | 6 | - | 4 | 2 | Практическая работа |
| 7.4 | Подготовка и публикация материалов | 2 | - | 2 | - | Практическая работа |
| 7.5 | Управление контентом | 2 | - | 2 | - | Практическая работа |
| 7.6 | Дизайн сайта | 4 | - | 4 | - | Практическая работа |
| 7.7 | Работа с плагинами WordPress | 2 | - | 2 | - | Практическая работа |
| 8 | Итоговая аттестация | 8 |  |  | - |  |
| 8.1 | Демонстрационный экзамен по компетенции | 8 |  |  | - | Практическая работа |

**7. Учебная (рабочая) программа повышения квалификации «**  Технологии веб-дизайна и разработки   **»**

**Модуль 1.** Современные профессиональные технологии **( \_**8**\_ час.)**

**Тема 1.1** Современные информационно-коммуникационные технологии в профессиональной деятельности **( \_\_**4**\_ час)**

**Содержание темы:**

Современная ИТ - информационная технология— это ИТ с "дружественным" интерфейсом работы пользователя, использующая персональные компьютеры и телекоммуникационные средства.

Три основных принципа компьютерной ИТ:

· Интерактивный (диалоговый) режим работы с компьютером;

· Интегрированность(стыковка, взаимосвязь) с другими программными продуктами;

· Гибкость процесса изменения, как данных, так и постановок задач.

Основу современных ИТ - информационных технологий составляют три технических достижения:

· появление новой среды накопления информации;

· развитие современные средства связи, в том числе и спутниковых;

· постоянное совершенствование компьютеров и ПО для автоматизированной обработки и передачи информации в реальном масштабе времени.

**Тема 1.2** Перспективы развития веб-технологий ( \_\_4\_ час)

Содержание темы

Progressive Web Apps (PWA – прогрессивные веб-приложения). В последнее время, популярность нативных мобильных приложений превзошла все ожидания. Прогрессивные веб-приложения нового поколения (PWA) собрали информацию о лучшем пользовательском опыте и воплотили это в мобильных приложениях. На самом деле, технология не новая, так как PWA появились еще в 2015 году. Тем не менее, разработчикам потребовалось два года, чтобы превратить PWA в одну из основных тенденций веб-программирования.

Чат боты и искусственный интеллект. В настоящее время «Искусственный интеллект» покинул сферу научно-фантастических технологий и уверенно вошел в число 10 лучших трендов веб разработки в 2020 году. Крупные компании, такие как Facebook, Google, Microsoft, IBM и Amazon, увидели перспективы для инвестиций в развитие ИИ и сделали все возможное для частичной интеграции технологий искусственного интеллекта на своих площадках. Они каждый день вкладывают значительные средства в развитие технологии чат ботов.

Тренды веб разработки: Блокчейн Блокчейн (blockchain) – это быстро развивающаяся технология, которая трансформирует всю суть бизнеса. Сама идея принадлежит человеку или по имени Сатоши Накамото. Первоначально эта технология поддерживала цифровую валюту Bitcoin, но теперь для нее нашли много других применений, и она стала действительно революционной.

Motion UI – анимация и переходы Следующий тренд веб разработки не столь далекий от обычного пользователя, как ИИ и блокчейны. Он предсказывает популярность и широкое распространение библиотеки Motion UI. Эта библиотека позволяет мгновенно анимировать пользовательский интерфейс вашего сайта используя моушн-дизайн.

**Модуль 2.** Графический дизайн веб-страниц **(** 8 **час.)**

**Тема 2.1.** Создание дизайна по предоставляемы инструкциям и спецификациям (1 час)

Содержание темы:

Объектом дизайна может стать практически любое новое техническое промышленное изделие (комплект, ансамбль, комплекс, система) в любой сфере жизнедеятельности людей. Согласно объектам деятельности разделяют виды дизайна. Различают

Графический дизайн

Индустриальный дизайн

Компьютерный дизайн

Веб-дизайн

Дизайн интерьеров (общественные здания, жилые помещения, промышленные здания)

Городской дизайн

Ландшафтный дизайн

Фитодизайн

Световой дизайн

Архитектурный дизайн

Звуковой дизайн

Визаж

Одежда

Прически

Экодизайн

Основные этапы дизайна:

Этап 1. Получение задания на проектирование (техническое задание)

Этап 2. Пред проектные исследования

Этап 3. Эскизное проектирование

Этап 4. Техническое проектирование.

Этап 5. Рабочее проектирование.

Тема 2.2. Выбора цвета, типографики и композиции. Современные стили и тенденции дизайна (1 час.)

Содержание темы:

Главные тренды веб-дизайна в 2020 году

1. Смелые, яркие цвета

2. Яркие и насыщенные градиенты  
3. Простота и удобство  
4. Экстра глубина (и полуплоский дизайн)

5. Геометрические элементы

Модуль 3. Верстка страниц ( 8 час.)

Тема 3.1. Структура HTML- Документа. Разметка текста. Ссылки и изображения. Работа с тэгами и атрибутами тэгов для разметки текста. (2 час)

Содержание темы:

HTML-теги — основа языка HTML. Теги используются для разграничения начала и конца элементов в разметке.

Каждый HTML-документ состоит из дерева HTML-элементов и текста. Каждый HTML-элемент обозначается начальным (открывающим) и конечным (закрывающим) тегом. Открывающий и закрывающий теги содержат имя тега.

Все HTML-элементы делятся на пять типов:

пустые элементы — <area>, <base>, <br>, <col>, <embed>, <hr>, <img>, <input>, <link>, <menuitem>, <meta>, <param>, <source>, <track>, <wbr>;

элементы с неформатированным текстом — <script>, <style>;

элементы, выводящие неформатированный текст — <textarea>, <title>;

элементы из другого пространства имён — MathML и SVG;

обычные элементы — все остальные элементы.

Модуль 4. Основы CSS ( 8 час.)

Тема 4.1. Работа со шрифтами в CSS (1 час)

Содержание темы:

Безопасные веб-шрифты

В CSS стиле для выбора типа шрифта применяется свойство [font-family](https://basicweb.ru/css/css_pr_font-family.php), в котором указывается интересующий Вас шрифт. Предположим, что вы хотите применить для абзацев страницы шрифт Courier. В этом случае Вам необходимо будет создать, например, селектор типа и воспользоваться свойством [font-family](https://basicweb.ru/css/css_pr_font-family.php):

p {

font-family : Courier; /\* устанавливаем тип шрифта – Courier \*/

}

Главная особенность данного способа заключается в том, что он будет работать, при условии, что у посетителя установлен подобный шрифт, иначе, страница будет отображена с использованием шрифта "встроенного" в браузер.

Так как вы заранее не знаете, есть у пользователя тот, или иной шрифт, то рекомендуется указывать не только основной шрифт, но и пару запасных (альтернативных) шрифтов, для того случая если у пользователя отсутствует основной шрифт.

Рекомендуется последним в списке шрифтов указывать и семейство шрифта (generic-family). Если у пользователя по каким-то причинам отсутствуют все перечисленные Вами шрифты, то в этом случае страница будет отображена хотя бы шрифтом того же семейства, а не шрифтом "встроенным" в браузер.

Типы веб-шрифтов и их поддержка браузерами

Все современные браузеры поддерживают использование определённых веб-шрифтов. Происходит это следующим образом: браузер пользователя загружает шрифт с указанного сервера и применяет его для отображения текущей страницы. В настоящее время существуют следующие виды веб-шрифтов:

TTF/OTF (True Type и Open Type Fonts) - эти шрифты имеют широкую поддержку. Разработаны Microsoft совместно с Adobe, с целью применения в различных операционных системах.

WOFF (Web Open Font Format) - сжатая версия шрифтов TTF/OTF. Формат включает в себя метаданные, в которые автор шрифта может добавить информацию об использовании шрифта. WOFF-формат имеет широкую поддержку со стороны браузеров.

WOFF2 (Web Open Font Format 2) - спецификация была разработана, чтобы обеспечить улучшенное сжатие и тем самым снизить использование пропускной способности сети, в то же время, позволяя быстро производить декомпрессию даже на мобильных устройствах.

SVG (Scalable Vector Graphic) – способ создания векторной графики. SVG-формат имеет очень ограниченную поддержку (IOS/Safari). Планируется, что он перестанет использоваться в Chrome.

EOT (Embedded Open Type) – шрифты, которые поддерживаются только в Internet Explorer/Edge (разработаны компанией Microsoft для использования в качестве встроенных шрифтов на веб-страницах).

Тема 4.2. Работа с текстом в CSS (1 час)

Содержание темы:

Работа с текстом в CSS



CSS дает широкие возможности по оперированию стилем текста. Можно влиять на шрифт, цвет, размер и другие свойства текста.

Так как эта страница расчитана на новичков, то сначала напомним базовый синтаксис CSS и используемую терминологию.

Каждое правило CSS состоит из селектора и определения. Селектор - обычно это тег, к которому мы применяем наш стиль, а определение - это само стилевое отображение. Определение состоит из свойства и его значения. Например: свойство - цвет, значение - красный.

Синтаксис CSS следующий:

селектор { свойство: значение }

Пример правила CSS:

p { color: blue }

В этом примере задано правило тегу <p> отображать текст синим цветом. Одному селектору можно указать несколько пар (свойство : значение).

p { color: blue; font-size: 10pt }

В этом примере задано правило тегу <p> отображать текст синим цветом и размером 10pt. То есть тут мы имеем две пары (свойство : значение) применённые к одному селектору, в данном случае к тегу абзаца <p>.

В дальнейшем, когда в тексте будут встречаться слова свойство и значение, будет иметься ввиду именно определение для селектора в правиле CSS.

Свойства шрифта

Если не указывать это свойство в CSS, то броузер будет показывать текст шрифтом по умолчанию. Обычно это Times New Roman.

Список шрифтов в значении font-family может включать одно или несколько названий, разделенных запятой. Если в имени шрифта содержатся пробелы, например, Trebuchet MS, оно должно заключаться в кавычки, одинарные или же двойные.

