

# MeetHere

## 期末展示

第12组 彭钧涛 李尚真 郭源杰



# 目录



## 需求说明

郭源杰



## 功能演示

郭源杰



## 测试思想

郭源杰 彭钧涛 李尚真



## 项目特色

彭钧涛



# 项目简介

## 开发环境

Windows 10  
MySQL 8.0  
Tomcat 9.0

## 测试环境

Windows 10  
MySQL 8.0  
Tomcat 9.0

## 测试工具

JUnit 5  
Selenium  
Allure  
ChromeDriver 79.0.3945.36  
Python 3.7

## 项目架构

Spring Framework 5.1.5

## 构建架构

Maven 4.0.0

## 测试持续集成框架

Jenkins 2.190.3





## 小组概况

小组成员	分工角色	完成任务列表
彭钧涛	项目经理、开发、测试	集成、系统测试、性能测试
郭源杰	测试	单元测试
李尚真	开发、测试	集成、系统测试、性能测试

# 功能需求说明-用户

## 用户角色

### 用户功能

注册

登录

个人信息管理

### 场馆功能

查看场馆介绍与预约信息

场馆预约

预定订单管理

### 新闻功能

查看新闻

### 留言功能

查看

发布

删除

修改

# 功能需求说明-管理员



## 管理员角色

### 用户功能

#### 用户管理

添加用户

删除用户

### 场馆功能

#### 场馆信息管理

场馆介绍

场馆位置

场馆租金

空余时间

#### 预约订单管理

预约订单审核

预约订单统计

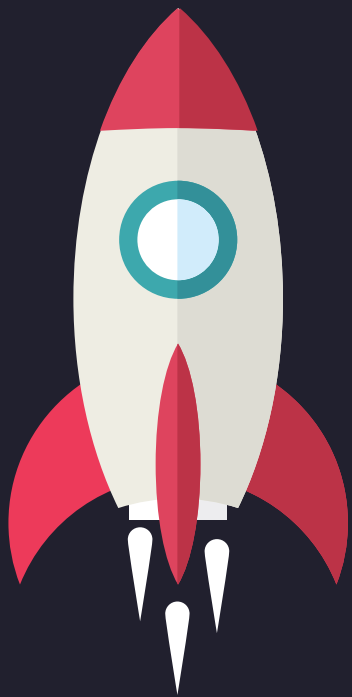
### 新闻功能

#### 新闻管理

发布新闻

修改新闻

删除新闻



# 单元测试

郭源杰



## 使用工具

```
assertAll(  
    () -> assertEquals( expected: "mainUser", result.getViewName()),  
    () -> assertEquals( expected: "user/newsDetail.jsp", result.getModelMap().get("mainPage")),  
    () -> assertEquals(news, result.getModelMap().get("news"))  
);
```

## JUnit5 用于判断程序输出的结果是否满足预期

```
private BuildingDao buildingDao;  
private Connection connection = mock(Connection.class);  
private PreparedStatement preparedStatement = mock(PreparedStatement.class);  
private SQLException test_sql_exception;  
private ResultSet rs = mock(ResultSet.class);  
private ByteArrayOutputStream outContent;  
private ByteArrayOutputStream errContent;  
private PrintStream originalOut;  
private PrintStream originalErr;
```

## Mockito 用来模拟对象并且为对象打桩





# 设计思想

```
private BuildingDao buildingDao;  
private Connection connection = mock(Connection.class);  
private PreparedStatement preparedStatement = mock(PreparedStatement.class);  
private SQLException test_sql_exception;  
private ResultSet rs = mock(ResultSet.class);  
private ByteArrayOutputStream outContent;  
private ByteArrayOutputStream errContent;  
private PrintStream originalOut;  
private PrintStream originalErr;
```

首先，Mock与数据库有关的对象，使单元测试能够不依赖外部资源

```
class TestableBuildingDaoImpl extends BuildingDaoImpl {  
    @Override  
    public Connection getConnection() { return connection; }  
}
```

构建子类重写getConnection方法，让它能返回自己mock的对象



# 设计思想

```
@BeforeEach
void init() {
    this.buildingDao = new TestableBuildingDaoImpl();
    this.test_sql_exception = new SQLException();
    this.outContent = new ByteArrayOutputStream();
    this.errContent = new ByteArrayOutputStream();
    this.originalOut = System.out;
    this.originalErr = System.err;
    System.setOut(new PrintStream(outContent));
    System.setErr(new PrintStream(errContent));
}
```

```
@AfterEach
void tear_down() throws IOException {
    System.setErr(this.originalErr);
    System.setOut(this.originalOut);
    this.outContent.close();
    this.errContent.close();
}
```

```
when(preparedStatement.executeQuery()).thenReturn(rs);
when(rs.next()).thenReturn(t: true, ...ts: false); /* First call returns t:
when(rs.getInt(columnLabel: "id")).thenReturn(id);
when(rs.getString(columnLabel: "name")).thenReturn(name);
when(rs.getString(columnLabel: "description")).thenReturn(description);
when(rs.getString(columnLabel: "price")).thenReturn(price);
```

为Mock对象打桩,

改变输出流, 重定向输出, 方便之后的异常判断



## 遇到的问题



### 问题1：无法为mock对象的静态方法打桩

例：待测方法调用System.currentTimeMillis

```
long currentTime = System.currentTimeMillis();
```

