

Институт кибербезопасности и цифровых технологий
Кафедра: КБ-14 «Цифровые технологии обработки данных»

Дисциплина: «Веб-программирование»

Требования к проектам

1. Внедрение технологии AJAX в проект

Необходимо внедрить технологию AJAX в ранее созданный проект (проект, который был защищен в конце первого семестра). Данная технология должна применяться минимум: в разделе администратора.

Первоначальный проект, должен был содержать в себе: 5 страниц, панель администратора, регистрацию и авторизацию пользователя. Функциональные возможности проекта: возможность добавлять, удалять и редактировать записи. Есть разграничение прав доступа: обычный пользователь, администратор.

В случае интернет-магазина была возможность добавления товаров в корзину и составление заказа – с записью его в БД.

Ко всем страницам сайта должны были быть применены css стили, которые подразумевают настройку: отступов – внешних и внутренних, работу с фоном, цветом текста и шрифтом. Сайт обладал единой стилистикой, с наличием навигации на каждой из страниц.

Внедрение технологии AJAX

Необходимо реализовать возможность добавления, редактирования, удаления информации с сайта, без перезагрузки страницы. Должна быть реализована возможность редактирования любого поля карточки.

На стороне сервера должны быть реализованы файлы обработчики в формате php, которые взаимодействуют с MySQL.

При настройке запроса на сервер, необходимо настроить его таким образом, чтобы после получить ответ с данными от сервера. Данные с стороны сервера должны быть возвращены в формате JSON. Пакет JSON должен содержать не менее 4 полей. В JSON не должны передаваться теги с информацией.

Если была произведена операция добавления статьи, то минимум необходимо вернуть данные, которые прошли проверку на стороне сервера + доп. информацию о выполненной операции.

Если была произведена операция редактирования статьи, то минимум необходимо вернуть данные, что ранее находилось в карточке и какие данные записали в БД в процессе данной операции + доп. информацию о выполненной операции.

Если была произведена операция удаления статьи, то минимум необходимо указать заголовок удалённой статьи + доп. информацию о выполненной операции.

После получения ответа на стороне клиента, необходимо вывести информация о внесённых изменениях.

Если была добавлена новая карточка, то она должна отобразиться на странице (без перезагрузки), с сохранением стилистики страницы.

Если были внесены изменения в карточку, то в карточке необходимо отразить блок с ранее существовавшей информацией в карточке + новая информация, которая была добавлена на стороне сервера.

Если произведена операция удаления карточки, то необходимо вывести информацию о удалённых записях в доп. блок на странице, который будет служить доп. панелью для информирования пользователя о внесённых изменениях.

Перед тем как отправить данные на сторону сервера, необходимо осуществить проверку заполненности полей на стороне клиента с использованием JS.

Блок ответа должен быть оформлен с использованием css.

2. Проект на Node.js

Проект должен быть разработан на платформе node.js.

Обязательными к использованию модулями являются:

- express
- path
- fs
- url
- view-engine (ejs)
- mysql2/mongodb

В проекте должна использоваться база данных. Допускается работа с MongoDB или MySQL.

Должны быть реализованы запросы на: получение данных, добавление, изменение и удаление.

В процессе настройки ответа обязательно необходимо настроить следующие поля: - код ответа (200, 404, 301), Content-Type.

В проекте должна быть настроена маршрутизация. Для работы используются следующие методы: use, get, post, put, delete.

Проект должен обладать единой стилистикой, содержать минимум 5 страниц, на которых выводится минимум 5 блоков. Исполнители тегов для работы с текстом: b, i, u, p и т.д. – не считаются как использованный отдельный элемент.

Добавление 5 картинок подряд не рассматриваются к защите. Страницы должны иметь логину выводимой информации.

На каждой странице должен быть добавлен блок с навигацией по сайту.

Страница регистрации и авторизации пользователя должна быть реализована в виде одной страницы, где используется JS.

Если у Вас интернет-магазин – должны быть представлены карточки товаров + блоки с полезной для пользователя информацией.

Если информационный сайт, новостной – структурированные блоки с информацией и работа с текстом.

Карточка товара или карточка новости, статьи – должна содержать минимум 4 поля. Все поля карточки, можно редактировать/заполнять из панели администратора.

Ко всем страницам сайта должны быть применены css стили, которые подразумевают настройку: отступов – внешних и внутренних, работу с фоном, цветом текста и шрифтом.

