



ВСЕРОССИЙСКОЕ
ЧЕМПИОНАТНОЕ
ДВИЖЕНИЕ
ПО ПРОФЕССИОНАЛЬНОМУ
МАСТЕРСТВУ

КОНКУРСНОЕ ЗАДАНИЕ КОМПЕТЕНЦИИ

«Веб-технологии»

Региональный этап Чемпионата по профессиональному
мастерству «Профессионалы» в 2025 г

2025 г

Конкурсное задание разработано экспертным сообществом и утверждено Менеджером компетенции, в котором установлены нижеследующие правила и необходимые требования владения профессиональными навыками для участия в соревнованиях по профессиональному мастерству.

Конкурсное задание включает в себя следующие разделы:

1. ОСНОВНЫЕ ТРЕБОВАНИЯ КОМПЕТЕНЦИИ	3
1.1. ОБЩИЕ СВЕДЕНИЯ О ТРЕБОВАНИЯХ КОМПЕТЕНЦИИ	3
1.2. ПЕРЕЧЕНЬ ПРОФЕССИОНАЛЬНЫХ ЗАДАЧ СПЕЦИАЛИСТА ПО КОМПЕТЕНЦИИ «ВЕБ-ТЕХНОЛОГИИ»	3
1.3. ТРЕБОВАНИЯ К СХЕМЕ ОЦЕНКИ	9
1.4. СПЕЦИФИКАЦИЯ ОЦЕНКИ КОМПЕТЕНЦИИ.....	10
1.5. КОНКУРСНОЕ ЗАДАНИЕ	10
1.5.1. Разработка/выбор конкурсного задания (https://disk.yandex.ru/i/XJfQr1jBmZ2rew).....	10
Разработанные задания должны быть размещены по адресу http://xxxxxx-m4.domain.ru/u/ , где xxxxxx - логин конкурсанта (указан на индивидуальной карточке), у — номер задачи. Медиафайлы задач должны быть скопированы в соответствующую папку задачи. Также в медиа возможны уточняющие требования, специфичные для конкретной задачи.....	40
2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ.....	40
2.1. Личный инструмент конкурсанта.....	40
2.2. Материалы, оборудование и инструменты, запрещенные на площадке	40
3. ПРИЛОЖЕНИЯ	42

ИСПОЛЬЗУЕМЫЕ СОКРАЩЕНИЯ

1. *IP – Информационный ресурс*

2. *SSH - SSH (от англ. secure shell — безопасная оболочка) — это защищённый сетевой протокол для удалённого управления сервером через интернет.*

3. *FTP - File Transfer Protocol», или «протокол передачи файлов» — это набор процедур или правил, позволяющих электронным устройствам взаимодействовать между собой.*

4. *CMS - Content Management System) — это система создания и управления сайтом. Это визуально удобный интерфейс, с помощью которого можно добавлять и редактировать содержимое сайта.*

1. ОСНОВНЫЕ ТРЕБОВАНИЯ КОМПЕТЕНЦИИ

1.1. ОБЩИЕ СВЕДЕНИЯ О ТРЕБОВАНИЯХ КОМПЕТЕНЦИИ

Требования компетенции (ТК) «Веб-технологии» определяют знания, умения, навыки и трудовые функции, которые лежат в основе наиболее актуальных требований работодателей отрасли.

Целью соревнований по компетенции является демонстрация лучших практик и высокого уровня выполнения работы по соответствующей рабочей специальности или профессии.

Требования компетенции являются руководством для подготовки конкурентоспособных, высококвалифицированных специалистов / рабочих и участия их в конкурсах профессионального мастерства.

В соревнованиях по компетенции проверка знаний, умений, навыков и трудовых функций осуществляется посредством оценки выполнения практической работы.

Требования компетенции разделены на четкие разделы с номерами и заголовками, каждому разделу назначен процент относительной важности, сумма которых составляет 100.

1.2. ПЕРЕЧЕНЬ ПРОФЕССИОНАЛЬНЫХ ЗАДАЧ СПЕЦИАЛИСТА ПО КОМПЕТЕНЦИИ «ВЕБ-ТЕХНОЛОГИИ»

Таблица №1

Перечень профессиональных задач специалиста

№ п/п	Раздел	Важность в %
1	Тестирование информационных ресурсов	7
	Специалист должен знать и понимать: – способы решения возникающих проблем, анализ проблемной ситуации, возникшей в ходе решения профессиональных задач, пути их решения с учетом этических норм и правил, опираясь на профессиональную этику; – принципы, лежащие в основе сбора и представления информации; – основные приемы и методы визуального представления информации (черновое макетирование страниц, объектно-событийное моделирование, создание блок-схем и др.);	

	<p>– английский язык в рамках чтения и понимания официальной технической документации по используемым технологиям и языкам программирования.</p>	
	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> – собирать, анализировать и оценивать информацию; – использовать навыки грамотности для толкования стандартов и требований; – составлять тестовую документацию для тестирования новой функциональности продукта – проводить ручное тестирование новых функциональностей – проводить регрессионное ручное тестирование – вести баг-репорты – составлять отчеты по итогам тестирования – автоматизировать регрессионное тестирование – общаться с заказчиком, командой разработки и тестирования 	
2	<p>Техническая поддержка и администрирование информационных ресурсов</p>	7
	<p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> – принципы и практики, которые позволяют продуктивно работать, в том числе в команде; – аспекты систем, которые позволяют повысить продуктивность и выработать оптимальную стратегию; – основные принципы выбора технологий и инструментария для решения поставленных задач (проектов); - основные подходы к планированию и документированию проекта. 	
	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> – формировать архитектуру проекта (программного продукта) в соответствии с последними отраслевыми решениями; – выбирать технологии и инструменты для решения поставленных задач; – планировать график рабочего дня с учетом требований; – планировать задачи, учитывать временные ограничения и сроки; 	