### 解决方法：

查阅资料，得知PowerMockito可以为静态方法打桩，但是PowerMockito暂不支持JUnit5，因此寻找其他办法。

### 妥协：

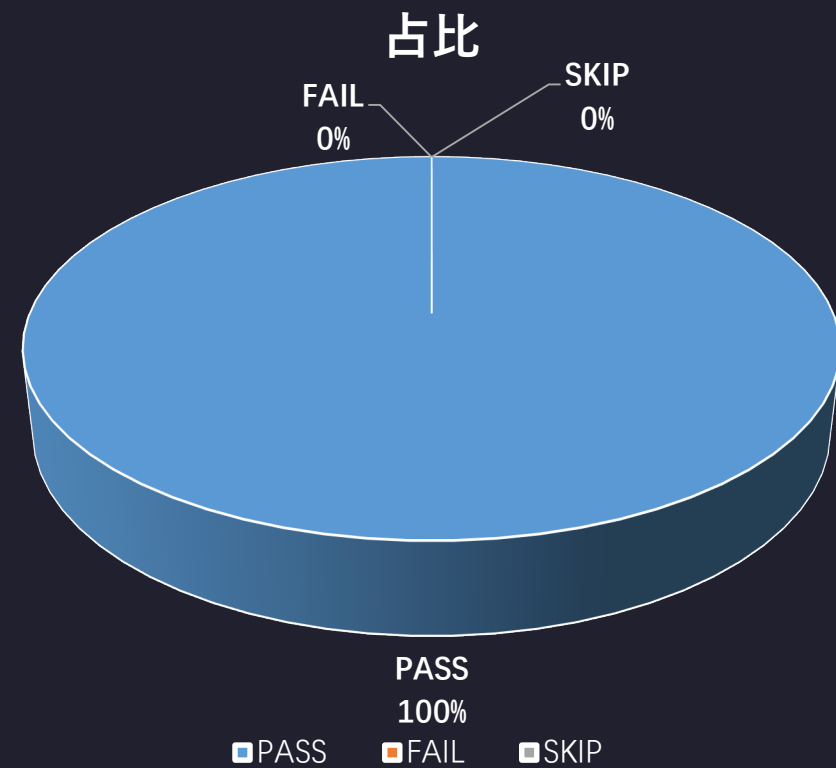
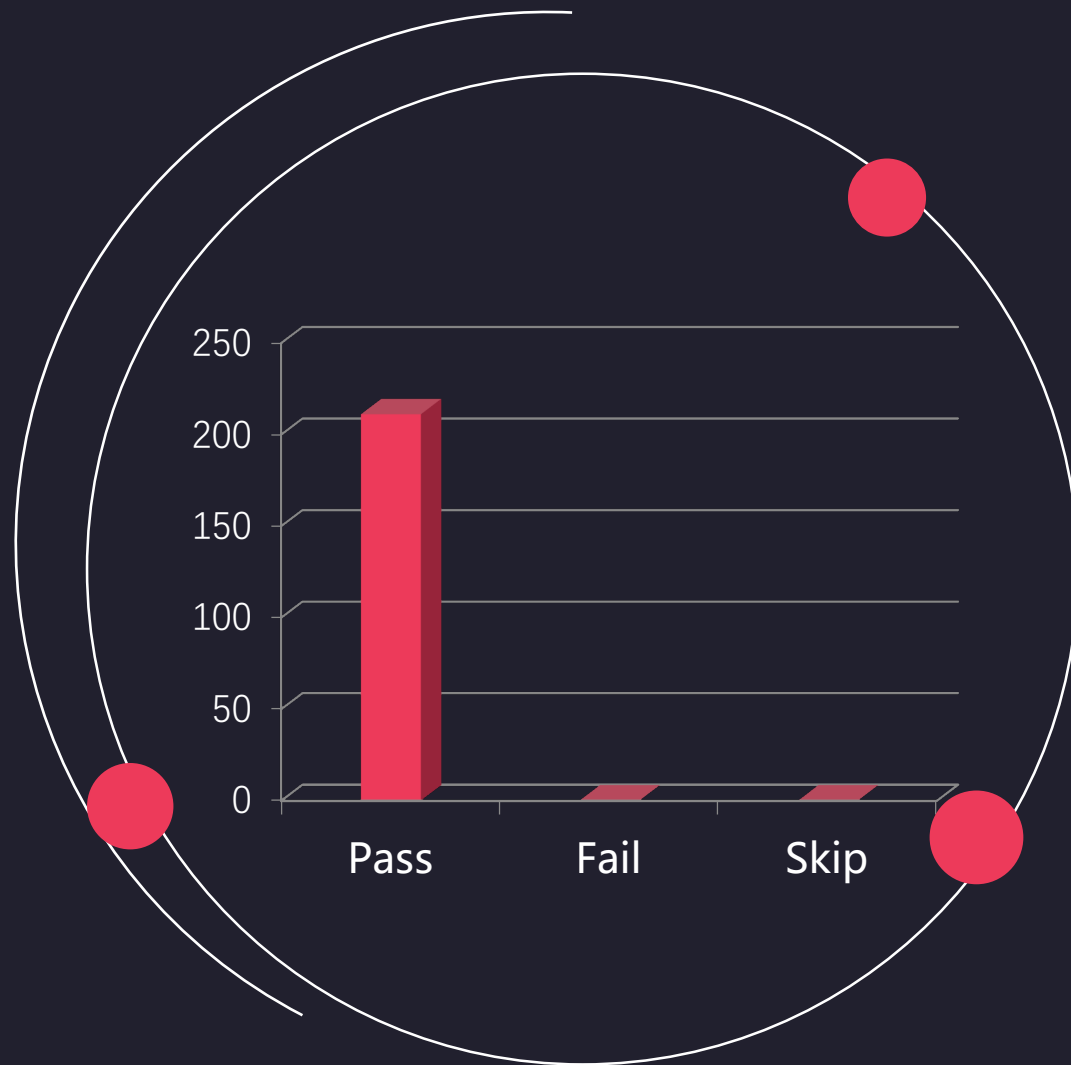
对涉及到System.currentTimeMillis的属性，在Assert时忽略对时间的判断。



