

# Лабораторная работа 7

---

## Создание своего простого клиент серверного приложения

### Задание

Написать клиент серверное приложение с использованием express на сервере, чистого JS на клиенте и node-telegram-bot-api на сервере телеграм бота.

#### Клиентское приложение:

Необходима создать js файл, который будет загружаться с помощью вызова из html файла. Всё взаимодействие с интерфейсом необходимо сделать через и добавление нативных функций "createElement()", ".append()" и других функций напрямую взаимодействующих с DOM-деревом. HTML документ должен рендериться при обращении по url "<ваш ip>".

#### Серверное приложение:

Необходимо отдавать .html, .js, .css и другие статические документы через серверное приложение при обращении к "<ваш ip>/<путь до документа>". Все запросы к api должны начинаться с /api через [отдельные сабруты](#) Данные приложения необходимо хранить в любом из двух вариантов: - Хранить изначально в файлике, при запуске сервера считывать данные и помещать их в объект JS. Дальнейшие манипуляции с данными на сервере происходят с JS объектом. При желании можно добавить сохранение данных в файлик при отключении сервера. - Хранить в созданной БД и обращаться к ней для получения и изменения данных.

#### Telegram сервер:

Необходимо дублировать весь функционал веб приложения также и через телеграм бота.

### Дополнительная информация

[хороший урок по nodeJS + ExpressJS](#)

### Терминология

CRUD (Create, Read, Update, Delete) - создание, чтение, обновление и удаление данных.

### Задания по вариантам

Вариант	Задание
1	Приложение сохранения картинок. Реализовать CRUD операции над картинками, также отображение всех картинок на отдельной веб странице.
2	Приложения заметок написания заметок. Через веб версию и телеграмм можно добавить, удалить, посмотреть или поменять заметку, а также посмотреть список всех заметок

Вариант	Задание
3	Приложение новостного сайта с telegram ботом для показа новостей. В приложении можно добавлять, удалять, редактировать, смотреть новость и список новостей, как в веб версии так и через бота
4	Приложение которое сохраняет прогноз погоды у себя отображает его в телеграмме и фронт приложения.
5	Приложение для сохранения всяких ссылок допустим видео на YouTube с телеграмм ботом. Реализовать CRUD.
6	Приложение с рецептами. Web + Telegram. CRUD + список любимых рецептов.
7	Список дел. Web + Telegram. CRUD + список выполненных и не выполненных дел.
8	Сервис напоминаний о событиях. Web + Telegram. CRUD + напоминания.
9	Маленькое CRM приложение для магазина продуктов. Web + Telegram. Реализовать CRUD + отображение списка продуктов в виде таблицы на grid для web версии, их наличия, стоимости и покупках. Через телеграмм бота реализовать механизм покупки без денег (игрушечный)
10	Приложение которое отображает стоимость активов, в котором можно добавить свои активы и через telegram бот оповещает, когда стоимость достигла вершины. В качестве списка активов взять популярные криптовалюты и по api отслеживать их динамику. Web + Telegram. CRUD + оповещения и список всех активов.
11	Telegram бот - посредник для ведения интернет магазина. Через приложение или другого телеграмм бота можно отправлять сообщения клиентам, общающимся с телеграмм ботом магазина. Web + Telegram.
12	Система отслеживания доходов и расходов. Web + Telegram. Реализовать CRUD.
13	Приложение для автоматизации очередей в Бауманскую библиотеку с возможностью записаться на получение книг и их сдачу. Web + Telegram. Реализовать CRUD.
14	Маленькое CRM приложение для кинотеатра. Web + Telegram. Реализовать CRUD + отображение списка билетов в виде таблицы на grid для web версии, их наличия и покупках. Через телеграмм бота реализовать механизм покупки без денег (игрушечный)
15	Маленькое CRM приложение для магазина книг. Web + Telegram. Реализовать CRUD + отображение списка книг в виде таблицы на grid для web версии, их наличия, стоимости и покупках. Через телеграмм бота реализовать механизм покупки без денег (игрушечный)
16	Приложение букинга квартир и домов. Web + Telegram. Реализовать CRUD + отображение списка жилищ в виде таблицы на grid для web версии. Реализовать механизм бронирования