



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 ( <a href="https://disk.yandex.ru/i/XJfQr1jBmZ2rew">https://disk.yandex.ru/i/XJfQr1jBmZ2rew</a> ) .....	10
1.5.2. Структура модулей конкурсного задания (инвариант/вариатив) <b>Ошибка!</b>	3
2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ .....	44
2.1. Личный инструмент конкурсанта .....	47
2.2. Материалы, оборудование и инструменты, запрещенные на площадке	48
3. ПРИЛОЖЕНИЯ.....	49

## 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	<b>Тестирование информационных ресурсов</b>	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.); – английский язык в рамках чтения и понимания	

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

	<p>разработки кода;</p> <ul style="list-style-type: none"> <li>– формировать тестовые наборы, применять инструменты автоматического тестирования;</li> <li>– производить отладку кода программ и находить ошибки;</li> <li>– оптимально использовать компьютерное оборудование и программное обеспечение для повышения эффективности своей работы;</li> <li>– использовать менеджеры пакетов при разработке проекта;</li> <li>- использовать систему контроля версий.</li> </ul>	
3	<p><b>Разработка интерфейса пользователя</b></p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> <li>– структуру и общепринятые элементы веб-страниц различных видов и назначений;</li> <li>– основные принципы организации контента веб-приложения;</li> <li>– основные правила выбора цвета, работы с типографикой и композицией;</li> <li>– принципы и методы создания и адаптации графики для использования ее на веб-сайтах;</li> <li>– ограничения, которые накладывают мобильные устройства и разрешения экранов при использовании их для просмотра веб-сайтов;</li> <li>– принципы построения эстетичного и креативного дизайна;</li> <li>– методы обеспечения доступа к страницам веб-сайтов аудитории с ограниченными возможностями;</li> <li>– World Wide Web Consortium (W3C) стандарты HTML и CSS;</li> <li>– методы верстки веб-сайтов и их стандартную структуру;</li> <li>– Web accessibility initiative (WAI) стандарт доступности активных Интернет-приложений для людей с ограниченными возможностями;</li> <li>– основные принципы применения соответствующих CSS правил и селекторов для получения ожидаемого результата;</li> <li>– лучшие практики для Search Engine Optimization (SEO) и интернет-маркетинга;</li> </ul> <p>Специалист должен уметь:</p> <ul style="list-style-type: none"> <li>– создавать, использовать и оптимизировать изображения для веб-сайтов;</li> </ul>	26

	<ul style="list-style-type: none"> <li>– выбирать дизайнерское решение, которое будет наиболее подходящим для целевого рынка;</li> <li>– принимать во внимание влияние каждого элемента, который добавляется в проект во время разработки дизайна;</li> <li>– использовать все требуемые элементы при разработке дизайна;</li> <li>– создавать «отзывчивый» дизайн, который будет отображаться корректно на различных устройствах и при разных разрешениях;</li> <li>– создавать html-страницы сайта на основе предоставленных графических макетов их дизайна;</li> <li>– корректно использовать CSS для обеспечения единого дизайна в разных браузерах;</li> <li>– создавать адаптивные веб-страницы, которые способны оставаться функциональными на различных устройствах при разных разрешениях;</li> <li>– создавать веб-сайты, полностью соответствующие текущим стандартам W3C (<a href="http://www.w3.org">http://www.w3.org</a>);</li> <li>– создавать и модифицировать веб-интерфейсы с учетом принципов Search Engine Optimization;</li> <li>– использовать препроцессоры.</li> </ul>	
4	<p><b>Разработка на стороне клиента</b></p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> <li>– основные принципы паттерновой разработки веб-приложений;</li> <li>– ECMAScript (JavaScript);</li> <li>– принципы, особенности и способы использования открытых фреймворков;</li> <li>– принципы разработка кода с использованием открытых библиотек;</li> <li>– различные интерфейсы взаимодействия с объектами браузера</li> </ul> <p>Специалист должен уметь:</p> <ul style="list-style-type: none"> <li>– создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;</li> <li>– манипулировать элементами страницы веб-приложения;</li> <li>– разрабатывать анимацию для повышения доступности и визуальной привлекательности веб-приложения;</li> </ul>	25

	<ul style="list-style-type: none"> <li>– применять открытые библиотеки и фреймворки;</li> <li>– тестировать веб-приложение.</li> </ul>	
	<p><b>Разработка веб приложения на стороне сервера</b></p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> <li>– процедурные и объектно-ориентированные языки PHP, Python, Node.js;</li> <li>– основные принципы и правила использования открытых библиотек и фреймворков;</li> <li>– распространенные модели организации и хранения данных;</li> <li>– основные принципы создания баз данных;</li> <li>– основные принципы обмена данными между клиентом и сервером;</li> <li>– методы работы с протоколами SSH/(s)FTP при подключении к серверам;</li> <li>– способы разработки программного кода в соответствии с паттернами проектирования;</li> <li>– основные принципы обеспечения безопасности веб-приложения.</li> </ul>	
5	<p>Специалист должен уметь:</p> <ul style="list-style-type: none"> <li>– разрабатывать процедурный и объектно-ориентированный программный код;</li> <li>– разрабатывать веб-сервисы с применением PHP, Python, Node.js в соответствии с техническим заданием;</li> <li>– создавать библиотеки и модули для выполнения повторяющихся задач;</li> <li>– разрабатывать веб-приложения с доступом к различным базам данных;</li> <li>– создавать SQL (Structured Query Language) запросы и конструкции;</li> <li>– обеспечивать безопасность (устойчивость веб-приложения к атакам и взломам);</li> <li>– интегрировать существующий и создавать новый программный код с API (Application Programming Interfaces);</li> <li>– использовать открытые библиотеки и фреймворки;</li> </ul>	25
6	<p><b>Разработка информационных ресурсов с использованием готовых решений</b></p> <p>Специалист должен знать и понимать:</p> <ul style="list-style-type: none"> <li>– преимущества и ограничения системы</li> </ul>	10



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

### 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 ТРЕБОВАНИЙ КОМПЕТЕНЦИИ
Разделы ТРЕБОВАНИЙ КОМПЕТЕНЦИИ		А	Б	В	Г	
	1	4	1,5	1,5		7
	2	4	1,5	1,5		7
	3	20	2		4	26
	4		19		6	25
	5			21	4	25
	6				10	10
Итого баллов за критерий/модуль		28	24	24	24	<b>100</b>

