

**Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут”**

**Факультет прикладної математики  
Кафедра спеціалізованих комп’ютерних систем**

**Лабораторна робота № 3**  
з дисципліни «ВЕБ-дизайн»  
«Розробка Web-додатка засобами Javascript/VueJS»

Виконав:  
студент групи КВ-92  
**Іванов Максим**

Київ 2022

## Зміст

Мета: ознайомитись із засобами фреймворка VueJS та навчитись створювати асинхронні запити до Web-сервера.

Загальне завдання: адаптувати програмний код ЛР№2 до вимог фреймворка VueJS та забезпечити завантаження необхідних даних з Web-сервера.

Інструменти розробки: Javascript, VueJS, NodeJS, Express.

Зауваження: студент може обрати інший серверний фреймворк (наприклад, Django).

## Завдання

**Мій Telegram: @HYMPYT**

**Github-репозиторій з лабораторної:**

Фронтенд:

<https://github.com/HYMPYT/WEB/tree/main/Lab3/frontend>

Бекенд:

<https://github.com/HYMPYT/WEB/tree/main/Lab3/backend>

**Тема: Адаптація лабораторної роботи №2 до вимог Vue. Створення веб-серверу за допомогою NodeJS + Express.**

У цій лабораторній роботі було створено додаток за допомогою фреймворка Vue. Було використано vue-router для навігації та vuex для взаємодії з даними.

Клієнт взаємодіє з сервером за допомогою бібліотеки fetchAPI.

Сервер реалізовано за допомогою NodeJS + Express. Як сховище даних використовується SQLite3. Для більш простої взаємодії з базою даних використано ORM Sequelize.

## Скріншоти

# Hello there !

It's a great day to call your friends or relatives :)

Name	Phone	City, Country	Edit/Delete
Evgen Levchuk	+38099332233	Kyiv, Ukraine	<a href="#">Edit</a> <a href="#">Remove</a>
Oleg	+380(50) 185 - 64 - 48	Kherson	<a href="#">Edit</a> <a href="#">Remove</a>
sashok	8-(800)-555-3535	kiev	<a href="#">Edit</a> <a href="#">Remove</a>

[Didn't find user? Add him!](#)

# Hello there !

It's a great day to call your friends or relatives :)

## Change contact

**Name**  
test

**Phone number**  
111

**City, Country**  
test

[Change contact](#)

[Back to contacts](#)

```

Executing (default): DELETE FROM `items`
Executing (default): INSERT INTO `items` (`id`,`name`,`tel`,`place`) VALUES (NULL,$1,$2,$3);
Executing (default): INSERT INTO `items` (`id`,`name`,`tel`,`place`) VALUES (NULL,$1,$2,$3);
Executing (default): INSERT INTO `items` (`id`,`name`,`tel`,`place`) VALUES (NULL,$1,$2,$3);
get
Executing (default): SELECT `id`,`name`,`tel`,`place` FROM `items` AS `items`;
get
Executing (default): SELECT `id`,`name`,`tel`,`place` FROM `items` AS `items`;
post
[
  {
    id: 71,
    name: 'Evgen Levchuk',
    tel: '+38099332233',
    place: 'Kyiv, Ukraine'
  },
  {
    id: 72,
    name: 'Oleg',
    tel: '+380(50) 185 - 64 - 48',
    place: 'Kherson'
  }
]
Executing (default): DELETE FROM `items`
Executing (default): INSERT INTO `items` (`id`,`name`,`tel`,`place`) VALUES (NULL,$1,$2,$3);
Executing (default): INSERT INTO `items` (`id`,`name`,`tel`,`place`) VALUES (NULL,$1,$2,$3);
get
Executing (default): SELECT `id`,`name`,`tel`,`place` FROM `items` AS `items`;

```

Акти  
Чтоб

## Код

### contact.vue

```

<template>
  <tr>
    <td>{{ contact.name }}</td>
    <td>{{ contact.tel }}</td>
    <td>{{ contact.place }}</td>
    <td class="d-none d-md-block">
      <button @click="toEdit()" type="button" class="btn btn-primary">
        Edit
      </button>
      <button @click="remove()" class="btn btn-warning remove">Remove</button>
    </td>
  </tr>
</template>

<script>
export default {
  props: {
    contact: { type: Object },
    id: { type: Number },
  },
  methods: {
    remove() {
      console.log(this.id);
      this.$store.dispatch("REMOVE_CONTACT", this.id);
    },
    toEdit() {
      //console.log(this.contact);
      this.$router.push({ name: "edit", params: { id: this.id } });
    },
  },
};
</script>