	<ul style="list-style-type: none"> – решать распространенные задачи веб-дизайна и разработки кода; – формировать тестовые наборы, применять инструменты автоматического тестирования; – производить отладку кода программ и находить ошибки; – оптимально использовать компьютерное оборудование и программное обеспечение для повышения эффективности своей работы; – использовать менеджеры пакетов при разработке проекта; - использовать систему контроля версий. 	
3	<p>Разработка интерфейса пользователя</p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> – структуру и общепринятые элементы веб-страниц различных видов и назначений; – основные принципы организации контента веб-приложения; – основные правила выбора цвета, работы с типографикой и композицией; – принципы и методы создания и адаптации графики для использования ее на веб-сайтах; – ограничения, которые накладывают мобильные устройства и разрешения экранов при использовании их для просмотра веб-сайтов; – принципы построения эстетичного и креативного дизайна; – 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; – основные принципы применения соответствующих CSS правил и селекторов для получения ожидаемого результата; – лучшие практики для Search Engine Optimization (SEO) и интернет-маркетинга; <p>Специалист должен уметь:</p>	26

	<ul style="list-style-type: none"> – создавать, использовать и оптимизировать изображения для веб-сайтов; – 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; – создавать «отзывчивый» дизайн, который будет отображаться корректно на различных устройствах и при разных разрешениях; – 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 веб-интерфейсы с учетом принципов Search Engine Optimization; – использовать препроцессоры. 	
4	Разработка на стороне клиента	25
	<p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> – основные принципы паттерновой разработки веб-приложений; – ECMAScript (JavaScript); – принципы, особенности и способы использования открытых фреймворков; – принципы разработка кода с использованием открытых библиотек; – различные интерфейсы взаимодействия с объектами браузера 	
	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> – создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта; – манипулировать элементами страницы веб-приложения; 	

	<ul style="list-style-type: none"> – разрабатывать анимацию для повышения доступности и визуальной привлекательности веб-приложения; – применять открытые библиотеки и фреймворки; – тестировать веб-приложение. 	
	<p>Разработка веб приложения на стороне сервера</p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> – процедурные и объектно-ориентированные языки PHP, Python, Node.js; – основные принципы и правила использования открытых библиотек и фреймворков; – распространенные модели организации и хранения данных; – основные принципы создания баз данных; – основные принципы обмена данными между клиентом и сервером; – методы работы с протоколами SSH/(s)FTP при подключении к серверам; – способы разработки программного кода в соответствии с паттернами проектирования; – основные принципы обеспечения безопасности веб-приложения. 	
5	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> – разрабатывать процедурный и объектно-ориентированный программный код; – разрабатывать веб-сервисы с применением PHP, Python, Node.js в соответствии с техническим заданием; – создавать библиотеки и модули для выполнения повторяющихся задач; – разрабатывать веб-приложения с доступом к различным базам данных; – создавать SQL (Structured Query Language) запросы и конструкции; – обеспечивать безопасность (устойчивость веб-приложения к атакам и взломам); – интегрировать существующий и создавать новый программный код с API (Application Programming Interfaces); – использовать открытые библиотеки и фреймворки; 	25

6	Разработка информационных ресурсов с использованием готовых решений	10
	<p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> – преимущества и ограничения системы управления контентом с открытым исходным кодом; – методы работы с плагинами/модулями; – способы реализации функциональных возможностей CMS; – основные принципы организации контента веб-приложения; – понимать необходимость поддержания и обновления для плагинов CMS и соответствующих модулей для безопасности системы; <p>основные принципы интеграции с внешними веб-приложениями.</p>	
	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> – устанавливать, настраивать и модифицировать систему управления контентом; – устанавливать, настраивать и обновлять плагины/модули CMS; – создать пользовательские темы/шаблоны для системы управления контентом; – создавать пользовательские плагины/модули и шаблоны/темы; – использовать встроенные методы и средства CMS при разработке веб-приложения. 	

1.3. ТРЕБОВАНИЯ К СХЕМЕ ОЦЕНКИ

Сумма баллов, присуждаемых по каждому аспекту, должна попадать в диапазон баллов, определенных для каждого раздела компетенции, обозначенных в требованиях и указанных в таблице №2.

Таблица №2

Матрица пересчета требований компетенции в критерии оценки

Критерий/Модуль						Итого баллов за раздел ТРЕБОВА НИЙ КОМПЕТЕ НЦИИ
Разделы ТРЕБОВА НИЙ КОМПЕТЕ НЦИИ		А	Б	В	Г	
	1	2	2	1	2	7
	2	2	2	1,5	1,5	7
	3	20,5		2	3,5	26
	4			25		25
	5		25			25
	6				10	10
Итого баллов за критерий/мод уль		24,5	29	29,5	17	100

1.4. СПЕЦИФИКАЦИЯ ОЦЕНКИ КОМПЕТЕНЦИИ

Оценка Конкурсного задания будет основываться на критериях, указанных в таблице №3:

Таблица №3

Оценка конкурсного задания

Критерий		Методика проверки навыков в критерии
А	Разработка интерфейса пользователя	Проверка результата выполнения модуля после его завершения
Б	Разработка Веб-приложения на стороне клиента	Проверка результата выполнения модуля после его завершения
В	Разработка Веб-приложения на стороне сервера	Проверка результата выполнения модуля после его завершения
Г	Спидтаски	Проверка результата выполнения модуля после его завершения

1.5. КОНКУРСНОЕ ЗАДАНИЕ

Общая продолжительность Конкурсного задания¹: 12 ч.

Количество конкурсных дней: 3 дня

Вне зависимости от количества модулей, КЗ должно включать оценку по каждому из разделов требований компетенции.

Оценка знаний конкурсанта должна проводиться через практическое выполнение Конкурсного задания. В дополнение могут учитываться требования работодателей для проверки теоретических знаний / оценки квалификации.

1.5.1. Разработка/выбор конкурсного задания

(<https://disk.yandex.ru/i/XJfQr1jBmZ2rew>)

Конкурсное задание состоит из 4 модулей, включает обязательную к выполнению часть (инвариант) - 3 модуля, и вариативную часть - 1 модуль. Общее количество баллов конкурсного задания составляет 100.

Обязательная к выполнению часть (инвариант) выполняется всеми регионами без исключения на всех уровнях чемпионатов.

