

南京大学软件测试中心在线业务平台测试报告

前端F组：

后端D组：杨茂琛 孙逸杨 戴显灏 姜宁 王岳 张世茂

南京大学软件测试中心在线业务平台测试报告

引言

1.1 编写目的

1.2 项目背景

1.3 系统简介

1.5 参考资料

测试概要

2.1 测试环境及配置

2.2 测试方法及工具

2.3 测试概要分析表

功能及安全性测试

3.1 测试概要-单元/集成测试

3.2 测试概要-前后端联调

兼容性测试

整体测试结果分析

5.1 测试总结分析

5.2 测试的缺陷和限制

引言

1.1 编写目的

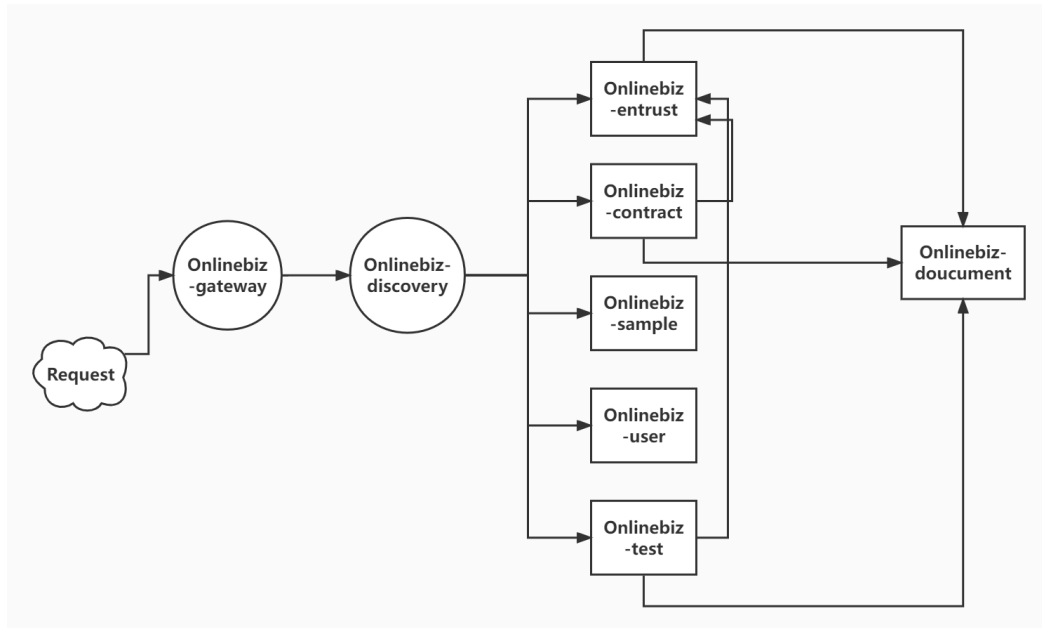
对于前后端分布式系统开发，为了发现和报告应用服务的错误和缺陷，开发人员需通过一系列测试代码，包括**后端单元测试**、**集成测试**及**前后端联调**，以确保系统的功能、互操作性等符合软件的设计要求，满足用户的使用要求。而在测试过程中，通过分析错误产生的原因和错误的分布特征，可以帮助开发人员发现当前所采用的软件过程的缺陷，以便对系统进行升级时进行改进。预期参考人员包括用户、后台管理人员、开发人员和需要阅读本报告的甲方人员（三位指导老师及南京大学软件测试中心项目负责人）。

1.2 项目背景

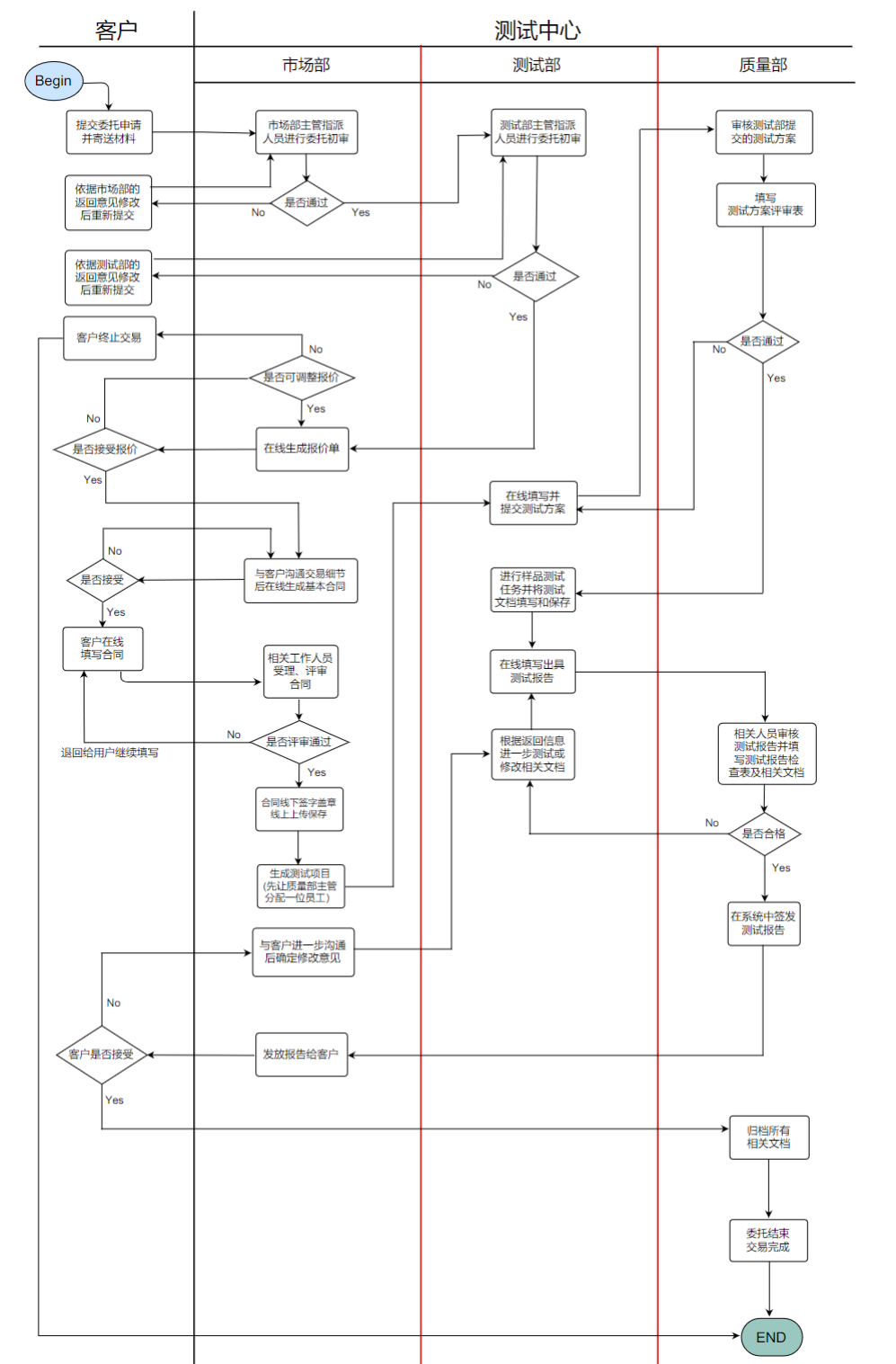
“南大测试”项目在线管理平台主要由三个部分构成，分别为：“南大测试”信息网站、“南大测试”在线业务WEB应用、“南大测试”在线业务微信公众号。主要是通过该项目在线管理平台提高业务处理的效率和管理方便性，优化业务流程，同时为机构的服务质量提质，从整体上全面地提升客户的体验。三个部分相互辅助，从不同维度为用户提供服务，信息网站可以提供用户对机构主体或业务信息的了解咨询需求；在线业务WEB应用可以使用户方便地完成对业务申请管理、信息查询等相关的全流程覆盖的服务；在线业务微信公众号符合当前移动互联网时代用户新的需求，作为在线业务应用的延伸，可以为客户提供一个方便快捷的访问接口应用，使客户可以随时随地方便地进行主要功能的在线访问。