Все промежуточные страницы, которые отвечают за ответ на запрос: удаления, редактирования, добавления – должны быть оформлены. Страница должна иметь кнопку возвращения в панель администратора, быть оформлена в соответствии с стилистикой сайта и на странице должно быть добавлено меню.

В момент добавления/редактирования информации, должна быть реализована проверка, что все поля заполнены корректно и, если пользователю все же удастся отправить запрос на сервер, на стороне сервера файл обработчик выведет пользователю сообщение с информацией, о том, что не все поля формы заполнены и необходимо до заполнить информацию.

В случае, если у Вас приложение, которое подразумевает планер задач или заметки, необходимо продумать следующий функционал:

- устанавливать задачи нескольким пользователям;
- устанавливать категорию и приоритет задачи;
- устанавливать видимость задачи (только для данного пользователя, всей команде, всем пользователям);
- менять статус задачи: в работе, заморожена, выполнена;
- добавлять/удалять/изменять задачи;
- выполненные задачи добавляются в отдельный раздел «Выполненные задачи».

Работы, которые из себя представляют по функциональным возможностям лишь выполненные на практиках практические задания, без оформления и доработки – будут сниматься с защиты.

Должно быть реализовано минимум 5 страницы:

1 страница – панель администратора (см. описание **«Раздел администратора»**)

2 страница – главная страница (см. описание **«Раздел Главная страница»**)

3 страница – форма для обратной связи (см. описание **Раздел «Обратная связь»**)

4 страница – регистрации и входа в систему (см. описание **«Раздел регистрации и входа в систему»**)

5 страница – страница 404 (см. описание **«Страница 404»**)

Раздел «администратора»

В данном разделе пользователь имеет доступ к меню, а также к панели управления контентом. На данной панели, администратор выбирает, что он хочет далее сделать: добавить, отредактировать, удалить.

Это возможно реализовать через следующие элементы: выпадающий список, checkbox, radio, кнопки. Или же, внедрить данную логику через события и привязать их к элементам.

Данные действия должны быть интуитивно понятный для пользователя.

При нажатии на раздел **«добавить информацию»** - на странице скрываются все остальные блоки и появляется блок, который позволяет добавить новую статью или карточку товара (зависит от того, какое приложение Вы реализуете).

При нажатии на раздел **«редактировать информацию»** - на странице скрываются все остальные блоки и появляется блок с карточками товаров/статей, где пользователь может двойным кликом мышки (или через кнопку редактировать), выбрать конкретную карточку, а далее приступить к редактированию её содержимого.

При нажатии на раздел **«удалить информацию»** - на странице скрываются все остальные блоки и появляется блок с карточками

товаров/статей, где пользователь может выбрать одну или несколько карточек и отправить запрос на удаление.

Должны редактироваться все блоки, созданные под карточку (статьи, новости, товара).

Во всех 3-х случаях, допускается реализация отправки информации без использования технологии Ajax.

Раздел «Главная страница»

На главной странице необходимо предоставить кратко информацию о возможностях Вашего проекте + кнопка, для скрытия данного блока.

Также на странице должен находиться блок для фильтрации карточек товаров/статей. Фильтрация осуществляется на стороне клиента, без отправки запроса на сервер + есть кнопка, показать все записи.

Также на главной странице есть доступ к панели меню.

Раздел регистрации и входа в систему

На данной странице, необходимо реализовать блок, где пользователь может ввести свой логин и пароль, а в случае, если у него нет аккаунта, добавить кнопку/ссылку «Регистрация».

При клике по данной кнопке/ссылке, на странице открывается блок с поля, для регистрации нового пользователя.

До того момента, пока пользователь не заполнит все поля формы, кнопка «Отправить» неактивна.

Если пользователь правильно ввёл логин и пароль, открывается доступ на остальные страницы сайта. В ином случае, доступ закрыт и настроена переадресация на страницу входа.

Страница 404

Тематически оформленная страница с добавлением меню и ссылкой на главную страницу. Код страницы должен быть 404.

Раздел «Обратная связь»

На данной странице необходимо вывести форму обратной связи для пользователя.

Минимальное кол-во полей – 3.

Пока пользователь не заполнил все поля формы, кнопка «отправить» неактивна. После того, как пользователь отправил форму, данные записываются в БД, а на экране выводится сообщение с информацией о том, что заявка принята.