Количество модулей из вариативной части, выбирается регионом самостоятельно в зависимости от материальных возможностей площадки соревнований и потребностей работодателей региона в соответствующих специалистах. В случае если ни один из модулей вариативной части не

¹ Указывается суммарное время на выполнение всех модулей КЗ одним конкурсантом.

подходит под запрос работодателя конкретного региона, то вариативный (е) модуль (и) формируется регионом самостоятельно под запрос работодателя. При этом, время на выполнение модуля (ей) и количество баллов в критериях оценки по аспектам не меняются.

1.5.2. Структура модулей конкурсного задания (инвариант/вариатив)

МОДУЛЬ А. РАЗРАБОТКА ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

Технологии этого модуля: граф. Дизайн, HTML5, CSS3

Время на выполнение: 4 часов

Задание: Вам необходимо использовать все имеющиеся навыки в дизайне и верстке чтобы создать Landing Page, а также все остальные страницы.

В современном мире с каждым днём появляется всё больше и больше новых технологий. Всё больше людей и организаций осознают важность сохранения окружающей среды и перехода к устойчивому развитию.

К вам обратилась компания «ЭкоГород» – новая и динамичная организация, направленная на поддержку экологических инициатив и развитие экологически чистых решений в городской среде. Главная цель компании – популяризация устойчивого образа жизни, разработка и реализация проектов, способствующих улучшению экологии, и предоставление комплексных решений для сохранения природных ресурсов.

Использование любых фреймворков и библиотек (Bootstrap, Tailwind, например) запрещено. Разрешены SASS, LESS, PostCSS.

Страницы должны быть адаптивны. При переходе от десктопной версии главной страницы (ширина экрана 1920px) к мобильной версии (ширина экрана 768px) должно быть реализовано не менее четырех структурных изменений контента.

На остальных разрешениях контент должен отображаться корректно.

ОПИСАНИЕ ПРОЕКТА И ЗАДАЧ

Ваша задача – сверстать следующие страницы веб-сайта:

- Главная страница - Landing Page

- Страница входа в личный кабинет
- Страница регистрации
- Страница мероприятий
- Страница пунктов приема отходов
- Страница личный кабинет пользователя
- Страница Сведения о мероприятии
- Страница акций и скидок
- Страница контактов
- Страница создания отзыва

Главная страница (Landing Page):

Главная страница должна содержать следующие блоки:

- **Шапка (Header):**
 - Логотип компании
 - Меню навигации:
 - Главная
 - Мероприятия
 - Пункты приема
 - О проекте
 - Контакты
 - Кнопка "Вход".
- **Поле поиска:**
 - Текст: *"Найдите ближайший пункт приема отходов"* (поиск в статичном списке).
 - Поле ввода: *Введите город.*
 - Кнопка "Найти".
- **Секция "О проекте":**
 - Заголовок: *"ЭкоГород — экологическое будущее уже сегодня!"*
 - Краткое описание
- **Секция "Наши достижения":**

- Заголовок: *"Мы делаем мир чище!"*
- Блок с информацией:
 - 100+ мероприятий.
 - 50,000 участников.
 - 1,200 тонн переработанных отходов.
- **Карта пунктов приема:**
 - Заголовок: *"Найдите ближайший пункт приема отходов"*
 - Статический макет карты с маркерами пунктов.
- **Секция "Подписка на рассылку":**
 - Заголовок: *"Присоединяйтесь к нам!"*
 - Форма с полями:
 - Email
 - Кнопка "Подписаться".
- **Подвал сайта:**
 - Логотип компании
 - Меню навигации
 - Мероприятия
 - Пункты приема
 - О проекте
 - Контакты
 - Связь
 - Телефон "8 (800) 111-11-11"
 - Email
 - Telegram
 - VK
 - WhatsApp

Страница регистрации

Страница должна содержать следующие блоки:

- **Заголовок:** *"Присоединяйтесь к ЭкоГороду!"*.
- **Форма регистрации:**
 - Фамилия

- Имя
- Отчество
- Телефон
- Email
- Дата рождения
- Пароль
- Повтор пароля
- Кнопка для регистрации
- Чекбокс *"Я согласен с политикой конфиденциальности"*.

- **Текст-подсказка:** "Уже есть аккаунт? Войти".

Страница входа в личный кабинет

На этой странице вам необходимо сделать форму со следующими полями:

- Телефон или email
- Пароль
- **Кнопка для входа**

Страница мероприятий:

Страница должна содержать следующую информацию:

- **Фильтр мероприятий:**
 - Поля фильтрации:
 - Дата (диапазон).
 - Тип мероприятия (*уборка, образование, посадка деревьев*).
- **Список мероприятий:**
 - Заголовок
 - Карточки мероприятий:
 - Изображение.
 - Название мероприятия.
 - Дата, место.
 - Краткое описание.
 - Кнопка "Подробнее".
 - Карточки должны быть интерактивными с hover-эффектами.

- **Пагинация:**

- Номера страниц с выделением текущей страницы.

Страница пунктов приема отходов

Страница должна содержать следующую информацию:

- **Заголовок:**

- **Фильтр пунктов:**

- Поля фильтрации:
 - Тип отходов (например, *пластик, бумага, металл, электроника*).
 - Ближайшие пункты (сортировка по удаленности).
- Кнопка "Применить фильтр".

- **Карта пунктов:**

- Интерактивная карта с отметками пунктов.
- При наведении на метку выводится краткая информация о пункте (название, адрес, принимаемые отходы).

- **Список пунктов:**

- Каждая карточка должна содержать:
 - Название пункта.
 - Адрес.
 - Контактный телефон.
 - Принимаемые отходы (иконками).
 - Кнопка "Подробнее".

Страница контактов

Страница должна содержать следующую информацию:

- **Секция "Общие контакты":**

- Телефон: +7 (800) 555-35-35.
- Email: info@ecogorod.ru.
- Социальные сети: иконки (VK, Telegram, Instagram).

- **Секция "Региональные представительства":**

- Карта с отметками региональных офисов.

- Карточки офисов:
 - Название города.
 - Адрес.
 - Телефон.