1.3 系统简介

服务间的相互调用



基本业务流程



1.5 参考资料

《南京大学软件测试中心在线业务平台需求文档》

《南京大学软件测试中心在线业务平台设计说明书》

《南京大学软件测试中心在线业务平台用户手册》

GB/T 25000.51-2016 《系统与软件工程 系统与软件质量要求和评 (SQuaRE) 第51部分: 就绪可用软件产品 (RUSP) 的质量要求和测试细则》

测试概要

本项目测试环节主要包含后端单元测试、后端集成测试与前后端联调部分。后端单元测试覆盖了所有服务的service层及DAO层，后端集成测试覆盖了所有项目流程主干及各分支，前后端联调部分覆盖所有服务的Controller层以及对后端测试进行了补充完善。

2.1 测试环境及配置

数据库服务器配置			
机器名 (IP)	CPU	内存	软件环境 (操作系统、应用软件)
124.221.156.98			MySql
124.221.156.98			MongoDB
应用服务器配置			
机器名 (IP)	CPU	内存	软件环境 (操作系统、应用软件)
210.28.133.13	Intel(R) Xeon(R) Silver 4114 CPU @ 2.20GHz, 8 Cores	16GB	Linux ubuntu 4.15.0-187-generic #198- Ubuntu SMP Tue Jun 14 03:23:51 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
客户端配置			
172.26.30.68	Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz, 4 Cores	16GB	Windows 10 家庭中文版 版本号 21H2 19044.1766

2.2 测试方法及工具

测试内容	测试方法	测试工具
功能	单元测试、集成测试、 前后端联调	Junit、 APIfox
可靠性	单元测试、集成测试、 前后端联调	Junit、 APIfox
安全性	静态分析	CodeQL
兼容性	跨系统平台运行软件	
安装和手册	实际安装内测	

2.3 测试概要分析表

模块名称	测试类型	内部测试结果说明
委托申请服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过
合同服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过
测试项目服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过
文档服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过
用户服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过
样品服务	功能测试	通过
	安全性测试	通过
	兼容性测试	通过

功能及安全性测试

3.1 测试概要-单元/集成测试

模块名称	用例数（个）	用例通过数（个）	问题数（个）	用例通过率
委托申请服务	56	56	0	100%
合同服务	33	33	0	100%
测试项目服务	127	127	0	100%
文档服务	16	16	0	100%
用户服务	62	62	0	100%
样品服务	13	13	0	100%

单元测试部分巧妙地运用了mock技术，将测试与数据层切割开来，当测试代码每次调用DAO层时，都会将该方法拦截，并返回一个我们预设的初值，这样我们就可以规避掉冗余的流程来生成每一个数据库中的数据样品，从而可以绕开流程限制，顺利地进行单元测试。

集成测试部分我们将单一服务模块中的流程抽象出来，模拟了整个模块流程的主干及分支情况，达到了较高的流程覆盖率。

MongoReportServiceTest x	
<div> <div>✓</div> <div>⊘</div> <div>↓_a</div> <div>↓_≡</div> <div>≡</div> <div>÷</div> <div>↑</div> <div>↓</div> <div>🕒</div> <div>↶</div> <div>↷</div> <div>⚙️</div> </div>	
✓ MongoReportServiceTest (com.njustc.onlinebiz.118毫秒)	
✓ updateReportInvalidStatus()	59毫秒
✓ updateReportByTester()	4毫秒
✓ findReportByMarketer()	3毫秒
✓ findReportByCustomerInvalidStage()	2毫秒
✓ createReportByTester()	2毫秒
✓ findReportByCustomer()	4毫秒
✓ deleteReportByTester()	1毫秒
✓ findReportByInvalidQa()	3毫秒
✓ deleteReportByAdmin()	5毫秒
✓ createReportByMarketer()	2毫秒
✓ findReportDaoNull()	2毫秒
✓ createReportByQa()	2毫秒
✓ deleteReportNotExisted()	1毫秒
✓ findReportByInvalidTester()	2毫秒
✓ deleteReportByQa()	1毫秒
✓ updateReportDaoNull()	2毫秒
✓ findReportByTester()	4毫秒
✓ updateReportByQa()	3毫秒
✓ findReportByQa()	2毫秒
✓ updateReportByMarketer()	2毫秒
✓ findReportByInvalidCustomer()	2毫秒
✓ updateReportByCustomer()	3毫秒
✓ deleteReportDaoFailure()	1毫秒
✓ updateReportDaoFailure()	2毫秒
✓ updateReportByInvalidTester()	1毫秒
✓ findReportInvalidStatus()	3毫秒