Браузер поочерёдно проверяет наличие шрифтов из списка на компьютере пользователя. Если первый шрифт в списке есть, то используется он, иначе браузер проверяет наличие второго шрифта из списка. Если ни одного нет, используется шрифт по умолчанию. В конце списка обычно прописывают ключевые слово, которое описывает тип желаемого шрифта — serif, sans-serif, fantasy, cursive или monospace.

Модуль 5. Программирование на стороне клиента и на стороне сервера 98 час.)

Тема 5.1. Основы программирования на JavaScript (1 час)

Содержание темы:

Изначально JavaScript был создан, чтобы «сделать веб-страницы живыми».

Программы на этом языке называются скриптами. Они могут встраиваться в HTML и выполняться автоматически при загрузке веб-страницы.

Скрипты распространяются и выполняются, как простой текст. Им не нужна специальная подготовка или компиляция для запуска.

Это отличает JavaScript от другого языка – [Java](https://ru.wikipedia.org/wiki/Java).

Почему JavaScript?

Когда JavaScript создавался, у него было другое имя – «LiveScript». Однако, язык Java был очень популярен в то время, и было решено, что позиционирование JavaScript как «младшего брата» Java будет полезно.

Со временем JavaScript стал полностью независимым языком со своей собственной спецификацией, называющейся [ECMAScript](http://ru.wikipedia.org/wiki/ECMAScript), и сейчас не имеет никакого отношения к Java.

Сегодня JavaScript может выполняться не только в браузере, но и на сервере или на любом другом устройстве, которое имеет специальную программу, называющуюся [«движком» JavaScript](https://ru.wikipedia.org/wiki/%D0%94%D0%B2%D0%B8%D0%B6%D0%BE%D0%BA_JavaScript).

У браузера есть собственный движок, который иногда называют «виртуальная машина JavaScript».

Разные движки имеют разные «кодовые имена». Например:

[V8](https://ru.wikipedia.org/wiki/V8_(%D0%B4%D0%B2%D0%B8%D0%B6%D0%BE%D0%BA_JavaScript)) – в Chrome и Opera.

[SpiderMonkey](https://ru.wikipedia.org/wiki/SpiderMonkey) – в Firefox.

…Ещё есть «Trident» и «Chakra» для разных версий IE, «ChakraCore» для Microsoft Edge, «Nitro» и «SquirrelFish» для Safari и т.д.

Эти названия полезно знать, так как они часто используются в статьях для разработчиков. Мы тоже будем их использовать. Например, если «функциональность X поддерживается V8», тогда «Х», скорее всего, работает в Chrome и Opera.

Как работают движки?

Движки сложны. Но основы понять легко.

Движок (встроенный, если это браузер) читает («парсит») текст скрипта.

Затем он преобразует («компилирует») скрипт в машинный язык.

После этого машинный код запускается и работает достаточно быстро.

Движок применяет оптимизации на каждом этапе. Он даже просматривает скомпилированный скрипт во время его работы, анализируя проходящие через него данные, и применяет оптимизации к машинному коду, полагаясь на полученные знания. В результате скрипты работают очень быстро.

[Что может JavaScript в браузере?](https://learn.javascript.ru/intro" \l "chto-mozhet-javascript-v-brauzere)

Современный JavaScript – это «безопасный» язык программирования. Он не предоставляет низкоуровневый доступ к памяти или процессору, потому что изначально был создан для браузеров, не требующих этого.

Возможности JavaScript сильно зависят от окружения, в котором он работает. Например, [Node.JS](https://ru.wikipedia.org/wiki/Node.js) поддерживает функции чтения/записи произвольных файлов, выполнения сетевых запросов и т.д.

В браузере для JavaScript доступно всё, что связано с манипулированием веб-страницами, взаимодействием с пользователем и веб-сервером.

Например, в браузере JavaScript может:

Добавлять новый HTML-код на страницу, изменять существующее содержимое, модифицировать стили.

Реагировать на действия пользователя, щелчки мыши, перемещения указателя, нажатия клавиш.

Отправлять сетевые запросы на удалённые сервера, скачивать и загружать файлы (технологии [AJAX](https://ru.wikipedia.org/wiki/AJAX) и [COMET](https://ru.wikipedia.org/wiki/Comet_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5))).

Получать и устанавливать куки, задавать вопросы посетителю, показывать сообщения.

Запоминать данные на стороне клиента («local storage»).

[Чего НЕ может JavaScript в браузере?](https://learn.javascript.ru/intro" \l "chego-ne-mozhet-javascript-v-brauzere)

Возможности JavaScript в браузере ограничены ради безопасности пользователя. Цель заключается в предотвращении доступа недобросовестной веб-страницы к личной информации или нанесения ущерба данным пользователя.

Примеры таких ограничений включают в себя:

JavaScript на веб-странице не может читать/записывать произвольные файлы на жёстком диске, копировать их или запускать программы. Он не имеет прямого доступа к системным функциям ОС.

Современные браузеры позволяют ему работать с файлами, но с ограниченным доступом, и предоставляют его, только если пользователь выполняет определённые действия, такие как «перетаскивание» файла в окно браузера или его выбор с помощью тега <input>.

Существуют способы взаимодействия с камерой/микрофоном и другими устройствами, но они требуют явного разрешения пользователя. Таким образом, страница с поддержкой JavaScript не может незаметно включить веб-камеру, наблюдать за происходящим и отправлять информацию в [ФСБ](https://ru.wikipedia.org/wiki/%D0%A4%D0%B5%D0%B4%D0%B5%D1%80%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%BB%D1%83%D0%B6%D0%B1%D0%B0_%D0%B1%D0%B5%D0%B7%D0%BE%D0%BF%D0%B0%D1%81%D0%BD%D0%BE%D1%81%D1%82%D0%B8_%D0%A0%D0%BE%D1%81%D1%81%D0%B8%D0%B9%D1%81%D0%BA%D0%BE%D0%B9_%D0%A4%D0%B5%D0%B4%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%B8).

Различные окна/вкладки не знают друг о друге. Иногда одно окно, используя JavaScript, открывает другое окно. Но даже в этом случае JavaScript с одной страницы не имеет доступа к другой, если они пришли с разных сайтов (с другого домена, протокола или порта).

Это называется «Политика одинакового источника» (Same Origin Policy). Чтобы обойти это ограничение, обе страницы должны согласиться с этим и содержать JavaScript-код, который специальным образом обменивается данными.

Это ограничение необходимо, опять же, для безопасности пользователя. Страница https://anysite.com, которую открыл пользователь, не должна иметь доступ к другой вкладке браузера с URL https://gmail.com и воровать информацию оттуда.

JavaScript может легко взаимодействовать с сервером, с которого пришла текущая страница. Но его способность получать данные с других сайтов/доменов ограничена. Хотя это возможно в принципе, для чего требуется явное согласие (выраженное в заголовках HTTP) с удалённой стороной. Опять же, это ограничение безопасности.

Тема 5.2. Основы программирования на PHP (1 час)

Содержание темы:

Данные - это объекты, которыми может оперировать PHP, к примеру, сообщение пользователя - это строка, а его возраст - это число. Данные могут быть разных типов.

Строки и числа

Самыми простыми типами данных в PHP являются строки и числа.

Числа обозначают сами себя, например 1, 12, 145, а вот строки требуется брать в кавычки (одинарные или двойные - без разницы):

<?php

'строка', "строка"; //это примеры строк

?>

Числа также могут быть дробями. Разделителем целой и дробной части служит знак "точка". К примеру, число 10.754 - это дробь.

Переменные

Одним из самых главных и распространенных объектов в программировании является переменная.

Переменная - это такой объект, который может хранить внутри себя различные данные, например, строки или числа.

Имя переменной должно состоять из английских букв: больших или маленьких, а также цифр и знака\_подчеркивания.

В PHP перед переменной обязательно должен быть написан знак доллара $:

<?php

$a, $a1, $isVar, $is\_var; //это примеры переменных

?>

Операция присваивания

Очень важным элементом программирования является операция присваивания.

С помощью присваивания мы можем положить в переменную результат какой-либо операции, например, строку или число, а затем достать эти данные в нужном нам месте и воспользоваться ими.

Пример присваивания:

<?php

$a = 4; //мы присвоили переменной $a значение 4

?>

Комментарии

К коде PHP, так же, как и в HTML и CSS, можно оставлять комментарии, которые будут проигнорированы PHP при выполнении программы.

Их обычно используют для того, чтобы оставлять какие-либо пометки в коде или временно убрать какие-то строки кода, которые вам мешают, а потом иметь возможность быстро их вернуть обратно.

Комментарии могут быть многострочными и однострочными:

<?php

$a = 4; //это пример однострочного комментария

?>

<?php

/\*

Это пример

многострочного комментария.

\*/

$a = 4;

?>

Функции

В программировании есть такое понятие, как функции. В отличии от переменных, которые хранят внутри себя некоторые данные, функции выполняют операции над этими данными.

Функция отличается от переменной тем, что перед именем функции не ставится знак доллара (он нужен только для переменных), а также тем, что у нее после названия стоят круглые скобки ( ), внутри которых через запятую можно перечислять параметры.

Параметры - это данные (например, строки или числа), которые мы даем функции, она производит над ними какие-то операции и возвращает нам результат.

Например, в PHP есть стандартная функция sqrt(), которая параметром принимает число, а возвращает квадратный корень этого числа.

То есть, если ей передать число 9, вот так - sqrt(9), то на выходе получим число 3. Полученное число мы можем записать в какую-либо переменную или вывести на экран.

Модуль 6. Базы данных. Основы MySQL (4час.)

Тема 6.1 Основные понятия MySQL (1 час.)

Содержание темы:

Основные характеристики MySQL

Клиентская программа MySQL представляет собой утилиту командной строки. Эта программа подключается к серверу по сети. Команды, выполняемые сервером, обычно связаны с чтением и записью данных на жестком диске.