Страница "Личный кабинет пользователя"

Страница должна содержать следующую информацию:

- **Секция профиля:**

- Отображение информации:
 - Имя пользователя.
 - Email.
 - Номер телефона.
 - Город.
- Кнопки:
 - "Редактировать данные".
 - "Выйти".

- **Секция "Мои мероприятия":**

- Заголовок: *"Записанные мероприятия"*.
- Список карточек мероприятий, каждая содержит:
 - Название.
 - Дата, место.
 - Статус (например, *Подтверждено*, *Ожидание подтверждения*).
 - Кнопка "Отменить участие".

- **Секция "История участия":**

- Заголовок: *"Прошедшие мероприятия"*.
- Список мероприятий с краткой информацией и возможностью оставить отзыв.

- **Форма редактирования профиля:**

- Поля:
 - Имя.

- Фамилия.
- Email.
- Телефон.
- Город.

- Кнопка "Сохранить изменения".

Страница "Сведения о мероприятии"

Страница должна содержать следующую информацию:

- Название.
- Изображение (баннер мероприятия).
- Дата и время проведения.
- Место проведения (ссылка на карту).
- Краткое описание.
- Контактное лицо (имя и номер телефона).
- **Секция "Программа мероприятия":**
 - Список пунктов программы с временем и активностями.
- **Секция "Отзывы участников":**
 - Блок отзывов (имя, текст, рейтинг).
 - Кнопка "Добавить отзыв".
- **Форма записи на мероприятие:**
 - Поля:
 - Имя.
 - Email.
 - Телефон.
 - Кнопка "Записаться".

Страница "Акции и скидки"

- **Секция "Текущие акции":**
 - Карточки акций:
 - Название акции.
 - Превью-изображение.
 - Старая цена (перечеркнута) и новая цена.

- Краткое описание.
- Кнопка "Подробнее".
- **Секция "Скоро заканчиваются":**
 - Карточки акций, отсортированные по дате завершения.
- **Секция "Архив акций":**
 - Карточки прошлых акций с возможностью просмотра подробностей.

Страница "Создание и редактирование отзыва"

- **Форма отзыва:**
 - Выбор мероприятия (выпадающий список с прошлыми мероприятиями).
 - Текст отзыва.
 - Рейтинг (выбор от 1 до 5 звезд).
 - Кнопка "Отправить".

ИНСТРУКЦИЯ ДЛЯ КОНКУРСАНТА

Ваш HTML/CSS должен быть валидным. Используйте лучшие практики верстки в соответствии с заданием.

Страницы должны быть согласованы между собой. Вам нужно использовать не менее одного из представленных шрифтов.

Оценка будет производиться в браузере Google Chrome.

Сверстаный веб-сайт должен быть доступен по адресу <http://xxxxxx-m1.domain.ru/>, где xxxxxx - логин участника (указан на индивидуальной карточке).

Сохраните сверстаные страницы и созданные дизайн-макеты со следующими именами:

- Главная - index.html, index.png
- Страница входа в личный кабинет - login.html, login.png
- Страница регистрации - register.html, register.png
- Страница личного кабинета пользователя - profile.html, profile.png
- Страница мероприятий – event.html, event.png

- Страница Сведения о мероприятии – information_event.html, information_event.png
- Страница пунктов приема отходов – points.html, points.png
- Страница акций и скидок – sale.html, sale.png
- Страница контактов– contact.html, contact.png
- Страница создания отзыва– add.html, add.png

Макеты должны быть расположены в отдельной папке (/design).

МОДУЛЬ Б. РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ КЛИЕНТА

Технологии этого модуля: Клиентское программирование

Время на выполнение: 2 часа

К вам обратилась компания «ЭкоГород» – новая и динамичная организация, направленная на поддержку экологических инициатив и развитие экологически чистых решений в городской среде. Главная цель компании – популяризация устойчивого образа жизни, разработка и реализация проектов, способствующих улучшению экологии, и предоставление комплексных решений для сохранения природных ресурсов.

Заказчик предоставляет вам полностью готовую верстку со всеми страницами и рабочее API. Вам необходимо использовать все имеющиеся навыки в клиентской разработке для создания Single Page Application, далее SPA.

Заказчик хочет, чтобы приложение можно было легко поддерживать, поэтому использование JavaScript фреймворков будет плюсом.

ВНИМАНИЕ! Проверяться будут только работы, загруженные на сервер!

ОПИСАНИЕ ПРОЕКТА И ЗАДАЧ

Ваша задача – реализовать SPA, которое будет работать с уже разработанным API.

Для вашего удобства, во всех URL будет использоваться переменная {host} которая обозначает хост адрес API: <http://??????.api.wsr.ru> – где ?????? Ваш логин.

Для тестирования работы запросов, требующих авторизации, используйте:

Пользователь

Телефон: 89046512337

Пароль: user2025

Оператор

Телефон: 89046512338

Пароль: admin-professional

Примечание: Все запросы к API должны включать заголовок Authorization: Bearer {accessToken}, Accept-Language: {ru, en}.

Ваше SPA должно состоять из следующих экранов:

- Страница входа в личный кабинет
- Страница личного кабинета пользователя
- Страница личного кабинета оператора
- Страница калькулятора расчета эко-товаров
- Страница с подтверждением заявки на сдачу эко-товаров

Для улучшения пользовательского опыта (UX) все взаимодействия с интерфейсом приложения должны сопровождаться анимацией, интерактивными сообщениями и т.д.

ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

Регистрация

На данном экране представлена форма для регистрации нового пользователя. При вводе некорректных значений у соответствующих полей формы отображаются тексты ошибок, а сами поля подсвечиваются красным. При успешной регистрации происходит переход на экран входа в систему.

Интерактивные элементы экрана

- По щелчку на кнопку регистрации происходит попытка зарегистрировать нового пользователя.

Вход в личный кабинет

На данном экране представлена форма для входа пользователя в личный кабинет. При вводе некорректных значений у соответствующих полей формы отображаются тексты ошибок, а сами поля подсвечиваются красным. При

успешной авторизации происходит переход на экран “Страница личного кабинета пользователя/Страница личного кабинета оператора” в зависимости от роли авторизованного пользователя, иначе выводится соответствующее сообщение.