#### 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
<b>А</b>	<b>Разработка интерфейса пользователя</b>	Проверка результата выполнения модуля после его завершения
<b>Б</b>	<b>Разработка Веб-приложения на стороне клиента</b>	Проверка результата выполнения модуля после его завершения
<b>В</b>	<b>Разработка Веб-приложения на стороне сервера</b>	Проверка результата выполнения модуля после его завершения
<b>Г</b>	<b>Спидтаски</b>	Проверка результата выполнения модуля после его завершения

#### 1.5. КОНКУРСНОЕ ЗАДАНИЕ

Общая продолжительность Конкурсного задания<sup>1</sup>: 13 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

<sup>1</sup> 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:** 4,5 часов

#### **Задание:**

К Вам обратилась компания «АвтоДоктор» - сеть авторемонтных мастерских. Руководством компании было принято решение об использовании в работе своих мастерских современных информационных технологий. Так как сеть мастерских достаточно развита в большом количестве городов, выбор был сделан в пользу веб - технологий.

Вам необходимо разработать дизайн и сверстать страницы веб-сайта, который будет привлекать потенциальных клиентов.

Потенциальные пользователи сайта предпочитают использовать мобильные устройства для доступа к интернету и поиска информации. Поэтому страницы должны корректно отображаться как на мобильных устройствах, так и на десктопных.

Страницы должны быть адаптивны. При переходе от десктопной версии главной страницы (ширина экрана 1920px) к мобильной версии (ширина экрана 960px) должно быть реализовано не менее двух структурных изменений контента (например, скрываются/добавляются, увеличиваются/уменьшаются элементы и др.).

В дизайне сверстанных страниц следует использовать и применять элементы, концепции и приемы, позволяющие привлечь внимание, создать положительный образ ресурса, реализовать современные концепции дизайна веб-страниц.

Также при верстке страниц вам следует учитывать существующие практики UI/UX, чтобы использование веб-ресурса было максимально удобным и продуктивным, а также творчески применять предоставленные или созданные медиаданные. Текстовый и графический контент должны быть согласованы между собой.

## **ОПИСАНИЕ ПРОЕКТА И ЗАДАЧ**

Ваша задача – разработать дизайн и сверстать следующие страницы веб-сайта:

1. Главная - index.html; index-dark.html (темная тема)
2. Страница с отзывами и рекомендациями - feedback.html
3. Страница с услугами ремонта - repair.html
4. Страница «О нас» - about.html
5. Страница «Контакты» - contacts.html
6. Личный кабинет – lk.html
7. Страница не найдена - 404.html

Все страницы должны быть сверстаны в темной теме. Однако для главной страницы также должен быть разработан дизайн и сверстана версия со светлой темой.

Сверстаный веб-сайт должен быть доступен по адресу <http://xxxxxx-m1.wsr.ru>, адрес указан в личном кабинете конкурсанта соревнований.

### **1. Главная страница – Landing Page**

Главная страница должна содержать следующие блоки (вы также можете добавить дополнительные блоки/секции, если считаете, что это усилит работу) – порядок блоков/секций может быть произвольным:

#### **1. Шапка сайта.**

##### **1.1. Логотип.**

Вам необходимо создать логотип, с учетом следующих пожеланий заказчика:

Логотип должен быть уникальным и легко узнаваемым. Он должен быть достаточно простым, чтобы визуально ассоциироваться с тематикой сайта. Логотип должен быть согласован с общей концепцией сайта и выполнен в высоком качестве. Логотип должен быть представлен на сайте в виде векторного или растрового графического файла.

##### **1.2. Меню навигации**

1.2.1 Сервис – представляет собой выпадающий список, элементы списка:

- Выхлопная система
- Двигатель
- Диагностика автомобиля
- Замена жидкостей

Кондиционеры и отопление  
Кузовной ремонт и покраска  
Подвеска  
Рулевое управление

На данный момент разработки нет необходимости, чтобы элементы списка вели на отдельные страницы сайта

1.2.2 Рекомендации – ссылка на отдельную страницу (ссылка на страницу с отзывами).

1.2.3 О нас – ссылка на отдельную страницу (ссылка на страницу «О нас»).

1.2.4 Магазин – по желанию заказчика, это ссылка-заглушка, реализация магазина планируется в дальнейшем.

1.2.5 Контакты – ссылка на отдельную страницу (ссылка на страницу «Контакты»).

1.2.6 Поиск

1.2.7 Элемент переключения между темной и светлой темами

1.2.8 Личный кабинет

1.3. Форма для быстрого входа в личный кабинет.

Форма входа должна предоставлять поле для ввода телефона и пароля для аутентификации, кнопка/ссылка для входа.

Дизайн формы заказчик оставляет на Ваше усмотрение, главное, чтобы форма корректно вписалась в шапку сайта.

2. Баннер.

Баннер представляет собой изображение, также на баннер содержать необходимую информацию:

2.1. Слоган компании - «Ваш автомобиль — наша забота»

2.2. Акция компании, актуальная на текущий момент времени (должна быть реализована возможность менять акции) – «Бесплатная замена масла с 15 марта. Только при наличии цифры "5" на вашем номерном знаке».

2.3. Кнопка/ссылка «Записаться» - ссылка на блок записи на ремонт.

3. Информационная секция/блок

Секция содержит разделы:

3.1. Наши преимущества:

бесплатная диагностика и проверка тормозов

сертифицированные услуги по ремонту

оценка ремонта по справедливой цене

3.2. Часы работы:

Будни: 05:00 – 22:00  
Суббота: 10:00 – 20:00  
Воскресенье: 10:00 – 18:00

3.3. Связаться с нами:

(415) 347-3160

Мобильный: 02 562-958

Факс: 02 562-95

#### 4. Секция/блок «Запись на сервис»

Секция должна содержать следующие поля:

4.1 Марка

4.2 Модель

4.3 Город

4.4 Имя

4.5 Телефон

4.6 Желаемая дата

4.7 Поле для указания неисправности автомобиля

4.8 Поле – согласие на обработку персональных данных

4.9 Кнопка «Записаться»

4.10 Информационное сообщение: «После подачи заявки с Вами свяжется менеджер для уточнения деталей. В течении нескольких минут Вам поступит СМС-сообщение с временным паролем от личного кабинета. Установить пароль Вы сможете в личном кабинете».

