|  |  |
| --- | --- |
|  |  |

КОНКУРСНОЕ ЗАДАНИЕ КОМПЕТЕНЦИИ

«Веб-технологии»

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](#_Toc189488875)

[1.1. ОБЩИЕ СВЕДЕНИЯ О ТРЕБОВАНИЯХ КОМПЕТЕНЦИИ 3](#_Toc189488876)

[1.2. ПЕРЕЧЕНЬ ПРОФЕССИОНАЛЬНЫХ ЗАДАЧ СПЕЦИАЛИСТА ПО КОМПЕТЕНЦИИ «ВЕБ-ТЕХНОЛОГИИ» 3](#_Toc189488877)

[1.3. ТРЕБОВАНИЯ К СХЕМЕ ОЦЕНКИ 10](#_Toc189488878)

[1.4. СПЕЦИФИКАЦИЯ ОЦЕНКИ КОМПЕТЕНЦИИ 12](#_Toc189488879)

[1.5. КОНКУРСНОЕ ЗАДАНИЕ 12](#_Toc189488880)

[2. СПЕЦИАЛЬНЫЕ ПРАВИЛА КОМПЕТЕНЦИИ 38](#_Toc189488882)

[2.1. Личный инструмент конкурсанта 38](#_Toc189488883)

[2.2. Материалы, оборудование и инструменты, запрещенные на площадке 39](#_Toc189488884)

[3. ПРИЛОЖЕНИЯ 40](#_Toc189488885)

**ИСПОЛЬЗУЕМЫЕ СОКРАЩЕНИЯ**

1. ИР – 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** | 8 |
| 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 официальной технической документации по используемым технологиями и языкам программирования. |
| Специалист должен уметь:  − собирать, анализировать и оценивать информацию;  − использовать навыки грамотности для толкования стандартов и требований;  − составлять тестовую документации для тестирования новых функциональностей 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 | **Техническая поддержка и администрирование информационных ресурсов** | 7 |
| Специалист должен знать и понимать:  − 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. |
| Специалист должен уметь:  − 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;  − 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 | **Разработка интерфейса пользователя** | 21,5 |
| Специалист должен знать и понимать:  − 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; |
| Специалист должен уметь:  − 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** | 36 |
| Специалист должен знать и понимать:  − 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 |
| Специалист должен уметь:  − создавать и модифицировать JavaScript код для улучшения функциональности и интерактивности сайта;  − манипулировать элементами страницы веб-приложения;  − разрабатывать анимацию для повышения доступности и визуальной привлекательности веб-приложения;  − применять открытые библиотеки и фреймворки;  − тестировать веб-приложение. |
| 5 | **Разработка веб приложения на стороне сервера** | 27,5 |
| Специалист должен знать и понимать:  − 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. |
| Специалист должен уметь:  − 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; |
|  |

### 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** |  | **A** | **Б** | **В** | **Г** |  |
| **1** | 2 | 3 | 1 | 2 | 8 |
| **2** | 2 | 2 | 1 | 2 | 7 |
| **3** | 20 |  |  | 1,5 | 21,5 |
| **4** |  | 23 |  | 13 | 36 |
| **5** |  |  | 17 | 10,5 | 27,5 |
| **Итого баллов за критерий/модуль** | | 24 | 28 | 19 | 29 | **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** | **Короткие задачи** | Проверка результата выполнения модуля после его завершения |

### 1.5. КОНКУРСНОЕ ЗАДАНИЕ

Общая продолжительность Конкурсного задания[[1]](#footnote-1): 15 ч.

Количество конкурсных дней: 3 дня

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

Оценка знаний участника 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

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 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.