Страница личного кабинета пользователя

На этом экране пользователь может просматривать свою историю заявок на сдачу эко-товаров. Если заявка была изменена оператором, то должна быть пометка, что заявка изменена и видна информация не только заявки, но и того, что изменил оператор. В разделе профиля можно выйти из системы. На экране также отображается кнопка «калькулятор» для расчета стоимости сдачи эко-товаров.

Интерактивные элементы экрана:

- По щелчку на калькулятор происходит переход на страницу калькулятора расчета эко-товаров.

Страница калькулятора расчета эко-товаров

На этом экране пользователи могут рассчитать стоимость сдачи эко-товаров. Для этого необходимо выбрать из списка данные о группе товара, на основании выбора группы товара, должны погрузиться данные о типе товара данной группы, после необходимо выбрать размер товара и ввести количество сдаваемых товаров. После ввода данных, калькулятор автоматически рассчитывает цену и отображает итоговую сумму.

Интерактивные элементы экрана

- По щелчку на кнопку оформить происходит вычисление стоимости и сохранение данных в БД.
- По щелчку на кнопку назад происходит возврат на страницу личного кабинета пользователя.

Страница личного кабинета оператора

На этом экране оператор может просматривать заявки пользователей на сдачу эко-товаров, а также подтверждать их. Оператор может просматривать и редактировать детали каждой заявки.

Интерактивные элементы экрана:

- По щелчку на заявку происходит переход к деталям заявки.

Страница с подтверждением заявки на сдачу эко-товаров

На данном экране оператор подтверждает заявку на сдачу эко-товаров. Отображаются все введенные данные о типах и количестве сдаваемых материалов, а также итоговая стоимость. При этом оператор может редактировать детали каждой заявки, включая группу, тип и количество сдаваемых товаров, статус заявки, а также может удалять материал из заявки.

Инструкция для конкурсанта

Разработанное приложение должно быть доступно по адресу <http://xxxxxx-m2.wsr.ru>, где xxxxxx - логин участника.

Для оценки качества кода необходимо выгружать на сервер также не скомпилированный вариант проекта.

Вам предоставляются следующие конфигурации фреймворков и библиотек:

- VueJS 2.x
- VueJS 3.x
- Vue Router 3.x
- Vue Router 4.x
- Vue CLI
- Vue Query
- React
- React Router
- React Query
- React Hook Form
- Yup, Zod, Joi, Vest, Ajv
- React CLI
- Angular CLI

- Axios

Вы можете использовать любой из представленных фреймворков. Для оценки качества кода необходимо выгружать на сервер также не скомпилированный вариант проекта.

МОДУЛЬ В. РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ СЕРВЕРА

Технологии этого модуля: REST API

Время на выполнение: 2 часа

К вам обратилась компания «ЭкоГород» – новая и динамичная организация, направленная на поддержку экологических инициатив и развитие экологически чистых решений в городской среде. Главная цель компании – популяризация устойчивого образа жизни, разработка и реализация проектов, способствующих улучшению экологии, и предоставление комплексных решений для сохранения природных ресурсов.

Заказчик хочет, чтобы api можно было легко поддерживать, поэтому использование фреймворков будет плюсом.

ВНИМАНИЕ! Проверяться будут только работы, загруженные на сервер! В медиафайлах вам предоставляется sql дамп с готовой базой данных. База данных уже содержит набор данных, которые НЕ должны быть изменены! Структуру БД менять также запрещается. Любое изменение или удаление предоставленных данных в БД будет влиять на вашу оценку.

ОПИСАНИЕ ПРОЕКТА И ЗАДАЧ

Ваша задача – реализовать REST API, которое будет отвечать требованиям заказчика. Для вашего удобства, во всех URL будет использоваться переменная {host} которая обозначает адрес `http://xxxxxxx-m3.wsr.ru/`, где xxxxxx - логин участника.

Примечание: Все запросы к API должны включать заголовок Authorization: Bearer {accessToken}, Accept-Language: {ru, en}.

При попытке доступа гостя к защищенным авторизацией функциям системы во всех запросах необходимо возвращать ответ следующего вида:

Status: 401

Content-Type: application/json

Body:

```
{  
  "error": "unauthorized",  
  "message": "Authentication credentials were missing or incorrect"  
}
```

При ошибке сервера во всех запросах необходимо возвращать ответ следующего вида:

Status: 500

Content-Type: application/json

Body:

```
{  
  "error": "server_error",  
  "message": "Internal server error occurred"  
}
```

При ошибке валидации во всех запросах необходимо возвращать ответ следующего вида:

Status: 400

Content-Type: application/json

Body:

```
{  
  "error": "validation_error",  
  "message": <message>,  
}
```

Описание общего функционала

Регистрация:

При успешной регистрации возвращается сгенерированный accessToken и refreshToken добавленного пользователя. Функция должна принимать следующие параметры:

- phone – номер телефона, маска ввода: +7XXXXXXXXXX
- password - пароль пользователя обязательный, состоит минимум из 3 символов, из которых как минимум одна строчная, одна прописная и одна цифра
- lastName - имя, обязательное
- firstName - фамилия обязательное

Request	Response
URL: {{host}}/auth/register Method: POST Headers - Content-Type: application/json - Authorization: Bearer-token Body: { "phone": "+1234567890", "password": "SecurePass123", "lastName": "Doe", "firstName": "John" }	<u>Successful</u> Status: 201 Content-Type: application/json Body: { "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." } <u>Ошибка (невалидный или истекший refresh token)</u> Status: 409 Content-Type: application/json Body: { "error": "invalid_token", "message": "Refresh token has expired" }

Авторизация:

При успешной авторизации возвращается сгенерированный access_token и

refresh_token,

Особенности:

- AccessToken срок действия – 15 минут, RefreshToken – 7 дней.
- Автоматическое обновление токенов при истечении accessToken.
- Ролевая модель:
 - Пользователь: доступ к /requests, /categories
 - Оператор: доступ к /operator/requests

Request	Response
URL: {{host}}/auth/login Method: POST Headers - Content-Type: application/json - Authorization: Bearer-token Body: { "phone": "+1234567890", "password": "SecurePass123!" }	<u>Successful</u> Status: 200 Content-Type: application/json Body: { "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." }
Request	Response
URL: {{host}}/auth/refresh Method: POST Headers - Content-Type: application/json - Authorization: Bearer-token Body: { "access_token": "<сгенерированный access_token>" }	<u>Successful</u> Status: 200 Content-Type: application/json Body: { "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." }

Описание функционала авторизованного пользователя

Сброс авторизации:

Запрос предназначен для очистки значение access_token и refresh_token пользователя.