## 测试结果

通过率100%

测试用例  
全部通过





# 测试总结

单元测试良好，通过率达到100%

覆盖率达到初步标准

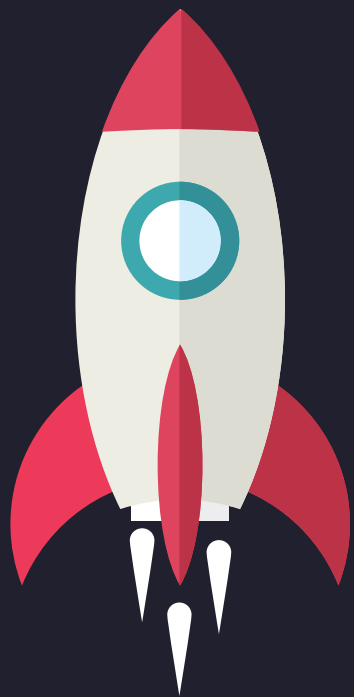
Suites						
order	name	duration	status	Filter by status:	0038410	
>	com.webapp.controller.BlankControllerIT				3	
>	com.webapp.controller.BlankControllerTest				3	
>	com.webapp.controller.BuildingControllerIT				23	
>	com.webapp.controller.BuildingControllerTest				13	
>	com.webapp.controller.CommentControllerIT				19	1
>	com.webapp.controller.IndexControllerIT				1	
>	com.webapp.controller.IndexControllerTest				1	
>	com.webapp.controller.LoginControllerIT				21	
>	com.webapp.controller.LoginControllerTest				3	
>	com.webapp.controller.NewsControllerIT				15	
>	com.webapp.controller.NewsControllerTest				10	
>	com.webapp.controller.PasswordControllerIT				6	
>	com.webapp.controller.PasswordControllerTest				5	
>	com.webapp.controller.RecordControllerIT				16	
>	com.webapp.controller.RecordControllerTest				9	
>	com.webapp.controller.SignupControllerIT				11	
>	com.webapp.controller.SignupControllerTest				3	
>	com.webapp.controller.UserControllerIT				34	
>	com.webapp.controller.UserControllerTest				13	

com.webapp.controller

100% (10/10)

100% (44/44)

99% (450/451)

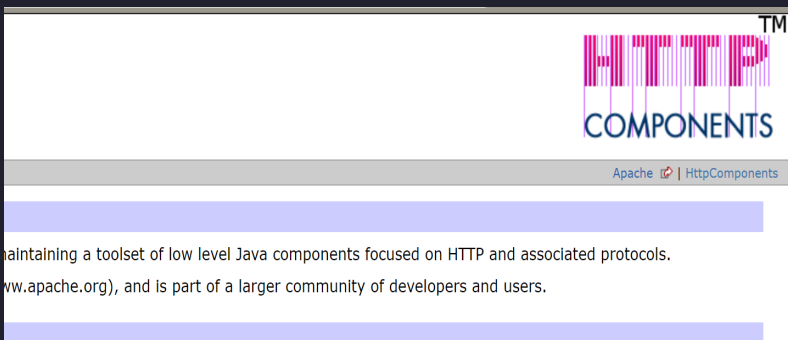


# 集成测试

彭钧涛 李尚真



## 设计思想



## 基于Apache HttpComponents的测试

JAVA代码发送Http请求，以  
测试系统功能



# Apache HttpComponents

# Java Development Kit

## 用于将录制的操作通过java代码回放在Chrome浏览器中

# JUnit

## 用于判断测试结果





## 难点

### Dao的配置

WebApp没有正式部署,  
Dao无法连接数据库



```
36  * @author Juntao Peng
37  */
38  @ExtendWith(SpringExtension.class)
39  @WebAppConfiguration
40  @ContextConfiguration(classes = MvcConfig.class)
41  public class SignupControllerIT {
42      @Autowired
43      WebApplicationContext wac;
44      MockMvc mockMvc;
45
46      @Autowired
47      UserDao userDao;
```