Клиентские программы могут работать не только в режиме командной строки. Есть и графические клиенты, например MySQL GUI, PhpMyAdmin и др. Но они – тема отдельного курса.

MySQL взаимодействует с базой данных на языке, называемом SQL (Structured Query Language — язык структурированных запросов).

SQL предназначен для манипуляции данными, которые хранятся в Системах управления реляционными базами данных (RDBMS). SQL имеет команды, с помощью которых данные можно извлекать, сортировать, обновлять, удалять и добавлять. Стандарты языка SQL определяет ANSI (American National Standards Institute). В настоящее время действует стандарт, принятый в 2003 году (SQL-3).

SQL можно использовать с такими RDBMS как MySQL, mSQL, PostgreSQL, Oracle, Microsoft SQL Server, Access, Sybase, Ingres. Эти системы RDBMS поддерживают все важные и общепринятые операторы SQL, однако каждая из них имеет множество своих собственных патентованных операторов и расширений.

SQL является общим языком запросов для нескольких баз данных различных типов. Данный курс рассматривает систему MySQL, которая является RDBMS c открытым исходным кодом, доступной для загрузки на сайте MySQL.com.

Вот как характеризуют MySQL её разработчики.

MySQL - это система управления базами данных.

База данных представляет собой структурированную совокупность данных. Эти данные могут быть любыми - от простого списка предстоящих покупок до перечня экспонатов картинной галереи или огромного количества информации в корпоративной сети. Для записи, выборки и обработки данных, хранящихся в компьютерной базе данных, необходима система управления базой данных, каковой и является ПО MySQL. Поскольку компьютеры замечательно справляются с обработкой больших объемов данных, управление базами данных играет центральную роль в вычислениях. Реализовано такое управление может быть по-разному - как в виде отдельных утилит, так и в виде кода, входящего в состав других приложений.

MySQL - это система управления реляционными базами данных.

В реляционной базе данные хранятся в отдельных таблицах, благодаря чему достигается выигрыш в скорости и гибкости. Таблицы связываются между собой при помощи отношений, благодаря чему обеспечивается возможность объединять при выполнении запроса данные из нескольких таблиц. SQL как часть системы MySQL можно охарактеризовать как язык структурированных запросов плюс наиболее распространенный стандартный язык, используемый для доступа к базам данных.

Программное обеспечение MySQL - это ПО с открытым кодом.

ПО с открытым кодом означает, что применять и модифицировать его может любой желающий. Такое ПО можно получать по Internet и использовать бесплатно. При этом каждый пользователь может изучить исходный код и изменить его в соответствии со своими потребностями.

Технические возможности СУБД MySQL

ПО MySQL является системой клиент-сервер, которая содержит многопоточный SQL-сервер, обеспечивающий поддержку различных вычислительных машин баз данных, а также несколько различных клиентских программ и библиотек, средства администрирования и широкий спектр программных интерфейсов (API).

Безопасность

Система безопасности основана на привилегиях и паролях с возможностью верификации с удаленного компьютера, за счет чего обеспечивается гибкость и безопасность. Пароли при передаче по сети при соединении с сервером шифруются. Клиенты могут соединяться с MySQL, используя сокеты TCP/IP, сокеты Unix или именованные каналы (named pipes, под NT)

Вместимость данных

Начиная с MySQL версии 3.23, где используется новый тип таблиц, максимальный размер таблицы доведен до 8 миллионов терабайт (263 bytes). Однако следует заметить, что операционные системы имеют свои собственные ограничения по размерам файлов. Ниже приведено несколько примеров:

- 32-разрядная Linux-Intel – размер таблицы 4 Гб.

- Solaris 2.7 Intel - 4 Гб

- Solaris 2.7 UltraSPARC - 512 Гб

- WindowsXP - 4 Гб

Как можно видеть, размер таблицы в базе данных MySQL обычно лимитируется операционной системой. По умолчанию MySQL-таблицы имеют максимальный размер около 4 Гб. Для любой таблицы можно проверить/определить ее максимальный размер с помощью команд SHOW TABLE STATUS или myisamchk -dv table\_name. Если большая таблица предназначена только для чтения, можно воспользоваться myisampack, чтобы слить несколько таблиц в одну и сжать ее. Обычно myisampack ужимает таблицу по крайней мере на 50%, поэтому в результате можно получить очень большие таблицы.

Тема 6.2. Основы языка SQL (1 час.)

Содержание темы:

SQL (обычно произносимый как "СИКВЭЛ" или "ЭСКЮЭЛЬ") символизирует собой Структурированный Язык Запросов. Это - язык, который дает Вам возможность создавать и работать в реляционных базах данных, являющихся наборами связанной информации, сохраняемой в таблицах.

Информационное пространство становится более унифицированным. Это привело к необходимости создания стандартного языка, который мог бы использоваться в большом количестве различных видов компьютерных сред. Стандартный язык позволит пользователям, знающим один набор команд, использовать их для создания, нахождения, изменения и передачи информации - независимо от того, работают ли они на персональном компьютере, сетевой рабочей станции, или на универсальной ЭВМ.

В нашем все более и более взаимосвязанном компьютерном мире, пользователь снабженый таким языком, имеет огромное преимущество в использовании и обобщении информации из ряда источников с помощью большого количества способов.

Элегантность и независимость от специфики компьютерных технологий, а также его поддержка лидерами промышленности в области технологии реляционных баз данных, сделало SQL (и, вероятно, в течение обозримого будущего оставит его) основным стандартным языком. По этой причине, любой, кто хочет работать с базами данных 90-х годов, должен знать SQL.

Стандарт SQL определяется ANSI (Американским Национальным Институтом Стандартов) и в данное время также принимается ISO (Международной Организацией по Стандартизации). Однако, большинство коммерческих программ баз данных расширяют SQL без уведомления ANSI, добавляя различные особенности в этот язык, которые, как они считают, будут весьма полезны. Иногда они несколько нарушают стандарт языка, хотя хорошие идеи имеют тенденцию развиваться и вскоре становиться стандартами "рынка" сами по себе в силу полезности своих качеств.

На данном уроке мы будем, в основном, следовать стандарту ANSI, но одновременно иногда будет показывать и некоторые наиболее общие отклонения от его стандарта.

Точное описание особенностей языка приводится в документации на СУБД, которую Вы используете. SQL системы InterBase 4.0 соответствует стандарту ANSI-92 и частично стандарту ANSI-III.

Состав языка SQL

Язык SQL предназначен для манипулирования данными в реляционных базах данных, определения структуры баз данных и для управления правами доступа к данным в многопользовательской среде.

Поэтому, в язык SQL в качестве составных частей входят:

язык манипулирования данными (Data Manipulation Language, DML)

язык определения данных (Data Definition Language, DDL)

язык управления данными (Data Control Language, DCL).

Подчеркнем, что это не отдельные языки, а различные команды одного языка. Такое деление проведено только лишь с точки зрения различного функционального назначения этих команд.

Модуль 7. Системы управления контентом (20час.)

Тема 7.1 Знакомство с CMS WordPress (2 час.)

Содержание темы:

Вордпресс — это система управления контентом, по английски сокращенно CMS, расшифровка Content Management System. Появился данный продукт в Америке в далеком 2003 году.

Система управления контентом  – это программа, которая устанавливается на хостинг, для упрощения администрирования и управления сайтом.  Хостинг – хранилище, на котором располагаются файлы сайта. Тот же компьютер, но подключенный к интернету и работающий 24 часа в сутки.

Основные задачи любой CMS

Упростить для чайников администрирование. CMS упрощает жизнь хозяину сайта: публикация контента, модерирование комментариев, вставка ссылок, загрузка видео и картинок, создание товаров и т.д.

Управление персоналом и подписчиками. Например в WordPress существует несколько типов участников от простого подписчика, до администратора сайта. Для каждого прописываются правила доступа к инструментам админки.

Отображать страницы сайта посетителям. Когда человек приходит на сайт, то страница начинает собираться из нескольких шаблонов: текста, картинок, комментариев, видео и других элементов. Такая технология называется “на лету”, элементы быстро соединяются в единый файл и отдаются на экран пользователю.

Почему эти задачи настолько важны, что для их решения создали такой сложный программный продукт? Давайте заглянем на несколько десятилетий назад. Раньше создание сайтов считалось сложной вещью. Лишь грамотные программисты могли сделать веб — ресурс, каждая страница писалась отдельно и загружалась на хостинг. Неудобств было много, например, нужно поменять в интернет магазине телефон, и тогда приходилось переписывать все страницы меняя на каждой номер.

Описание преимуществ и недостатков WordPress

В списке 6 пунктов с преимуществами WordPress:

Огромная база дополнений, расширяющих функционал сайта

Чем популярнее продукт, тем больше о нём написано в интернете. Описание ошибок, как добавить что-то или убрать. Поэтому решить проблему не составит труда.

Быстрая установка, все современный хостинги поддерживают быструю [установку WordPress](https://wpcourses.ru/install-wordpress-on-hosting/)

Интуитивно понятная структура файлов и административной панели

Полностью бесплатный и уже больше 10 лет поддерживаемый движок

Открытый исходный код

Недостаток только один это ограниченность сферы применения. Это блоговый движок, и достигнуть чего-то большего приходиться посредством установки дополнений, которые не всегда оказываются полезными и безопасными.

