

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бек-енд разработка

Отчет

Лабораторная работа № 1

Выполнил:

Легин Денис

Группа

К33402

Проверил:

Добряков Д.И.

Санкт-Петербург

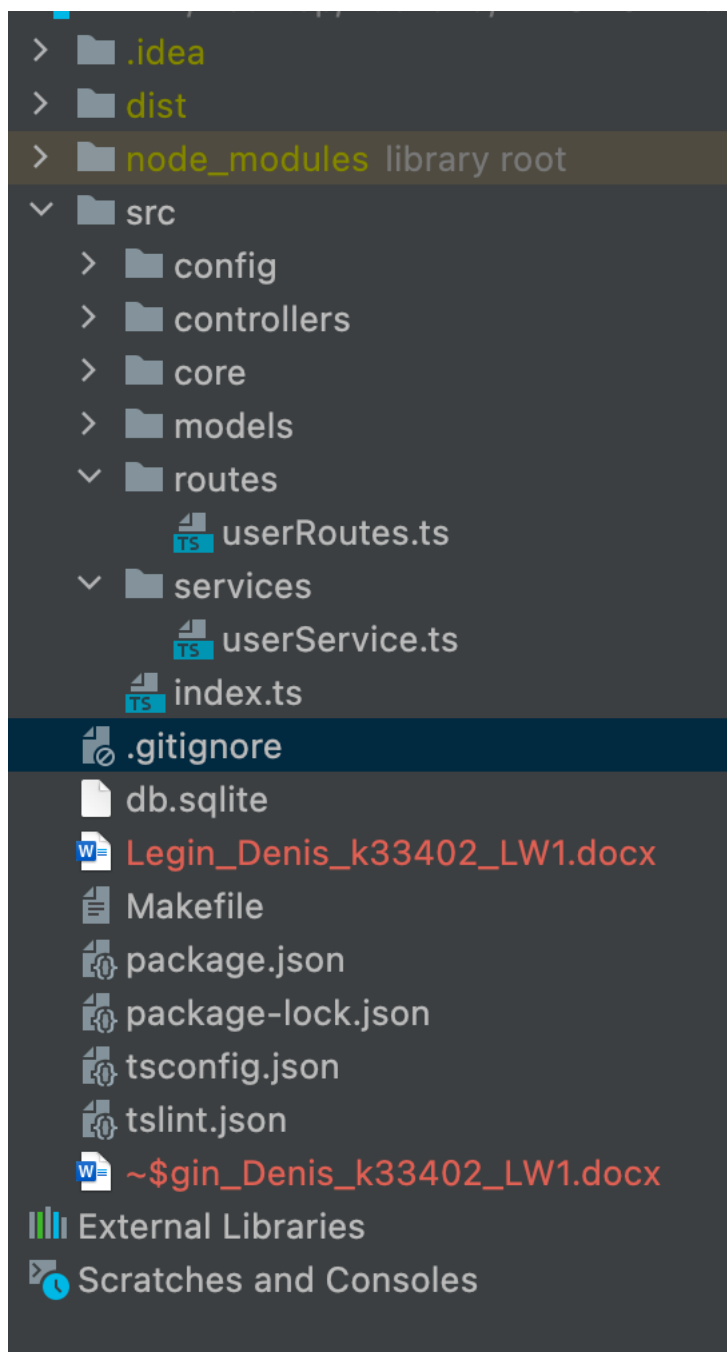
2022 г.

Задача

Написать свой boilerplate на express + sequelize / TypeORM + typescript.
Должно быть явное разделение на:

- модели
- контроллеры
- роуты
- сервисы для работы с моделями (реализуем паттерн “репозиторий”)

Ход работы



Структура бойлерплейта

```
export default class ExampleController {  
  
  private service = new DefaultService()  
  
  post = async (request: any, response: any) => {  
    try {  
      const user = request.body  
      await this.service.add(user.nickname, user.password, user.email)  
      response.send('Added' + " '" + user.nickname + "'" + "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)  
    }  
  }  
}
```

Контроллеры

```
import { Table, Column, Model } from 'sequelize-typescript'

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

  @Column
  password: string

  @Column
  email: string
}
```

Модель пользователя

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

const exampleController = new ExampleController()

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

export default router
```

Роуты

```

export default class DefaultService {

    private repo = sequelize.getRepository(User)

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

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

```

Сервисы

POST запрос:

GET ▼ http://127.0.0.1:8008/v1

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (7) Test Results 🌐 200 OK 19 ms 369 B

Pretty Raw Preview Visualize JSON ▼

```

1  [
2    {
3      "id": 1,
4      "nickname": null,
5      "password": null,
6      "email": null,
7      "createdAt": "2022-06-13T19:50:16.348Z",
8      "updatedAt": "2022-06-13T19:50:16.348Z"
9    }
10 ]

```

GET-запрос:

POST

⌵

http://127.0.0.1:8008/v1

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE	DESCRIPTION
	Key	Value	Description

Body

Cookies

Headers (7)

Test Results

200 OK

64 ms

248 B

S

Pretty

Raw

Preview

Visualize

HTML ⌵

1 Added 'undefined' user

Вывод:

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