```

## model.js

```
export default class Model {
  constructor() {
    this.currentUser = JSON.parse(localStorage.getItem("currentUser"));
  }

  getContact(id) {
    return this.currentUser ? this.currentUser.contacts[id] : null;
  }

  getContacts() {
    return this.currentUser ? this.currentUser.contacts : null;
  }

  async loadUser() {
    //this.currentUser = JSON.parse(localStorage.getItem("currentUser"));
    // console.log(this.currentUser);
    await this.getContactsFromServer();
  }

  async addContact(contact) {
    this.currentUser.contacts.push(contact);
    // localStorage.setItem("currentUser", JSON.stringify(this.currentUser));
    await this.sendContactsOnServer();
    await this.getContactsFromServer();
  }

  async editContact(edit) {
    this.currentUser.contacts[edit.id] = edit.edited;
    //localStorage.setItem("currentUser", JSON.stringify(this.currentUser));
    await this.sendContactsOnServer();
    await this.getContactsFromServer();
  }

  async removeContact(id) {
    console.log(id);
    this.currentUser.contacts = this.currentUser.contacts.filter(
      (_, idx) => idx !== id
    );
    //localStorage.setItem("currentUser", JSON.stringify(this.currentUser));
    await this.sendContactsOnServer();
    await this.getContactsFromServer();
  }

  async getContactsFromServer() {
    await fetch("http://localhost:3000/")
      .then((res) => res.json())
      .then((res) => {
        this.currentUser.contacts = res.items;
      });
  }

  async sendContactsOnServer() {
    fetch("http://localhost:3000/set", {
      method: "POST",
      body: JSON.stringify({ currentUser: this.currentUser }),
      headers: {
        "Content-Type": "application/json",
      },
    });
  }
}
```

## router.index.js

```
import { createRouter, createWebHistory } from "vue-router";
import Home from "../views/Home.vue";
```

```

import edit from "../views/edit.vue";
import addContact from "../views/addContact.vue";
const routes = [
  {
    path: "/",
    name: "Home",
    component: Home,
    props: { newsletterPopup: false },
  },
  {
    path: "/edit",
    name: "edit",
    component: edit,
    props: (route) => ({
      ...route.params,
    }),
  },
  {
    path: "/addContact",
    name: "addContact",
    component: addContact,
  },
];

const router = createRouter({
  history: createWebHistory(process.env.BASE_URL),
  routes,
});

export default router;

```

## store.index.js

```

import { createStore } from "vuex";
import model from "@model/model.js";
export default createStore({
  state: {
    model: new model(),
  },
  mutations: {
    ADD_CONTACT(state, contact) {
      state.model.addContact(contact);
    },
    EDIT_CONTACT(state, edit) {
      state.model.editContact(edit);
    },
    REMOVE_CONTACT(state, id) {
      state.model.removeContact(id);
    },
    LOAD_USER(state) {
      state.model.loadUser();
    },
  },
  actions: {
    ADD_CONTACT({ commit }, contact) {
      commit("ADD_CONTACT", contact);
    },
    EDIT_CONTACT({ commit }, edit) {
      commit("EDIT_CONTACT", edit);
    },
    REMOVE_CONTACT({ commit }, id) {
      commit("REMOVE_CONTACT", id);
    },
    LOAD_USER({ commit }) {
      commit("LOAD_USER");
    },
  },
  modules: {},
});