#### 5. Секция/блок «Услуги по ремонту транспортных средств»

Секция содержит:

5.1. Текстовое сообщение «Мы являемся одной из ведущих автомастерских, обслуживающих клиентов по всему региону. Все ремонтные услуги выполняются высококвалифицированными механиками»

5.2. Изображение

5.3. Иконки (не менее 4 штук) с подписями, означающие возможные варианты услуг по ремонту

5.4. Кнопка/ссылка «Все услуги», ведущая на отдельную страницу (ссылка на страницу с услугами ремонта).

#### 6. Секция/блок «О нас»

Секция содержит:

6.1. Текстовое сообщение «Мы предлагаем полный спектр услуг владельцам транспортных средств. Наши профессионалы знают, как

справиться с широким спектром ремонта авто. Наши механики позаботятся о том, чтобы Ваш автомобиль работал наилучшим образом еще до того, как покинет нашу ремонтную мастерскую. Мы знаем, как ухаживать за Вашим автомобилем» с заголовком «Мы работаем с 2002 года».

6.2. Кнопка/ссылка «Все услуги», ведущая на отдельную страницу (ссылка на страницу с услугами ремонта).

6.3. Слайдер, состоящий из отзывов клиентов. Слайдер должен содержать следующую информацию:

Фото клиента

Имя клиента

Город клиента

Отзыв клиента

Элементы управления слайдером

Необходимо представить в слайдере минимум 3 отзыва

## 7. Секция/блок «Популярные товары»

Секция содержит карточки товаров магазина, реализация которого планируется позже. Каждая карточка должна содержать следующие элементы:

7.1 Изображение

7.2 Название

7.3 Цена

7.4 Кнопка/ссылка «В корзину»

В секции должно отображаться не менее 6-х карточек

8. Секция/блок с картой региона/города с указанием расположения сети мастерских (допустимо использовать изображение-заглушку из медиафайлов)

## 9. Подвал сайта

Подвал состоит из следующих блоков:

9.1 блок «О нас», содержащий сообщение «Автосервис предоставляет надежные услуги по техническому обслуживанию автомобилей всех размеров и производителей с 2002 года» и иконки соц.сетей

9.2 блок «Навигация», содержащий ссылки «о нас», «часто задаваемые вопросы», «контакты», «цены», «блог», «конфиденциальность», «ресурсы», «карта сайта»

9.3 блок «Свяжитесь с нами», содержащий электронную почту info@doc-auto.ru и телефон (415) 347-3160



9.4 блок «Подписка на рассылку», содержащий поле для ввода электронной почты и кнопки «Подписаться»

9.5 Копирайт – 2025 АвтоДоктор.

## **2. Страница с отзывами и рекомендациями**

Данная страница содержит отзывы и рекомендации клиентов, пользующимися услугами компании. Каждый блок отзыва содержит следующую информацию:

Фото клиента

Имя клиента

Город клиента

Отзыв клиента

Заказчик желает, чтобы при просмотре страницы на десктопной версии отзывы выводились в две колонки. Необходимое минимальное количество отзывов на странице – 6 отзывов.

## **3. Страница с услугами ремонта**

Данная страница содержит услуги ремонта, оказываемые автомастерской. Карточка каждой услуги содержит следующую информацию:

Изображение

Заголовок

Описание

Кнопка «Узнать больше»

Заказчик желает, чтобы при просмотре страницы на десктопной версии карточки услуг выводились в четыре колонки. Необходимое минимальное количество услуг на странице – 6 услуг.

## **4. Страница «О нас»**

Данная страница должна содержать следующие блоки:

### **1. Раздел «Наша короткая история»**

Раздел должен содержать изображение и краткую историю компании: «Когда 24 года назад 3 механика собрались вместе, они решили, что их любовь к автомобилям может открыть им новую сферу деятельности! С тех пор Авторемонтная мастерская отремонтировала тысячи новых и ретро-автомобилей...

За этот длительный период мы увеличили нашу команду всего с 3 механиков до более чем 200 человек. Кроме того, теперь у нас есть сильная

база постоянных клиентов, которым мы всегда выражаем признательность дополнительными бонусами и скидками на будущие услуги!»

## 2. Раздел «Чем мы отличаемся от других»

Данный раздел содержит 5 отличительных черты компании. Отличительная черта представлена иконкой, названием и описанием. Названия и описания Вам предоставляет заказчик:

### 1) Клиентоориентированный сервис

Мы ценим оказанное нам доверие, и наши постоянные клиенты, которые возвращаются к нам снова и снова, всегда могут рассчитывать на некоторую благодарность от нас, например, скидку на будущие услуги или подарок.

### 2) Опытная команда

Когда дело доходит до ремонта или технического осмотра вашей автомеханики или электроники, вся наша команда, вместе взятая, обладает почти столетним практическим опытом.

### 3) Доступные цены

Мы стремимся сделать наши услуги максимально доступными. Именно поэтому мы сокращаем все основные коммерческие расходы.

### 4) Высококачественные автомобильные запчасти

Все автомобильные запчасти, имеющиеся у нас на складе, были приобретены у самых надежных дилеров и производителей.

## 3. Информационный баннер.

Баннер представляет собой изображение, с указанным на нем текстом акции, и кнопкой «Записаться» - ссылка на отдельную страницу (ссылка на главную страницу).

Текст акции

**«ПОЛУЧИТЕ СКИДКУ В размере 10% НА УКАЗАННЫЕ НИЖЕ УСЛУГИ!**

Плановое техническое обслуживание от производителя

Настройка и ремонт двигателя

Замена жидкостей»

## 5. Страница «Контакты»

Данная страница содержит форму обратной связи «Связаться с нами». Форма содержит поле для ввода имени, поле для ввода email, поле для ввода телефона, поле для ввода текстового сообщения, кнопка «Отправить».