# 集成测试脚本展示

## 示例：注册失败的测试

填写用户名、密码  
等信息

断言注册失败后的提示

```
118 @ParameterizedTest
119 @MethodSource("invalidUsernamePasswordPairProvider")
120 void shouldErrorMessage_WhenSignupFailed(String username, String password, String name,
121                                         String sex, String tel) throws Throwable {
122     this.mockMvc.perform(post( urlTemplate: "/signupSubmit")
123                          .contentType(MediaType.APPLICATION_FORM_URLENCODED)
124                          .content(EntityUtils.toString(new UrlEncodedFormEntity(Arrays.asList(
125                              new BasicNameValuePair( name: "username", username),
126                              new BasicNameValuePair( name: "password", password),
127                              new BasicNameValuePair( name: "name", name),
128                              new BasicNameValuePair( name: "sex", sex),
129                              new BasicNameValuePair( name: "tel", tel)
130                          )))))
131     .andExpect(status().isOk())
132     .andExpect(view().name( expectedViewName: "signup"))
133     .andExpect(forwardedUrl( expectedUrl: "/WEB-INF/jsp/signup.jsp"))
134     .andExpect(model().attribute( name: "error", value: "该用户名已存在"));
135 }
136 }
```



# 集成测试脚本展示

```
@ParameterizedTest
@MethodSource("illegalUserTypeAndActionProvider")
void givenUserTypeWhenActionIsIllegalThenList(String userType, String action)
    throws Exception {
    Map<String, Object> sessionAttrs = new HashMap<>();
    sessionAttrs.put("currentUserType", userType);
    final String viewName;
    if ("admin".equals(userType)) {
        viewName = "mainAdmin";
    } else {
        viewName = "mainUser";
    }
}
```

参数化测试

用于简化测试代码

```
static Stream<Arguments> illegalUserTypeAndActionProvider() {
    return Stream.of(
        arguments("user", "add"),
        arguments("user", "delete"),
        arguments("user", "modify"),
        arguments("user", "save"),
    );
}
```



# 缺陷报告

Via GitHub Issues

☐  Find bug during RecordController integration test **bug**

#8 by Renn was closed 9 days ago

☐  MeetHere #33 failed

#7 by UncooleBen was closed 9 days ago

☐  Find bug during NewsController integration test **bug**

#6 by Renn was closed 9 days ago

☐  A bug during CommentController integration test **bug**

#5 by Renn was closed 10 days ago

☐  MeetHere #10 failed

#4 by UncooleBen was closed 15 days ago



# 缺陷报告

## Via GitHub Issues

UncooleBen / MeetHere

Unwatch 1Unstar 4Fork 0

<> Code

Issues 0

Pull requests 0

Actions

Projects 0

Wiki

Security

Insights

Settings

Find bug during RecordController integration test #8

EditNew issue

Closed

Renn opened this issue 9 days ago · 1 comment

Renn commented 9 days ago

Collaborator + 🧑

When testing with action save, a null pointer exception was thrown.

🔍

Renn added the **bug** label 9 days ago

👤

Renn self-assigned this 9 days ago

Renn commented 9 days ago

Collaborator Author + 🧑

The reason is that there is no field named 'userId' in the request. So I changed the way controller getting the userId.

🔒

Renn closed this 9 days ago

Assignees

Renn

Labels

bug

Projects

None yet

Milestone

No milestone

Notifications

Customize

🔊 Unsubscribe

You're receiving notifications because you're watching this repository.



# 集成测试结果

## Allure报告

Allure	
Overview	
Categories	
Suites	
Graphs	
Timeline	
Behaviors	
Packages	
> com.webapp.controller.IndexControllerIT	1
> com.webapp.service.database.DatabaseServiceIT	2
> com.webapp.controller.PasswordControllerIT	6
> com.webapp.controller.BuildingControllerTest	13
> com.webapp.controller.NewsControllerIT	15
> com.webapp.service.database.dao.impl.BuildingDaoImplIT	4
> com.webapp.service.database.dao.impl.CommentDaoImplIT	4
> com.webapp.controller.RecordControllerTest	9
> com.webapp.service.database.dao.impl.LoginDaoImplIT	3
> com.webapp.controller.SignupControllerIT	11
> com.webapp.service.database.dao.impl.BuildingDaoImplTest	13
> com.webapp.controller.UserControllerIT	34
> com.webapp.controller.BlankControllerIT	3
> com.webapp.controller.RecordControllerIT	16
> com.webapp.controller.CommentControllerIT	19 1
> com.webapp.controller.LoginControllerIT	21
> com.webapp.service.database.dao.impl.UserDaoImplIT	5
> com.webapp.service.database.dao.impl.NewsDaoImplIT	4
> com.webapp.service.database.dao.impl.RecordDaoImplIT	8
> com.webapp.controller.BuildingControllerIT	23