```

## addContact.vue

```
<template>
  <div class="container">
    <div class="rounded text-dark justify-content-center">
      <div
        class="rounded border border-info bg-light border-3 text-center fw-bold"
        action="../index.html"
        id="createForm"
      >
        <h2 class="display-4 fw-bold">Create contact</h2>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="name">Name </label>
          <input
            v-model="name"
            required
            placeholder="Name"
            type="text"
            id="name"
            class="form-control"
          />
        </div>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="phone">Phone number </label>
          <input
            v-model="tel"
            required
            placeholder="Phone number"
            type="tel"
            id="phone"
            class="form-control"
          />
        </div>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="city_country">City, Country </label>
          <input
            v-model="place"
            required
            placeholder="City, Country"
            type="text"
            id="city_country"
            class="form-control"
          />
        </div>
        <button
          @click="addContact(name, tel, place)"
          type="submit"
          class="btn btn-primary btn-block mb-4"
        >
          Create contact
        </button>
      </div>
    </div>
    <router-link to="/" class="fs-5 text-light mb-3">
      >Back to contacts</router-link>
  </div>
</template>

<script>
export default {
  name: "addContact",
  data() {
    return {
      name: "",
      tel: "",
      place: "",
    };
  },
  mounted() {
    this.$store.dispatch("LOAD_USER");
  },
}
```

```

methods: {
  addContact(name, tel, place) {
    console.log(name, tel, place);
    if (name.length && tel.length && place.length) {
      this.$store.dispatch("ADD_CONTACT", { name, tel, place });
      this.name = "";
      this.tel = "";
      this.place = "";
    } else {
      alert("Fill all inputs!");
    }
  },
},
};
</script>

```

## edit.vue

```

<template>
  <div class="container">
    <div class="rounded text-dark justify-content-center">
      <div
        class="rounded border border-info bg-light border-3 text-center fw-bold"
        id="createForm"
      >
        <h2 class="display-4 fw-bold">Change contact</h2>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="name">Name </label>
          <input
            v-model="name"
            required
            placeholder="Name"
            type="text"
            id="name"
            class="form-control"
          />
        </div>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="phone">Phone number </label>
          <input
            v-model="tel"
            required
            placeholder="Phone number"
            type="tel"
            id="phone"
            class="form-control"
          />
        </div>
        <div class="form-outline mb-4 fs-3">
          <label class="form-label" for="city_country">City, Country </label>
          <input
            v-model="place"
            required
            placeholder="City, Country"
            type="text"
            id="city_country"
            class="form-control"
          />
        </div>
        <button
          @click="editContact(name, tel, place)"
          type="submit"
          class="btn btn-primary btn-block mb-4"
        >
          Change contact
        </button>
      </div>
    </div>
    <router-link to="/" class="fs-5 text-light mb-3"

```



```

    >Back to contacts</router-link
  >
</div>
</template>
<script>
export default {
  name: "edit",
  props: ["id"],
  data() {
    return {
      name: "",
      tel: "",
      place: "",
    };
  },
  methods: {
    editContact(name, tel, place) {
      const edit = { id: this.id, edited: { name, tel, place } };
      this.$store.commit("EDIT_CONTACT", edit);
    },
  },
};
</script>

```

## Home.vue

```

<template>
  <div>
    <table
      class="table-bordered rounded table table-light"
      v-if="$store.state.model.currentUser.contacts != null"
    >
      <thead>
        <tr>
          <th scope="col">Name</th>

          <th scope="col">Phone</th>
          <th scope="col">City, Country</th>
          <th class="d-none d-md-block" scope="col">Edit/Delete</th>
        </tr>
      </thead>
      <tbody>
        <contact
          v-for="(contact, id) in $store.state.model.currentUser.contacts"
          v-bind:key="contact"
          v-bind:contact="contact"
          v-bind:id="id"
        />
      </tbody>
    </table>
    <div v-else>Any contacts</div>
    <router-link to="/addContact" class="fs-5 text-light mb-3"
      >Didn't find user? Add him!</router-link>
  >
</div>
</template>

<script>
// @ is an alias to /src
import contact from "../components/contact.vue";

export default {
  components: {
    contact,
  },
  created() {
    this.$store.dispatch("LOAD_USER");
  }
};

```

```
},  
});  
</script>
```

## App.vue

```
<template>  
  <div>  
    <router-view />  
  </div>  
</template>  
  
<style>  
</style>
```

## main.js

```
import { createApp } from "vue";  
import App from "./App.vue";  
import router from "./router";  
import store from "./store";  
  
createApp(App).use(store).use(router).mount("#app");
```