MongoReportDAOIT x		
✓	✓ MongoReportDAOIT (com.njustc.onlinebiz.t	5秒 671毫秒
✓	insertReport()	2秒 479毫秒
✓	findReportByldNonexistent()	45毫秒
✓	findReportByld()	32毫秒
✓	updateContentByldNonexistent()	30毫秒
✓	updateContentByld()	1秒 572毫秒
✓	deleteReportByldNonexistent()	26毫秒
✓	deleteReportByld()	1秒 487毫秒

MongoSchemeReviewDAOIT x		
✓	✓ MongoSchemeReviewDAOIT (com.njustc.or	8秒 191毫秒
✓	insertSchemeReview()	2秒 303毫秒
✓	findSchemeReviewByldNonexistent()	46毫秒
✓	findSchemeReviewByld()	1秒 512毫秒
✓	updateSchemeReviewByldNonexistent()	25毫秒
✓	updateSchemeReviewByld()	1秒 407毫秒
✓	updateScannedCopyPathByldNonexistent()	22毫秒
✓	updateScannedCopyPathByld()	1秒 443毫秒
✓	deleteSchemeAuditByldNonexistent()	25毫秒
✓	deleteSchemeAuditByld()	1秒 408毫秒

MongoSchemeServiceTest	
✓	✓ MongoSchemeServiceTest (com.njustc.onlinebiz. 97毫秒)
✓	updateScheme() 73毫秒
✓	removeScheme() 5毫秒
✓	findScheme() 15毫秒
✓	createScheme() 4毫秒

MongoTestRecordServiceTest	
✓	✓ MongoTestRecordServiceTest (com.njustc.onlin 103毫秒)
✓	updateTestRecordList() 76毫秒
✓	createTestRecordList() 5毫秒
✓	removeTestRecordList() 8毫秒
✓	findTestRecordList() 14毫秒

MongoTestIssueServiceTest	
✓	✓ MongoTestIssueServiceTest (com.njustc.onlinebi 83毫秒)
✓	createTestIssueList() 44毫秒
✓	findTestIssueList() 20毫秒
✓	updateTestIssueList() 14毫秒
✓	removeTestIssueList() 5毫秒

MongoTestcaseDAOIT x		
           		
✓	MongoTestcaseDAOIT (com.njustc.onlinebiz	6秒 632毫秒
✓	insertTestcase()	2秒 165毫秒
✓	findTestcaseByIdNonexistent()	44毫秒
✓	findTestcaseById()	1秒 475毫秒
✓	updateTestcaseByIdNonexistent()	19毫秒
✓	updateTestcaseById()	1秒 473毫秒
✓	deleteTestcaseByIdNonexistent()	22毫秒
✓	deleteTestcaseById()	1秒 434毫秒

运行: DefaultContractServiceTest



✓ DefaultContractServiceTest (com.njustc.onlin 457毫秒)

✓ testCreateContractNoAuthority() 230毫秒

✓ testCreateContractSuccess() 85毫秒

✓ testCreateContractInconsistent() 21毫秒

✓ ModifyContractTest 121毫秒

✓ testUpdateStatusNoAuthority() 6毫秒

✓ testRemoveContractNoAuthority() 3毫秒

✓ testUpdateMarketerIdNoAuthority() 4毫秒

✓ testUpdateContractInvalidStage() 5毫秒

✓ testFindContractSuccess() 5毫秒

✓ testRemoveContractSuccess() 3毫秒

✓ testUpdateStatusSuccess() 5毫秒

✓ testUpdateContractNoAuthority() 4毫秒

✓ testUpdateCustomerIdNoAuthority() 4毫秒

✓ testApproveContractInvalidStage() 4毫秒

✓ testFindContractNoAuthority() 4毫秒

✓ testUpdateNonDisclosureSuccess() 5毫秒

✓ testRemoveContractNotExist() 5毫秒

✓ testUpdateCustomerIdSuccess() 4毫秒

✓ testUpdateContractSuccess() 4毫秒

✓ testApproveContractNoAuthority() 4毫秒

✓ testFindContractNotExist() 4毫秒

✓ testUpdateMarketerIdSuccess() 4毫秒

✓ testDenyContractNoAuthority() 3毫秒

✓ testDenyContractInvalidStage() 3毫秒

✓ testUpdateNonDisclosureInvalidStage() 4毫秒

✓ testUpdateNonDisclosureNoAuthority() 2毫秒


✓ testDenyContractSuccess() 3毫秒

✓ testApproveContractSuccess() 4毫秒

✓ testUpdateContractInconsistent() 3毫秒

✓ ScannedCopyTest 22毫秒

✓	testSaveScannedCopyInvalidStage()	5毫秒
✓	testGetScannedCopyNoAuthority()	4毫秒
✓	testGetScannedCopyInvalidStage()	6毫秒

MongoContractDAOIT ×		
		
✓	MongoContractDAOIT (com.njustc.onlinebiz	3秒 890毫秒
✓	testInsertContract()	2秒 564毫秒
✓	testFindContractByIdNonExist()	203毫秒
✓	testFindContractByIdExist()	76毫秒
✓	testUpdateContractNonExist()	111毫秒
✓	testUpdateContractExist()	134毫秒
✓	testUpdateMarketerIdNonExist()	72毫秒
✓	testUpdateMarketerIdExist()	121毫秒
✓	testUpdateCustomerIdNonExist()	71毫秒
✓	testUpdateCustomerIdExist()	119毫秒
✓	testUpdateScannedCopyPathNonExist()	59毫秒
✓	testUpdateScannedCopyPathExist()	46毫秒
✓	testUpdateNonDisclosureNonExist()	38毫秒
✓	testUpdateNonDisclosureExist()	53毫秒
✓	testUpdateStatusNonExist()	42毫秒
✓	testUpdateStatusExist()	52毫秒
✓	deleteContractByIdNonExist()	101毫秒
✓	deleteContractByIdExist()	28毫秒

MongoEntrustTestReviewDAOIT		
✓	✓	MongoEntrustTestReviewDAOIT (com.njustc 6秒 717毫秒)
✓	✓	insertEntrustTestReview() 2秒 226毫秒
✓	✓	findEntrustTestReviewByldNonexistent() 43毫秒
✓	✓	findEntrustTestReviewByld() 1秒 506毫秒
✓	✓	updateEntrustTestReviewByldNonexistent() 29毫秒
✓	✓	updateEntrustTestReviewByld() 1秒 459毫秒
✓	✓	deleteEntrustTestReviewByldNonexistent() 23毫秒
✓	✓	deleteEntrustTestReviewByld() 1秒 431毫秒

