

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэк-энд разработка

Отчет

Практическая/Лабораторная работа

Выполнил:

До Ван Тхиен

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. index.js

```
1  const express = require('express')
2  const db = require('./models')
3  const bp = require('body-parser')
4  const app = express()
5  const port = 3000
6  app.use(bp.json())
7  app.use(bp.urlencoded({ extended: true }))
8  app.get('/', (req, res) => {
9    res.send('Hello World!')
10 })
11 app.post('/users', async (req, res) => {
12   const user = await db.User.create(req.body)
13   console.log(req.body)
14   res.send(user.toJSON())
15 }) //create user
16 app.get('/users/:id', async (req, res) => {
17   const user = await db.User.findByPk(req.params.id)
18   console.log('user is ', user)
19   res.send(user)
20 }) // find by id
21 app.get('/users/email/:email', async (req, res) => {
22   const user = await db.User.findOne({ where: { email: req.params.email } })
23   console.log('user is ', user)
24   res.send(user)
25 }) //find by email
26 app.get('/users', async (req, res) => {
27   const user = await db.User.findAll(req.params)
28   console.log('user is ', user)
29   res.send(user)
30 })// get all users
31 app.put('/users/:id', async (req, res) => {
32   let user = await db.User.findByPk(req.params.id)
33   await db.User.update(req.body, { where: { id: req.params.id } })
34   res.send(user)
35 }) // update user
36 app.get('/users/delete/:id', async (req, res) => {
37   const user = await db.User.destroy({ where: { id: req.params.id } })
38   console.log('user was deleted!')
39   res.send('User was deleted!')
40 }) // delete user
41 app.listen(port, () => {
42   console.log(`Example app listening on port ${port}`)
43 })
```

## 2. createUser.js

```
1  const db = require('./models')
2
3  async function main() {
4      await db.User.create({
5          firstName: 'Test4',
6          lastName: 'Test4',
7          email: 'test4@example.com'
8      })
9  }
10
11  main()
```

## 3. user.js

```
'use strict';
const {
  Model
} = require('sequelize');
module.exports = (sequelize, DataTypes) => {
  class User extends Model {
    /**
     * Helper method for defining associations.
     * This method is not a part of Sequelize lifecycle.
     * The `models/index` file will call this method automatically.
     */
    static associate(models) {
      // define association here
    }
  }
  User.init({
    firstName: DataTypes.STRING,
    lastName: DataTypes.STRING,
    email: DataTypes.STRING
  }, {
    sequelize,
    modelName: 'User',
  });
  return User;
};
```

## **Вывод**

Узнал создать простого пользователя с помощью Nodejs, использовать инструменты Express и Sequelize для создания методов CRUD для работы с пользователями и написания запросов для получения информации о пользователях.