## 6. Страница «Личный кабинет»

На данной странице должны быть расположены следующие блоки:

1. Информационное сообщение «Для участия в наших акциях и спецпредложениях, пожалуйста, заполните всю информацию в личном кабинете»

2. Блок с информацией о клиенте. Должен содержать следующие элементы:

2.1 Фамилия

2.2 Имя

2.3 Отчество

2.4 Фото

2.5 Дата рождения

2.6 Город проживания

2.7 Телефон

2.8 Электронная почта

2.9 Установить пароль

2.10 Повторить пароль

2.11 Согласие на обработку персональных данных

2.12 Кнопка «Обновить данные пользователя»

3. Блок с информацией о транспортных средствах клиента. Должен содержать следующую информацию:

3.1 Марка транспортного средства

3.2 Модель транспортного средства

3.3 VIN номер ТС

3.4 Регистрационный номер ТС

3.5 Год выпуска ТС

3.6 Фотография ТС

3.7 Кнопка «Добавить автомобиль»

4. Блок с историей ремонта автомобиля клиента. Должен содержать следующую информацию:

4.1 Марка транспортного средства

4.2 Модель транспортного средства

4.3 VIN номер ТС

4.4 Регистрационный номер ТС

4.5 Год выпуска ТС

4.6 Оказанная услуга по ремонту

4.7 Дата проведения ремонта

4.8 Стоимость ремонта

**7. Страница не найдена - 404.html**

Отображается сообщение, что такая страница не существует.

Оформление страницы соответствует дизайн-решениям других страниц.

## ИНСТРУКЦИЯ ДЛЯ КОНКУРСАНТА

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

Сохраните сверстаные страницы со следующими именами:

1. Главная - `index.html`; `index-dark.html` (темная тема)
2. Страница с отзывами и рекомендациями – `feedback.html`
3. Страница с услугами ремонта - `repair.html`
4. Страница «О нас» - `about.html`
5. Страница «Контакты» - `contacts.html`
6. Личный кабинет – `lk.html`
7. Страница не найдена - `404.html`

Все страницы указанные выше должны быть доступны к просмотру по соответствующим адресам: <http://xxxxxx-m1.wsr.ru/index.html> , <http://xxxxxx-m1.wsr.ru/feedback.html> и т.д.

Проверяются только работы, загруженные на сервер! Страницы расположенные в других местах или с ошибками в названии проверяться не будут!

Ваш HTML/CSS должен быть валидным.

Оценка будет производиться в браузере Google Chrome.

Использование любых фреймворков и библиотек (bootstrap, например) запрещено.

## **Модуль Б, В. Программирование на стороне сервера (2,5 часа) и на стороне клиента (2,5 часа)**

**Технологии этого модуля:** PHP, Python, backend-фреймворки, JS, frontend-фреймворки

**Время на выполнение:** 5 часов

К Вам обратилась сеть автомастерских АвтоДоктор. У данной сети автомастерских возникли трудности с поддержкой их внутренней информационной системой.

**Для лучшего масштабирования информационной системы необходимо реализовать отдельно серверную часть в виде API и клиентскую часть в виде одностраничного приложения (SPA).**

В дальнейшем планируется развитие проекта, поэтому использование фреймворков будет плюсом.

От вас требуется реализовать клиентскую и серверную части приложения, основываясь на ресурсах, предоставляемой компанией. Основным требованием к информационной системе является реализация учета и управления заказами, менеджмент персонала, также формирование отчетной документации. Пользователями системы будут являться сотрудники автомастерской.

Заказчик предоставляет вам верстку, дополнительно, заказчик желал бы улучшения дизайна для большего удобства пользования приложением. Также Вам предоставляется набор данных в виде дампа базы данных.

**ВНИМАНИЕ! Проверяться будут только работы, загруженные на сервер!**

**ВНИМАНИЕ! В медиафайлах вам предоставляется sql дамп с готовой базой данных. База данных уже содержит набор данных, которые заказчик не рекомендует изменять.**

**ВНИМАНИЕ! Для упрощения последующей автоматической проверки не хешируйте пароли пользователей в БД!**

**Описание проекта и задач**

**Общее описание проекта**

Все пользователи системы подразделяются на три группы:

1. Администратор;
2. Начальник смены;
3. Механик.

Пользователи получают доступ к функциям ИС только после успешной авторизации.

Ваше SPA должно состоять из следующих экранов:

1. Экран авторизации.
2. Экран просмотра сотрудников (доступен только администратору).
3. Экран добавления сотрудника (доступен только администратору).
4. Экран управления сменами (доступен только администратору). Должен включать в себя следующий функционал:
  - a. создание смены;
  - b. открытие смены;
  - c. закрытие смены;
  - d. добавление сотрудника на смену;
  - e. просмотр наряд-заказов конкретной смены.
5. Экран управления наряд-заказами начальником смены. Должен включать в себя следующий функционал:
  - a. создание наряд-заказа;
  - b. просмотр наряд-заказов;
  - c. просмотр конкретного наряд-заказа;
  - d. изменение статуса наряд-заказа (смена со статуса «принят» на «отменен», а также со статуса «готов» на статус «оплачен»);
  - e. добавление позиций в наряд-заказ;
  - f. удаление позиций из наряд-заказа.
6. Экран управления наряд-заказами механиком. Должен включать в себя следующий функционал:
  - a. просмотр наряд-заказов текущей смены;

в. изменение статуса наряд-заказа (смена со статуса «принят» на «в работе», а также со статуса «в работе» на статус «готов»).

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

Разработанное приложение должно быть доступно по адресу `http://xxxxxx-m2.wsr.ru/`, где `xxxxxx` - логин участника (указан на индивидуальной карточке).

### **Описание API**

В примерах будет использоваться переменная `{{host}}` которая обозначает адрес `http://xxxxxx-m3.wsr.ru/`, где `xxxxxx` - логин участника.

### **Общие требования к API**

Все функции, кроме аутентификации доступны только авторизованным пользователям. Идентификацию пользователя организуйте посредством Bearer Token.