DefaultEntrustServiceTest		
✓	✓	DefaultEntrustServiceTest (com.njustc.onlinebiz 438毫秒)
✓	✓	testCreateEntrustSuccess() 54毫秒
✓	✓	testCreateEntrustNoAuthority() 3毫秒
>	✓	ModificationTest 205毫秒
>	✓	FindEntrustOutlinesTest 172毫秒
✓	✓	FindEntrustTest 4毫秒
✓	✓	testFindEntrustNotExist() 1毫秒
✓	✓	testFindEntrustNoAuthority() 2毫秒
✓	✓	testFindEntrustSuccess() 1毫秒



✓	✓	MongoEntrustDAOIT (com.njustc.onlinel	40秒 516毫秒
✓		testInsertEntrust()	2秒 679毫秒
✓		testFindEntrustByldNotExist()	57毫秒
✓		testFindEntrustByldExist()	1秒 755毫秒
✓		testFindAllEntrusts()	90毫秒
✓		testCountAll()	31毫秒
✓		testUpdateCustomerIdNotExist()	38毫秒
✓		testUpdateCustomerIdExist()	1秒 898毫秒
✓		testCountByCustomerIdNotExist()	23毫秒
✓		testCountByCustomerIdExist()	176毫秒
✓		testFindEntrustsByCustomerIdNotExist()	29毫秒
✓		testFindEntrustsByCustomerIdExist()	440毫秒
✓		testUpdateMarketerIdNotExist()	19毫秒
✓		testUpdateMarketerIdExist()	1秒 947毫秒
✓		testCountByMarketerIdNotExist()	20毫秒
✓		testCountByMarketerIdExist()	102毫秒
✓		testFindEntrustsByMarketerIdNotExist()	23毫秒
✓		testFindEntrustsByMarketerIdExist()	202毫秒
✓		testUpdateTesterIdNotExist()	20毫秒
✓		testUpdateTesterIdExist()	1秒 773毫秒
✓		testCountByTesterIdNotExist()	19毫秒
✓		testCountByTesterIdExist()	726毫秒
✓		testFindEntrustsByTesterIdNotExist()	16毫秒
✓		testFindEntrustsByTesterIdExist()	1秒 496毫秒
✓		testUpdateContractIdNotExist()	22毫秒
✓		testUpdateContractIdExist()	2秒 832毫秒
✓		testUpdateContentNotExist()	20毫秒
✓		testUpdateContentExist()	3秒 69毫秒

✓ testUpdateSoftwareDocReviewNotExist()	32毫秒
✓ testUpdateSoftwareDocReviewExist()	3秒 807毫秒
✓ testUpdateReviewNotExist()	23毫秒
✓ testUpdateReviewExist()	3秒 529毫秒
✓ testUpdateQuoteNotExist()	26毫秒
✓ testUpdateQuoteExist()	3秒 571毫秒

安全性测试

安全性测试主要指测试代码是否能够通过权限校验与阶段校验，将服务器接收到的请求在service层筛选，拦截错误权限访问及错误阶段访问请求。这一部分主要在单元测试中完成，通过mock 不同权限的用户、不同阶段发出的请求来确保平台系统的安全性。


3.2 测试概要-前后端联调

按照角色搜寻用户

▼ 常规

请求 URL: http://localhost:7777/api/user/search?userRole=MARKETER

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×

标头

负载

预览

响应

发起程序

计时

Cookie

▼ [{userId: 3, userName: "test", userRole: "MARKETER"},...]

▼ 0: {userId: 3, userName: "test", userRole: "MARKETER"}

userId: 3

userName: "test"

userRole: "MARKETER"

▶ 1: {userId: 4, userName: "MARKETER", userRole: "MARKETER"}

▶ 2: {userId: 23, userName: "m", userRole: "MARKETER"}

▶ 3: {userId: 29, userName: "mytest", userRole: "MARKETER"}

▶ 4: {userId: 6886, userName: "MARKETER114", userRole: "MARKETER"}

按照用户名搜寻用户

▼ 常规

请求 URL: http://localhost:7777/api/user/search?userName=SSM1

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 负载 预览 响应 发起程序 计时 Cookie

1 [{"userId":6893,"userName":"SSM1","userRole":"CUSTOMER"}]

[查看测试项目完整信息](#)

▼ 常规

请求 URL: http://localhost:7777/api/test/62c7835fe33a743f0fc580ac

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie

```
▼ {id: "62c7835fe33a743f0fc580ac",...}
  id: "62c7835fe33a743f0fc580ac"
  ▼ projectBaseInfo: {entrustId: "62c7723c4fcabd15f0b7193c8", marketerId: 4, te
    customerId: 10
    entrustId: "62c7723c4fcabd15f0b7193c8"
    marketerId: 4
    qaId: 6
    serialNumber: "1"
    softwareName: "1"
    testerId: 5
  }
  ▼ projectFormIds: {testSchemeId: "62c785d9e33a743f0fc580ad", testReportId: "
    testIssueListId: "62c785d9e33a743f0fc580b2"
    testRecordListId: "62c785d9e33a743f0fc580b0"
    testReportCecklistId: "62c785d9e33a743f0fc580b1"
    testReportId: "62c785d9e33a743f0fc580ae"
    testSchemeChecklistId: "62c785d9e33a743f0fc580b4"
    testSchemeId: "62c785d9e33a743f0fc580ad"
    testcaseListId: "62c785d9e33a743f0fc580af"
    workChecklistId: "62c785d9e33a743f0fc580b3"
  }
  ► status: {stage: "SCHEME_UNFILLED", message: ""}
```


▼ 常规

请求 URL: http://localhost:7777/api/test/report/62c785d9e33a743f0fc580ae

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie

```
▼ {content: {...}, entrustId: "62c7723c4fcbd15f0b7193c8", id: "62c785d9e33a743f0fc580ae",...}
  ▼ content: {...}
    approver: "1"
    approverDate: "2022-07-06"
    auditor: "1"
    auditorDate: "2022-07-05"
    ▶ clientContact: {companyCH: "1", companyEN: null, companyPhone: "15896257308", compan
    ▶ efficiencyTests: [{content: "1", description: "1", result: "1"}]
    ▶ functionalTests: [{content: "1", description: "1", result: "1"}]
    hardwareConfig: "1"
    hardwareName: "1"
    hardwareNum: "1"
    hardwareType: "1"
    id: "1"
    mainTester: "1"
    mainTesterDate: "2022-07-04"
    ▶ maintainabilityTests: [{content: "1", description: "1", result: "1"}]
    networkEnvironment: "1"
    osSoftwareName: "1"
    osVersion: "1"
    ▶ portableTests: [{content: "1", description: "1", result: "1"}]
    projectSerialNumber: "1"
    ▶ referenceMaterials: ["1"]
    ▶ reliabilityTests: [{content: "1", description: "1", result: "1"}]
    reportDate: "2022-07-08"
    sampleDate: "2022-07-01"
    sampleList: "1"
    sampleName: "1"
    sampleStatus: "1"
    sampleVersion: "1"
    ▶ softwareEnvironments: [{softwareType: "1", softwareName: "1", softwareVersion: "1"}]
    softwareName: "1"
    ▶ testBases: ["1"]
    testBasis: "1"
    testConclusion: "1"
    testEndTime: "2022-07-06"
```