*Таблица №4* **Матрица конкурсного задания**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Обобщенная трудовая функция | Трудовая функция | Нормативный документ/ЗУН | Модуль | Константа/вариатив | ИЛ | КО |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |

Инструкция по заполнению матрицы конкурсного задания **(Приложение № 1)**

**1.5.2. Структура модулей конкурсного задания (инвариант/вариатив)**

**Модуль А. Разработка интерфейса пользователя**

**Технологии этого модуля:** HTML5, CSS3, графический Дизайн

*Время на выполнение модуля:* 6 часов

**Задания:**

В современном мире с каждым днём появляется всё больше и больше новых технологий, которые позволяют решать повседневные задачи.

В Ненецком автономном округе не особо развита повседневная доставка внутри населённых пунктов, только межрегиональная доставка. Но в последнее время вырос спрос на доставку не только горда Нарьян-Мар, но и в соседние населенные пункты.

К вам обратилась организация, оказывающая доставку в городе Нарьян-Мар, поселок Искателей и село Тельвиска.

Подобных ресурсов недостаточно и заказчик хочет в кратчайшие сроки занять лидирующие позиции в этой нише (максимально увеличить количество пользователей, использующих данный сервис). Для этого вам необходимо использовать в дизайне сверстанных страниц 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.

Подразумевается, что пользователями сайта являются жители Ненецкого автономного округа, которые обращаются к ресурсу с помощью смартфонов и персонального компьютера.

Использование любых фреймворков и библиотек (Bootstrap, Tailwind, например) запрещено.

**ОПИСАНИЕ ПРОЕКТА И ЗАДАЧ**

Ваша задача - сверстать следующие страницы веб-сайта:

1. Главная
2. Страница входа в личный кабинет
3. Страница регистрации в личном кабинете
4. Страница личного кабинета пользователя
5. Страница с активными заказами пользователя.
6. Страница оформления доставки
7. Страница с тарифами
8. Страница с историей заказов

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

Страницы должны быть адаптивны. При переходе от версии для персонального компьютера, главная страница (разрешение экрана 1920 пикселей), к мобильной версии (ширина экрана минимум 768 пикселей) должно быть реализовано не менее четырех структурных изменений контента. На остальных разрешениях контент должен отображаться корректно.

**Основные роли пользователей**

Гость

1. Регистрация
2. Авторизация
3. Просмотр тарифов и расчет доставки

Авторизованный пользователь

1. Создание заказа
2. Возможность отмены заказа
3. Просмотр статуса активных заказов
4. Просмотр истории заказов
5. Просмотр информации о себе с возможностью редактирования некоторых данных.
6. Оценить оставить комментарий по выбранному заказу
7. Открытие споров по заказу
8. Выход из профиля

Оператор

1. Просмотр активных заказов
2. Просмотр истории заказов за смену
3. Ответ на отзыв
4. Решение споров
5. Завершение заказов
6. Назначение заказов
7. Выход

**Главная - Landing Page**

Главная страница должна содержать следующие блоки:

1. Шапка сайта
   1. Логотип.

Вам предоставлен существующий логотип компании logo.png, однако заказчик просит разработать и использовать на сайте новый логотип. Учтите, что новый логотип должен содержать основные цвета сайта, сохранять преемственность с существующим, но, тем не менее, иметь отличия от предоставленного.

* 1. Меню навигации.
     1. Поиск.
     2. Регистрация.
     3. Личный кабинет.
     4. Вход
     5. Заказы

1. Вывод новостей в виде слайдера с индикацией активного слайда (работа слайдера не проверяется, только дизайн и верстка). Слайд должен содержать следующую информацию:
   1. Изображение.
   2. Название новости
   3. Краткое описание
   4. Кнопка «подробнее» для перехода на страницу с полным описанием.
   5. Подписка на новости
2. Секция с быстрым расчетом цены заказа. Предусматривает наличие одного поля для ввода начала маршрута и одного поля для ввода конечного пункта. В поле ввода должны выводится подсказки по поиску.
3. Факты о компании и сайте
   1. Осуществили доставок (за неделю, месяц, год, все время)
   2. Примерный товарооборот.
   3. Наиболее распространённые маршруты

Около каждого из пунктов заказчик просит разработать и разместить соответствующее графическое изображение.