**Описание практико-ориентированных заданий и кейсов**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Номер темы/модуля** | **Наименование практического занятия** | **Описание** |
| **1.1** | 2.1. | Разработка дизайна одностраничника | Выполнение практического задания по разработке дизайна страницы сайта |
| **1.2** | 2.2. | Разработка композиции веб-страницы в различных редакторах | Выполнение практического задания по разработке композиции веб-страницы в различных редакторах |
| 1.3 | 2.2. | Создание макета сайта согласно требованиям ТЗ. | Выполнение практического задания составлению макета сайта |
| 2.1 | 3.1. | Практическая работа. Создание HTML-Документов | Выполнение практического задания по разработке HTML-документа |
| 2.2. | 3.1. | Практическая работа. Основы HTML5 Создание форм HTML 5 | Выполнение практического задания по разработке HTML5 форм |
| 2.3 | 3.1 | Практическая работа. Форматирование документа. Списки. Гиперссылки | Выполнение практического задания по созданию структуры документа |
| 3.1 | 4.1 | Практическая работа. Работа со шрифтами в CSS | Выполнение практического задания по разработке таблицы стилей CSS |
| 3.2 | 4.2 | Практическая работа. Работа с текстом в CSS | Выполнение практического задания по разработке таблицы стилей CSS |
| 3.3 | 4.2 | Практическая работа. Оформление списков в CSS | Выполнение практического задания по разработке таблицы стилей CSS |
| 3.4 | 4.2 | Практическая работа. Дополнительные элементы управления элементами в CSS | Выполнение практического задания по разработке таблицы стилей CSS |
| 3.5 | 4.2 | Практическая работа. Введение в анимацию | Выполнение практического задания по разработке таблицы стилей CSS |
| 4.1 | 5.1 | Практическая работа. Разбор примеров | Рассмотрение примеров задач на JavaScript |
| 4.2 | 5.2 | Практическая работа. Разбор примеров | Рассмотрение примеров задач на PHP |
| 5.1 | 6.1 | Практическая работа. Создание базы данных MySQL | Выполнение практического задания по разработке базы данных |
| 5.2 | 6.2 | Практическая работа. Создание запросов | Выполнение практического задания по созданию запросов |
| 6.1 | 7.1 | Практическая работа. Развертывание WordPress на сервере. Выбор доменного имени и хостинга. | Выполнение практического задания по работе с регистраторами доменных имен и хостинг-провайдерами |
| 6.2 | 7.1 | Практическая работа. Установка и настройка WordPress | Выполнение практического задания по установке и настройке WordPress |
| 6.3 | 7.1 | Практическая работа. Дополнительная настройка WordPress | Выполнение практического задания по настройке WordPress |
| 6.4 | 7.1 | Практическая работа. Подготовка и публикация материалов | Выполнение практического задания по работе в CMS |
| 6.5 | 7.1 | Практическая работа Управление контентом | Выполнение практического задания по работе в CMS |
| 6.6 | 7.1 | Практическая работа Дизайн сайта | Выполнение практического задания по работе в CMS |
| 6.7 | 7.1 | Практическая работа Создание структуры сайта | Выполнение практического задания по работе в CMS |
| 6.8 | 7.1 | Практическая работа Работа с плагинами WordPress | Выполнение практического задания по работе в CMS |

**8.Оценочные материалы по образовательной программе**

**8.1. Вопросы тестирования по модулям**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ модуля** | **Вопросы входного тестирования** | **Вопросы промежуточного тестирования** | **Вопросы итогового тестирования** |
| **1.1** | Ваш уровень знаний современных информационно-коммуникационных технологий в профессиональной деятельности  1) низкий  2) средний  4) высокий | Какие технологии появились за последние 3 года? | Какие технологии показали наибольший рост за последний год? |
| **1.2** | Ваш уровень знаний перспектив развития веб-технологий  1) низкий  2) средний  4) высокий | Какие веб-технологии анонсированы к выпуску? | Какой стек технологий лучше всего использовать для веб-разработки? |
| 2.1 | Ваш уровень знаний и умений в создании дизайна по предоставляемым инструкциям и спецификациям  1) низкий  2) средний  4) высокий | Какие программные средства используются для создания дизайна? | Какой порядок стадий создания дизайна? |
| 2.2 | Ваш уровень знаний и умений в выборе цвета, типографики и композиции. Современных стилей и тенденций дизайна  1) низкий  2) средний  4) высокий | Какие шрифты и цвета являются универсальными? | Как пользоваться колесами цветов? |
| 3 | Ваш уровень знаний структуры HTML- Документа, разметке текста, ссылки и изображения. Работа с тэгами и атрибутами тэгов для разметки текста  1) низкий  2) средний  4) высокий | Какие теги предназначены для создания блочных элементов? | Какие основные принципы построения HTML-документов? |
| 4.1 | Ваш уровень знаний и умений в работе со шрифтами в CSS  1) низкий  2) средний  4) высокий | Какие селекторы CSS используются для оформления элементов? | Какие основные свойства CSS применяются при оформлении сайтов? |
| 4.2 | Ваш уровень знаний и умений в работе с текстом в CSS  1) низкий  2) средний  4) высокий | Какие семейства свойств используются для работы с текстом? | Как добавить на сайт собственные шрифты, отсутствующие на компьютерах пользователя? |
| 5.1 | Ваш уровень знаний основ программирования на JavaScript  1) низкий  2) средний  4) высокий | Какое основное назначение языка JavaScript? | Какие события JavaScript используются при разработке сайтов чаще всего? |
| 5.2 | Ваш уровень знаний основ программирования на PHP  1) низкий  2) средний  4) высокий | Какое основное назначение языка PHP? | Какие способы работы с данными, отправляемыми на сервер есть в PHP? |
| 6.1 | Ваш уровень знаний основных понятий MySQL  1) низкий  2) средний  4) высокий | Как создаются таблицы в MySQL? | Как работать с базами данных MySQL с помощью PHP? |
| 6.2. | Ваш уровень знаний основ языка SQL  1) низкий  2) средний  4) высокий | Какой состав языка SQL? | Какой синтаксис запросов SELECT, DELETE, UPDATE, INSERT? |
| 7.1 | Ваш уровень знанийсистем управления контентом  1) низкий  2) средний  4) высокий | Что такое CMS? | Какие этапы разработки сайта при помощи CMS? |

**8.2.**  **описание показателей и критериев оценивания, шкалы оценивания**

Сформированность компетенции оценивается на следующих уровнях:

0 — не владеет (компетенция не сформирована)

1 — начальный уровень дает общее представление о виде деятельности, основных закономерностях функционирования объектов профессиональной деятельности, методов и алгоритмов решения практических задач;

2 — базовый уровень позволяет решать типовые задачи, принимать профессиональные и управленческие решения по известным алгоритмам, правилам и методикам;

3 — продвинутый уровень предполагает готовность решать практические задачи повышенной сложности, нетиповые задачи, принимать профессиональные и управленческие решения в условиях неполной определенности, при недостаточном документальном, нормативном и методическом обеспечении **.**

**8.3.**  **примеры контрольных заданий по модулям или всей образовательной программе** **.**

Задание для демонстрационного экзамена по комплекту оценочной документации № 1.4 по компетенции №17 «Вебдизайн и разработка» (образец)

Задание включает в себя следующие разделы: 1. Формы участия 2. Модули задания, критерии оценки и необходимое время 3. Необходимые приложения

Продолжительность выполнения задания: 8 ч.

1. ФОРМА УЧАСТИЯ

Индивидуальная

Модули с описанием работ

Модуль 1:

Введение

В последнее время проблем в городах становится все больше и больше. Не каждую проблему удается решить быстро, потому что компании ответственные за решение проблем не всегда знают о них. Вам необходимо создать городской портал «Сделаем лучше вместе!» по приему заявок на устранение проблем в городе: ямочный ремонт дорог, ремонт детских площадок, зданий сооружений и т.д. Вам предстоит продемонстрировать свои знания в разработке веб-портала с использованием таких технологий как дизайн, верстка, программирование на стороне клиента и сервера. Также необходимо позаботиться о защите проекта: от взлома, от несанкционированного доступа к административным функциям и т.д. Вы можете использовать предоставленные фреймворки и библиотеки: Yii2, Laravel 6.5, Django 2.2.4, jQuery, jQuery UI, Vue.js.

Вам необходимо продумать дизайн и сверстать его, а также реализовать функциональные возможности сервиса в соответствии с текущим заданием. Стоит принять во внимание тот факт, что основная часть пользователей использует сайт через мобильный телефон. Обязательно позаботьтесь об удобстве использования сервиса и обратной связи между порталом и пользователями. Целью создания сайта является привлечение молодежи от 14 до 25 лет к проблемам благоустройства города. Также вам необходимо разработать логотип, отвечающий требованиям проекта. Дополнительным преимуществом будет реализация интерактивных взаимодействий с пользователем, анимация и микроанимация.

ВНИМАНИЕ! Проверяться будут только работы, загруженные на сервер! описание проекта и задач Ваша задача – разработать дизайн сайта, которым удобно пользоваться на смартфонах с разрешением 720х1440px.

Несмотря на то, что основная целевая группа будет пользоваться вашим веб-ресурсом со смартфонов, вы должны предусмотреть возможность удобной работы на сайте с компьютеров. Поэтому ваша верстка должна быть адаптивна и корректно открываться со следующих устройствах:

● смартфоны с разрешением 720x1440px;

● компьютеры с шириной экрана от 1200px. Структура страниц для смартфонов и компьютеров должна отличаться.

Следует обратить внимание, что дизайн сайта должен быть удобен и понятен для использования, соответствовать современным тенденциям, а также учитывать особенности платформ, для которых создается сайт. Все созданные страницы должны иметь согласованный интерфейс.

При разработке верстки необходимо использовать один из предоставленных шрифтов.

На сайте должны быть реализованы следующие страницы:

● Главная страница, предоставляющая возможности авторизации и регистрации пользователей.

● Личные кабинеты администратора и авторизованных пользователей

Дополнительная информация по каждой из страниц представлена ниже. Вы можете создавать дополнительные страницы по своему усмотрению.

Учтите, что дизайн и верстка будут оцениваться в первую очередь по страницам:

● Главная страница

● Личный кабинет авторизованного пользователя Доступ к этим страницам должен быть очевидным.