Request	Response
URL: {{host}}/logout Method: GET	<u>Successful</u> Status: 204 Content-Type: application/json Body (не отдается)

Описание функционала пользователя

Получение информации о себе

Request	Response
URL: {{host}}/users/me Method: GET	<u>Successful</u> Status: 200 Content-Type: application/json Body: <pre>{ "phone": "+1234567890", "lastName": "Doe", "firstName": "John" }</pre>

Изменение информации о себе

Request	Response
URL: {{host}}/users/me Method: PUT Headers - Content-Type: application/json - Authorization: Bearer-token Body: <pre>{ "phone": "+1987654321", "lastName": "Smith", "firstName": "Jane" }</pre>	<u>Successful</u> Status: 200 Content-Type: application/json Body (не отдается)

Получение материалов для калькуляции эко-товара

1. Клиент запрашивает список категорий товаров ('/categories'). Сервер возвращает доступные категории товаров.

2. Клиент использует categoryId из предыдущего ответа для запроса типов товаров, относящихся к категории (`/categories/{categoryId}/items`). Сервер возвращает список доступных типов товаров.
3. Клиент использует itemId из предыдущего ответа для запроса размеров и цен, относящихся к типу товара (`/items/{itemId}/variants`). Сервер возвращает список размеров и их соответствующих цен.

Request	Response
URL: {{host}}/categories Method: GET Headers - Content-Type: application/json - Authorization: Bearer-token	<u>Successful</u> Status: 200 Content-Type: application/json Body: <pre>[{ "id": "metals", "name": "Metals" }]</pre>
Request	Response
URL: {{host}}/categories/{categoryId}/items Method: GET Headers - Content-Type: application/json - Authorization: Bearer-token	<u>Successful</u> Status: 200 Content-Type: application/json Body: <pre>[{ "id": "aluminum", "name": "Aluminum", "description": "Various aluminum products" }]</pre>
Request	Response
URL: {{host}}/items/{itemId}/variants Method: GET Headers - Content-Type: application/json - Authorization: Bearer-token	<u>Successful</u> Status: 200 Content-Type: application/json Body: <pre>[{ "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }]</pre>

Калькулятор расчета эко-товаров. Сохранение данных

Request	Response
<p>URL: {{host}}/ requests</p> <p>Method: POST</p> <p>Headers</p> <ul style="list-style-type: none">- Content-Type: application/json- Authorization: Bearer {token} <p>Body:</p> <pre>{ "items": [{ "variantId": "alu_can", "quantity": 15 }, { "variantId": "alu_foil", "quantity": 8 }], "scheduledDate": "2024-03-15" }</pre>	<p><u>Successful</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>{ "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:15:22.454Z" }], "status": "PENDING", "modifiedBy": "USER", }</pre>

	<pre>"modifiedAt": "2025-02-12T09:15:22.454Z" }</pre>
--	---

Просмотр своих заявок

Request	Response
<p>URL: {{host}}/ requests</p> <p>Method: GET</p> <p>Headers</p> <ul style="list-style-type: none">- Content-Type: application/json-Authorization: Bearer-token	<p><u>Successful</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>[{ "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:16:36.305Z" }], "status": "PENDING", "modifiedBy": "USER",</pre>

	<pre>"modifiedAt": "2025-02-12T09:16:36.305Z" }]</pre>
--	---

Просмотр деталей конкретной заявки

Request	Response
<p>URL: {{host}}/requests/{requestId}</p> <p>Method: GET</p> <p>Headers</p> <ul style="list-style-type: none">- Content-Type: application/json- Authorization: Bearer-token	<p><u>Successful</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>{ "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:18:03.325Z" }], "status": "PENDING", "modifiedBy": "USER",</pre>

	"modifiedAt": "2025-02-12T09:18:03.325Z" }
--	---

Изменение деталей заявки

Request	Response
<p>URL: {{host}}/requests/{requestId} Method: PUT</p> <p>Headers - Content-Type: application/json - Authorization: Bearer-token</p> <p>Body: { "items": [{ "variantId": "alu_can", "quantity": 20 }, { "variantId": "alu_foil", "quantity": 5 }], "scheduledDate": "2024-03-16" }</p>	<p><u>Successful</u></p> <p>Status: 200 Content-Type: application/json Body: { "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:18:27.309Z" }], "status": "PENDING", "modifiedBy": "USER", "modifiedAt": "2025-02-12T09:18:27.309Z" }</p>

	}
--	---

Описание функционала оператора

Просмотр всех заявок

Request	Response
<p>URL: {{host}}/operator/requests</p> <p>Method: GET</p> <p>Headers</p> <ul style="list-style-type: none">- Content-Type: application/json- Authorization: Bearer-token	<p><u>Successful</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>[{ "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:20:13.891Z" }], "status": "PENDING",</pre>

	<pre>"modifiedBy": "USER", "modifiedAt": "2025-02-12T09:20:13.891Z" }]</pre>
--	---

Изменение деталей заявки

Request	Response
<p>URL: { {host} }/operator/requests/{requestId}</p> <p>Method: PATCH</p> <p>Headers - Content-Type: application/json - Authorization: Bearer-token</p> <p>Body: { "status": "PROCESSING", "adjustments": [{ "variantId": "alu_can", "finalQuantity": 18, "comment": "Damaged items removed" }, { "variantId": "pet_bottles", "finalQuantity": 10, "comment": "Added during processing" }] }</p>	<p><u>Successful</u></p> <p>Status: 200 Content-Type: application/json Body: { "originalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "quantity": 15, "calculatedPrice": 37.5 }], "finalItems": [{ "variant": { "id": "alu_can", "name": "Aluminum Cans", "pricePerUnit": 2.5 }, "initialQuantity": 0, "finalQuantity": 0, "calculatedPrice": 0, "comment": "string" }], "adjustments": [{ "variantId": "string", "initialQuantity": 0, "finalQuantity": 0, "comment": "string", "adjustedBy": "SYSTEM", "adjustedAt": "2025-02-12T09:20:40.088Z" }], "status": "PENDING", }</p>


