

Разработка системы проведения опросов аудитории во время публичных выступлений

Выпускная квалификационная работа

Е.А. Тактаров

Научный руководитель — к.ф.-м.н., доцент Е. М. Андреева

Южный Федеральный Университет

Институт математики, механики и компьютерных наук им. И.И. Воровича
02.03.02 — Фундаментальная информатика и информационные технологии

Ростов-на-Дону — 2019

Содержание

- 1 Постановка задачи
- 2 Обзор инструментов разработки
- 3 Аспекты Реализации
 - Структура веб-приложения
 - Веб-фреймворк Express
 - Разработка модели данных
 - Использование WebSocket
 - Отложенный вызов с Debounce
 - Интерфейс пользователя
- 4 Результаты

Постановка задачи

Создать веб-сервис, отвечающий следующим требованиям:

- Функция создания и проведения опросов.
- Динамическое отображение результатов опроса на странице.
- Параллельное проведение нескольких опросов на одном развернутом веб-сервисе.
- Каждый опрос доступен по коротким ссылкам для голосования и просмотра результатов.
- Защита от вредоносного искажения результатов.
- Открытый исходный код под свободной лицензией.

Обзор инструментов разработки

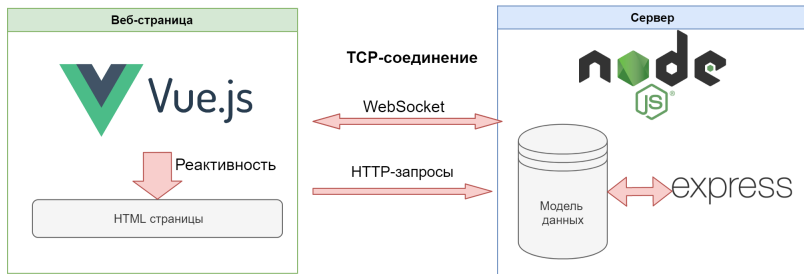
Node.js Программная платформа общего назначения для языка JavaScript.

Express Веб-фреймворк Node.js для создания серверной части веб-приложения.

WebSocket Протокол связи поверх TCP-соединения, предназначенный для обмена сообщениями между браузером и веб-сервером в режиме реального времени.

Vue.js Веб-фреймворк для создания пользовательского интерфейса в браузерах.

Структура веб-приложения



Использование Express

```
router.get("/get_polled/:poll",function(req, res, next){  
  let id = db.create_polling_session(req.params.poll);  
  res.status(200).json({link:db.sessions[id].link});  
});
```

Требования к модели данных

- 1 В модели может существовать неограниченное количество параллельных сессий, которые могут перемещаться между своими опросами.
- 2 Каждая сессия имеет две короткие ссылки для просмотра и участия в опросе.
- 3 Имея короткую ссылку, код должен уметь быстро переходить к данным о сессии, которой она принадлежит.
- 4 Код должен быстро получать список пользователей, показывающих опрос или в нем участвующих.
- 5 Пользователь может голосовать и перезагружать страницу неограниченное число раз, не вызывая подтасовку результатов.
- 6 Модель должна быть устойчивой к добавлению новых видов взаимодействия пользователей с сервисом.

Разработка модели данных

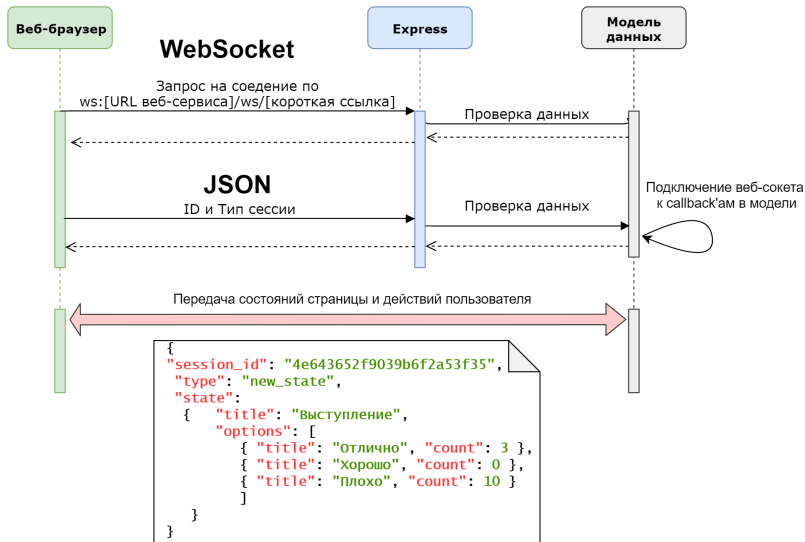
Особенности модели данных в приложении:

- JavaScript объект.
- Внутри рекурсивно списки, словари, объекты со свойствами и методами.
- Методы объектов инкапсулируют любое взаимодействие пользователей и модели.
- Каждый пользователь определяется объектом Websocket.
- События WebSocket'ов вызывают методы модели.
- Модель определяется и инициализируется в отдельном модуле **database.js**.
- Модуль экспортирует один объект модели в любой точке кода(аналогично синглтону).