Логотип должен быть реализован в соответствии с требованиями:

 В логотипе должны быть использованы основные цвета сайта;

 Логотип представляет собой изображение;

 Разработанный логотип должен быть сохранен на сервере в корневом каталоге в директории logo со следующим именем - logo.png

Портал должен поддерживать возможности 3 типов пользователей:

 Гость

 Авторизованный пользователь

 Администратор

Возможности гостя

 Вход в личный кабинет по логину и паролю

 Регистрация

 Просмотр главной страницы

Возможности авторизованного пользователя

 Регистрация, авторизация, выход;

 Создание заявки на решение проблемы;

 Просмотр своих заявок;

 Удаление своей заявки.

Разделы администратора

 Смена статуса заявки на «Решена» или «Отклонена».

 Управление категориями заявок (например, «ремонт дорог», «уборка мусора» и др.)

Авторизация пользователя и администратора

При вводе неправильной пары логин-пароль пользователю отображается сообщение об ошибке. При успешной авторизации пользователь должен перенаправляться в личный кабинет с возможностью просмотра своих заявок. Личный кабинет доступен только авторизованному пользователю. Панель управления сайтом доступна только администратору.

Регистрация пользователя

Вся валидация должна работать без перезагрузки страницы, все поля обязательные для заполнения:

 ФИО - только кириллические буквы, дефис и пробелы – проверка на стороне клиента;

 Логин – только латиница, уникальный - проверка на стороне сервера, без перезагрузки страницы;

 Email - валидный формат email-адрес - проверка на стороне клиента;

 Пароль;

 Повтор пароля – введенное значение должно совпадать с паролем;

 Согласие на обработку персональных данных - должно быть отмечено.

В случае несоответствия любым требованиям выводится анимированное сообщение об ошибке, поля с ошибками выделяются, данные на сервер не отправляются. Создайте учетную запись администратора с логином admin и паролем adminWSR. Все авторизованные пользователи могут выйти из профиля по нажатию на ссылку/кнопку «Выход».

Создание заявки (все поля обязательны):

 Название;

 Описание;

 Категория (выбор из существующих категорий);

 Фото, демонстрирующее проблему в одном из форматов (jpg, jpeg, png, bmp) максимальный размер 10Мб; при невыполнении хотя бы одного из требований, заявка не сохраняется на сервере, выводится сообщения об ошибке. Временная метка добавления заявки создается автоматически при добавлении заявки в базу данных. При добавлении заявки она должна автоматически получить статус «Новая».

Удаление заявки При удалении заявки пользователю должно быть выведено сообщение с просьбой подтвердить желаемое действие. Пользователь может удалить только свою заявку, статус которой не был изменен администратором на «решено» или «отклонена».

Просмотр своих заявок На странице просмотра своих заявок необходимо отобразить список своих заявок со следующими полями: • Временная метка • Название заявки • Описание заявки • Категория заявки • Статус заявки (Новая, Решена, Отклонена). По умолчанию отображаются все заявки в порядке добавления заявок (недавно добавленные отображаются в начале таблицы). Вам необходимо добавить возможности фильтрации заявок по статусу, например, отображение только заявок со статусом «Новая».

Главная страница

На главной странице выводится не более 4 последних решенных проблем (фото решенной проблемы) со следующими полями: • Временная метка • Название • Категория заявки • Фотография

Изначально видна фотография «после». При наведении указателя мыши на фото «после», вместо него анимировано должна отображаться фотография «до». Стиль анимации: «масштабирование». После вывода указателя мыши с изображения появляется начальное изображение «после» с тем же стилем анимации. Пример реализации анимированной смены изображений доступен в видеофайле «hover/масштабирование.mp4».

Добавьте на главную страницу счетчик:  Количество решенных заявок. Информация о количестве решенных задачах обновляется автоматически, без перезагрузки страницы не реже чем 1 раз в 5 секунд. В случае изменения значения счетчика должно звучать оповещение (Notif.mp3). Значение счетчика должно обновляться с анимацией.

Смена статуса заявки Заявке со статусом «Новая» можно сменить статус на «Решена» с обязательным прикреплением (добавлением) фотографии – доказательства решения проблемы (фотография «ПОСЛЕ»). Заявке со статусом «Новая» можно сменить статус на «Отклонена» с обязательным указанием причины отказа. Смена статуса с «Решена» или «Отклонена» невозможна.

Управление категориями заявок

Администратор может добавить или удалить категорию заявок. При удалении категории должны быть удалены все заявки данной категории. Заявки этой категории не отображаются.

ИНСТРУКЦИЯ ДЛЯ УЧАСТНИКОВ

Портал должен быть доступен по адресу: http://xxxxxx-m1.wsr.ru/, где xxxxxx – логин пользователя, полученный участником на индивидуальное карточке Административная панель должна быть доступна по адресу: <http://xxxxxx-m1.wsr.ru/admin>

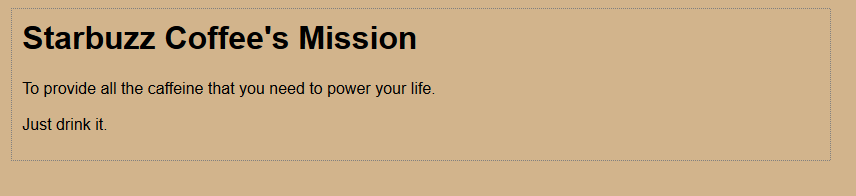
Внимание! Доступ ко всем страницам должен быть реализован с помощью элементов пользовательского интерфейса. Медиа файлы располагаются на сервере в папке public. Вы можете изменять предоставленные файлы и создавать новые для улучшения приложения. Проверяются только работы, загруженные на сервер. Ваш HTML/CSS должен быть валидным, а в консоли браузера и на сайте отсутствовать ошибки/предупреждения/оповещения, создаваемые средствами JavaScript и PHP. Оценка будет производиться в браузере Google Chrome.

**8.4.**  **тесты и обучающие задачи (кейсы), иные практикоориентированные формы заданий**

Разработать две страницы сайта о кофейне, используя указания в файле «Лабораторная работа №96.pdf»

Результат работы:





1. Изучить структуру исходной html-страницы и имеющийся CSS код.

Скопируйте файлы в новую папку на диске D. Откройте в текстовом редакторе Notepad++ файлы index.html и style.css. Внимательно ознакомьтесь с их содержимым.

2. Дописать недостающий CSS код, чтобы страница выглядела аналогично предоставленному образцу.

В файле style.css допишите недостающие фрагменты кода для придания исходной страницы внешнего вида, максимально приближенного к предлагаемому образцу. МЕНЯТЬ HTML-КОД ЗАПРЕЩЕНО!!!

3. Защитить лабораторную работу.

Оформить отчет по лабораторной работе согласно стандарту. В отчет поместить код измененного файла style.css и скриншот получившегося результата. При защите лабораторной работы необходимо пояснить, для чего предназначаются те или иные стили и какие свойства они изменяют.

Файлы для лабораторной работы:

1. index.html – страница, которую необходимо обработать.

2. style.css – каскадная таблица стилей для страницы index.html.

3. Каталок images – содержит необходимый графический материал.

4. Образец.png – конечный вид страницы.

1. Определитесь с тематикой web-страницы.

Необходимо выбрать тему, для которой будет создаваться web-страница.

К примеру: сайт кафе, спортивной школы, книжного интернет-магазина.

2. Установить CMS Wordpress (файл в папке с л/р).

3. Найти тему внешнего вида, соответствующую тематике сайта.

4. Создать меню пользователя, отражающее выбранную предметную область.

5. Создать страницы сайта, в соответствии с разработанным меню пользователя.

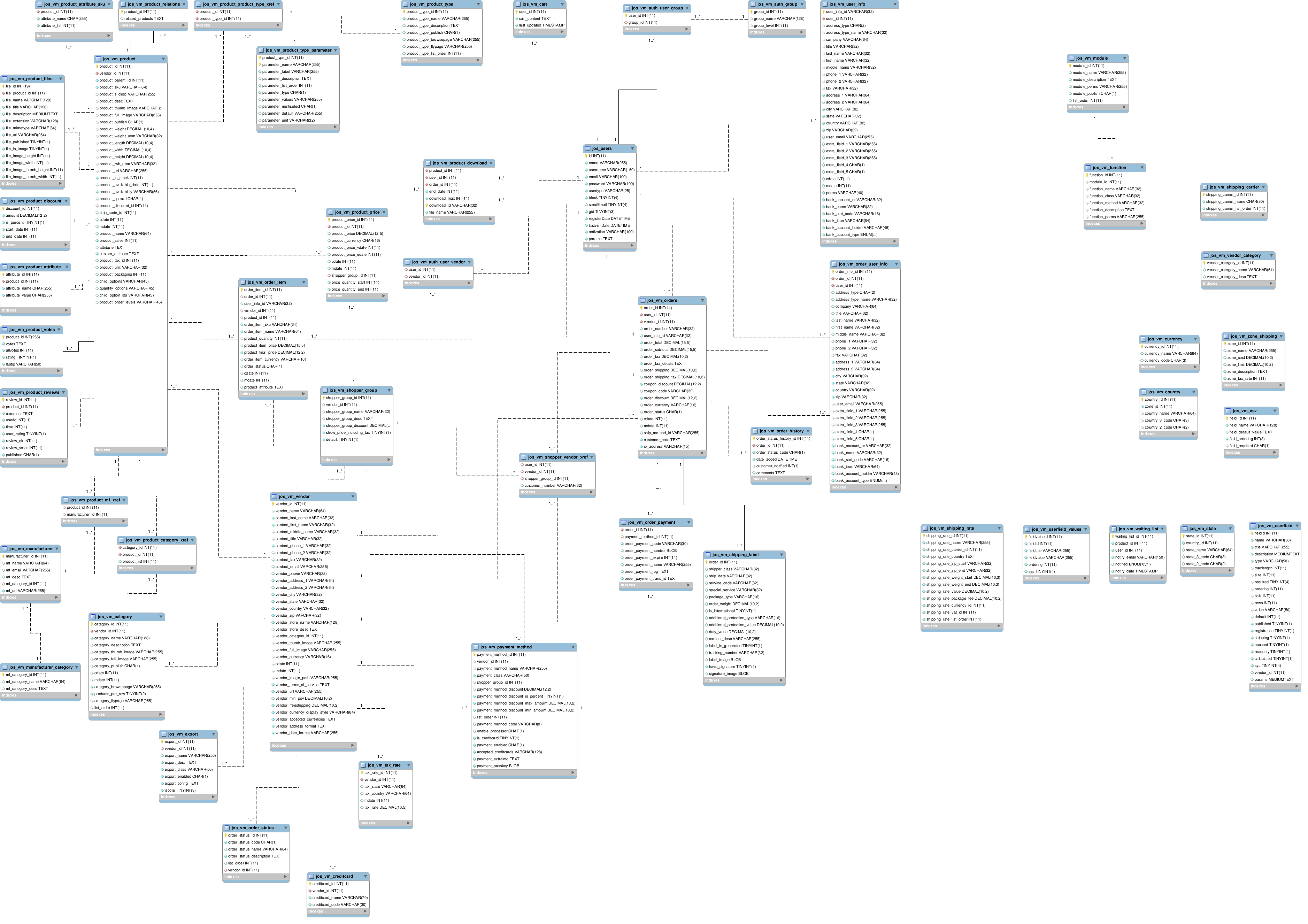
6. Добавить плагины:

- Электронного магазина или каталога товаров;

- для вывода галереи изображений;

- для отображения интерактивной карты

Разработать базу данных, согласно приведенной в приложении схемы.