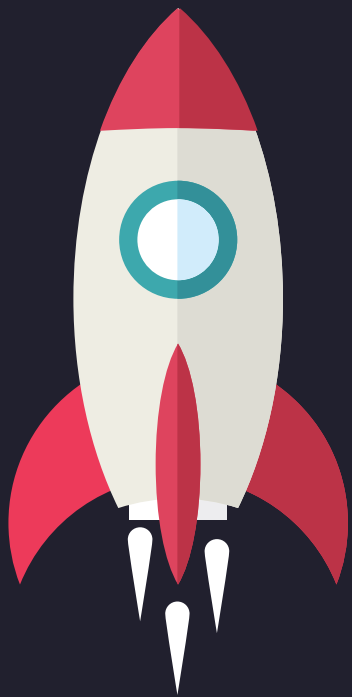
# 集成测试总结

## 自动化测试

测试时间成本降低，但是配置环境复杂

## Setup与TearDown

基于空表的测试需要添加数据库表项



# 系统测试

彭钧涛 李尚真

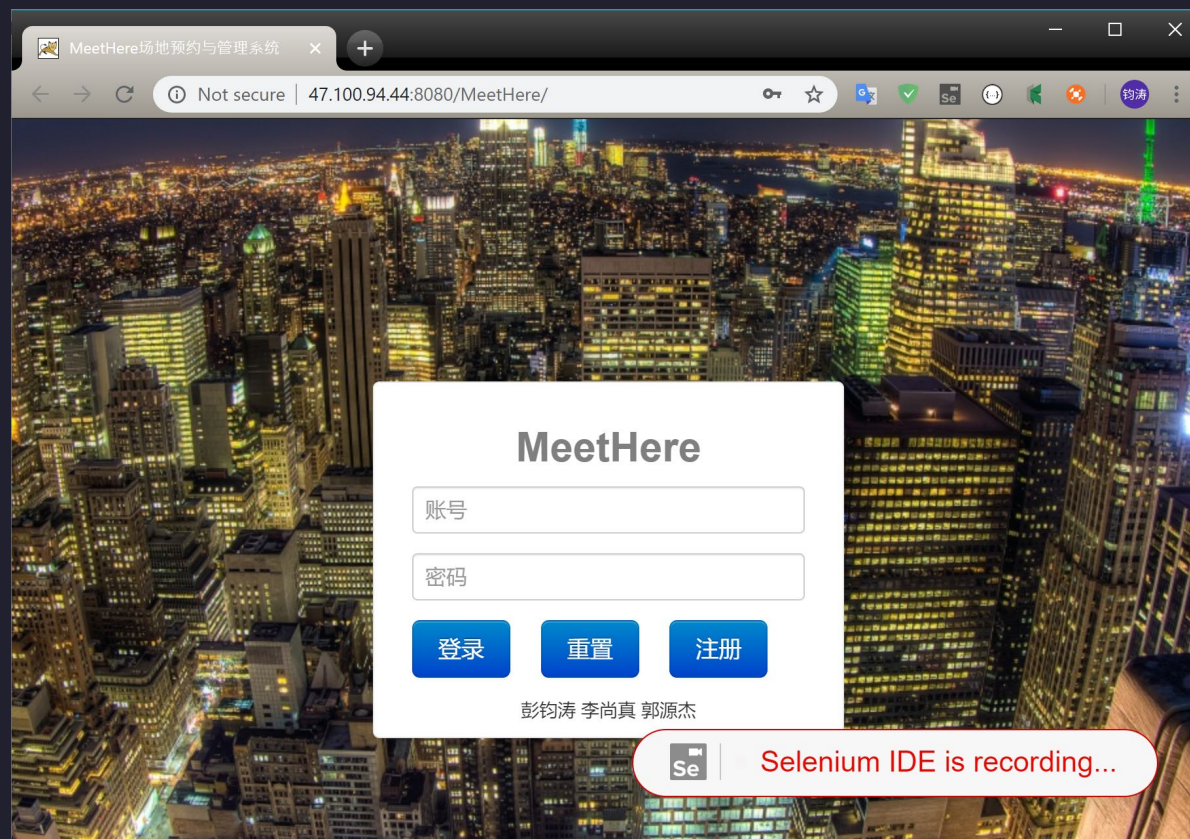




# 设计思想

## 基于Selenium的自动化测试

通过自动化的录制回放工具，  
模拟用户和浏览器前端的点击  
互动





# 使用工具

Selenium IDE - Test\*

Project: Test\*

Tests +

Search tests...

Test Case 1\*

Command	Target
1 open	http://47.100.94.44:8080/MeetHere/
2 set window size	925x730
3 click	id=#id_signup
4 click	id=username
5 type	id=username
6 type	id=name
7 type	id=password
8 click	id=tel
9 type	id=tel
10 click	css= btn:nth-child(1)

```
@ParameterizedTest
@MethodSource("wrongUserProvider")
@Order(1)
public void loginFail(String username, String password, String error) {
    // Test name: LoginFail
    // Step # | name | target | value | comment
    // 1 | open | http://localhost:8080/MeetHere war/ | |
    driver.get("http://localhost:8080/MeetHere war/");
    // 2 | setWindowSize | 1052x554 | |
    driver.manage().window().setSize(new Dimension( width: 1920, height: 1080));
    // 3 | click | id=username | |
    driver.findElement(By.id("username")).click();
    // 4 | type | id=username | 123 |
    driver.findElement(By.id("username")).sendKeys(username);
    // 5 | type | id=password | 123 |
    driver.findElement(By.id("password")).sendKeys(password);
    // 6 | click | css= btn:nth-child(1) | |
    driver.findElement(By.cssSelector(".btn:nth-child(1)")).click();
    assertEquals(error, driver.findElement(By.id("error")).getText());
}
```

## Selenium IDE

用于录制前端互动操作

## ChromeDriver + Selenium WebDriver

用于将录制的操作通过java代码回放在Chrome浏览器中

## JUnit

用于判断测试结果



## 难点

### HTML元素的获取

没有指定id的元素默认通过CSS样式获取很容易出错

```
// 6 | click | css= btn:nth-child(1) | |  
driver.findElement(By.cssSelector(".btn:nth-child(1)")).click();  
assertEquals(error, driver.findElement(By.id("error")).getText());  
}
```

