

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа №3

Выполнил:

Захарчук Александр

Группа К3441

**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 повторяющиеся функционала в отдельные файлы

Ход работы

В качестве API для данной работы использовался сервис рецептов, разработанный в прошлом семестре в рамках бэкенд разработки.

На странице регистрации была добавлена отправка данных из формы на бэкенд по нажатию кнопки. После успешной регистрации происходит редирект на страницу логина.

The screenshot shows a registration form titled "Создать аккаунт" (Create account). The form is part of a website for a culinary community. At the top, there is a header with a search icon, the text "Кулинарное сообщество", and navigation links for "Главная", "Поиск рецептов", "Вход", and "Регистрация". The main form area has a light gray background and contains the following fields:

- "Имя пользователя *": A text input field containing "ivan". Below it, a note says "Это имя будет отображаться в вашем профиле".
- "Email *": A text input field containing "ivan@mail.ru".
- "Ссылка на фото": A placeholder text "Ваше фото" in a text input field.
- "Ваша биография": A text input field containing "My awesome bio".
- "Пароль *": A text input field with several dots indicating the password. Below it, a note says "Минимум 6 символов".
- "Подтверждение пароля *": A text input field with several dots indicating the confirmation password.
- A checkbox labeled "Я соглашаюсь с [условиями использования](#)".
- A large red button at the bottom right labeled "Зарегистрироваться" (Register).

Рисунок 1 - Страница регистрации

Код, обрабатывающий регистрацию, показан на рисунке 2.

```
<script setup>
import FormContainer from "@/components/FormContainer.vue";
import Navbar from "@/components/Navbar.vue";
importFormField from "@/components/FormField.vue";
import { ref } from "vue";
import { useUserStore } from "@/stores/user";

const form = ref({
  username: "",
  email: "",
  imageLink: "",
  bio: "",
  password: "",
  confirmPassword: "",
  termsAccepted: false,
});

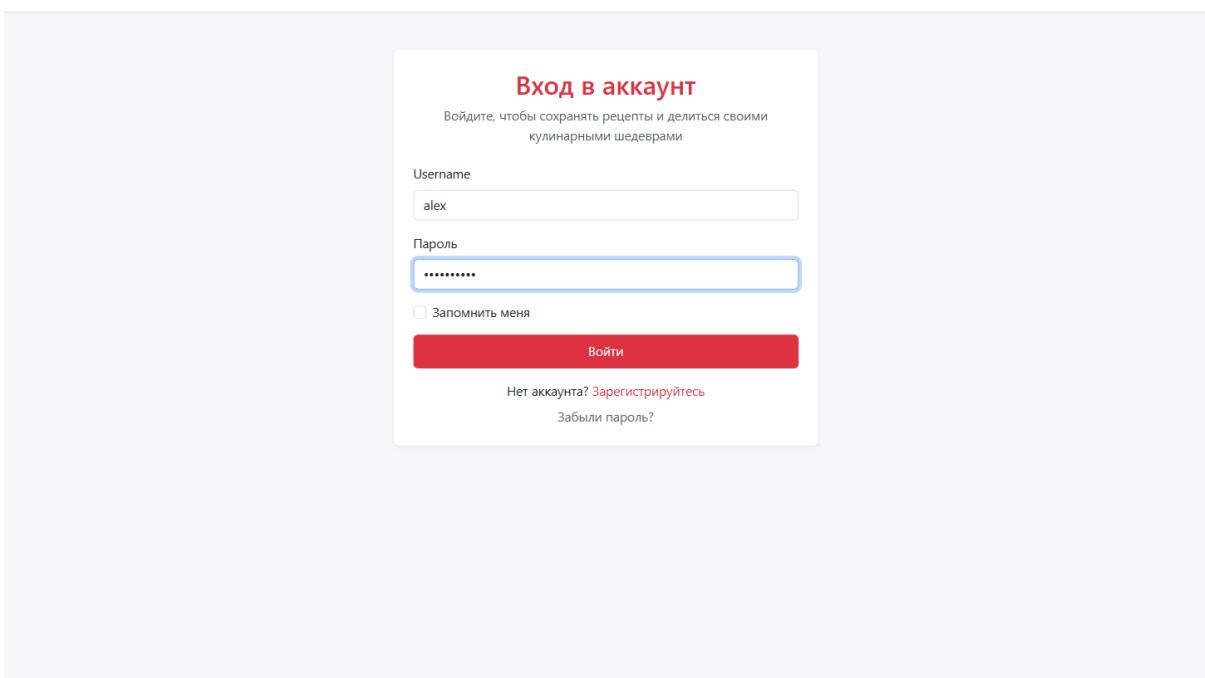
const userStore = useUserStore();

const handleRegister = async () => {
  if (form.value.password !== form.value.confirmPassword) {
    return;
  }
  try {
    await userStore.register(form.value.username, form.value.email, form.value.password, form.value.imageLink);
  } catch (error) {
    console.error("Login failed:", error);
  }
};
</script>
```

Рисунок 2 - Код регистрации

На странице логина была добавлена обработка формы. После ввода логина и пароля отправляется запрос на бэкенд. В случае успешного входа полученный токен сохраняется в localstorage.

