

Клиент-сайд

Список книг

Собрать страницу с нижеописанным функционалом:

Слева форма для добавления новой книги, справа - список книг. В форме должны быть поля: автор, год издания, название книги, кол-во страниц. В списке - автор и название книги. У каждого элемента в списке книг должны быть кнопки edit и remove. При remove книга удаляется из списка, при edit - ее данные подставляются в форму для редактирования. При добавлении, редактировании и удалении страница не должна перезагружаться.

В качестве хранилища нужно использовать localStorage (при закрытии окна браузера и повторном открытии - данные должны восстанавливаться). Можно использовать любые css-библиотеки, из js - нельзя использовать MV*-библиотеки. Страница должна быть адаптивной (на мобильных устройствах (при недостаточной ширине окна браузера) список книг должен уходить под форму добавления). Страница должна открываться с диска в браузере.

Сервер-сайд с базовым знанием клиент-сайд

Многопользовательский чат

Необходимо реализовать многопользовательский чат. Сервером должен служить node.js процесс, запущенный в одном экземпляре и обрабатывающий запросы из браузера. node.js не должен использовать базу данных, всю информацию нужно обрабатывать в памяти процесса.

При первом обращении из браузера к node.js-процессу - должна произойти авторизация (простой ввод имени в форму), после чего любым способом создаётся временная сессия (до закрытия окна браузера) и пользователь видит перед собой экран, состоящий из трёх зон:

- основное окно (содержит сообщения участников чата)
- список присутствующих собеседников (справа от основного окна)
- строка ввода нового сообщения (снизу от основного окна чата)

также нужно реализовать кнопки:

- отправки сообщения (также происходит по нажатию Enter в строке ввода)
- logout (выход)

После входа пользователь видит только новые сообщения, историю сообщений до появления пользователя хранить и отображать не нужно. Список пользователей должен отражать в реальном времени изменения в составе участников чата (реагировать на вошёл/вышел/закрыв окно браузера).

Введённое сообщение должен увидеть каждый участник диалога, в т.ч. автор в виде:

[15:28:31] Username: Текст сообщения

Элементы интерфейса в сумме должны занимать всю экранную площадь, реагировать на изменение размеров окна браузера. Для оформления и организации адаптивности приветствуется применение bootstrap или других аналогичных css-библиотек, интерфейс в целом должен выглядеть аккуратно.

Для организации js-кода должен быть использован jQuery, применение MV* фреймворков не допускается.

Чат должен работать в современной версии браузера Google Chrome, поддержка кроссбраузерности не требуется.