1. Карточки с типом курьерского транспорта.

Общее число типов на странице не более 6, верстка не должна нарушаться если типов транспорта менее 6. При наведении на транспорт должно появляться краткое описание или визуальное выделение его.

1. Форма расчета. Должна содержать следующие поля ввода:
   1. Место отправления
   2. Место назначения
   3. Дата отправления
   4. Кнопка расчета.
2. Отзывы в виде слайдера с прокруткой по 1 элементу. Реализация JS не требуется, но визуально должно быть понятно, что подразумевается поочередная смена отзывов.
   1. Имя автора отзыва
   2. Фото
   3. Отзыв
   4. Дата отзыва

Порядок полей может быть любым.

1. Подвал сайта
   1. Телефон “8 (800)123-45-67”
   2. e-mail: [mail@newlife.ru](mailto:mail@newlife.ru)
   3. Навигация по сайту:
      1. Главная
      2. Регистрация
      3. Авторизация
      4. Личный кабинет
      5. Создать заказ

**Страница входа в личный кабинет**

На этой странице вам необходимо сделать форму со следующими полями:

* e-mail
* Пароль
* Кнопка для входа
* Кнопка восстановления пароля

**Страница регистрации в личном кабинете**

На этой странице вам необходимо сделать форму со следующими полями, обязательными для заполнения:

* e-mail
* Имя
* Фамилия
* Возраст
* Телефон
* Пароль
* Повтор пароля
* Элемент, подтверждающий согласие на обработку персональных данных
* Кнопка для регистрации

Отобразите сообщения, если пользователь указал некорректные данные, не заполнил обязательные поля и т.д.

**Страница личного кабинета пользователя**

* Общая информация о пользователе (в любом порядке):
* номер телефона и кнопка «Изменить»,
* e-mail и кнопка «Изменить»,
* история заказов
* Фамилия, Имя и кнопка «Изменить»,
* Краткая сводка о сумме затрат, созданных маршрутах, закрытых и отмененных заказов.

**Страница добавления информации о новом виде транспорта**

Информацию о новом виде транспорте может добавлять только оператор.

Ниже представлены поля, которые необходимо заполнить.

* Вид транспорта
* Перевозимый груз
* Габаритные размеры
* Грузоподъемность
* Населенные пункты, в которые возможна доставка
* Фотографии
* Дополнительная информация

Отобразите варианты сообщений, если пользователь указал некорректные данные, не заполнил обязательные поля и т.д.

**Страница личного кабинета оператора**

На странице должны отображаться заказы пользователей. Для каждого заказа должна отображаться вся информация, указанная пользователем при заполнении.

Модератор может сортировать по дате, по пользователю, по начальной и конечной точке - вам необходимо предусмотреть соответствующие элементы управления.

По умолчанию у новых заказов установлен статус «Активен». Должны быть доступны элементы управления, позволяющие изменить статус на «Назначен транспорт», «Завершен», «Отменен».

**Страница с заказами**

Попасть на эту страницу можно с главной (из формы быстрого расчета суммы или соответствующей вкладки). Вам необходимо создать форму. На этой странице необходимо отобразить в виде таблицы информацию обо всех найденных заказах пользователя, а именно:

* Точка отправления,
* Точка назначения
* Время отправления
* Дата отправления
* Вид транспорта
* Контакты водителя
* Вид груза
* Стоимость
* Статус

Дополнительная информация, например, фотографии, описание, и другое

**ИНСТРУКЦИЯ ДЛЯ КОНКУРСАНТА**

Ваш HTML/CSS должен быть валидным.

Страницы должны быть согласованы между собой.

Вам нужно использовать не менее одного из представленных шрифтов.

Оценка будет производиться в браузере Google Chrome.

Сверстанный веб-сайт должен быть доступен по адресу   
http://xxxxxx-ml.domain.ru/, где хххххх - логин участника (указан на индивидуальной карточке).