▼ 常规

请求 URL: http://localhost:7777/api/test/scheme/62c785d9e33a743f0fc580ad

请求方法: GET


状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×	标头	预览	响应	发起程序	计时	Cookie
▼	{id: "62c785d9e33a743f0fc580ad", creatorId: 4, entrustId: "62c7723c4fcbd15f0b7193c8", content: {version: "1",...}baseline: "1"designSchedule: {workload: "1", startDate: "2022-07-05", endDate: "2022-07-06", documentSummary: "1"}evaluateSchedule: {workload: "1", startDate: "2022-07-05", endDate: "2022-07-06", documentSummary: "1"}executeSchedule: {workload: "1", startDate: "2022-07-05", endDate: "2022-07-06", expectedTime: null, hardware: "1", logo: "1"}modificationList: [{version: "2", modificationDate: "2022-07-05", method: "1", organization: "1", otherEnvironment: "1", participant: "1"}]planSchedule: {workload: "1", startDate: "2022-07-05", endDate: "2022-07-06", software: "1", systemSummary: "1", testCondition: "1", testLevel: "1", testSample: "1", testToBeExecuted: "1", testType: "1", traceability: "1", version: "1"}creatorId: 4entrustId: "62c7723c4fcbd15f0b7193c8"id: "62c785d9e33a743f0fc580ad"projectId: "62c7835fe33a743f0fc580ac"}					

查看任意测试记录表的详细信息

▼ 常规
请求 URL: http://localhost:7777/api/test/testRecord/62c785d9e33a743f0fc580b0
请求方法: GET
状态代码:  200 OK
远程地址: 127.0.0.1:7777
引用站点策略: strict-origin-when-cross-origin
响应头: 无
响应体: 无


×	标头	预览	响应	发起程序	计时	Cookie
▼	<pre>{id: "62c785d9e33a743f0fc580b0", entrustId: "62c7723c4fcdbd15f0b7193c8", entrustId: "62c7723c4fcdbd15f0b7193c8", id: "62c785d9e33a743f0fc580b0", projectId: "62c7835fe33a743f0fc580ac", testRecords: [{category: "1", testCaseId: "1", designInstruction: "1", actualResult: "1", bugId: "1", caseExecutor: "1", category: "1", confirmationPerson: "1", designInstruction: "1", designer: "1", executionProcess: "1", expectedResult: "1", isConsistent: "1", prerequisites: "1", statute: "11", testCaseId: "1", time: "1"}, {category: "2", testCaseId: "2", designInstruction: "2", statute: "12", testCaseId: "2", time: "2"}]}</pre>					

查看任意测试问题清单的详细信息

▼ 常规

请求 URL: http://localhost:7777/api/test/testIssue/62c785d9e33a743f0fc580b2

请求方法: GET


状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×	标头	预览	响应	发起程序	计时	Cookie
▼	<pre>{id: "62c785d9e33a743f0fc580b2", entrustId: "62c7723c4fcbd15f0b7193c8",...} entrustId: "62c7723c4fcbd15f0b7193c8" id: "62c785d9e33a743f0fc580b2" projectId: "62c7835fe33a743f0fc580ac" testIssues: [{testIssueId: "1", description: "1", correspondingRequirement: "1", ▼ 0: {testIssueId: "1", description: "1", correspondingRequirement: "1", initial associatedCase: "1" correspondingRequirement: "1" description: "1" findTime: "1" initialConditions: "1" responsiblePerson: "1" specificOperation: "1" suggestion: "1" testIssueId: "1" ► 1: {testIssueId: "2", description: "2", correspondingRequirement: "2", initial</pre>					

查看任意测试用例表的详细信息

▼ 常规
请求 URL: http://localhost:7777/api/test/testcase/62c785d9e33a743f0fc580af
请求方法: GET
状态代码:  200 OK
远程地址: 127.0.0.1:7777
引用站点策略: strict-origin-when-cross-origin

×	标头	预览	响应	发起程序	计时	Cookie
▼	<pre>{id: "62c785d9e33a743f0fc580af", entrustId: "62c7723c4fcbd15f0b7193c8",...} entrustId: "62c7723c4fcbd15f0b7193c8" id: "62c785d9e33a743f0fc580af" projectId: "62c7835fe33a743f0fc580ac" testcases: [{category: "1", testCaseId: "1", designInstruction: "1", statute: " ▼ 0: {category: "1", testCaseId: "1", designInstruction: "1", statute: "1", exp category: "1" designInstruction: "1" designer: "1" expectedResult: "1" statute: "1" testCaseId: "1" time: "1" ► 1: {category: "2", testCaseId: "2", designInstruction: "2", statute: "2", exp</pre>					

创建合同

▼ 常规

请求 URL: http://localhost:7777/api/contract?entrustId=62c7723c4fcbd15f0b7193c8

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

登出账号

▼ 常规

请求 URL: http://localhost:7777/api/logout

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

登陆账号

▼ 常规

请求 URL: http://localhost:7777/api/login?userName=SSM&userPassword=123456

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

分配测试部人员

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcbd15f0b7193c8/tester?testerId=5

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

分配市场部人员

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcbd15f0b7193c8/marketer?marketerId=4

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

分配质量部人员

▼ 常规

请求 URL: http://localhost:7777/api/test/62c7835fe33a743f0fc580ac/qa?qaId=6

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新测试方案检查表内容

▼ 常规

请求 URL: http://localhost:7777/api/review/scheme/62c785d9e33a743f0fc580b4

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新合同内容

▼ 常规

请求 URL: http://localhost:7777/api/contract/62c77fb2c0de6c2cb1803812

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新检查表内容

▼ 常规

请求 URL: http://localhost:7777/api/review/report/62c785d9e33a743f0fc580b1

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新软件评审文档

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdb15f0b7193c8/software_doc_review