	<pre>"modifiedBy": "USER", "modifiedAt": "2025-02-12T09:20:40.088Z" }</pre>
--	---

МОДУЛЬ Г. СПИДТАСКИ (ВАРИАТИВ)

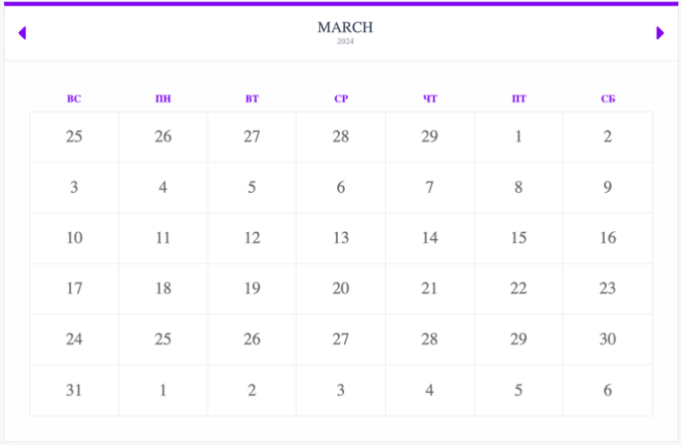
Время на выполнение: 4 часов

Ваша задача - решить максимальное количество небольших задач.

№	Задача
Разработка интерфейса пользователя	
1	<p>Дано изображение, сделайте его в оттенках серого, наложите эффект размытия. <i>Задачу можно выполнить как с помощью HTML+CSS так и с помощью графического редактора</i></p> <p><i>Резушьтат должен быть представлен в файле: task1.html или task.jpg/png</i></p>
2	<p>Создайте UI-комплект из следующих элементов: текстовое поле, кнопка, выпадающий список, чекбокс, радиокнопка, метка с чекбоксом, метка с радиокнопкой, выпадающий календарь ввода, диапазон, прогресс-бар.</p> <p><i>Резушьтат должен быть представлен в папке: task2</i></p>
3	<p>С помощью CSS/HTML сделать 3D-куб, который медленно бесконечно вращается. Каждая грань куба должна дополнительно поделена на 9 частей. (нельзя использовать изображения) <i>Резушьтат должен быть представлен в файле: task3.html</i></p>
4	<p>Вам предоставляется спрайт таблица (sprite sheet) с персонажем ее необходимо анимировать. Персонаж должен бежать слева на права</p> <p><i>Резушьтат должен быть представлен в файле: task4.html</i></p>
5	<p>Сделать адаптивный макет (только секции) для блога: шапка, основная (в основной секции необходимо реализовать блоки для</p>

	<p>постов. Ширина постов составляет $\frac{1}{3}$ ширины контейнера), боковая панель, футер. Макет должен корректно отображаться в следующих значениях ширины: $\geq 576\text{px}$, $\geq 768\text{px}$, $\geq 992\text{px}$.</p> <p><i>Резушьтат должен быть представлен в файле: task5.html</i></p>
6	<p>Создание тумблера для IOS с использованием только HTML и CSS. Посмотрите видео, чтобы узнать подробности анимации и детали.</p>  <p><i>Резушьтат должен быть представлен в файле: task6.html</i></p>
Разработка на стороне клиента	
7	<p>Создать генератор цветов RGB. При перемещении соответствующих ползунков должен изменяться цвет и выводится RGB код Используйте файл 7.html</p> <p><i>Резушьтат должен быть представлен в файле: task7.html</i></p>
8	<p>Создайте экранную клавиатуру. При клике по кнопке в поле ввода должен записаться соответствующий символ. Shift переключает регистр. Используйте файл 8.html</p> <p><i>Резушьтат должен быть представлен в файле: task8.html</i></p>
9	<p>Создайте арифметическую функцию, которая при вводе строки выражения возвращает вычисленный результат. Эта функция должна включать следующие операторы, перечисленные в порядке по приоритету:</p> <ol style="list-style-type: none"> 1. Круглые скобки: () 2. Знак минус: - 3. Экспоненция: ^ 4. Умножение, деление, модуль: * / % 5. Сложение, вычитание: + - <p>Пример 1: Ввод функции: 1+2*3 Выход функции: 7</p>

	<p>Пример 2: Вход функции: $3^2\%5$ Выход функции: 4</p> <p>Пример 3: Вход функции: (-1-2)</p> <p><i>Резушьтат должен быть представлен в файле: task9.html</i></p>
10	<p>Вам необходимо реализовать часы, показывающие текущее местное системное время. Фон должен выглядеть так, как показано в медиафайле «clock.png», содержащем границы часов и тики для часов и минут. Также стрелки часов должны выглядеть так, как показано на изображении. Секундная стрелка часов обновляется каждую секунду, минутная стрелка обновляется каждую минуту, часовая стрелка обновляется каждую минуту.</p> <p>Изображение «clock.png» можно использовать по своему усмотрению.</p> <p><i>Резушьтат должен быть представлен в файле: task10.html</i></p>
11	<p>Попробуйте использовать холст, чтобы создать блокнот для рисования со следующими возможностями:</p> <ul style="list-style-type: none"> - Рисование с помощью нажатия и перемещения мыши - Переключение 3 цветов <p><i>Резушьтат должен быть представлен в файле: task11.html</i></p>
Разработка на стороне сервера	
12	<p>С помощью PHP/Python для создания изображения проверочного кода (captcha), которое должно соответствовать следующим правилам:</p> <ol style="list-style-type: none"> Случайным образом генерировать четыре символа, состоящих из английского алфавита (A-Z) или цифр (0-9). Четыре символа должны быть слегка повернуты. Четыре символа не должны располагаться в одном ряду. Изображение содержит не менее 3 случайных линий. Не менее 3 точек смешанного шума. <p><i>Резушьтат должен быть представлен в папке: task12</i></p>
13	