**.**

**8.5.**  **описание процедуры оценивания результатов обучения**

В основе оценивания лежит экспертная оценка.

Для задания разрабатывается схема оценки, где каждому действию соответствует свой критерий. Итоговый балл считается как сумма выполненных критериев.

**.**

**9.Организационно-педагогические условия реализации программы**

**9.1. Кадровое обеспечение программы**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№**  **п/п** | **Фамилия, имя, отчество (при наличии)** | **Место основной работы и должность, ученая степень и ученое звание (при наличии)** | **Ссылки на веб-страницы с портфолио (при наличии)** | **Фото в формате jpeg** | **Отметка о полученном согласии на обработку персональных данных** |
| **1** | Саладай Алексей Владимирович | Преподаватель специальных дисциплин, ГАПОУ «ОНТ им. В.А. Сорокина» | - | Pnh1glBWHko | https://sun9-73.userapi.com/Qy4Txdi4NgulydIJDNnkM4Pnh4oAPBRNVRA2ZA/RwzRVUYu5Y8.jpg |

**9.2.Учебно-методическое обеспечение и информационное сопровождение**

|  |  |
| --- | --- |
| **Учебно-методические материалы** | |
| Методы, формы и технологии | Методические разработки,  материалы курса, учебная литература |
| Методы обучения: словесные, наглядные, практические, репродуктивные и проблемные, индуктивные и дедуктивные, самостоятельной работы и работы под руководством преподавателя  формы обучения: лекции, практические занятия, контрольные работы, консультации, самостоятельная работа экзамен.  Технологии:  Индивидуальное и дифференцированное обучение.  Модульное обучение. | 1. Техническое описание компетенции;  2. Комплект оценочной документации по компетенции;  3. Раздаточные материалы для слушателей;  4. Учебные пособия, изданных по отдельным разделам программы:  1) «Фонд оценочных средств по разделам: Графический дизайн веб-страниц, Верстка веб-страниц;  2) Методические указания по проведению практических работ «Проектирование и разработка веб-приложений»  5. Профильная литература;  1) Тузовский, А. Ф. Проектирование и разработка web-приложений: учебное пособие для академического бакалавриата / А. Ф. Тузовский. – М.: Издательство Юрайт, 2017. –218 с.  2) Никсон Робин. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Никсон Робин. – Питер, 2017. – 768 с.  6. отраслевые и другие нормативные документы; |
|  |  |

|  |  |
| --- | --- |
| **Информационное сопровождение** | |
| Электронные  образовательные ресурсы | Электронные  информационные ресурсы |
| hortal.ontvkr.ru | https://stepik.org/ |
|  | https://www.coursera.org/ |

**9.3.Материально-технические условия реализации программы**

|  |  |
| --- | --- |
| Вид занятий | Наименование оборудования,  программного обеспечения |
| Лекции | Камера, микрофон.  Компьютер  Монитор  Компьютерная мышь  Клавиатура  Notepad ++ или аналог  Sublime Text 3 или аналог  Web Browser - Firefox Developer Edition или аналог  Web Browser - Chrome или аналог  GIMP или аналог  XAMPP или аналог  Windows 10 или аналог |
| Лабораторные и практические занятия, тестирование, демонстрационный экзамен | Камера, микрофон.  Компьютер  Монитор  Компьютерная мышь  Клавиатура  Notepad ++ или аналог  Sublime Text 3 или аналог  Web Browser - Firefox Developer Edition или аналог  Web Browser - Chrome или аналог  GIMP или аналог  XAMPP или аналог  Windows 10 или аналог |
|  |  |

**III.Паспорт компетенций (Приложение 2)**

ПАСПОРТ КОМПЕТЕНЦИИ

(Технологии веб-дизайна и разработки)

(Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | Использование современных систем управления контентом | |
| 2. | Указание типа компетенции | общекультурная/  универсальная | профессиональная | |
| общепрофессиональ ная |
| профессиональная |
| профессионально-специализированная |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | Под компетенцией понимается наличие знаний в области современных технологий управления контентом, основ управления сайтом.  Слушатель должен  знать:  современные современных систем управления контентом  уметь:  - работать в системе CMS WordPress  Владеть навыками:  создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  поиска, выбора и подключения нужных плагинов/модулей; | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформированности компетенции  обучающегося | Индикаторы |
|  | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материаллов  Умеет:  Применять нормативные документы, определяющие требования к оформлению страниц ИР;  Применять специализированное программное обеспечение для верстки страниц ИР; использовать язык разметки страниц ИР;  Применять выбранные языки программирования для написания программного кода;  Владеет навыками:  выбора цвета, типографии, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  поиска, выбора и подключения нужных плагинов /модулей; |
|  | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределённости, сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Владеет:  навыками создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции; |
|  | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  разработки серверной части сайта; разработка сайта с использованием CMS;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Применять открытые библиотеки; создавать библиотеки и модули для выполнения повторяющихся задач;  Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  Применять нормативные документы, определяющие требования к оформлению страниц ИР;  Определять возможности отображения веб-страниц в размерах рабочего пространства устройств для разных видов дизайн-макетов;  Применять специализированное программное обеспечение для верстки страниц ИР; использовать язык разметки страниц ИР;  Применять выбранные языки программирования для написания программного кода;  Использовать выбранную среду программирования и средства системы управления базами данных;  Владеет:  навыками создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  поиска, выбора и подключения нужных плагинов/модулей; |
|  | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  разработки клиентской части сайта;  разработки серверной части сайта; разработка сайта с использованием CMS;  порядок организации и проведения групповых и массовых мероприятий;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Создавать веб-сайты полностью соответствующие текущим стандартам W3C (http://www.w3.org);  Создавать и модифицировать сайты с учетом Search Engine Optimization.  Применять нормативные документы, определяющие требования к оформлению страниц ИР;  Применять специализированное программное обеспечение для верстки страниц ИР; использовать язык разметки страниц ИР;  Применять выбранные языки программирования для написания программного кода;  Использовать выбранную среду программирования и средства системы управления базами данных;  Использовать возможности имеющейся программной архитектуры ИР; использовать существующие типовые решения и шаблоны ИР;  размещать текстовую и графическую информацию на страницах сайта;  Редактировать гипертекстовую разметку сайта с использованием систем администрирования;  Применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов;  Осуществлять коммуникации с заинтересованными сторонами;  Разрабатывать программный код клиентской и серверной части веб-приложений;  Осуществлять оптимизацию веб-приложения с целью повышения его рейтинга в сети Интернет.  Владеет навыками: создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  поиска, выбора и подключения нужных плагинов/модулей; |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Выполнять разработку спецификаций отдельных компонент. | |
| 6. | Средства и технологии оценки | | Выполнение практических работ | |

ПАСПОРТ КОМПЕТЕНЦИИ

(Технологии веб-дизайна и разработки)

(Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | Разработка графического дизайна веб-страниц | |
| 2. | Указание типа компетенции | общекультурная/  универсальная | профессиональная | |
| общепрофессиональ ная |
| профессиональная |
| профессионально-специализированная |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | Под компетенцией понимается наличие способностей в подборе цветов, шрифтов и графики, для поддержки эффективной рабочей коммуникации с профессиональными дизайнерами, а также при разработке структуры сайта  Слушатель должен  знать:  - структуру и общепринятые элементы веб-страниц различных видов и назначений;  - вопросы, связанные с когнитивными, социальными, культурными, технологическими и экономическими условиями при разработке дизайна;  - как создавать и оптимизировать графику для сети Интернет;  - как создавать дизайн по предоставляемым инструкциям  и спецификациям;  - как производить выбора цвета, работать с типографикой и композицией;  - принципы и методы адаптации графики для использования ее на веб-сайтах;  - правила поддержания фирменного стиля, бренда и стилевых инструкций;  - ограничения, которые накладывают мобильные устройства и разрешения экранов при использовании их для просмотра веб-сайтов;  - принципы построения эстетичного и креативного дизайна;  - современные стили и тенденции дизайна.  уметь:  - создавать и анализировать разработанные визуальные ответы на поставленные вопросы, в том числе об иерархии, типографики, эстетики и композиции;  - создавать, использовать и оптимизировать изображения для веб-сайтов;  - анализировать целевой рынок и продукцию, которую продвигает, используя дизайн;  - выбирать дизайнерское решение, которое будет наиболее подходящим для целевого рынка;  - принимать во внимание влияние каждого элемента, который добавляется в проект во время разработки дизайна;  - использовать все требуемые элементы при разработке дизайна;  - учитывать существующие правила корпоративного стиля;  - создавать «отзывчивый» дизайн, который будет отображаться корректно на различных устройствах и при разных разрешениях;  - придерживаться оригинальной концепции дизайна проекта и улучшать его визуальную привлекательность;  - превращать идею в эстетичный и креативный дизайн  Владеть навыками :  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформирован ности компетенции  обучающегося | Индикаторы |
|  | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Порядок работы с оргтехникой и правила технической безопасности;  Правила оформления информационно-презентационных материалов;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Владеет:  Навыками выбора цвета, типографики, композиции; |
|  | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределён-ности, сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Порядок работы с оргтехникой и правила технической безопасности;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Владеет навыками:  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS; |
|  | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Порядок работы с оргтехникой и правила технической безопасности;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Владеет навыками:  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  навыками создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем; |
|  | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Порядок работы с оргтехникой и правила технической безопасности;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Владеет навыками:  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем; |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Выполнять отладку программного продукта с использованием специализированных программных средств | |
| 6. | Средства и технологии оценки | | Выполнение практических работ | |