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新项目状态

▼ 常规

请求 URL: http://localhost:7777/api/test/62c7835fe33a743f0fc580ac/status

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

更新样品集

▼ 常规

请求 URL: http://localhost:7777/api/sample/62c77a337f08cd34847926ea

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

获取当前登陆的账号

▼ 常规

请求 URL: http://localhost:7777/api/account

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie

1 [{"userId":6893,"userName":"SSM","userRole":"CUSTOMER"}]

获取合同详情

▼ 常规

请求 URL: http://localhost:7777/api/contract/62c77fb2c0de6c2cb1803812

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

```
×  标题  预览  响应  发起程序  计时  Cookie

▼ {id: "62c77fb2c0de6c2cb1803812", marketerId: 4, customerId: 10, entrustId: "62c7723c4fcbd15f0b7193c8",...}
  customerId: 10
  entrustId: "62c7723c4fcbd15f0b7193c8"
  id: "62c77fb2c0de6c2cb1803812"
  marketerId: 4
  nonDisclosureAgreement: null
▼ partyA: {companyCH: "1", companyEN: "1", companyPhone: "1", companyWebsite: "1", companyAddress: "1",...}
  account: "1"
  accountName: "1"
  bankName: "1"
  companyAddress: "1"
  companyCH: "1"
  companyEN: "1"
  companyPhone: "1"
  companyWebsite: "1"
  contact: "1"
  contactEmail: "1196953351@qq.com"
  contactPhone: "1"
  fax: "1"
  representative: "1"
  sigDate: "2022-07-08"
  zipCode: "1"
▶ partyB: {companyCH: "1", companyEN: "1", companyPhone: "1", companyWebsite: "1", companyAddress: "1",...}
  price: 1
  projectName: "1"
  rectificationDaysEachTime: 1
  rectificationLimit: 1
  scannedCopyPath: null
  serialNumber: null
  signedAt: "1"
  signedDate: "2022-07-08"
▶ status: {stage: "MARKETER_CHECKING", message: null}
  targetSoftware: "1"
  totalWorkingDays: 1
```

获取检查表详情

▼ 常规

请求 URL: http://localhost:7777/api/review/entrustTest/62c785d9e33a743f0fc580b3

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

▼ 常规

请求 URL: http://localhost:7777/api/review/entrustTest/62c785d9e33a743f0fc580b3

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×

标头

预览

响应

发起程序

计时

Cookie


[illegible]

获取委托概要列表

▼ 常规

请求 URL: <http://localhost:7777/api/entrust?page=1&pageSize=20>

请求方法: GET


状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×	标头	负载	预览	响应	发起程序	计时	Cookie
▼	<pre>{page: 1, pageSize: 20, total: 30,...} ▼ list: [{id: "62bea8217d497c2893cab95", customerId: 10, marketerId: 23, testerId: 24}, ▼ 0: {id: "62bea8217d497c2893cab95", customerId: 10, marketerId: 23, testerId: 24, contractId: null, customerId: 10, id: "62bea8217d497c2893cab95", marketerId: 23, projectId: null, softwareName: "1", softwareVersion: "1", ▶ status: {stage: "CUSTOMER_ACCEPT_QUOTE", message: null}, testerId: 24}, ▶ 1: {id: "62beae3b7d497c2893cab96", customerId: 10, marketerId: 4, testerId: 24}, ▶ 2: {id: "62beaea37d497c2893cab98", customerId: 10, marketerId: 23, testerId: 24}, ▶ 3: {id: "62beb0e37d497c2893cab99", customerId: 10, marketerId: 4, testerId: 24}, ▶ 4: {id: "62beb15d7d497c2893cab9a", customerId: 10, marketerId: null, testerId: 24}, ▶ 5: {id: "62beb1c37d497c2893cab9b", customerId: 10, marketerId: null, testerId: 24}, ▶ 6: {id: "62beb2057d497c2893cab9c", customerId: 10, marketerId: null, testerId: 24}, ▶ 7: {id: "62beb2667d497c2893cab9d", customerId: 10, marketerId: null, testerId: 24}, ▶ 8: {id: "62beb3ae7d497c2893cab9e", customerId: 10, marketerId: null, testerId: 24}, ▶ 9: {id: "62beb4077d497c2893cab9f", customerId: 10, marketerId: null, testerId: 24}, ▶ 10: {id: "62beb4db7d497c2893cabca0", customerId: 10, marketerId: null, testerId: 24}, ▶ 11: {id: "62beb5c77d497c2893cabca1", customerId: 10, marketerId: null, testerId: 24}, ▶ 12: {id: "62beb62a7d497c2893cabca2", customerId: 10, marketerId: null, testerId: 24}, ▶ 13: {id: "62beb6857d497c2893cabca3", customerId: 10, marketerId: 23, testerId: 24}, ▶ 14: {id: "62beb7667d497c2893cabca4", customerId: 10, marketerId: null, testerId: 24}, ▶ 15: {id: "62bebc857d497c2893cabca5", customerId: 10, marketerId: null, testerId: 24}, ▶ 16: {id: "62bebd057d497c2893cabca6", customerId: 10, marketerId: null, testerId: 24}, ▶ 17: {id: "62bebdba67d497c2893cabca7", customerId: 10, marketerId: null, testerId: 24}, ▶ 18: {id: "62bebe3c7d497c2893cabca8", customerId: 10, marketerId: null, testerId: 24}, ▶ 19: {id: "62befafd7d497c2893cabca9", customerId: 10, marketerId: 4, testerId: 24}, page: 1, pageSize: 20, total: 30}</pre>						

获取委托详情

▼ 常规
请求 URL: http://localhost:7777/api/entrust/62c7723c4fcbd15f0b7193c8
请求方法: GET
状态代码:  200 OK
远程地址: 127.0.0.1:7777
引用站点策略: strict-origin-when-cross-origin

×	标头	预览	响应	发起程序	计时	Cookie
▼	{id: "62c7723c4fcbd15f0b7193c8", customerId: 10, marketerId: null, testerId: null, contractId: null,...}	▼	content: {testType: ["软件确认测试"],...} document: "q" expectedTime: "1" principal: {companyCH: "阿布", companyEN: "ab", companyPhone: "15896257308", companyWebsite: "1",...} sampleHandling: "1" software: {name: "1", version: "1", developer: "1", developerType: "1", userDescription: "1", functionIntro: "1",...} softwareMedium: "1" techIndex: ["功能性"] testStandard: ["GB/T 25000.51-2010"] testType: ["软件确认测试"] contractId: null customerId: 10 id: "62c7723c4fcbd15f0b7193c8" marketerId: null projectId: null quote: null review: null softwareDocReview: null status: {stage: "WAIT_FOR_MARKETER", message: "等待市场部主管分配市场部人员"} testerId: null			

