

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет

Практическая работа 2

Выполнила:

Герулайтите Габриэля
К33402

Проверил:

Добряков Д. И.

Санкт-Петербург
2022 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

1. Модель пользователя:

```
JS user.js x
1  'use strict';
2  const {
3    Model
4  } = require('sequelize');
5  module.exports = (sequelize, DataTypes) => {
6    class User extends Model {
7      /**
8       * Helper method for defining associations.
9       * This method is not a part of Sequelize lifecycle.
10      * The `models/index` file will call this method automatically.
11      */
12      static associate(models) {
13        // define association here
14      }
15    }
16    User.init({
17      firstName: DataTypes.STRING,
18      lastName: DataTypes.STRING,
19      email: DataTypes.STRING
20    }, {
21      sequelize,
22      modelName: 'User',
23    });
24    return User;
25  };
```

2. Реализация CRUD-методов:

Создание (Create):

```
app.post('/create', async (req,res) => {  
  const user = await db.User.create(req.body)  
  res.send(user.toJSON())  
})
```

Обновление (Update):

```
app.put('/update/:id', async (req,res) => {  
  const user = await db.User.findByPk(req.params.id)  
  if (user) {  
    await user.update(req.body)  
    res.send(user.toJSON())  
  } else {  
    res.send({"msg": "user is not found"})  
  }  
})
```

Удаление (Delete):

```
app.delete('/delete/:id', async (req, res) => {  
  const user = await db.User.destroy({ where: { id: req.params.id } })  
  if (user) {  
    res.send({"msg": "user deleted"})  
  }  
  else {  
    res.send({"msg": "user not found"})  
  }  
})
```

3. Запрос получения пользователя по e-mail:

```
app.get('/user/email/:email', async (req, res) => {
  const user = await db.User.findOne({
    where: {
      email: req.params.email
    }
  })
  if (user) {
    res.send(user.toJSON())
  }
  else {
    res.send({"msg": "user not found"})
  }
})
```

← → ↺ localhost:3000/users/email/test1@example.com

{"id":1,"firstName":"Test1","lastName":"Test1","email":"test1@example.com","createdAt":"2022-03-17T17:49:28.053Z","updatedAt":"2022-03-17T17:49:28.053Z"}

Вывод

В ходе работы ознакомилась с express и sequelize. Сделана модель пользователя, состоящая из id, имени, фамилии и почты. Реализован набор методов по управлению пользователями, добавлены возможности создать, обновить и удалить пользователя (CRUD-методы). Сделана возможность получить пользователя по его почте.