### 填充内容的获取

Input tag内容和标签内容获取方式有所不同

```
ext("修改密码")).click();  
密码", driver.findElement(By.id("#id_title")).getText());  
child(1) | |  
tor(".btn:nth-child(1)")).click();  
findElement(By.id("username")).getAttribute( name: "value");  
findElement(By.id("password")).getAttribute( name: "value");
```



## 难点

### 测试准备阶段

部分测试准备代码需要  
基于空表

```
@BeforeEach
public void setUp() {
    // Protect original data in database
    List<User> originUserList = this.userDao.queryAllUsers();
    Set<Integer> originUserIdSet = new HashSet<>();
    for (User user : originUserList) {
        originUserIdSet.add(user.getId());
    }
    this.originNewsIdSet = new HashSet<>();
    List<News> originalNewsList = this.newsDao.listNews( size: 20);
    for (News news : originalNewsList) {
        this.originNewsIdSet.add(news.getId());
    }

    List<Comment> originUnverifiedCommentList = this.commentDao.listComment( size: 20, verified: false);
    List<Comment> originVerifiedCommentList = this.commentDao.listComment( size: 20, verified: true);
    Set<Integer> originUnverifiedCommentIdSet = new HashSet<>();
    Set<Integer> originVerifiedCommentIdSet = new HashSet<>();
}
```



# 系统测试脚本展示

## 示例：测试管理员成功登录

▶ service.database  
▼ systemtest  
    AdminST  
    LoginSignUpST  
    UserST

```
@ParameterizedTest
@MethodSource("correctAdminProvider")
@Order(2)
public void loginSucceedAdmin(String username, String password) {
    // Test name: LoginSucceedAdmin
    // Step # | name | target | value | comment
    // 1 | open | http://localhost:8080/MeetHere_war/ | |
    driver.get("http://localhost:8080/MeetHere_war/");
    // 2 | setWindowSize | 1052x554 | |
    driver.manage().window().setSize(new Dimension(width: 1920, height: 1080));
    // 3 | click | id=username | |
    driver.findElement(By.id("username")).click();
    // 4 | type | id=username | root |
    driver.findElement(By.id("username")).sendKeys(username);
    // 5 | type | id=password | root |
    driver.findElement(By.id("password")).sendKeys(password);
    // 6 | click | css=.btn:nth-child(1) | |
    driver.findElement(By.cssSelector(".btn:nth-child(1)")).click();
    // 7 | findElement | By.id("welcome") | |
    assertEquals("欢迎您,系统管理员!", driver.findElement(By.id("welcome")).getText());
}
```

填写用户名密码

断言登录成功后  
页面出现欢迎文字



# 系统测试脚本展示

## 示例：测试修改新闻

- ▶ service.database
- ▼ systemtest
  - AdminST
  - LoginSignUpST
  - UserST

设定窗口大小，否则  
无法按按钮

让线程休眠一段时间

```
@ParameterizedTest
@MethodSource("newsProvider")
public void modifyNews(
    String title, String author, String detail, boolean success, String errorMessage)
    throws Exception {
    driver.manage().window().setSize(new Dimension( width: 1920, height: 1080));
```

```
for (int newsId : this.addedNewsIdSet) {
    driver.get("http://localhost:8080/MeetHere_war/news?action=list");
    driver.findElement(By.id("modify" + newsId)).click();
    driver.findElement(By.cssSelector(".data_form")).click();
    driver.findElement(By.id("title")).clear();
```

```
    driver.findElement(By.id("author")).sendKeys(author);
    driver.findElement(By.id("detail")).clear();
    driver.findElement(By.id("detail")).sendKeys(detail);
    driver.findElement(By.cssSelector(".btn:nth-child(1)")).click();
    if (success) {
        Thread.sleep( millis: 500);
```

```
static Stream<Arguments> newsProvider() {
    return Stream.of(
        arguments("correct title", "correct author", "correct detail", true, ""),
        arguments("", "correct author", "correct detail", false, "标题不能为空!"),
        arguments("correct title", "", "correct detail", false, "作者不能为空!"),
        arguments("correct title", "correct author", "", false, "内容不能为空!"),
        arguments("", "", "", false, "内容不能为空!"));
}
```



# 缺陷报告

Via GitHub Issues

o.com/UncooleBen/MeetHere/issues?q=is%3Aissue+is%3Aclosed



☐ ⓘ 0 Open ✓ 13 Closed Author ▾ Label ▾ Projects ▾ Milestones ▾ Assignee ▾

☐ ⓘ MeetHere #89 failed  
#14 by UncooleBen was closed yesterday

☐ ⓘ Find bug in login&signup system test bug  
#13 by UncooleBen was closed 4 days ago



☐ ⓘ Find bug during user system test bug  
#12 by Renn was closed 4 days ago



☐ ⓘ MeetHere #81 failed  
#11 by UncooleBen was closed 4 days ago




# 缺陷报告

## Via GitHub Issues

### Find bug in login&signup system test #13


Edit New issue


**Closed** UncooleBen opened this issue 4 days ago · 1 comment


 UncooleBen commented 4 days ago

Owner +😊 ...