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

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": "Authorization failed"
  }
}
```

При попытке доступа авторизованным пользователем к функциям недоступным для своей группы во всех запросах необходимо возвращать ответ следующего вида:

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": "Access denied"
  }
}
```

В случае ошибок связанных с валидацией данных во всех запросах необходимо возвращать следующее тело ответа:

```
{
  "error": {
    "code": <code>,
    "message": <message>,
    "errors": {
      <key>: [ <error message>]
    }
  }
}
```

Обратите внимание, что вместо `<code>` и `<message>` необходимо указывать соответствующее значение, определенное в описании ответа на соответствующий запрос. В свойстве `error.errors` необходимо перечислить те свойства, которые не прошли валидацию, а в их значениях указать массив с ошибками валидации.

Например, если отправить пустой запрос на сервер, где проверяется следующая валидация:

- `phone` – обязательно поле



- password – обязательное поле

то тело ответа будет следующим:

```
{
  "error": {
    "code": 422,
    "message": "Validation error",
    "errors": {
      phone: [ "field phone can not be blank" ],
      password: [ "field password can not be blank" ]
    }
  }
}
```

Примите во внимание, что code и message могут быть определены иначе, если в запросе указано иное. В значениях свойств errors вы можете использовать любые сообщения об ошибках (если не указана конкретная ошибка), но они должны описывать возникшую проблему.

## Специфические требования к API

### Аутентификация

Запрос для аутентификации пользователя в системе. При отправке запроса необходимо передать объект с логином и паролем. Если клиент отправил корректные данные, то необходимо вернуть сгенерированный токен, а иначе сообщение об ошибке.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/login</p> <p><b>Method:</b> POST</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p> <p><b>Body:</b></p>	<p style="text-align: right;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "user_token": &lt;сгенерированный token&gt;   } }</pre>

<pre>{   "login": "admin",   "password": "admin" }</pre>	<pre>} }</pre> <hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p> <hr/> <p style="text-align: center;"><u>Неправильные логин или пароль</u></p> <p><b>Status:</b> 401</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 401,     "message": "Authentication failed"   } }</pre>
--	--

**Выход**

Запрос предназначен для очистки значения токена пользователя.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/logout</p> <p><b>Method:</b> GET</p>	<p style="text-align: center;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "message": "logout"   } }</pre>

## Функционал администратора

### Просмотр списка всех сотрудников

Запрос предназначен для получения списка всех сотрудников автомастерской.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/user</p> <p><b>Method:</b> GET</p>	<p style="text-align: right;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": [     {       "id": 1,       "name": "Bell",       "login": "admin",       "status": "working",       "group": "Администратор"     },     {       "id": 2,       "name": "Shana",       "login": "supervisor",       "status": "working",       "group": "Начальник смены"     },     {       "id": 3,       "name": "Estrella",       "login": "worker",       "status": "working",       "group": "Механик"     }   ] }</pre>

	<pre>] }</pre>
--	----------------

### Добавление сотрудника

Запрос для регистрации нового пользователя в системе. При отправке запроса необходимо передать FormData со следующими полями:

- name – обязательное поле, строка;
- surname – не обязательное поле, строка;
- patronymic – не обязательное поле, строка;
- login – обязательное и уникальное поле, строка;
- password – обязательное поле, строка;
- photo\_file – файл с фото нового сотрудника (загружается в папку photos), не обязательное поле, файл jpeg, png;
- role\_id – обязательное поле, число, роль с данным идентификатором должна быть.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/user</p> <p><b>Method:</b> POST</p> <p><b>Headers</b></p> <p>- Content-Type: multipart/form-data</p> <p><b>FormData:</b></p> <pre>"name": "Ivan", "login": "Ivanov", "password": "paSSword" "role_id": 1</pre>	<p style="text-align: right;"><u>Успех</u></p> <p><b>Status:</b> 201</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "id": 6,     "status": "created"   } }</pre> <hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p>

## Создание смены

Запрос предназначен для создания новой смены. При отправке запроса необходимо отправить объект со следующими полями:

- start – обязательное поле, дата и время начала смены, не должно быть раньше, чем текущая дата и время, формат Y-m-d H:i;
- end – обязательное поле, дата и время начала смены, не должно быть раньше, чем дата и время начала смены, формат Y-m-d H:i.

Request	Response
<b>URL:</b> {{host}}/api-auto/work-shift <b>Method:</b> POST  <b>Headers</b> - Content-Type: application/json  <b>Body:</b> { "start": "2021-03-26 08:00", "end": "2021-03-26 17:00" }	<p style="text-align: center;"><u>Успех</u></p> <b>Status:</b> 201 <b>Content-Type:</b> application/json <b>Body:</b> { "start": "2021-03-26 08:00", "end": "2021-03-26 17:00" }
	<hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p>

## Открытие смены

Запрос открывает смену. Открыть смену можно, только если все смены закрыты.

Request	Response
<b>URL:</b> {{host}}/api-auto/work-shift/{{id}}/open <b>Method:</b> GET  <b>Headers</b> - Content-Type: application/json	<p style="text-align: center;"><u>Успех</u></p> <b>Status:</b> 200 <b>Content-Type:</b> application/json <b>Body:</b> {

	<pre>"data": {   "id": 18,   "start": "2021-03-28 08:00:00",   "end": "2021-03-29 17:00:00",   "active": true } }</pre> <hr/> <p style="text-align: center;"><u>Попытка открыть вторую смену</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": "Denied. There are open shifts!"   } }</pre>
--	--

### Заккрытие смены

Запрос закрывает смену. Закрывать уже закрытую смену нельзя.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/work-shift/{{id}}/close</p> <p><b>Method:</b> GET</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p>	<p style="text-align: center;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "id": 18,     "start": "2021-03-28 08:00:00",     "end": "2021-03-29 17:00:00",     "active": false   } }</pre>

	<pre> } } </pre> <hr/> <p style="text-align: center;"><u>Попытка закрыть закрытую смену</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre> {   "error": {     "code": 403,     "message": " Denied. The shift is already closed!"   } } </pre>
--	--

### Добавление сотрудников на смену

Запрос позволяет добавить работников на смену. Два раза на одну и ту же смену сотрудника добавить нельзя.

При отправке запроса необходимо отправить объект со следующими полями:

- **user\_id** - обязательное, пользователь с таким идентификатором должен быть среди работников и не быть уволенным.

