

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Фронт-энд разработка

Отчет

Лабораторная работа 2

Выполнили:  
Дмитриева Лина  
Анатольевна

Мильберг Кристина  
Алексеевна

Группа К3342, К3344

Проверил:  
Добряков Д. И.

Санкт-Петербург

2024 г.

## Задача

Теперь Вам нужно привязать то, что Вы делали в ЛР1 к внешнему API средствами fetch/axios/xhr. Реализуйте моковое API средствами JSON-сервера и подключите к нему авторизацию, как в примерах, которые мы рассматривали в рамках тем "Имитация работы с API".

Например, для приложения для просмотра прогнозов погоды задание выглядит следующим образом:

Реализовать получение погоды (прогноз на ближайшие 7 дней) из открытого API OpenWeatherMap, в зависимости от геолокации пользователя. Реализовать вывод полученного прогноза в виде 7 карточек в три ряда (первый ряд - крупная карточка, второй ряд - три карточки в меньшем размере, третий ряд - четыре карточки в маленьком размере).

Выбранный и согласованный вариант:

Сервис для предсказания будущего и раскрытия тайн 🌌

1. Вход
2. Регистрация
3. Личный кабинет пользователя (личные данные, нужные для каждого запроса)
4. Поиск информации про арканы
5. Поиск натальной карты + фильтрация по планетам
6. Поиск матрицы судьбы + расшифровка
7. Поиск матрицы совместимости + расшифровка
8. Информационная страница

## Ход работы

### 1. Создание mock API

Для выполнения задания был настроен файл db.json, который содержит следующие данные:

Пользователи: реализован функционал для авторизации по логину и паролю.

Данные приложения: для примера, использованы прогнозы гороскопов.

Запуск JSON-server осуществлялся с помощью команды:

```
json-server --watch db.json --port 3000
```

### 2. Реализация интерфейса авторизации и регистрации

На следующем этапе работы была разработана веб-страница с формами авторизации и регистрации пользователей. Для оформления использовался фреймворк Bootstrap. Основной функционал реализован с помощью JavaScript.

#### Описание функционала:

- **Переключение между формами:**

На странице предусмотрены две формы: для авторизации и регистрации. Пользователь может переключаться между ними с помощью ссылок.

- **Авторизация:**

Пользователь вводит логин и пароль.

Скрипт проверяет данные с помощью запроса к эндпоинту `http://localhost:3000/users`.

При успешной проверке выводится сообщение об успешном входе, и пользователь перенаправляется на другую страницу (`profile.html`).

- **Регистрация:**

Пользователь вводит логин и пароль для нового аккаунта.

Скрипт проверяет, существует ли пользователь с таким именем в базе.

Если такого пользователя нет, то создается новая запись через POST-запрос.

- **Профиль:**

С авторизации пользователя перекидывает на страницу профиля

Имя введено из формы регистрации (подтягивается по id пользователя)

Далее пользователь заполняет личную информацию, необходимую далее при раскладах

Введенная информация добавляется в db.json и сохраняется при дальнейшем входе в профиль.

## **Вывод**

Реализована регистрация и авторизация через JSON сервер, теперь авторизация работает полноценно и записывает данные в базу данных. Кроме того через id пользователя соединены страница авторизации и профиля пользователя, данные из него также сохраняются в базу данных.