

Chai 和 Mocha 为API编写测试



天行无忌 Lv4

2021年08月18日 20:17 · 阅读 208

关注



【这是我参与8月更文挑战的第 18 天，活动详情查看：[8月更文挑战](#)】

昨天在文章《[JavaScript单元测试的“抹茶”组合：Mocha和Chai](#)》介绍了JavaScript单元测试常用组合，本文展示使用 Chai 和 Mocha 为API编写简单的测试用例。

本文将除用到主要框架 Chai 和 Mocha 外，还需要用到 Chai HTTP 。

Chai HTTP 是一个使用 Chai 断言进行 HTTP 集成测试的框架，详细文档参阅[官方文档](#)。

首先，通过运行以下命令来安装相关的依赖项：

```
npm install mocha --save-dev
```

复制代码



探索稀土掘金



登录

修改项目文件 `package.json`，修改 `scripts` 项，增加 `test` 命令：

[复制代码](#)

```
"scripts": {  
  "start": "node ./app.js",  
  "dev": "nodemon ./app.js",  
  "test": "mocha"  
}
```

接下来开始编写测试用例，本文以用户登录接口为例，三个测试条件：

1. 登录成功
2. 用户名不存在
3. 密码错误

测试用例

第 1 步

在项目目录中创建一个名为 `test` 的文件夹，然后在其中创建一个文件 `administrators.js`。

第 2 步

接下来编写单元测试用例，编辑文件 `administrators.js`，代码如下：

[复制代码](#)

```
const chai = require("chai");  
const chaiHttp = require("chai-http");  
const apiUrl = "http://127.0.0.1:4088";  
  
chai.should();  
chai.use(chaiHttp);  
  
describe("测试用户登录模块", () => {  
  describe("POST /api/v1/auth/login", () => {  
    it("用户名不存在", (done) => {  
      chai.request(apiUrl)  
        .post("/api/v1/auth/login")  
        .end(done);  
    });  
  });  
});
```

```
        password: "dev126",
      })
      .end((error, response) => {
        response.should.have.status(200);
        response.body.should.be.a("object");
        response.body.should.have.property("code").eq(30004);
        done();
      });
    });

    it("登录密码错误", (done) => {
      chai.request(apiUrl)
        .post("/api/v1/auth/login")
        .send({
          username: "administrator",
          password: "dev126",
        })
        .end((error, response) => {
          response.should.have.status(200);
          response.body.should.be.a("object");
          response.body.should.have.property("code").eq(30001);
          done();
        });
    });

    it("登录成功", (done) => {
      chai.request(apiUrl)
        .post("/api/v1/auth/login")
        .send({
          username: "administrator",
          password: "devpoint",
        })
        .end((error, response) => {
          response.should.have.status(200);
          response.body.should.be.a("object");
          response.body.should.have.property("code").eq(0);
          response.body.should.have.property("data").a("object");
          response.body.data.should.have
            .property("username")
            .eq("administrator");
          done();
        });
    });
  });
});
```

在上面的代码片段中：

- `describe` 是一个保存测试集合的函数，需要两个参数，第一个是被测功能的名称或者描述，第二个是包含一个或多个测试逻辑的函数，也可以嵌套描述。
- `it` 也是一个函数，它实际上是一个测试主题，接受两个参数，第一个参数是测试的名称，第二个参数是函数，实现主要的测试逻辑。

第 3 步

现在运行以下命令以查看测试结果：

```
npm run test
```

复制代码

运行后，将能看到上面编写的3个单元测试用例结果，如图：



总结

单元测试为应用程序的最小部分，在开发API接口的时候可以为预期的结果编写相应的测试用例。在项目开发中，普遍是使用前后端分离的模式，可以使用 Chai 和 Mocha 为后台接口编写测试用例，以最大程度的覆盖接口的逻辑，可以降低前后端联调出现异常的概率，进而提高项目开发效率。

分类： 前端 标签： [Mocha](#) [单元测试](#)

评论

输入评论（Enter换行，⌘ + Enter发送）

相关推荐

勾崽 11月前 Mocha

H5 活动页面快速生成神器——Mocha

2234 14 3

zhangwinwin 1月前 HTTP

为什么少有人使用RESTful API?

3.1w 150 59

biaomingzhong 7月前 后端

推荐一款丧心病狂的API工具：Apifox!

1.3w 113 30

coderyi 5年前 Go Gin

GIN API Framework - 专为 Go Gin 打造的 API Framework

3466 46 2

仁风H 4年前 API 测试 单元测试

做好自动化 API 测试（二） - 如何高效编写测试

1030 25 评论

lzg9527 3月前 前端 JavaScript

这几个高级前端常用的API，你用到了吗？

1.9w 294 14

Jerry \ Hu 5月前 Vue.js 前端

Vue3.2 setup语法糖、Composition API归纳总结

1.9w 537 67

华为云开发者社区 1年前 API

【API进阶之路】太秃然了，老板要我一周内检测并导入一万个小时的视频

456 2 评论

Crap-API接口管理系统

005 1 2018

老学 2年前 RESTful

五分钟用hapi和mongoDB构建一个RESTful的CRUD接口API

1076 2 评论

Anwen 2年前 Elasticsearch

Elasticsearch入门及掌握其JavaAPI

1.3w 89 9

JS菌 3年前

基于 hapi 的博客系统 api

287 3 1

猫力molly 1年前 JavaScript

【真香】我同事开发了一款媲美postMan的免费api管理工具

2.1w 318 162

最爱妈妈本尊 4年前 API 安全

Api接口加密策略

1.7w 172 14

Violet-Jack 4年前 Vue.js 测试 Mocha

Vue.js学习系列六——Vue单元测试Karma+Mocha学习笔记

1681 44 评论

king哥二 2年前 Go

go基于echo、gorm实现增删改查,从请求到落库

1229 1 评论

Charph 7月前 单元测试 前端

mocha + chai 单元测试

522 2 评论

积墨开源 11月前 开源

149 点赞 评论

Brook Shi 4年前 API 单元测试 开源

开源API测试工具 Hitchhiker v0.7更新 - Schedule的对比diff

311 点赞 评论



3



评论



收藏