获取样品集概要列表

▼ 常规

请求 URL: http://localhost:7777/api/sample?page=1&pageSize=20

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×	标头	负载	预览	响应	发起程序	计时	Cookie
▼	{page: 1, pageSize: 20, total: 24, list: [...]}	▼	list: [...] 0: {id: "62bc130512a03b26f5ca37a9", entrustId: "62bc12c0b3b31312b2f2a38b", marketerId: 4, name: "测试测试",...} entrustId: "62bc12c0b3b31312b2f2a38b" id: "62bc130512a03b26f5ca37a9" marketerId: 4 name: "测试测试" samples: [{storageLocation: "北京", sampleCode: "114", sampleName: "无意识", sampleType: "食品", sampleDesc: "stage: "RECEIVED" 1: {id: "62beaf36eef5756025736ee", entrustId: "62beae467d497c2893cab97", marketerId: 4, name: "这是啥",...} 2: {id: "62befbd7eedf5756025736ef", entrustId: "62befafd7d497c2893cabca9", marketerId: 4, name: null,...} 3: {id: "62c0476b7f08cd34847926d6", entrustId: "62bea8217d497c2893cab95", marketerId: 23, name: null,...} 4: {id: "62c145e87f08cd34847926d7", entrustId: "62beae3b7d497c2893cab96", marketerId: 4, name: null,...} 5: {id: "62c1b5567f08cd34847926d8", entrustId: "62c1a8554fcbd15f0b7193ae", marketerId: 4, name: null,...} 6: {id: "62c324437f08cd34847926d9", entrustId: "62c31fac4fcbd15f0b7193af", marketerId: 4, name: null,...} 7: {id: "62c56e8d7f08cd34847926da", entrustId: "62c561b64fcbd15f0b7193b3", marketerId: 4, name: "1",...} 8: {id: "62c692f07f08cd34847926db", entrustId: "62c691994fcbd15f0b7193b4", marketerId: 4, name: null,...} 9: {id: "62c69d527f08cd34847926dc", entrustId: "62c69c914fcbd15f0b7193b5", marketerId: 4, name: null,...} 10: {id: "62c6a56f7f08cd34847926dd", entrustId: "62c6a4944fcbd15f0b7193b7", marketerId: 4, name: null,...} 11: {id: "62c6af7c7f08cd34847926de", entrustId: "62c6af3d4fcbd15f0b7193b8", marketerId: 4, name: null,...} 12: {id: "62c6bb3f7f08cd34847926df", entrustId: "62c6baf64fcbd15f0b7193b9", marketerId: 4, name: null,...} 13: {id: "62c6cca47f08cd34847926e0", entrustId: "62c6cc714fcbd15f0b7193bb", marketerId: 4, name: null,...} 14: {id: "62c6dcdf7f08cd34847926e1", entrustId: "62c6d9f14fcbd15f0b7193bc", marketerId: 6886, name: null,...} 15: {id: "62c6ee607f08cd34847926e2", entrustId: "62c6edbc4fcbd15f0b7193bd", marketerId: 23,...} 16: {id: "62c715507f08cd34847926e3", entrustId: "62c712fb4fcbd15f0b7193c1", marketerId: 6886, name: null,...} 17: {id: "62c71bbd7f08cd34847926e4", entrustId: "62c71b934fcbd15f0b7193c2", marketerId: 6886, name: null,...} 18: {id: "62c7239d7f08cd34847926e5", entrustId: "62c723734fcbd15f0b7193c3", marketerId: 6886, name: null,...} 19: {id: "62c729cb7f08cd34847926e6", entrustId: "62c729a14fcbd15f0b7193c4", marketerId: 6886, name: null,...} page: 1 pageSize: 20 total: 24				

获取样品集详情

▼ 常规

请求 URL: http://localhost:7777/api/sample/62c77a337f08cd34847926ea

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie


▼ {id: "62c77a337f08cd34847926ea", entrustId: "62c7
entrustId: "62c7723c4fcdbd15f0b7193c8"
id: "62c77a337f08cd34847926ea"
marketerId: 4
name: null
samples: null
stage: "RECEIVED"}

接受委托报价

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdbd15f0b7193c8/quote/acceptance

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

接受委托申请

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdbd15f0b7193c8/content/acceptance

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

拒绝合同

▼ 常规

请求 URL: http://localhost:7777/api/contract/62c77fb2c0de6c2cb1803812/denial

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

拒绝委托报价

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcbd15f0b7193c8/quote/denial

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

拒绝委托申请

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcbd15f0b7193c8/content/denial

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

列表获取测试项目概要

▼ 常规

请求 URL: http://localhost:7777/api/test?page=1&pageSize=20

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin


×	标头	负载	预览	响应	发起程序	计时	Cookie
▼	{page: 1, pageSize: 20, total: 10, list: [,...]}						
▼	list: [,...]						
▶	0: {id: "62bf00e6a91ee23a84ba37af", entrustId: "62befafd7d497c2893cabca9",						
▶	1: {id: "62c0405f406080390e6e9f01", entrustId: "62beae467d497c2893cabca9",						
▶	2: {id: "62c145d3406080390e6e9f0b", entrustId: "62beae3b7d497c2893cabca9",						
▶	3: {id: "62c1ba3423a8502042f40440", entrustId: "62c1a8554fcabd15f0b7193ae",						
▶	4: {id: "62c58bd7e33a743f0fc5809f", entrustId: "62c561b64fcabd15f0b7193b3",						
▶	5: {id: "62c68dcee33a743f0fc580a8", entrustId: "62c31fac4fcabd15f0b7193af",						
▶	6: {id: "62c6cdebe33a743f0fc580a9", entrustId: "62c6cc714fcabd15f0b7193bb",						
▶	7: {id: "62c7362be33a743f0fc580aa", entrustId: "62c72bd84fcabd15f0b7193c5",						
▶	8: {id: "62c738bfe33a743f0fc580ab", entrustId: "62c737ed4fcabd15f0b7193c7",						
▼	9: {id: "62c7835fe33a743f0fc580ac", entrustId: "62c7723c4fcabd15f0b7193c8",						
	entrustId: "62c7723c4fcabd15f0b7193c8"						
	id: "62c7835fe33a743f0fc580ac"						
	marketerId: 4						
	qaId: 6						
	softwareName: "1"						
▶	status: {stage: "SCHEME_UNFILLED", message: ""}						
	testerId: 5						
	page: 1						
	pageSize: 20						
	total: 10						