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

Главная - index.html, index.png

Страница входа в личный кабинет - login.html, login.png

Страница регистрации в личном кабинете - register.html, registering

Страница личного кабинета пользователя - profile.html, profile.png

Страница добавления информации о новом транспорте - add.html, add.png

Страница личного кабинета оператора - orders-management.html,   
orders-management.png

Страница заказов - order.html, order.png

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

Все страницы указанные выше должны быть доступны к просмотру по соответствующим адресам: <http://xxxxxx-ml.domain.ru/index.html>,

http://xxxxxx-ml.domain.ru/login.html и т.д.

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

Работы, выполненные с использованием CSS-фреймворков, проверяться не будут.

**Модуль Б. Разработка Веб-приложения на стороне клиента**

*Время на выполнение модуля: 3 часа*

Вам необходимо разработать Single Page Application (далее SPA) для организации организующая перемещение товаров (грузов) из (в) города Нарьян-Мар и прилегающие населенные пункты. Заказчик предоставляет вам рабочее API.

*Общее*

Ваша задача - реализовать SPA приложение, которое будет взаимодействовать с уже разработанным API.

В примерах будет использоваться переменная {{host}} которая обозначает адрес API: http://xxxxxx-m2.domain.ru/.

Ваше SPA должно состоять из следующих экранов:

* Главная
* Вход в личный кабинет
* Регистрация в личном кабинете
* Личный кабинет
* Создание заказа

ВНИМАНИЕ! Вся логика приложения должна быть организована через взаимодействие с API.

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 подсвечиваться красным. При успешной регистрации происходит переход на экран входа в систему.

*Интерактивные элемента экрана*

По щелчку на кнопку регистрации происходит попытка зарегистрировать нового пользователя.

Вход в систему

На данном экране представлена форма для входа пользователя в систему. При вводе некорректных значений у соответствующих полей формы отображаются тексты ошибок, а сами поля подсвечиваться красным. При успешной аутентификации происходит переход на экран «Заказы пользователя», иначе выводится соответствующее сообщение.

*Интерактивные элементы экрана*

По щелчку на кнопку входа происходит попытка аутентифицировать пользователя.

Личный кабинет

На данном экране отображаются список заказов, созданных пользователем. Для каждого заказа отображается информация откуда, куда, вид транспорта, время в пути. Возле каждого маршрута есть кнопки изменения статуса. Количество заказов. Комментарии, оставленные пользователем.

*Интерактивные элементы экрана*

По щелчку на кнопку маршрут завершается заказ.

При создании заказа должна быть предусмотрена возможность загрузки картинки или фотографии товара.

Создание заказа

На данном экране отображаются, поля для ввода начальной точки маршрута, конечной точки маршрута, тип транспорта (в том числе любой) и дата предполагаемого маршрута, указание груза кнопка создать.

Список заказов, напротив каждого маршрута кнопка «Завершить».

*Интерактивные элементы экрана*

По щелчку на кнопку «Все заказы» выводиться список всех заказов. Вверху списка активные, внизу завершенные.

**Описание предоставленного API**

Документация API идентична описанной в Модуле Б.

Инструкция для конкурсанта

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

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

Вам предоставляются следующие конфигурации фреймворков и библиотек:

jQuery 3.x

VueJS 2.x

VueJS 3.x

Vue Router 3.x

Vue Router 4.x

React

React Router

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**

Время на выполнение модуля: 3 часа

Ваша задача – Сервис по перевозке заказов по территории города Нарьян-Мара, поселок Красное и село Тельвиска.

**ОСНОВНОЙ ФУНКЦИОНАЛ**

Для неавторизованного пользователя:

* авторизация
* регистрация

Для авторизованного пользователя:

* возможность сброса авторизации
* Поиск и выбор заказа
* Добавление заказа
* Редактирование заказа
* Завершение заказа

Функционал авторизованного пользователя не должен быть доступен гостю.

