САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Отчет

Практическая работа

Выполнил:

Пырков Владислав

К33402

Проверил:
Добряков Д. И.

Санкт-Петербург

2024 г.

Задача

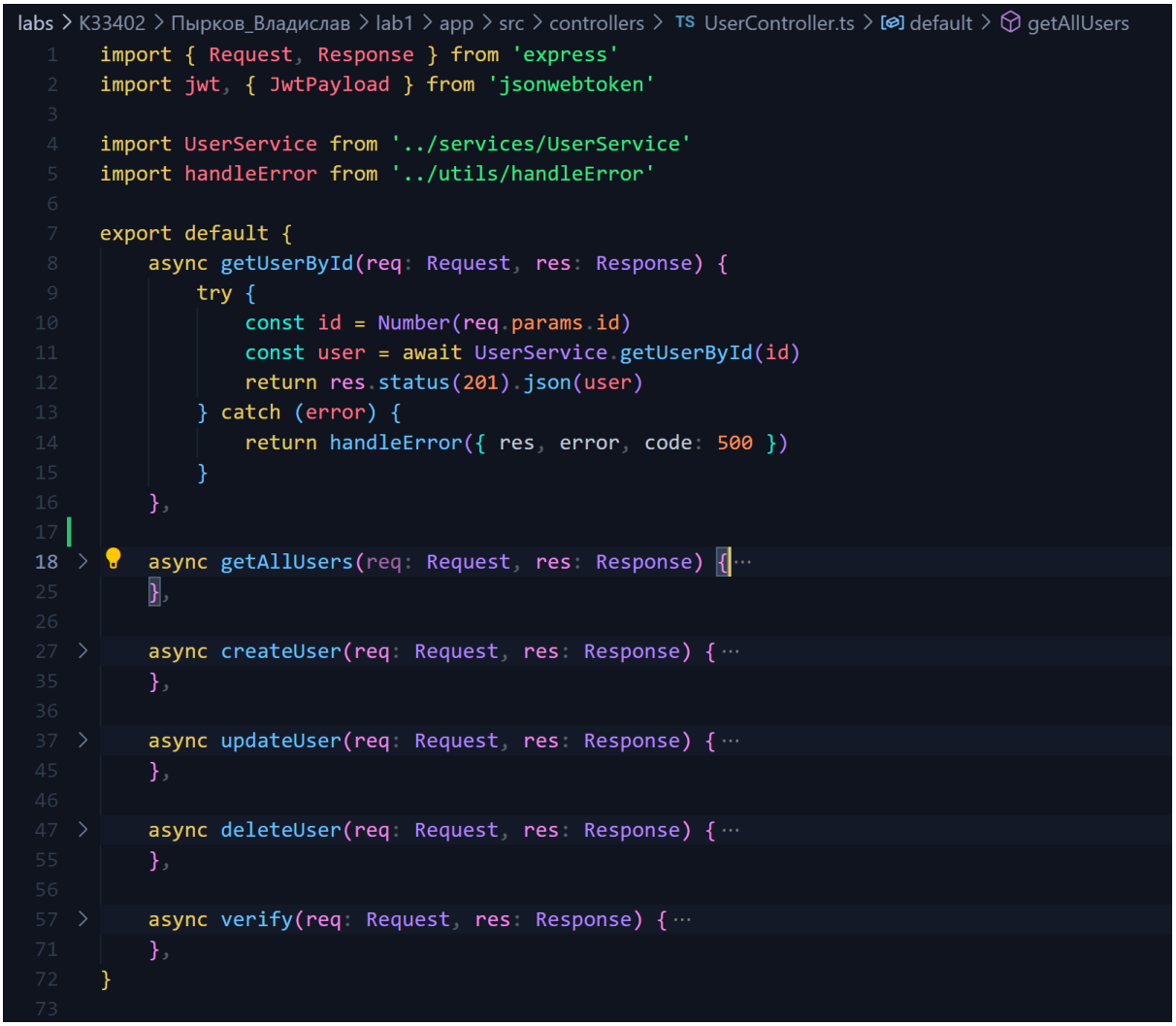
- Продумать свою собственную модель пользователя
- Реализовать набор из CRUD-методов для работы с пользователями средствами Express + Sequelize
- Написать запрос для получения пользователя по id/email

Ход работы

Модель пользователя в выбранном сервисе должна содержать данные об аккаунте (логин и пароль), а также почта, которая должна быть уникальной у каждого пользователя.

```
@Table
class User extends Model {
    @PrimaryKey
    @AutoIncrement
    @Column(DataType.INTEGER)
    declare id: number

    @Column(DataType.STRING)
    declare name: string

    @Unique
    @Column(DataType.STRING)
    declare email: string

    @Unique
    @Column(DataType.STRING)
    declare password: string
}
```

Далее создаем Controller для пользователя, подсоединяя методы из соответствующего сервиса.



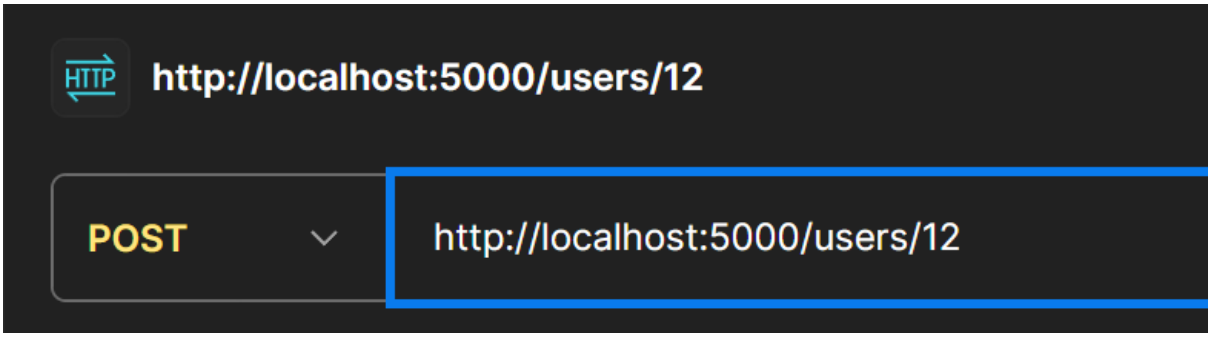
Внутри сервиса описываем методы взаимодействия с пользователем, используя паттерн Репозиторий для прослойки между сервисом и моделью.

labs > K33402 > Пырков_Владислав > lab1 > app > src > services > TS UserService.ts >

```ts
import sequelize from '../database'
import User from '../database/models/User'
import serviceHandleError from '../utils/serviceHandleError'

const userRepository = sequelize.getRepository(User)

class UserService {
    static async getUserById(id: number) {
        return userRepository.findByPk(id)
    }

    static async getAllUsers() {···
    }

    static async createUser(userData: any) {···
    }

    static async updateUser(id: number, userData: any) {···
    }

    static async deleteUser(id: number) {···
    }
}

export default UserService
```

В итоге получаем удобное описание всех свойств модели пользователя и взаимодействия с ним.

Для создания запроса, обращаемся к эндпоинту users/:id с методом get, тем самым получая данные о пользователе по его id.



Вывод

В процессе выполнения продумали собственную модель пользователя, поработали с express и Sequelize, получили опыт составления RESTful API для модели пользователя