上传测试方案评审表扫描件

▼ 常规

请求 URL: http://localhost:7777/api/review/scheme/62c785d9e33a743f0fc580b4/upload

请求方法: PUT

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

上传合同扫描件

▼ 常规

请求 URL: http://localhost:7777/api/contract/62c77fb2c0de6c2cb1803812/upload

请求方法: PUT

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

上传扫描件

▼ 常规

请求 URL: http://localhost:7777/api/review/report/62c785d9e33a743f0fc580b1/upload

请求方法: PUT

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

▼ 常规

请求 URL: http://localhost:7777/api/entrust/

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×	标头	负载	预览	响应	发起程序	计时	Cookie
1	62c7723c4fcbd15f0b7193c8						

同意合同

▼ 常规

请求 URL: http://localhost:7777/api/contract/62c77fb2c0de6c2cb1803812/acceptance

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

下载 JS001

▼ 常规

请求 URL: http://localhost:7777/api/doc/JS001

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

×

标头

预览

响应

发起程序

计时

Cookie

1

https://oss.syh1en.asia/doc/JS001_out.pdf

下载JS003

▼ 常规

请求 URL:

http://localhost:7777/api/doc/JS003/62c7723c4fcbd15f0b7193c8

请求方法:

GET

状态代码:

✔ 200 OK

远程地址:

127.0.0.1:7777

引用站点策略:

strict-origin-when-cross-origin

×

标头

预览

响应

发起程序

计时

Cookie

1

https://oss.syh1en.asia/doc/JS003_62c7723c4fcbd15f0b7193c8.pdf

下载JS004

▼ 常规

请求 URL:

http://localhost:7777/api/doc/JS004/62c77fb2c0de6c2cb1803812

请求方法:

GET

状态代码:

✔ 200 OK

远程地址:

127.0.0.1:7777

引用站点策略:

strict-origin-when-cross-origin

×

标头

预览

响应

发起程序

计时

Cookie

1

https://oss.syh1en.asia/doc/JS004_62c77fb2c0de6c2cb1803812.pdf

下载JS005

▼ 常规

请求 URL: http://localhost:7777/api/doc/JS005/62c77fb2c0de6c2cb1803812

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie


1 | https://oss.syh1en.asia/doc/JS005_62c77fb2c0de6c2cb1803812.pdf

下载 JS007

▼ 常规

请求 URL: http://localhost:7777/api/doc/JS007/62c785d9e33a743f0fc580ae

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie

1 | https://oss.syh1en.asia/doc/JS007_62c785d9e33a743f0fc580ae.pdf

下载 JS014

▼ 常规

请求 URL: http://localhost:7777/api/doc/JS014/62c7723c4fcbd15f0b7193c8

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie


1 | https://oss.syh1en.asia/doc/JS014_62c7723c4fcbd15f0b7193c8.pdf

下载 委托报价单 PDF

▼ 常规

请求 URL: http://localhost:7777/api/doc/entrustQuote/62c7723c4fcbd15f0b7193c8

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 预览 响应 发起程序 计时 Cookie


1 | https://oss.syh1en.asia/doc/EntrustQuote_62c7723c4fcbd15f0b7193c8.pdf

新增测试项目

▼ 常规

请求 URL: http://localhost:7777/api/test?entrustId=62c7723c4fcbd15f0b7193c8

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

× 标头 负载 预览 响应 发起程序 计时 Cookie

1 | 62c7835fe33a743f0fc580ac

修改测试报告

▼ 常规

请求 URL: http://localhost:7777/api/test/report/62c785d9e33a743f0fc580ae/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改测试方案信息

▼ 常规

请求 URL: http://localhost:7777/api/test/scheme/62c785d9e33a743f0fc580ad/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改测试记录表

▼ 常规

请求 URL: http://localhost:7777/api/test/testRecord/62c785d9e33a743f0fc580b0/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改测试问题清单

▼ 常规

请求 URL: http://localhost:7777/api/test/testIssue/62c785d9e33a743f0fc580b2/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改测试用例表

▼ 常规

请求 URL: http://localhost:7777/api/test/testcase/62c785d9e33a743f0fc580af/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改委托报价

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdb15f0b7193c8/quote

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改委托评审结果

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdb15f0b7193c8/review

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改委托申请

▼ 常规

请求 URL: http://localhost:7777/api/entrust/62c7723c4fcdbd15f0b7193c8/content

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

修改用户密码

▼ 常规

请求 URL: http://localhost:7777/api/account/password?oldValue=123456&newValue=123

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

修改用户名

▼ 常规

请求 URL: http://localhost:7777/account/settings

请求方法: GET

状态代码:  200 OK

远程地址: 127.0.0.1:7777


引用站点策略: strict-origin-when-cross-origin

注册账号

▼ 常规

请求 URL: http://localhost:7777/api/register?userName=SSM&userPassword=123456

请求方法: POST

状态代码:  200 OK

远程地址: 127.0.0.1:7777

引用站点策略: strict-origin-when-cross-origin

兼容性测试

测试情况	检查结果
是否自适应浏览器Firefox	是
是否自适应浏览器Edge	是
是否自适应浏览器Chrome	是
是否可以部署在Windows服务器	是
是否可以部署在Linux服务器	是
是否可以部署在MacOS主机	是
客户端是否可以在Windows操作系统下访问	是
客户端是否可以在Linux操作系统下访问	是
客户端是否可以在MacOS操作系统下访问	是

我们将平台系统分别部署在指导老师提供的Linux服务器、后端组同学租用的Windows服务器、后端组同学本地MacOS主机上，并通过不同系统（Windows、Linux、MacOS）上的不同浏览器（Firefox、Edge、Chrome）访问该平台，发现兼容性良好，证明平台具有可观的兼容性。

整体测试结果分析

5.1 测试总结分析

本软件在经过测试联调及代码修改后，测试结构已达到预期（测试用例通过率100%，绝大部分达到90%覆盖率）

5.2 测试的缺陷和限制

本软件的测试内容并不能达到很好的压力测试及性能测试，即我们尚不能保证在所有环境的运行状态下服务器能够达到的访问容量，我们只能保证在现有条件下（即指导老师提供的服务器上），平台系统能够容纳高额的访问率及吞吐量