Request	Response
<p><b>URL:</b> {{host}}/api-auto/work-shift/{{id}}/user</p> <p><b>Method:</b> POST</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p> <p><b>Body:</b></p> <pre> {   "user_id":4 </pre>	<p style="text-align: center;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre> {   "data": {     "id_user": 1,     "status": "added"   } } </pre>

}	<hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p> <hr/> <p style="text-align: center;"><u>Попытка два раза добавить работника</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": " Denied. The worker is already on shift!"   } }</pre>
---	--

### Просмотр заказов за конкретную смену

Запрос предназначен для просмотра наряд-заказов конкретной смены.

Request	Response
URL: {{host}}/api-auto/work-shift/{{id}}/order Method: GET  Headers - Content-Type: application/json	<u>Успех</u>  Status: 200 Content-Type: application/json Body: <pre>{   "data": {     "id": 1,     "start": "2021-03-30 08:00:00",     "end": "2021-03-31 17:00:00",     "active": 1,     "orders": [       {         "id": 1,         "car_box": "Бокс №1",         "shift_workers": "Jaiden",         "create_at": "2021-03- 03T08:34:43.000000Z",         "status": "В работе",         "price": 2934       },       {         "id": 2,         "car_box": " Бокс №2",</pre>



	<pre> "shift_workers": "Trudie", "create_at": "2021-03- 11T08:34:57.000000Z", "status": "В работе", "price": 0 } ], "amount_for_all": 2934 } } </pre>
--	---

### Функционал начальника смены

#### Создание наряд-заказа

Запрос предназначен для создания наряд-заказа начальником смены. Начальник смены может создать наряд-заказ только на активную смену и только если он в ней работает. При отправке запроса необходимо отправить объект со следующими полями:

- **work\_shift\_id** - идентификатор смены, обязательное, смена должна существовать и быть активной, пользователь должен работать на смене;
- **car\_box\_id** - идентификатор автомобильного бокса, должен существовать;

Request	Response
<p><b>URL:</b> {{host}}/api-auto/order</p> <p><b>Method:</b> POST</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p> <p><b>Body:</b></p> <pre>{   "work_shift_id":1,   "car_box_id":2 }</pre>	<p style="text-align: right;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "id": 10,     "car_box": "Бокс №2",     "shift_workers": "Trudie",     "create_at": "2021-03-27T04:03:37.000000Z",     "status": "Принят",     "price": 0   } }</pre>

	<pre>} </pre> <hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p> <hr/> <p style="text-align: center;"><u>Попытка добавить заказ в неактивную смену</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": " Denied. The shift must be active!"   } }</pre> <hr/> <p style="text-align: center;"><u>Попытка добавить заказ начальником смены не этой смены</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": " Denied. You don't work this shift!"   } }</pre>
--	--

### **Просмотр принятого наряд-заказа**

Запрос предназначен для просмотра деталей конкретного наряд-заказа, принятого начальником смены. Если начальник смены не принимал данный наряд-заказ, запрос не должен быть разрешен.

Request	Response
---------	----------

<p>URL: {{host}}/api-auto/order/{{id}}</p> <p>Method: GET</p> <p>Headers</p> <p>- Content-Type: application/json</p>	<p style="text-align: right;"><u>Успех</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>{   "data": {     "id": 1,     "car-box": "Бокс №1",     "shift_workers": "Allan",     "create_at": null,     "status": "Принят",     "positions": [       {         "id": 1,         "count": 5,         "position": "Aut sit ut et reprehenderit sed cumque.",         "price": 8135       },       {         "id": 2,         "count": 1,         "position": "Ut similique dolorum eos culpa officiis.",         "price": 1993       }     ],     "price_all": 10128   } }</pre> <hr/> <p><u>Попытка доступа к не своему наряд-заказу</u></p> <p>Status: 403</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre>{   "error": {     "code": 403,     "message": " Denied. You did not accept this order!"   } }</pre>
--	---

### **Просмотр всех принятых наряд-заказов за смену**

Запрос предназначен для получения всех заказов, выданных всеми начальниками смен за смену. Если начальник смены не работал на смене, запрос не должен быть разрешен.

Request	Response
URL: {{host}}/api-auto/work-shift/{{id}}/order Method: GET  Headers - Content-Type: application/json	<p style="text-align: center;"><u>Успех</u></p> <p style="text-align: center;">Формат как у соответствующей функции у администратора</p> <hr/> <p style="text-align: center;"><u>Попытка получить заказы не своей смены</u></p> Status: 403 Content-Type: application/json Body: <pre>{   "error": {     "code": 403,     "message": " Denied. You did not accept this order!"   } }</pre>

### **Изменение статуса наряд-заказа**

Запрос предназначен для смены статуса наряд-заказа активной смены, который он принимал. Для начальника смены доступна смена со статуса «принят» на «отменен», а также со статуса «готов» на статус «оплачен».

Request	Response
<b>URL:</b> {{host}}/api-auto/order/{{id}}/change-status <b>Method:</b> PATCH  <b>Headers</b> - Content-Type: application/json  <b>Body:</b> <pre>{   "status": "paid-up" }</pre>	<p style="text-align: center;"><u>Успех</u></p> <b>Status:</b> 200 <b>Content-Type:</b> application/json <b>Body:</b> <pre>{   "data": {     "id": 1,     "status": "paid-up"   } }</pre> <hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p>

**Формат ответа из общих требований**

---

Попытка изменить на неразрешенный статус

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": "Forbidden! Can't change
existing order status"
  }
}
```

---

Попытка изменить не свой наряд-заказ

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": " Denied! You did not
accept this order!"
  }
}
```

---

Попытка изменить наряд-заказ закрытой смены

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
```

	<pre> "code": 403, "message": "You cannot change the order status of a closed shift!" } } </pre>
--	--

### Добавление позиций из списка услуг в наряд-заказ

Запрос предназначен для добавления позиции услуг в наряд-заказ. Начальник смены может добавить позиции только в свой наряд-заказ текущей смены, если статус наряд-заказа “принят” или “готовится”.

При отправке запроса необходимо отправить объект со следующими полями:

- **service\_id** - идентификатор позиции в списке услуг, обязательное, позиция должна существовать;

Request	Response
<p><b>URL:</b> {{host}}/api-auto/order/{{id}}/position</p> <p><b>Method:</b> POST</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p> <p><b>Body:</b></p> <pre> {   "service_id":3 } </pre>	<p style="text-align: center;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre> {   "data": {     "id": 8,     "car_box": "Бокс №7",     "shift_workers": "Palma",     "create_at": "2021-03-28T02:52:57.000000Z",     "status": "Принят",     "price": 19530,     "positions": [       {         "id": 3,         "position": "Ut similique dolorum eos culpa officis.", </pre>