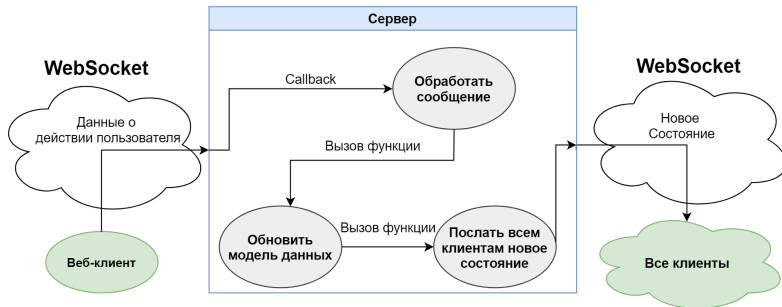
Использование WebSocket

- 1 Пакет **express-ws** добавляет в Express обработчики запросов на WebSocket-соединение.
- 2 Клиент запрашивает соединение по URL:
ws:[URL веб-сервиса]/ws/[короткая ссылка]
- 3 Данные отправляются в JSON.
- 4 Первое сообщение от клиента всегда содержит его идентификатор и тип сессии.
- 5 Если данные имеют неправильный формат или не совпадают с моделью данных, то сервер закрывает соединение.

Алгоритм соединения через WebSocket



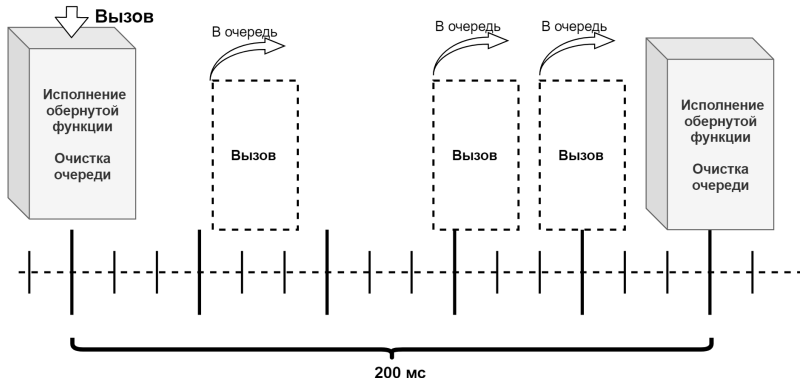
Проблема переполнения запросов



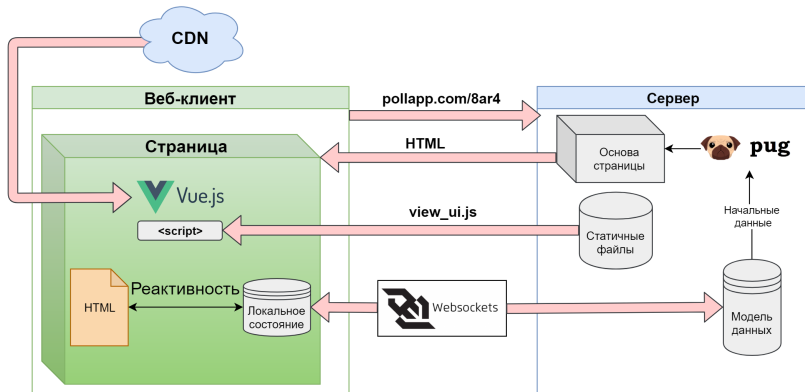
Что будет если 10 пользователей проголосуют одновременно?
Каждому пользователю придет 10 сообщений с новым состоянием, и только последнее из них будет актуальным.

Использование отсроченного вызова

Debounce пакет для Node.js, предоставляющий обертку для функций, которая откладывает их исполнение на указанный промежуток.



Устройство клиентской части



Со стороны пользователя



Оцените это выступление

ПЛОХО

СРЕДНЕЕ

ХОРОШО

ОТЛИЧНО

Полученные результаты

- Веб-сервис позволяет создавать, проводить и управлять опросами.
- Одновременно может проводится неограниченное количество опросов.
- Пользователи могут сразу же поучаствовать в опросе со своего мобильного устройства, перейдя по короткой ссылке, указанной на экране опроса.
- Результаты голосования отображаются динамически на странице просмотра результатов.
- Реализована основная защита от вредоносного вмешательства в процесс опроса, не позволяющая испортить результаты или остановить ход опроса.
- Веб-сервис имеет открытые исходные коды на сайте **GitHub.com** и доступен для развертывания, использования и модификации по свободной лицензии MIT.