

Описание задания экзамена

“Веб-разработка на стороне клиента”

Ваша задача - написать клиентскую часть для вашего курсового проекта, где упор идет на серверную разработку. Клиентская часть состоит из 2х страниц - главная страница и детальная страница конкретного объекта предметной области. Соединять клиент и сервер задачи нет, вы берете только предметную область. В случае, если вы осенью сменили предметную область, для текущего задания можете выбрать одну из двух на свое усмотрение. При необходимости можно создать дополнительные страницы.

Для успешной сдачи вам пригодятся знания по верстке веб-страниц (html/css, адаптивная верстка, использование CSS Grid или CSS Flexbox), SPA на Vue.js и Vue Router, Vuetify, Vuex хранилище. По согласованию с преподавателем допускается использовать аналогичные указанным в задании технологии.

Допускается использование справочников, учебных материалов, однако копирование кода у других студентов запрещается.

Данные для сайта должны приходить из предварительно сохраненного файла (например, в формате json), соединения с базой данных не требуется.

Итог задания - репозиторий с кодом (необходимо периодически делать комиты как подтверждение регулярности выполнения в течение дня), выложить на любое хранилище репозитория, открыть доступ для просмотра.

Веб-страница должна быть размещена в личном кабинете на сервере факультета fit.mospolytech.ru с использованием системы контроля версий Git. Все основные тексты на сайте должны быть на русском языке.

Для защиты задания необходимо предоставить отчет в формате pdf с пояснениями выполнения пунктов задания в структурированном виде (с указанием пункта задания, скриншота).

Основное задание

1. Созданы две веб-страницы, есть очевидный переход между страницами в меню. Первая страница - главная страница проекта. Вторая страница - детальная страница конкретного объекта вашего проекта (любой на выбор, на котором удобно показать работоспособность других пунктов задания).
2. Обе страницы размещены на хостинге (хостинг университета или любой другой доступный вам). URL должен содержать “webexam2023”.
3. В меню на обеих страницах присутствует ссылка на json файл для проверки в веб-среде (при открытии в браузере можно посмотреть содержимое файла).
4. В меню на обеих страницах присутствует ссылка на открытый git репозиторий.
5. В LMS курс <https://online.mospolytech.ru/course/view.php?id=10230> в раздел “Экзамен “Веб-разработка на стороне клиента”” загружен отчет по экзаменационному заданию. В отчете необходимо описать выполнение пунктов задания и приложить скриншот.

Скриншот должен содержать адрес страницы и текущие дату и время. При необходимости можно дополнительно добавить уточняющий скриншот (т.е. если мелко и нечитабельно - создать без адреса, даты и времени).

6. Отчет должен содержать титульный лист. На второй странице отчета необходимо указать предметную область и приложить скриншот-подтверждение, что эта область попала вам по варианту. **Использовать другую предметную область не допускается, работы не принимаются к проверке.** Также на второй странице разместить ссылки на страницы, git репозиторий.

Задание на удовлетворительно

1. Для создания страниц использовать семантические теги HTML5.
2. Выбрать целевое действие пользователя на сайте для своего проекта. Описать выбранное действие в отчете.
3. Учесть принципы доступности страниц.
4. Разместить на странице четко выраженный призыв к целевому действию.
5. Разработать эффектные релевантные заголовки, отражающие тему сайта.
6. Дизайн сайта простой, не перегруженный информацией.
7. Дизайн сайта единообразный (единый размер элементов, размер и оформление блоков и кнопок, заголовков, подзаголовков, основного текста и т.д.).
8. Написан подталкивающий к действию текст с релевантными ключевыми словами.
9. Используются формы и функциональные кнопки.
10. На сайте используются изображения, иконки, при этом обработка изображений не учитывается.
11. Гармоничный дизайн сайта (информация наглядна и легко воспринимается - элементы корректно расположены друг относительно друга, отступы достаточны, но не чрезмерны, шрифт соответствует стилю сайта).
12. Меню веб-страниц содержит вывод информации о “залогиненном” пользователе.
13. Адаптировать детальную страницу для печати.
14. Добавить на страницу форму поиска информации на сайте.
15. Добавить на страницу фильтрацию элементов на странице.
16. Добавить на страницу сортировку элементов на странице.
17. Добавить на страницу элементы доверия (комментарии, положительные отзывы, сертификаты, счетчики покупок и т.д.)
18. Добавить на страницу форму для добавления отзыва.
19. Добавить на страницу форму добавления или редактирования объекта (использование тега form, различных типов элементов форм, использование атрибута inputmode).
20. Создать функционирующую SPA.
21. Использовать виртуальную маршрутизацию vue-router.
22. Получать необходимые для SPA данные из файла main.json.
23. Главная страница - заполняется из файла main.json (для загрузки можно использовать axios/fetch). При необходимости можно использовать дополнительные json файлы.
24. Детальная страница объекта - идентификатор отображаемого объекта выбирается из динамического параметра url. Данные об объекте загружаются из файла main.json. При необходимости можно использовать дополнительные json файлы.
25. Создать компонент для отображения списка элементов, который использует полученные данные.
26. Создать компонент формы, который отправляет данные формы, введенные пользователем.

Задание на хорошо

1. Выполнено задание на удовлетворительно.
2. Продемонстрировать использование фреймворка Vuetify и его компонентов. Использовать минимум 5 компонентов.

Задание на отлично

1. Выполнено задание на хорошо.
2. Продемонстрировать использование Vuex хранилища.
3. Создать необходимые свойства state, getters, mutations, и actions.
4. Внедрить геттеры и действия в компоненты проекта.

Штрафные баллы

1. Страница содержит ошибки валидации HTML или CSS. Штрафные баллы за каждую ошибку.
2. Файл с данными недостаточно содержательный и разнообразный, мало данных, данные не соответствуют содержанию сайта.
3. Название ключей (переменных) в файле с данными отражают содержание.
4. Отсутствие регулярных коммитов в репозиторий (до экзамена и во время).
5. Слабое использование возможностей CSS. Не используются различные типы селекторов, возможности оформления текстов, изображений, фона, форм, таблиц и пр. Не используются сложные селекторы, использование классов и т.д., селекторы элемента, класса, контекстные селекторы, псевдоклассы, псевдоэлементы.
6. Текст на сайте не соответствует теме, нелогичен и неграмотно написан.
7. Отсутствует “резиновая” верстка для просмотра сайта при различных размерах браузера.
8. Отсутствует выполнение пунктов 1-4 из “Основного задания”. Штраф за каждое невыполненное задание.

Ссылки на справочные материалы

1. Ссылка на лабораторные работы
https://drive.google.com/drive/folders/11rxPSEZdHo_hSSW4wqzjC6Ogu4unvz4Z
2. <https://html5css.ru/tags/default.php>
3. <https://html5css.ru/css/default.php>
4. <https://a11ymyths.com/ru/>
5. <https://www.w3.org/WAI/ARIA/apg/patterns/>
6. <https://ru.vuejs.org/v2/style-guide/index.html>
7. <https://my-json-server.typicode.com/>
8. <https://v3.vuex.vuejs.org/ru/guide/state.html#%D0%B5%D0%B4%D0%B8%D0%BD%D0%BE%D0%B5-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE-%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8E>