```
"price": 3986
}
],
"price_all": 3986
}
}
```

---

Ошибки валидации полей

**Формат ответа из общих требований**

---

Попытка добавить не в свой заказ

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": " Denied! You did not accept this
order!"
  }
}
```

---

Попытка добавить в неактивную смену

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": " Denied! You cannot change the
order status of a closed shift!"
  }
}
```

---

	<p><u>Попытка добавить в заказ с запрещенными статусами</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": " Denied! Cannot be added to an order with this status "   } }</pre>
--	---

### Удаление позиций из наряд-заказа

Запрос предназначен для удаления позиции из наряд-заказа. Начальник смены может удалять позиции только в свой наряд-заказ текущей смены, если статус наряд-заказа «принят».

Request	Response
<p><b>URL:</b> {{host}}/api-auto/order/{{id}}/position/{{id}}</p> <p><b>Method:</b> DELETE</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p>	<p><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "data": {     "id": 8,     "car_box": "Бокс №7",     "shift_workers": "Palma",     "create_at": "2021-03- 28T02:52:57.000000Z",     "status": "Принят",     "price": 3986,     "positions": [       {</pre>



```
"id": 3,
  "position": "Ut similique dolorum
eos culpa officiis.",
  "price": 3986
}
],
"price_all": 3986
}
}
```

---

Ошибки валидации полей

**Формат ответа из общих требований**

---

Попытка удалить не из своего заказа

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": "Forbidden! You did not
accept this order!"
  }
}
```

---

Попытка удалить из наряд-заказа  
неактивной смены

**Status:** 403

**Content-Type:** application/json

**Body:**

```
{
  "error": {
    "code": 403,
    "message": " Denied! You cannot
```

	<p>change the order status of a closed shift!"</p> <pre> } } </pre> <hr/> <p><u>Попытка добавить в наряд-заказ с запрещенными статусами</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre> {   "error": {     "code": 403,     "message": " Denied! Cannot be added to an order with this status "   } } </pre>
--	---

#### Функционал механика

### Просмотр наряд-заказов текущей смены

Запрос предназначен для получения списка заказов активной смены со статусом «принят» или «в работе».

Request	Response
<p>URL: {{host}}/api-auto/order/taken Method: GET</p> <p>Headers - Content-Type: application/json</p>	<p style="text-align: center;"><u>Успех</u></p> <p>Status: 200</p> <p>Content-Type: application/json</p> <p>Body:</p> <pre> {   "data": [     {       "id": 7,       "car_box": "Бокс №5",       "shift_workers": "Palma",       "create_at": "2021-03- 28T02:52:52.000000Z", </pre>

	<pre> "status": "Принят", "price": 0 }, { "id": 11, "table": "Бокс №7", "shift_workers": "Palma", "create_at": "2021-03- 28T03:23:52.000000Z", "status": "В работе", "price": 0 } ] } </pre>
--	--

### Изменение статуса наряд-заказа

Запрос предназначен для смены статуса наряд-заказа активной смены. Для механика доступна смена со статуса «принят» на «в работе», а также со статуса «в работе» на статус «готов».

Request	Response
<p><b>URL:</b> {{host}}/api-auto/order/{{id}}/change-status</p> <p><b>Method:</b> PATCH</p> <p><b>Headers</b></p> <p>- Content-Type: application/json</p> <p><b>Body:</b></p> <pre> {   "status": "preparing" } </pre>	<p style="text-align: center;"><u>Успех</u></p> <p><b>Status:</b> 200</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre> {   "data": {     "id": 1,     "status": "preparing"   } } </pre> <hr/> <p style="text-align: center;"><u>Ошибки валидации полей</u></p> <p style="text-align: center;"><b>Формат ответа из общих требований</b></p>

	<hr/> <p><u>Попытка изменить на неразрешенный статус</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": " Denied! Can't change existing order status"   } }</pre> <hr/> <p><u>Попытка изменить заказ закрытой смены</u></p> <p><b>Status:</b> 403</p> <p><b>Content-Type:</b> application/json</p> <p><b>Body:</b></p> <pre>{   "error": {     "code": 403,     "message": "You cannot change the order status of a closed shift!"   } }</pre>
--	--

Разработанное API должно быть доступно по адресу <http://xxxxxx-m3.wsr.ru/api-auto>, где xxxxxx - логин участника (указан на индивидуальной карточке).

Форматы запросов и ответов, а также форматы дат и времени должен соответствовать примерам из задания.

## Модуль Г. Спидтаски

**Время на выполнение:** 3,5 часа

**Ваша задача - решить максимальное количество небольших задач**

№	ЗАДАЧА
<b>Разработка интерфейса пользователя</b>	
1	<p>Дана фотография с цветным фоном. Необходимо сделать так, чтобы фон стал прозрачным, а само изображение было приведено к оттенкам серого. Кроме того, нужно применить эффект размытия к изображению, чтобы оно приобрело мягкий, размытый вид. Задачу можно выполнить как с помощью графических редакторов, так и с помощью программирования на языках HTML и CSS.</p> <p>Результат должен быть представлен в файле: task1.html или task.jpg/png</p>
2	<p>Создайте интерфейс пользователя, содержащий следующие элементы: поле для ввода текста, кнопку, выпадающий список, флажок, переключатель, метку с флажком, метку с переключателем, календарь ввода, ползунок, индикатор прогресса.</p> <p>Результат должен быть представлен в папке: task2</p>
3	<p>С помощью CSS/HTML создать трехмерную фигуру в виде пирамиды, которая медленно вращается в бесконечном цикле. Каждая грань пирамиды должна быть разделена на 8 равных частей, а каждая часть должна иметь свой уникальный цвет.</p> <p>Результат должен быть представлен в файле: task3.html</p>
4	<p>С помощью CSS/HTML анимировать персонажа, изображенного на спрайт-таблице, бегущего справа налево. Анимация должна состоять из 8 кадров, каждый из которых представляет собой отдельную фазу бега. Длительность анимации должна быть 0,5 секунды на каждый кадр, а анимация должна повторяться бесконечно.</p> <p>Результат должен быть представлен в файле: task4.html</p>
5	<p>Создать адаптивный макет для веб-сайта, состоящий из следующих элементов: шапка, основная секция, боковая панель и футер. В основной секции должны быть реализованы блоки для постов, ширина которых составляет <math>\frac{1}{3}</math> ширины контейнера. Макет должен корректно отображаться на следующих значениях ширины экрана: <math>\geq 320\text{px}</math>, <math>\geq 480\text{px}</math>, <math>\geq 768\text{px}</math>.</p> <p>Результат должен быть представлен в файле: task5.html</p>
6	<p>Создать тумблер (switch) для Android с использованием только HTML и CSS. Тумблер должен иметь два положения (включено/выключено) и</p>