ПАСПОРТ КОМПЕТЕНЦИИ

(Технологии веб-дизайна и разработки)

(Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | Осуществлять вёрстку веб-страниц | |
| 2. | Указание типа компетенции | общекультурная/  универсальная | профессиональная | |
| общепрофессиональ ная |
| профессиональная |
| профессионально-специализированная |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | Под компетенцией понимается наличие знаний совместимости конечного продукта с современными версиями наиболее распространенных веб-браузеров, программами и устройстви.  Слушатель должен  знать:  - методы обеспечения доступа к страницам веб-сайтов аудитории с ограниченными возможностями;  - World Wide Web Consortium (W3C) стандарты HTML и CSS;  - методы верстки веб-сайтов и их стандартную структуру;  - Web accessibility initiative (WAI) стандарт доступности активных Интернет-приложений для людей с ограниченными возможностями;  - как применять соответствующие CSS правила и селекторы для получения ожидаемого результата;  - лучшие практики для Search Engine Optimization (SEO) и интернет-маркетинга;  - как встраивать и интегрировать анимацию, аудио, видео и другую мультимедийную информацию, управлять поведением остальных элементов на странице;  - как использовать предпроцессоры..  уметь:  - создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  - корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  - создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  - создавать веб-сайты полностью соответствующие текущим стандартам W3C (http://www.w3.org);  - создавать и модифицировать сайты с учетом Search Engine Optimization  владеть навыками:  создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS; | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформирован ности компетенции  обучающегося | Индикаторы |
|  | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Владеет навыками: выбора цвета, типографики, композиции; |
|  | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределён-ности, сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Владеет навыками: выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS; |
|  | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  разработки клиентской части сайта;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Владеет навыками: создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  выбора цвета, типографики, композицию;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS; |
|  | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает:  Cпецификации стандарта компетенции «Веб-дизайн разработка»;  Требования охраны труда и техники безопасности;  Порядок работы с оргтехникой и правила технической безопасности;  Наиболее востребованные информационно-коммуникационные технологии;  Правила оформления информационно-презентационных материалов;  разработки дизайна сайта; верстки сайта по макету;  разработки клиентской части сайта;  Умеет:  Создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;  Корректно использовать CSS для обеспечения единого дизайна в разных браузерах;  Создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;  Создавать веб-сайты полностью соответствующие текущим стандартам W3C (http://www.w3.org);  Владеет навыками:  создания и оптимизации графики для сети Internet, в том числе по предложенным инструкциям и спецификациям;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS; |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Разработка программных модулей программного обеспечения для компьютерных систем | |
| 6. | Средства и технологии оценки | | Выполнение практических заданий | |

ПАСПОРТ КОМПЕТЕНЦИИ

(Технологии веб-дизайна и разработки)

(Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | Программировать на стороне клиента и на стороне сервера | |
| 2. | Указание типа компетенции | общекультурная/  универсальная | профессиональная | |
| общепрофессиональ ная |
| профессиональная |
| профессионально-специализированная |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | Под компетенцией понимается наличие знаний в области технологий и методов программирования на стороне сервера и клиента, наличие умений разбираться в основах программной архитектуры и базах данных для хранения информации и организации сложных веб-сервисов.  Слушатель должен  знать:  - клиентский языка программирования JavaScript;  - принципы, особенности и способы использования открытых фреймворков;  - принципы разработка кода с использованием открытых библиотек;  - синтаксис и симантику языка, построение грамотного и структурированного кода;  - как взаимодействовать с объектной моделью документа (DOM)  - как разрабатывать PHP, Python, Node.js код на процедурном и объектно-ориентированном уровнях;  - как использовать открытые библиотеки и Фреймворки;  - распространенные модели организации и хранении данных и реализацию их с применением SQL подобных баз данных;  - FTP (File Transfer Protocol), особенности использования его на стороне сервера и клиента, а также необходимое для  этого программное обеспечение;  - SSH, производить удалённое управление операционной системой и настройку необходимых служб  - как разрабатывать веб-сервисы с применением PHP, Python, Node.js, XML (Extensible Markup Language) и JSON;  - различные методы программирования;  - как разрабатывать программный код в соответствии с паттернами (например, MVC (Model View Controller);  - как разрабатывать безопасное веб-приложение  уметь:  - разрабатывать анимацию для повышения его доступности и визуальной привлекательности;  - создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  - применять открытые библиотеки.  - разрабатывать полноценные веб приложения для возможности использования их в различных областях деятельности  - создавать библиотеки и модули для выполнения повторяющихся задач;  - разрабатывать веб-приложения с доступом к базе данных SQL подобных баз данных и веб-сервисы по требованиям клиента;  - интерпретировать ER (Entity-Relationship) диаграммы в функционирующую базу данных;  - создавать SQL (Structured Query Language) запросы, используя корректный синтаксис (классический и PDO (PHP Data Object));  - обеспечивать безопасность (устойчивость веб- приложения к атакам и взлому);  - интегрировать существующий и создавать новый программный код с API (Application Programming Interfaces), библиотеками и фреймворками;  - разрабатывать объектно-ориентированный программный код  владеть навыками:  обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применение World Wide Web Consortium (W3C) стандартов HTML и CSS;  создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем;  разработки кода с использованием открытых библиотек и фреймворков, объектно-ориентированной парадигмы JavaScrip, Yii2, Laravel;  разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON;  разработки программного кода в соответствии с паттернами (например, MVC (Model View Controller);  поиска, выбора и подключения нужных плагинов/модулей; | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформирован ности компетенции  обучающегося | Индикаторы |
|  | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Знает:  Понятия системы программирования;  Умеет:  Создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  Владеет навыками: обеспечения безопасности и защищенности созданных веб-приложений; |
|  | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределён-ности, сложности.) | Знает:  Понятия системы программирования;  Подпрограммы, составление библиотек программ;  Умеет:  Создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  Применять открытые библиотеки;  Владеет навыками: обеспечения безопасности и защищенности созданных веб-приложений;  разработки кода с использованием открытых библиотек и фреймворков, объектно-ориентированной парадигмы JavaScrip, Yii2, Laravel; |
|  | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Знает:  Правила разработки клиентской части сайта;  Понятия системы программирования;  Подпрограммы, составление библиотек программ;  Умеет:  Создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  Применять открытые библиотеки;  создавать библиотеки и модули для выполнения повторяющихся задач;  Владеет навыками: обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем;  разработки кода с использованием открытых библиотек и фреймворков, объектно-ориентированной парадигмы JavaScrip, Yii2, Laravel;  разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; |
|  | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает:  Правила разработки клиентской части сайта;  Правила разработки серверной части сайта;  Понятия системы программирования;  Подпрограммы, составление библиотек программ;  Умеет:  Создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  Применять открытые библиотеки;  создавать библиотеки и модули для выполнения повторяющихся задач;  Применять выбранные языки программирования для написания программного кода;  Использовать выбранную среду программирования и средства системы управления базами данных;  Использовать возможности имеющейся программной архитектуры ИР; мспользовать существующие типовые решения и шаблоны ИР;  Владеет навыками: обеспечения безопасности и защищенности созданных веб-приложений;  выбора цвета, типографики, композиции;  применения World Wide Web Consortium (W3C) стандартов HTML и CSS;  создания дизайн-концепции и техники чернового макетирования (wireframing), объектно-событийного моделирования (storyboarding), создания блок-схем;  разработки кода с использованием открытых библиотек и фреймворков, объектно-ориентированной парадигмы JavaScrip, Yii2, Laravel;  разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON;  разработки программного кода в соответствии с паттернами (например, MVC (Model View Controller);  поиска, выбора и подключения нужных плагинов/модулей; |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Разработка программных модулей программного обеспечения для компьютерных систем | |
| 6. | Средства и технологии оценки | | Выполнение практических работ | |