Для оператора:

* Просмотр заказа пользователей
* Изменение статуса заказов
* Назначение прав

ОБЩЕЕ

Вам необходимо реализовать REST API заданной структуры.

В примерах будет использоваться переменная {{host}} которая обозначает адрес <http://xxxxxx-m3.domain.ru/>, где хххххх - логин участника.

Идентификацию пользователя организуйте посредством Bearer Token.

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

**Status:** 403

**Content-Type:** application/json

**Body:**

{

"message": "Login failed"

}

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

**Status:** 403

**Content-Type:** application/json

**Body:**

{

"message": "Forbidden for you"

}

При попытке получить не существующий ресурс необходимо возвращать ответ следующего вида:

**Status:** 404

**Content-Type:** application/json

**Body:**

{

"message": "Not found"

}

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

**Status:** 422

**Content-Type:** application/json

**Body:**

{

"success": false,

"message": {

<key>: [<error message>]

}

}

В свойстве message.<key> 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:

{

"success": false,

"message": {

"phone": [ "field phone can not be blank"],

"password": ["field password can not be blank"]

}

}

В поле message могут быть любые сообщения об ошибках (если не указана конкретная ошибка), но они должны описывать возникшую проблему.

Вы должны завести тестовый аккаунт с email test@gmail.com и паролем test.

Серверная часть должна позволять загрузку картинок, редактирования ресурсов и заказов, удаление и поиск.

**ОПИСАНИЕ ФУНКЦИОНАЛА ГОСТЯ**

Аутентификация:

При успешной аутентификации возвращается сгенерированный токен пользователя.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host})/authorization **Method:** POST  **Headers**  - Content-Type:  application/json  **Body:**  { "email": "[admin@admin.ru](mailto:admin@admin.ru)", "password": "Qa1" } | Успех  **Status: 200**  **Content-Type:** application/json **Body:**  {  "success": true,  "message": "Success",  "token": <сгенерированный token>  } |
| Неправильные логин или пароль  **Status:** 401  **Content-Type:** application/json  **Body:**  {  "success": false,  "message": "Login failed"  } |

Регистрация:

При успешной регистрации возвращается сгенерированный токен добавленного пользователя. Функция должна принимать следующие параметры:

* email - email пользователя обязательный, валидный e-mail адрес, уникальный
* password - пароль пользователя
* обязательный, состоит минимум из 3 символов, из которых как минимум одна строчная, одна прописная и одна цифра
* first\_name - имя, обязательное
* last\_name - фамилия обязательное

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}} /registration  **Method:** POST  **Headers**  - Content-Type: application/json  **Body:**  {  "email": "admin@admin.ru",  "password": "Qa2",  "first\_name": "name",  "last\_name": "last\_name"  } | Успех  **Status:** 200  **Content-Type:** application/json  **Body:**  {  "success": true,  "message”: "Success",  "token": <сгенерированный token>  Ошибки валидации  **Формат ответа из общих требований** |

**ОПИСАНИЕ ФУНКЦИОНАЛА АВТОРИЗИРОВАННОГО ПОЛЬЗОВАТЕЛЯ**

Сброс авторизации:

Запрос предназначен для очистки значение токена пользователя.

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/logout **Method:** GET | Успех  **Status: 200**  **Content-Type:** application/json  **Body:**  {  "success": true,  "message": "Logout"  } |
| Попытка обратиться не авторизованным пользователем  **Формат ответа из общих требований** |

Получения информации об маршрутах пользователя

Запрос предназначен для получения информации о выбранных маршрутах, авторизированного пользователя:

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/user/favorites **Method:** GET  **Headers**  **- Content-Type:** application/json | Успех  **Status: 200**  **Content-Type:** application/json  **Body:**  {  "data": [  {  "treck\_id": "1",  "from": {  "Cpoint\_start": "НМ",  "Spoint\_start": "Оленная",  "Hpoint\_start": "3",  "date": "2023-10-01",  "time": "10:05"  },  "to": {  "Cpoint\_end": "Красное",  "Spoint\_end": "Оленная",  "Hpoint\_end": "4",  "date": "2023-10-01",  "Tcargo": "Железо"  "mass": "100"  },  "delta": "2h",  "type\_avto": "а455ам83"  ………"status":"Activ"  },  {  "treck\_id": "13",  "from": {  "Cpoint\_start": "Красное",  "Spoint\_start": "Усинская",  "Hpoint\_start": "15",  "date": "2023-11-01",  "time": "10:15"  },  "to": {  "Cpoint\_end": "Тельвиска",  "Spoint\_end": "Речная",  "Hpoint\_end": "4",  "date": "2023-11-01",  "Tcargo": "Вата"  "mass": "10"  },  "delta": "3h",  "type\_avto": "а458ом83"  ………"status":"Ending"  },  ]  } |
| Попытка обратиться не авторизованным пользователем  **Status:** 401  **Content-Type:** application/json  **Body:**  {  "error": {  "code": 401,  "message": "Unauthorized"  }  } |

Добавление заказа

|  |  |
| --- | --- |
| Request | Response |
| **URL:** {{host}}/api-tort/treck  **Method:** POST    **Headers**  - Content-Type: application/json    **Body:**  {  {  "from": {  "Cpoint\_start": "НМ",  "Spoint\_start": "Матросова",  "Hpoint\_start": "4",  "date": "2023-10-01",  "time": "10:05"  },  "to": {  "Cpoint\_end": "Красное",  "Spoint\_end": "Речная",  "Hpoint\_end": "5",  "date": "2023-10-01",  "Tcargo": "Оленина"  "mass": "10"  } | Успех  Успех  **Status:** 200  **Content-Type:** application/json  **Body:**  {  "data": {  …"message ":"completed"  }  } |
| Попытка обратиться не авторизованным пользователем  **Status:** 401  **Content-Type:** application/json  **Body:**  {  "error": {  "code": 401,  "message": "Unauthorized"  }  } |

Инструкция для конкурсанта

Вам предоставляются следующие конфигурации фреймворков и библиотек:

jQuery 3.x

VueJS 2.x

VueJS 3.x

Vue Router 3.x

Vue Router 4.x

React

React Router

Laravel 10.x

Yii Basic 2.0.x

Django 4.0.x (включая пакеты: djangorestframework, django-cors-headers, pillow, django-filter, mysqlclient)

Вы можете использовать любой из представленных фреймворков.

**Модуль Г. Короткие задачи**

Время на выполнение модуля: 3 часа

Ваша задача - решить максимальное количество небольших задач.