	<p>менять свой внешний вид в зависимости от текущего положения. При нажатии на тумблер его состояние должно меняться. Результат должен быть представлен в файле: task6.html</p>
<b>Разработка на стороне клиента</b>	
7	<p>Создать генератор цветов HEX. При перемещении соответствующих ползунков должен изменяться цвет и выводиться HEX код. Также, при нажатии на кнопку "Скопировать", должен происходить автоматический копипаст HEX кода в буфер обмена.</p> <p>Результат должен быть представлен в файле: task7.html</p>
8	<p>Создайте онлайн-калькулятор для расчета площади прямоугольника, круга и треугольника. При вводе пользователем значений сторон фигур, калькулятор должен автоматически вычислять и отображать площадь выбранной фигуры. Также, должна быть возможность переключения языка интерфейса (например, русский/английский).</p> <p>Результат должен быть представлен в файле: task8.html</p>
9	<p>Создайте функцию, которая принимает в качестве входных данных строку с арифметическим выражением, содержащую следующие операторы в порядке приоритета:</p> <p>Квадратные скобки: [] Унарный минус: - Экспоненция: ** Умножение, деление: * / Сложение, вычитание: + -</p> <p>Функция должна возвращать вычисленный результат выражения. Если в выражении есть ошибки, функция должна вернуть сообщение об ошибке.</p> <p>Пример 1: Ввод функции: [1+2]*3 Выход функции: 9</p> <p>Пример 2: Вход функции: 3**2/5 Выход функции: 1.8</p> <p>Пример 3: Вход функции: -[1-2] Выход функции: 1</p> <p>Результат должен быть представлен в файле: task9.html</p>
10	<p>Реализовать таймер обратного отсчета, который будет показывать оставшееся время до какого-либо события. Таймер должен отображать часы, минуты и секунды. При старте таймера, он должен запускаться с заданного времени и отсчитывать время до нуля. При достижении нуля, должен срабатывать триггер (например, звуковой сигнал или анимация). Таймер должен быть остановлен пользователем вручную или автоматически по истечении времени.</p> <p>Результат должен быть представлен в файле: task10.html</p>

11	<p>Создать онлайн-редактор для рисования, используя HTML5 Canvas. Редактор должен позволять рисовать на холсте, перемещая мышь, и иметь возможность переключаться между тремя цветами для рисования. Также, должен быть доступен инструмент для очистки холста.</p> <p>Результат должен быть представлен в файле: task11.html</p>
<b>Разработка на стороне сервера</b>	
12	<p>Используя PHP/Python, создать изображение проверочного кода (CAPTCHA), которое должно соответствовать следующим правилам:</p> <ul style="list-style-type: none"> <li>a. Случайным образом генерировать шесть символов, состоящих из английского алфавита (A-Z, a-z) или цифр (0-9).</li> <li>b. Каждый символ должен быть повернут на случайный угол от -15 до 15 градусов.</li> <li>c. Символы должны быть размещены на изображении в две строки по три символа в каждой.</li> <li>d. Изображение должно содержать не менее 5 случайных линий, пересекающихся друг с другом.</li> <li>e. Не менее 5 точек смешанного шума должны быть добавлены на изображение.</li> </ul> <p>Результат должен быть представлен в папке: task12</p>
13	<p>Реализовать функцию загрузки файлов на сервер с автоматической сортировкой по папкам в зависимости от расширения файла (txt, csv, json, xml). Файлы должны быть переименованы в соответствии с датой и временем их загрузки, например: 2024_05_28_15_13.txt</p> <p>Результат должен быть представлен в папке: task13</p>
14	<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, а также количество общих элементов..</p> <p>Результат должен быть представлен в папке: task14</p>
<b>Разработка информационных ресурсов с использованием готовых решений</b>	
15	<p>Установите WordPress. Данные для авторизации администратора: Логин: admin2026 Пароль: prof2026 Установите и активируйте тему Twenty Twenty-Three</p>
16	<p>Создайте плагин, который показывает время, оставшееся до Нового года. При активации плагина создается виджет. При добавлении виджета в сайдбар должно отображаться оставшееся время в правом верхнем углу на каждой странице. Дата не должна быть статичной. Формат даты (ММ: ДД: ЧЧ: СС) Название плагина: "Новогодний обратный отсчет"</p>
17	<p>Сделайте плагин, который позволит загружать несколько медиафайлов в галерею WordPress с помощью кнопки в админ-панели под названием</p>

	"Добавить в галерею" и отображать их в слайдере на главной странице сайта.
18	Создайте плагин, который выводит на Dashboard кнопку «Включить/Выключить режима обслуживания». При включении режима обслуживания, у клиентской части сайта должна отображаться страница «Сайт временно недоступен». При выключении режима обслуживания, сайт должен быть снова доступен для посетителей.
19	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 вверху форма (поле + кнопка) для добавления новой категории. Карточки товаров (заголовки+цена+категория+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. Остальные задачи необходимо решить без использования фреймворков и сторонних библиотек.

Разработанные задания должны быть размещены по адресу <http://xxxxxx-m4.wsr.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 КОМПЕТЕНЦИИ<sup>2</sup>

### 2.1. Личный инструмент конкурсанта

Конкурсанту разрешается использовать собственные:

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



- 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

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