1 4	<p>Реализуйте загрузку файлов с автоматической сортировкой по папкам в зависимости от типа файла (images, videos, documents (docx, pdf, xls) audio, fonts). Файлы должны быть названы датой и временем его загрузки. Например: 2024_05_28_15_13.png</p> <p><i>Резушьтат должен быть представлен в папке: task14</i></p>
1 5	<p>Вам предоставляется несколько массивов в медиафайлах, реализуйте PHP/Python-функцию, которая сравнивает два заданных массива и возвращает новый массив, содержащий общие элементы из обоих заданных массивов.</p> <p><i>Резушьтат должен быть представлен в папке: task15</i></p>
1 6	<p>Реализуйте календарь</p> <ol style="list-style-type: none"> 1. Вы можете создать index.php, используя index.html 2. Когда вы откроете index.php, в верхней области календаря должны отображаться текущий месяц и текущий год, а сегодняшняя дата должна быть выделена, как в index.html. 3. При нажатии на кнопку со стрелкой влево должен отображаться календарь предыдущего месяца. 4. При нажатии на кнопку со стрелкой вправо должен быть показан календарь на следующий месяц.  <p><i>Резушьтат должен быть представлен в папке: task16</i></p>
<p>Разработка информационных ресурсов с использованием готовых решений</p> <p><i>Каждая задача может быть выполнена как в составе одного проекта WordPress, так и в отдельном проекте.</i></p>	
1 7	<p>Установите wordpress. Данные для авторизации администратора: Логин: admin2024 Пароль: profmaster404 Установите и активируйте тему BlankSlate</p>

1 8	Приведите внешний вид темы в соответствие с предоставленным шаблоном. Все стили должны быть сохранены, как и интерактивные элементы. Измените изображение тему
1 9	Создайте плагин, который показывает время, оставшееся до 1 сентября. При активации плагина создается виджет. При добавлении виджета в сайдбар должно отображаться оставшееся время в левом нижнем углу на каждой странице. Дата не должна быть статичной. Формат даты (ДД: ЧЧ: ММ: СС) Название плагина: «Обратный отсчет»
2 0	Сделайте плагин, который позволит загружать несколько медиафайлов в галерею wordpress с помощью пункта меню (в шаблоне) под названием «Загрузка галереи»
2 1	Создайте плагин, который выводит на Dashboard кнопку «Статус технических работ», на странице должна быть возможность переключить сайт в статус «технические работы» при этом у клиентской части сайта должна отображаться страница «Идут технические работы». При отключении технических работ, сайт должен быть снова доступен
2 2	Создайте плагин интернет-магазина. При активации плагина в панели администратора добавляется пункт меню «Магазин» при нажатии на который открывается еще два пункта «Добавить товар» и «Добавить категорию». При нажатии на кнопку «Добавить товар» открывается страница, где размещена форма добавления товара (название, описание, цена, изображение, категории). Категории должны быть в виде выпадающего списка, который ранее были добавлены на странице «Добавить категорию» При нажатии на кнопку «Добавить категорию» открывается страница, где выводится список категорий и сверху форма (поле + кнопка) для добавления новой категории. Карточки товаров (заголовок+цена+категори+кнопка «купить») должны выводиться на страницу «Товары»

Инструкция для конкурсанта

Для выполнения заданий по технологии “Разработка информационных ресурсов с использованием готовых решений” вам предоставляется CMS Wordpress 6.2.x. Остальные задачи необходимо решить без использования фреймворков и сторонних библиотек.

Разработанные задания должны быть размещены по адресу <http://xxxxxx-m4.domain.ru/y/>, где xxxxxx - логин конкурсанта (указан на индивидуальной карточке), у — номер задачи. Медиафайлы задач должны быть скопированы в соответствующую папку задачи. Также в медиа возможны уточняющие требования, специфичные для конкретной задачи.

Разработанные задания должны быть размещены по адресу <http://xxxxxx-m4.domain.ru/y/>, где xxxxxx - логин конкурсанта (указан на индивидуальной карточке), у — номер задачи. Медиафайлы задач должны быть скопированы в соответствующую папку задачи. Также в медиа возможны уточняющие требования, специфичные для конкретной задачи

2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ²

2.1. Личный инструмент конкурсанта

Конкурсанту разрешается использовать собственные:

- клавиатуру на любом языке. Если конкурсант пользуется своей клавиатурой, и она выходит из строя, организатор предоставляет ему замену.;
- языковые файлы для клавиатуры;
- мышь;
- графический планшет;
- наушники;
- аудиофайлы с музыкальными композициями (не более 30 файлов в формате mp3). Файлы предоставляются на флеш-носителях в день С-1 техническому эксперту на проверку.

Все оборудование не должно содержать встроенной памяти.

2.2. Материалы, оборудование и инструменты, запрещенные на площадке

Всё оборудование, принесенное конкурсантами, может быть проверено экспертами на наличие внутренних запоминающих устройств. В случае обнаружения материалы будут изыматься.

Экспертам допускается использовать персональные компьютеры, но в специальной зоне. В помещениях для проведения оценки использование любых

² Указываются особенности компетенции, которые относятся ко всем возрастным категориям и чемпионатным линейкам без исключения.

электронных устройств запрещено, кроме специально организованных для оценки.

Также запрещено приносить:

- дополнительные программы и библиотеки, не предусмотренные инфраструктурным листом;
- мобильные телефоны;
- фото/видео устройства;
- карты памяти и другие носители информации;
- внутренние устройства памяти в собственном оборудовании.

3. ПРИЛОЖЕНИЯ

Приложение №1 Инструкция по заполнению матрицы конкурсного задания

Приложение №2 Матрица конкурсного задания

Приложение №3 Инструкция по охране труда по компетенции «Веб-технологии».

Приложение № 4 Медиа файлы