|  |  |  |  |
| --- | --- | --- | --- |
| № | | Задача | Технологии |
| 1 | | Написать JS функцию, имитирующую движение секундной стрелки | Разработка на стороне клиента |
| 2 | | Написать JS функцию, которая получает параметр N, создает поле из квадратов NxN, четные столбцы и четные строки имеют цвет вона зелёный, остальные желтый | Разработка на стороне клиента |
| 3 | | Написать функцию, создающий календарь (на месяц.) | Разработка на стороне клиента |
| 4 | | Написать функцию JS, получающий параметр N (ширина квадрата в пикселях) и параметр Z количество квадратов. Функция должна вывести Z Квадратов, вложенных друг в друга, и имеющий случайного цвета фон | Разработка на стороне клиента |
| 5 | | Написать JS Функцию. Которая произвольно закрашивала бы 10 клеток на поле 10\*10, так, чтобы между ними было как минимум 1 клетка | Разработка на стороне клиента |
| 6 | | Нарисовать с помощью Canvas снеговика. | Разработка на стороне клиента |
| 7 | | Написать функцию JS заставляющая двигаться контейнер <div> размером 30х30 пикселей. Внутри контейнера <div> размером 600х600. При столкновении со стенкой отскакивает под тем же углом | Разработка на стороне клиента |
| 8 | | Функция JS получает параметры день месяц и год. На основании данных величин рассчитать-  Рассмотрим расчет на примере даты 18.11.1991 года.  Суммируем все цифры даты рождения и получаем первое дополнительное число: 1+8+1+1+1+9+9+1=31  Суммируем цифры первого дополнительного числа и получаем второе дополнительное число: 3+1=4  Вычитаем из первого дополнительного числа первую цифру числа рождения, умноженную на 2. Получаем третье дополнительное число. 31-(1х2) = 29  Суммируем цифры третьего дополнительного числа и получаем четвертое дополнительное число. 2+9=11 | Разработка на стороне клиента |
| 9 | | С помощью JS функции, вывести модальную форму, запрашивающая Имя, Номер телефона, Номер и серию паспорта, электронный адрес, и проверить на корректность ввода данных | Разработка на стороне клиента |
| 10 | | Написать функцию JS, выводящую поле NxN (N – аргумент) фон каждого элемента поле случайный. | Разработка на стороне клиента |
| 11 | | Создать форму анкеты с верификацией данных (без использования JS) (Имя, Логин, дата рождения, электронная почта, номер телефона). Все поля обязательны | Разработка на стороне клиента |
| 12 | | Макет страницы Шапка, Левая панель навигации, Правая панель виджетов, контент, подвал. Верстка должна быть адаптивной. | Разработка на стороне клиента |
| 13 | | Создать шахматную доску. Доска должна генерироваться на стороне коиента. (подписи обязательно) | Разработка на стороне клиента |
| 14 | | Создать форму регистрации пользователя, данные должны быть верифицированы, и записаться в БД. Пароль хешируется (шифруется), одинаковые логины не допускаются | Разработка на стороне клиента |
| 15 | | Вводиться текстовая строка не более 255 символов. Выводиться таблица букв с указанием сколько данная буква встречается в строке. Пробел и символы не учитываются в подсчёте. | Разработка на стороне сервера |
| 16 | | Дано три таблицы Таблица городов, Таблица Стран, Таблица Принадлежности города к той или иной стране. Вывести выпадающий список стран. И таблицу городов, принадлежащих выбранному из списку стране. | Разработка на стороне сервера |
| 17 | | Вывести на экран поле 600 на 600 пикселей, при клике на поле в БД записываются координаты и время клика | Разработка на стороне сервера |
| 18 | | Создать Форму для заполнения БД расписания прибывающих поездов. После ввода данных выводиться таблица расписания | Разработка на стороне сервера |
| 19 | | На сервере генерируется календарь на выбранный месяц и год с указанием дней недели и красным помечаются выходные дни | Разработка на стороне сервера |
| 20 | Форма ввода данных читательского дневника. Автор, произведение, дата начало чтения, дата, когда закончил, тип книги (бумага, электронная, аудио). | | Разработка на стороне сервера |

Инструкция для конкурсанта

Задачи необходимо решить без использования фреймворков и сторонних библиотек.

### 2.1. Личный инструмент конкурсанта

Конкурсанту разрешается использовать собственное оборудование и инструменты:

● клавиатуру на любом языке. Если конкурсант пользуется своей клавиатурой без поддержки макросов и быстрых команд;

● языковые файлы для клавиатуры;

● мышь без поддержки макросов и дополнительных кнопок;

● графический планшет;

● наушники проводные;

● аудиофайлы с музыкальными композициями (не более 30 файлов в формате mp3). Файлы предоставляются на флеш-носителях в день Д-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, должно 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 Инфраструктурный лист

Приложение №4 Критерии оценки

Приложение №5 План застройки

Приложение №6 Инструкция по охране труда и технике безопасности по компетенции «Веб-технологии».

Приложение № 7 Медиа файлы

1. *Указывается суммарное время на выполнение всех модулей КЗ одним конкурсантом.* [↑](#footnote-ref-1)