

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа №3

Выполнил:

Акулов Даниил

К3439

Проверил:

Добряков Д. И.

Санкт-Петербург

2026 г.

Задача

Мигрировать ранее разработанное приложение (в рамках ЛР1 и ЛР2) на фреймворк Vue.JS.

Каким требованиям ваше приложение должно соответствовать:

Должен быть подключён роутер

Должна быть реализована работа с внешним API (желательно посредством axios)

Разумное деление на компоненты (продемонстрируйте понимание компонентного подхода)

Использование composable для выделения повторяющегося функционала в отдельные файлы

Ход работы

Установил необходимые зависимости. Настроил импорт стилей Bootstrap и кастомных стилей из fitnessApp/css/styles.css в src/assets/main.css .

Создал src/api/instance.js с настройкой Axios на <http://localhost:8080/api>.

Реализовал API сервисы:

- auth.js : вход и регистрация.
- workout.js: получение списка тренировок, одной тренировки, добавление тренировки пользователю.
- blog.js : получение списка постов и одного поста.

Создал src/stores/auth.js на Pinia для управления авторизацией (сохранение токена и пользователя в localStorage).

Компоненты и Layout:

- Создал src/layouts/BaseLayout.vue , который включает в себя:
- TheNavbar.vue: навигационная панель с логикой отображения ссылок в зависимости от авторизации.
- TheFooter.vue: подвал сайта.
- SVG иконки, используемые в проекте.

Views (Страницы):

- `HomeView.vue` (из `index.html`): главная страница с фильтрацией тренировок.
- `LoginView.vue` (из `login.html`): страница входа.
- `RegisterView.vue` (из `register.html`): страница регистрации.
- `DashboardView.vue` (из `dashboard.html`): личный кабинет пользователя со списком добавленных тренировок.
- `WorkoutView.vue` (из `workout.html`): детальная страница тренировки с возможностью добавить её в свой план.
- `BlogView.vue` (из `blog.html`): список статей блога.
- `BlogPostView.vue` (из `blogpost.html`): детальная страница статьи.

Роутинг:

- Настроил `src/router/index.js` с маршрутами для всех страниц.
- Добавил проверку авторизации (`requiresAuth`) для страницы `Dashboard`.

Код создания инстансов апи:

```
import axios from "axios";

const apiURL = "http://localhost:8080/api";

const $host = axios.create({
  baseURL: apiURL,
})

const $authHost = axios.create({
  baseURL: apiURL,
})

const authInterceptor = config => {
  config.headers.authorization = 'Bearer ' + localStorage.getItem('token')
  return config
}

$authHost.interceptors.request.use(authInterceptor)
```

```
export {  
  $host,  
  $authHost,  
}
```

Код файла [main.js](#):

```
import { createApp } from "vue";  
  
import App from "@App.vue";  
import router from "@router";  
import store from "@stores";  
  
import "bootstrap/dist/css/bootstrap.min.css";  
import "bootstrap";  
import "@assets/main.css";  
  
const app = createApp(App);  
  
app.use(store);  
app.use(router);  
  
app.mount("#app");
```

Вывод

В ходе работы перенес HTML сайт в Vue приложение.