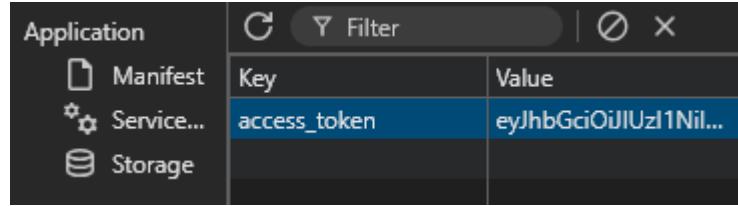
Страница логина показана на рисунке 3.



The screenshot shows a login form titled "Вход в аккаунт" (Login). The form asks the user to log in to save recipes and share their culinary masterpieces. It includes fields for "Username" (containing "alex") and "Пароль" (containing "*****"). There is a "Запомнить меня" (Remember me) checkbox, which is unchecked. A red "Войти" (Enter) button is at the bottom. Below the form are links for "Нет аккаунта? Зарегистрируйтесь" (No account? Register) and "Забыли пароль?" (Forgot password?).

Рисунок 3 - Страница логина

Токен доступа, сохраненный в localStorage, показан на рисунке 4.



Application	Key	Value
Manifest	access_token	eyJhbGciOiJIUzI1Ni...

Рисунок 4 - Токен в localStorage

Код, обрабатывающий форму логина показан на рисунке 5.

```

<script setup>
import FormContainer from "@/components/FormContainer.vue";
import FormField from "@/components/FormField.vue";
import Navbar from "@/components/Navbar.vue";
import { useUserStore } from "@/stores/user";
import { ref } from "vue";

const form = ref({ username: "", password: "" });

const userStore = useUserStore();

const handleLogin = async () => {
  try {
    await userStore.login(form.value.username, form.value.password);
  } catch (error) {
    console.error("Login failed:", error);
  }
};

</script>

```

Рисунок 5 - Код логина

На странице поиска рецептов была реализована загрузка информации с сервера. При открытии страницы на бэкенд отправляется запрос, данные из которого отрисовываются на странице. Страница поиска рецептов показана на рисунке 6.

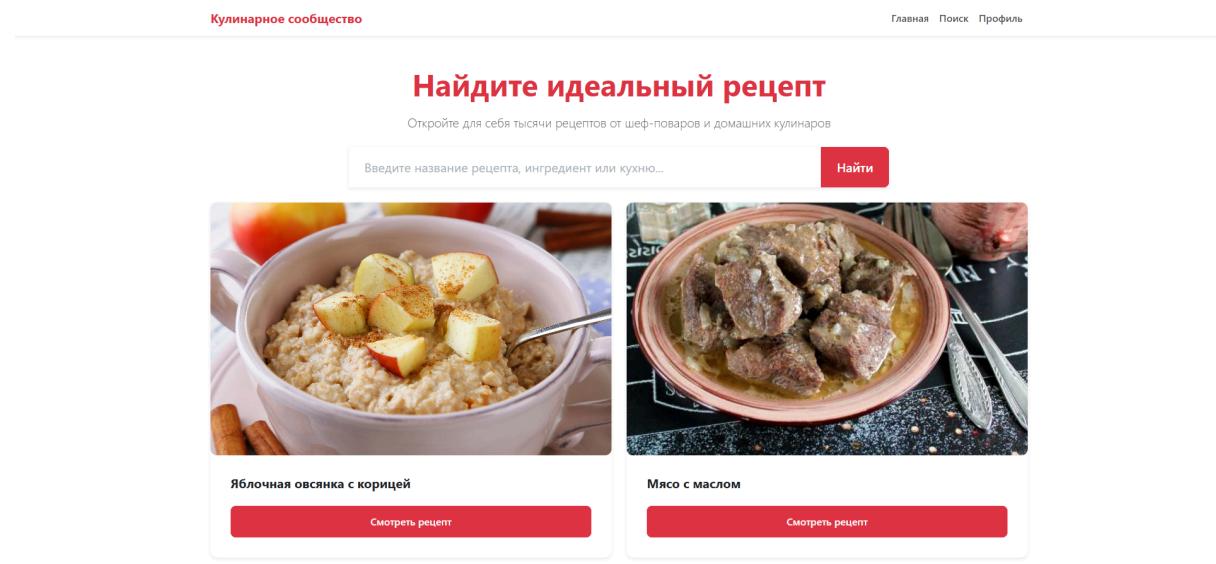


Рисунок 6 - Страница поиска рецептов

Код, который загружает рецепты с сервера и отрисовывает их, показан на рисунке 7.

```
<script setup>
import Navbar from "@/components/Navbar.vue";
import RecipeCard from "@/components/RecipeCard.vue";
import { useRecipeStore } from "@/stores/recipe";
import { ref } from "vue";

const searchText = ref();
const recipeStore = useRecipeStore();

const handleSearch = async () => {
  const filter = searchText.value.trim();

  if (filter) {
    await recipeStore.load(filter);
  }
};

</script>
```

Рисунок 7 - Код поиска рецептов

По нажатию на какой-либо рецепт происходит переход на страницу с информацией об этом рецепте. Информация о нем также запрашивается с сервера. Идентификатор рецепта извлекается из адреса страницы.