The javascript `checkForm()` function won't work when one of the if statement contains bug.


 UncooleBen added the **bug** label 4 days ago


 UncooleBen self-assigned this 4 days ago

 UncooleBen commented 4 days ago

Owner Author +😊 ...

Bug fixed!


 UncooleBen closed this 4 days ago

Assignees 

 UncooleBen

Labels 

**bug**


Projects 

None yet

Milestone 

No milestone

Notifications [Customize](#)

 Unsubscribe

You're receiving notifications because you're watching this repository.





# 系统测试结果

## Allure报告

> com.webapp.systemtest.AdminST	33
> com.webapp.systemtest.LoginSignUpST	37
> com.webapp.systemtest.UserST	6

Allure		
<div>Overview</div> <div>Categories</div> <div>Suites</div> <div>Graphs</div> <div>Timeline</div>		
▼ com.webapp.systemtest.AdminST 33 ▲		
✓ #5	[1] addbuildingname1, 300, addbuildingdescription1	6s 057ms
✓ #22	[1] correct title, correct author, correct detail, true,	8s 219ms
✓ #28	[1] correct title, correct author, correct detail, true,	6s 666ms
✓ #8	[1] newbuildingname1, 300, newbuildingdescription1	6s 180ms
✓ #2	[1] newusername, newpassword, newname, MALE, newtel	6s 746ms
✓ #14	[1] passwordpassword1, root, root, true,	5s 987ms
✓ #23	[2] , correct author, correct detail, false, 标题不能为空!	6s 707ms
✓ #29	[2] , correct author, correct detail, false, 标题不能为空!	6s 357ms
✓ #6	[2] addbuildingname1, 500, addbuildingdescription2	5s 770ms
✓ #9	[2] newbuildingname1, 500, newbuildingdescription2	6s 097ms
✓ #3	[2] newusername, newpassword, newname, FEMALE, newtel	6s 823ms
✓ #15	[2] wrongOldPassword, root, root, false, 旧密码错误	6s 028ms
✓ #24	[3] correct title, , correct detail, false, 作者不能为空!	6s 662ms
✓ #30	[3] correct title, , correct detail, false, 作者不能为空!	6s 715ms
✓ #16	[3] passwordpassword1, root, root1, false, 密码填写不一致!	6s 221ms
✓ #17	[4] , root, root, false, 信息填写不完整!	6s 084ms
✓ #25	[4] correct title, correct author, , false, 内容不能为空!	6s 892ms
✓ #31	[4] correct title, correct author, , false, 内容不能为空!	6s 332ms
✓ #26	[5] , , , false, 内容不能为空!	6s 511ms
✓ #32	[5] , , , false, 内容不能为空!	6s 260ms
✓ #18	[5] passwordpassword1, , root, false, 信息填写不完整!	6s 196ms