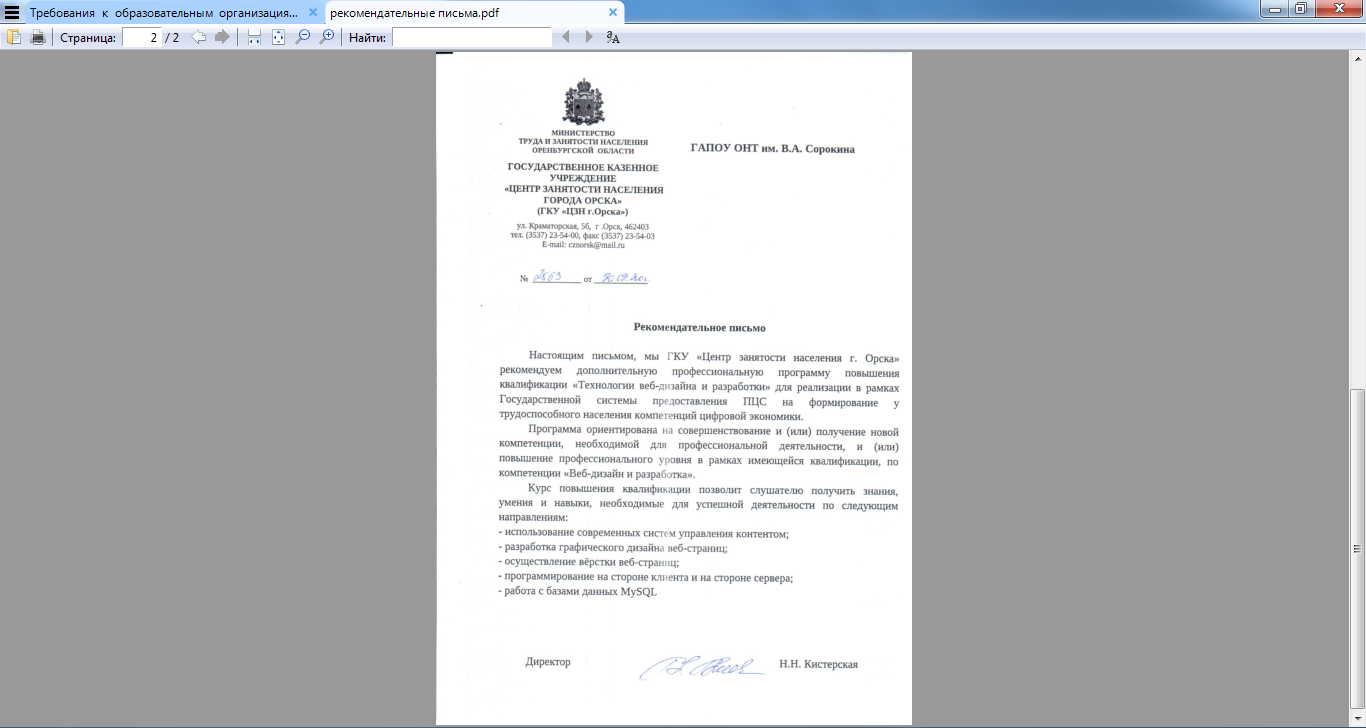
ПАСПОРТ КОМПЕТЕНЦИИ

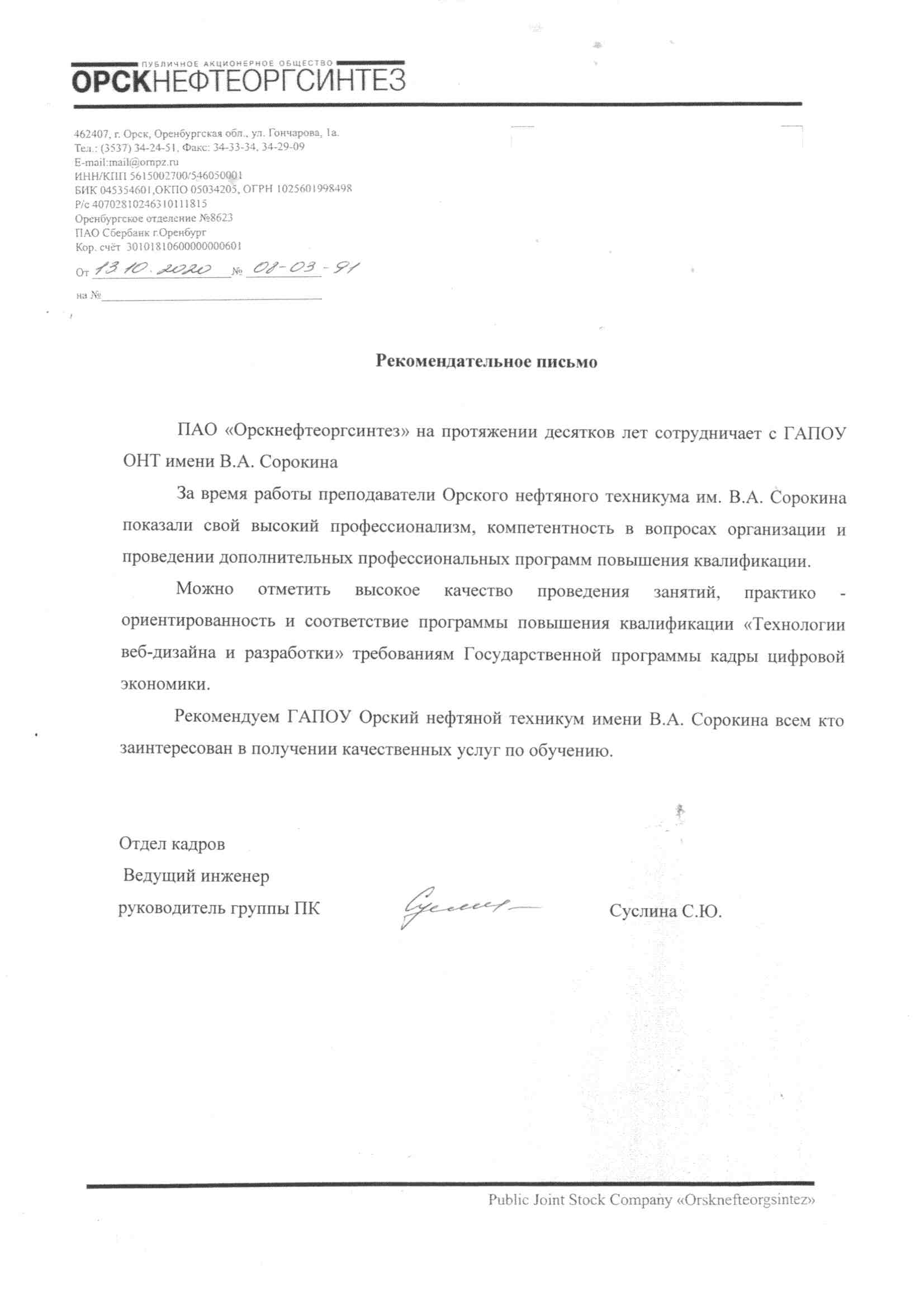
(Технологии веб-дизайна и разработки)

(Государственное автономное профессиональное образовательное учреждение «Орский нефтяной техникум им. Героя Советского Союза В.А. Сорокина»)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Наименование компетенции | | Работать с базами данных MySQL | |
| 2. | Указание типа компетенции | общекультурная/  универсальная | профессиональная | |
| общепрофессиональ ная |
| профессиональная |
| профессионально-специализированная |
| 3. | Определение, содержание и основные сущностные характеристики компетенции | | Под компетенцией понимается наличие умений работы с базами данных MySQL  Слушатель должен  знать:  - основные понятия MySQL  - основы языка SQL  уметь:  - разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  - использовать выбранную среду программирования и средства системы управления базами данных;  владеть:  навыками разработка веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; | |
| 4. | Дескриптор знаний, умений и навыков по уровням | | Уровни сформирован ности компетенции  обучающегося | Индикаторы |
|  | | Начальный уровень  (Компетенция недостаточно развита. Частично проявляет навыки, входящие в состав компетенции. Пытается, стремится проявлять нужные навыки, понимает их необходимость, но у него не всегда получается.) | Знает:  Объектно-ориентированную модель программирования, понятия классов и объектов, их свойства и методы.  Умеет:  Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  Владеет навыками: разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; |
|  | | Базовый уровень  (Уверенно владеет навыками, способен, проявлять соответствующие навыки в ситуациях с элементами неопределён-ности, сложности.) | Знает:  Понятия системы программирования;  Объектно-ориентированную модель программирования, понятия классов и объектов, их свойства и методы.  Умеет:  Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  Создавать SQL (Structured Query Language) запросы, используя корректный синтаксис (классический и PDO (PHP Data Object));  Владеет навыками:  разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; |
|  | | Продвинутый  (Владеет сложными навыками, способен активно влиять на происходящее, проявлять соответствующие навыки в ситуациях повышенной сложности.) | Знает:  Программу разработки серверной части сайта;  Понятия системы программирования;  Объектно-ориентированную модель программирования, понятия классов и объектов, их свойства и методы.  Умеет:  Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  Создавать SQL (Structured Query Language) запросы, используя корректный синтаксис (классический и PDO (PHP Data Object));  Владеет навыками:  разработка веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; |
|  | | Профессиональный  (Владеет сложными навыками, создает новые решения для сложных проблем со многими взаимодействую-щими факторами, предлагает новые идеи и процессы, способен активно влиять на происходящее, проявлять соответствующие навыки  в ситуациях повышенной сложности.) | Знает:  Программу разработки серверной части сайта;  Понятия системы программирования;  Объектно-ориентированную модель программирования, понятие классов и объектов, их свойства и методы.  Умеет:  Разрабатывать веб-приложения с доступом к базе данных MySQL и веб-сервисы по требованиям клиента;  Создавать SQL (Structured Query Language) запросы, используя корректный синтаксис (классический и PDO (PHP Data Object));  Владеет навыками:  разработки веб-сервисов с применением PHP, XML (Extensible Markup Language) и JSON; |
| 5. | Характеристика взаимосвязи данной компетенции с другими компетенциями/ необходимость владения другими компетенциями для формирования данной компетенции | | Разработка и администрирование баз данных | |
| 6. | Средства и технологии оценки | | Выполнение практических работ | |

**Рекомендаций к программе от работодателей**:





**VI.Указание на возможные сценарии профессиональной траектории граждан** по итогам освоения образовательной программы (в соответствии с приложением)

Сценарий профессиональной траектории граждан

|  |  |
| --- | --- |
| Цели получения персонального цифрового сертификата | |
| Развитие компетенций в текущей сфере занятости | |
| Работающий по найму в организации | Сохранение текущего рабочего места |
| Работающий по найму в организации | Развитие профессиональных качеств |

Директор ГАПОУ

ОНТ имени В.А. Сорокина Т.Б. Кочеткова

**VIII.Приложенные Скан-копии**

Утвержденной рабочей программа (подпись, печать, в формате pdf)