# САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа 3

Выполнил:

Старовойтова Елизавета

Группа:

K33402

Проверил: Добряков Д. И.

Санкт-Петербург

2023 г.

### Задача

Task manager: нужно реализовать фронтенд на vue.js для приложения по управлению задачами. Бэкенд реализован на Django на дисциплине Вебразработка.

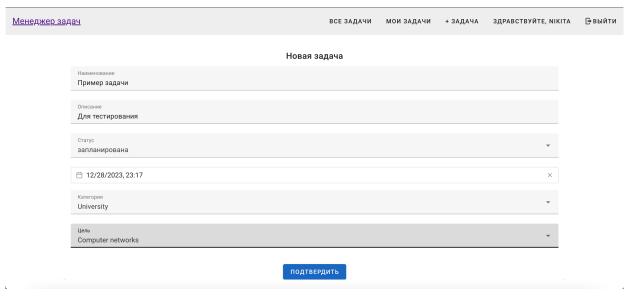
# Ход работы

В работе использовался vue-router для маршрутизации. Vuex для управления состоянием и централизованного хранилища данных для всех компонентов. Axios для HTTP-запросов. Vuetify для UI.

Реализован следующий функционал:

- Вход/регистрация и выход
- Просмотр списка всех задач и каждой задачи
- Просмотр мои задач и управление статусом задачи
- Создание новой задачи
- Назначение задачи исполнителю
- Возможность оставить/удалить комментарий
- Удаление задачи

Продемонстрирую работу в браузере, и пройдемся по коду на примере реализации создания новой задачи.



1. Poyrep (router/index.html)

Тут настройка маршрута для создания новой задачи и при переходе по адресу /add\_task будет отображаться компонент CreateTask.vue.

#### 2. Компонент (CreateTask.vue)

```
<v-alert
   type="error"
   <v-card-title>Новая задача</v-card-title>
   <v-card-text>
      <v-text-field
       outlined
        type="text"
      ></v-text-field>
      <v-select
        v-model="completed"
      <VueDatePicker class="mb-4" outlined required placeholder="Cpok</pre>
       outlined
        v-model="aim"
```

```
label="Цель"
    outlined
    required
    item-value="id"
    ></v-select>
    </v-card-text>
    <v-flex class="text-xs-center" mt-5>
        <v-btn color="primary" type="submit">Подтвердить</v-btn>
        </v-flex>
        </v-card>
        </v-form>
        </v-responsive>
        </v-container>
</template>
```

Код компонента, который предоставляет форму для создания новой задачи. Используются компоненты vuetify.

#### 3. Создание задачи (methods)

```
methods: {
    handleCreateTask() {
        this.loading = true;
        let formattedDate = ""

        if (this.deadline !== "" && this.deadline !== null)
            formattedDate = this.deadline.toISOString().split('T')[0] + ' ' +

this.deadline.toTimeString().split(' ')[0];

let newTask = {
        title: this.title,
        description: this.description,
        deadline: formattedDate,
        completed: this.completed,
        category: this.astegory,
        aim: this.aim
    }

this.$store.dispatch("task/addTask", newTask).then(
        () => {
            this.frouter.push("/tasks");
        ).catch(error => {
            this.loading = false;
            if (error.response && error.response.data &&
            error.response.data.message) {
            this.errorMessages.push(error.response.data);
        } else {
            const errors = Object.values(error.response.data).flat();
            errors.forEach(errorMessage => {
                  this.errorMessages.push(errorMessage);
            });
        });
    }
});
});
```

В этом методе обрабатываем создание задачи. Данные из формы собираются в объект newTask, и вызывается действие из хранилища (this.\$store.dispatch("task/addTask", newTask)), чтобы отправить данные на сервер.

4. Сервис для работы с задачами (task.service.js)

Сервис отправляет запрос на сервер для создания новой задачи. Axios.post для httpзапроса.

#### 5. Хранилище (task.module.js)

```
addTask({ commit }, task) {
   return TaskService.addTask(task).then(
    task => {
      commit('addTask');
      return Promise.resolve(task);
   },
   error => {
      commit('addTaskFailure');
      return Promise.reject(error);
   }
  );
},
```

Обработка результата запроса на сервер и, если все успешно, то вызывается мутация для обновления состояния хранилища.

#### 6. Интеграция компонента (view)

```
<template>
      <CreateTask />
      </template>

<script setup>
      import CreateTask from '@/components/task/CreateTask.vue'
</script>
```

Здесь происходит интеграция компонента создания задачи в представление. Я использую <CreateTask /> для отображения формы создания задачи.

## Вывод:

В данной лабораторной работе была реализована frontend-часть для сервиса Task Manager.