Страница рецепта показана на рисунках 8 и 9.

Куриный салат с овощами



Ингредиенты

- Куриная грудка - 300 г
- Помидор - 2 piece
- Огурец - 1 piece
- Лук репчатый - 0.5 piece
- Оливковое масло - 2 tbsp
- Соль - 0.5 tsp
- Черный перец - 0.25 tsp
- Лимон - 1 tbsp

Информация о рецепте

Общее время	15 минут
Активное время	20 минут
Тип блюда	salad
Сложность	easy

Рисунок 8 - Страница рецепта

Постапное приготовление

1. Куриную грудку промойте и обсушите бумажным полотенцем. Посолите и поперчите с обеих сторон.
2. Разогрейте сковороду с 1 столовой ложкой оливкового масла. Обжарьте курицу на среднем огне по 7-8 минут с каждой стороны до золотистой корочки и полной готовности. Дайте остыть и нарежьте кубиками.
3. Помидоры и огурцы вымойте, нарежьте кубиками среднего размера. Лук очистите и мелко нарежьте.
4. В большой миске смешайте все нарезанные ингредиенты.
5. Приготовьте заправку: смешайте оставшееся оливковое масло, лимонный сок, соль и перец.
6. Заправьте салат, аккуратно перемешайте. Дайте настояться 10 минут перед подачей.

Комментарии

Минимум 10 символов

Пользователь #1

Получилось прекрасно, буду готовить еще раз!

Рисунок 9 - Страница рецепта

В нижней части страницы отображаются комментарии к рецепту, которые также загружаются с сервера. Есть возможность добавить собственный комментарий, который сразу же отобразится на странице и сохранится в базе данных на бэкенде.

Код страницы рецепта показан на рисунке 10.

```
onMounted(async () => {
  try {
    await recipeStore.loadById(recipeId);
    await commentStore.getByRecipeId(recipeId);
    await userStore.getAll();
  } catch (error) {
    console.error("Ошибка загрузки рецепта:", error);
  }
});

const commentText = ref();

const getCommentAuthor = (userId) => {
  const user = userStore.getById(userId);
  if(user) {
    return user.username;
  }
  return "Guest";
};

const handleCommentCreate = async () => {
  const text = commentText.value.trim();
  if (text && text.length >= 10) {
    await commentStore.create(recipeId, text);
  }
};
```

Рисунок 10 - Код страницы рецепта

Авторизованный пользователь может перейти на страницу своего профиля, на которой будет отображаться информация о нем. Если пользователь не авторизован, то он будет перенаправлен на страницу логина. Страница профиля показана на рисунке 11.

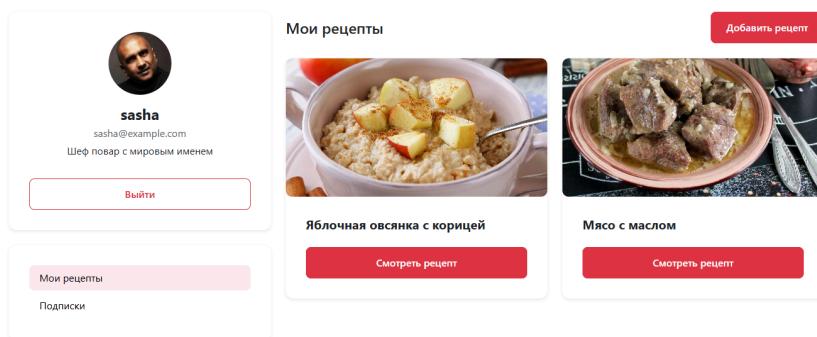


Рисунок 11 - Страница пользователя

Код страницы пользователя показан на рисунке 12.

```
<script setup>
import Navbar from "@/components/Navbar.vue";
import { useUserStore } from "@/stores/user";
import { onMounted } from "vue";
import router from "@/router";
import UserCard from "@/components/UserCard.vue";
import { useRecipeStore } from "@/stores/recipe";
import RecipeCard from "@/components/RecipeCard.vue";

const userStore = useUserStore();
const recipeStore = useRecipeStore();

onMounted(async () => {
  try {
    await userStore.getCurrentUser();
    await userStore.getSubs();
    await recipeStore.load(null, userStore.currentUser.id);
  } catch (error) {
    console.error("Ошибка загрузки пользователя:", error);
    userStore.logout();
  }
});

if (!userStore.isAuthenticated) {
  router.push("/login");
}
```

Рисунок 12 - Код страницы пользователя

Вывод

В ходе выполнения данной лабораторной работы был изучен фреймворк Vue, а также выполнен перенос реализованного ранее приложения на него.