

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет по лабораторной работе №1

Выполнил:

Галиновский Роман

Группа
К33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2023г.

Задача

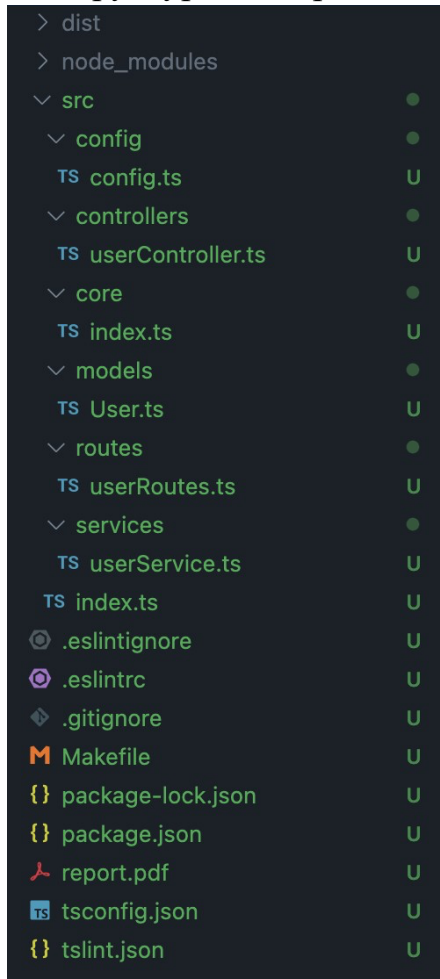
Написать свой boilerplate на express + sequelize / TypeORM + typescript.

Должно быть явное разделение на:

- контроллеры
- модели
- роуты
- репозиторий для работы с моделями

Ход работы

Структура boilerplate



Модель

```

import { Table, Column, Model } from 'sequelize'

@Table
export default class User extends Model {
  @Column
  name: string

  @Column
  password: string

  @Column
  email: string
}

```

Контроллеры

```

export default class ExampleController {

  private service = new DefaultService()

  post = async (request: any, response: any) => {
    try {
      const user = request.body
      await this.service.add(user.name, user.password, user.email)
      response.send('Added' + " " + user.name + " " + "user")
    } catch (error: any) {
      response.status(400).send(error.message)
    }
  }

  get = async (request: any, response: any) => {
    try {
      const data = await this.service.get()
      response.send(data)
    } catch (error: any) {
      response.status(400).send(error.message)
    }
  }
}

```

Сервисы

```

export default class DefaultService {

  private repo = sequelize.getRepository(User)

  add(name: string, password: string, email: string) {
    this.repo.create({ name: name, password: password, email: email })
  }

  get() {
    return this.repo.findAll()
  }
}

```

Роуты

```
const router: express.Router = express.Router()

const exampleController = new ExampleController()

router
  .route('/')
  .get(exampleController.get)
  .post(exampleController.post)

export default router
```

Вывод

В ходе работы был создан boilerplate на express, Sequelize, typescript. Созданный boilerplate можно использовать для создания полноценного backend приложения.