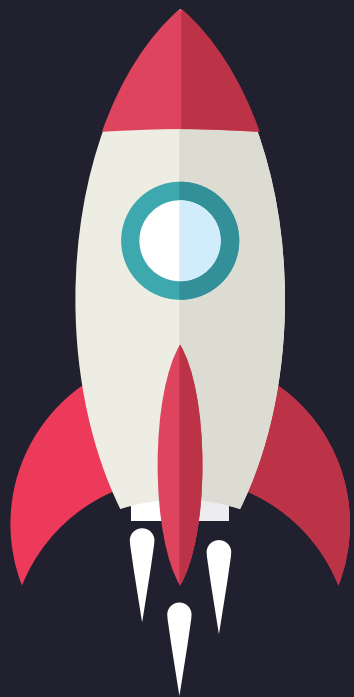
# 系统测试总结

## 自动化测试

测试时间成本降低，但是配置环境复杂

## 无法测试安全性

非法输入由JS屏蔽，但是无法阻止直接通过Chrome Console发送的非法请求



# 性能测试

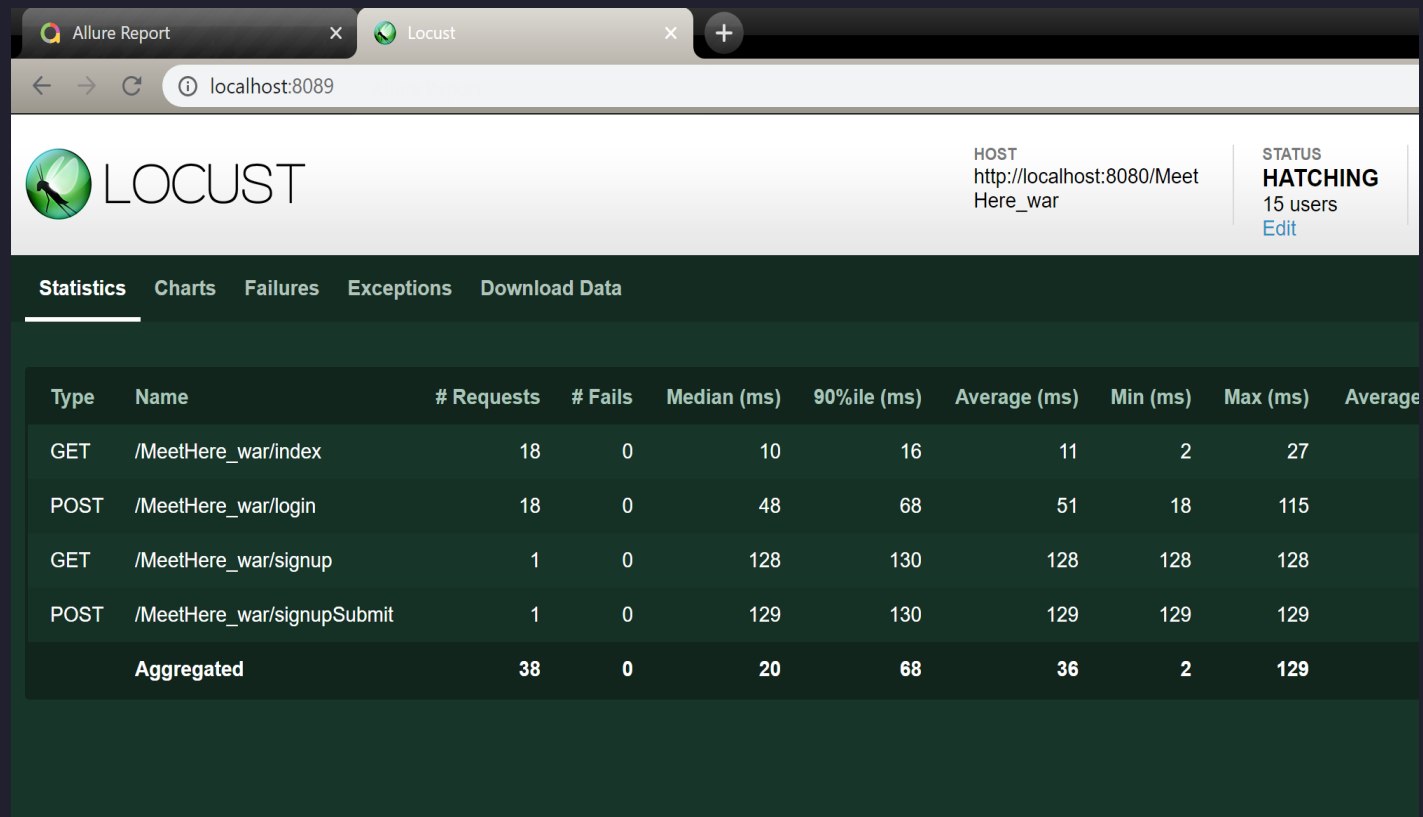
彭钧涛 李尚真



# 设计思想

## 基于Locust的压力测试

一个基于Python脚本的压力测试工具





# 使用工具

## Locust

一个基于Python脚本的压力测试工具

## Python

用于编写测试脚本

```
changPasswordWhenOldPasswordsWrong.py — C:\Users\Juntao Peng\Desktop\Software Testing\project\MeetHere\src\test
File Edit View Selection Find Packages Help

changPasswordWhenOldPass... x

1  from locust import HttpLocust, TaskSet, task, between
2
3  class WebsiteTasks(TaskSet):
4      def on_start(self):
5          self.client.post("/login", {
6              "username": "user",
7              "password": "user"
8          })
9
10     @task
11     def showNewsDetail(self):
12         self.client.post("/password?action=save",{
13             "userId": "2241",
14             "oldPassword": "wrongOldPassword",
15             "newPassword": "user001"
16         })
17
18     class WebsiteUser(HttpLocust):
19         task_set = WebsiteTasks
20         wait_time = between(5, 15)
21
```



## 难点

### 加压时间的计算

没有实际生产经验

Table 3 独立场景设计

并发数	响应时间要求	TPS 要求	加压时间	加压方式
1	<1 秒	N/A	3 分钟	直接加压
10	<1 秒	N/A	5 分钟	1 个用户/5 秒
20 (异常流)	<1 秒	>40 笔/秒	5 分钟	1 个用户/5 秒
50	<1 秒	>100 笔/秒	10 分钟	1 个用户/5 秒
100	<1 秒	>200 笔/秒	10 分钟	1 个用户/5 秒
200 (正常流)	<1 秒	>500 笔/秒	10 分钟	1 个用户/1 秒



# 性能测试脚本展示

## 示例：更改密码

填写旧密码、新  
密码

```
changPasswordWhenOldPass...
1  from locust import HttpLocust, TaskSet, task, between
2
3  class WebsiteTasks(TaskSet):
4      def on_start(self):
5          self.client.post("/login", {
6              "username": "user",
7              "password": "user"
8          })
9
10     @task
11     def change_password(self):
12         self.client.post("/password?action=save",{
13             "userId":"2241",
14             "oldPassword":"wrongOldPassword",
15             "newPassword":"user001"
16         })
17
18     class WebsiteUser(HttpLocust):
19         task_set = WebsiteTasks
20         wait_time = between(5, 15)
21
```



# 性能测试脚本展示

## 示例：混合场景-预约场馆

从用户登录到用户退出



```
from locust import HttpLocust, TaskSet, task, between

class WebsiteTasks(TaskSet):
    def on_start(self):
        pass

    @task
    def showNewsDetail(self):
        self.client.post("/login", {
            "username": "user",
            "password": "user"
        })
        self.client.get("/building?action=list")
        self.client.post("/building?action=book",{
            "buildingId":"632",
            "startDate":"2020-01-02",
            "duration":"2"
        })
        self.client.get("/index")

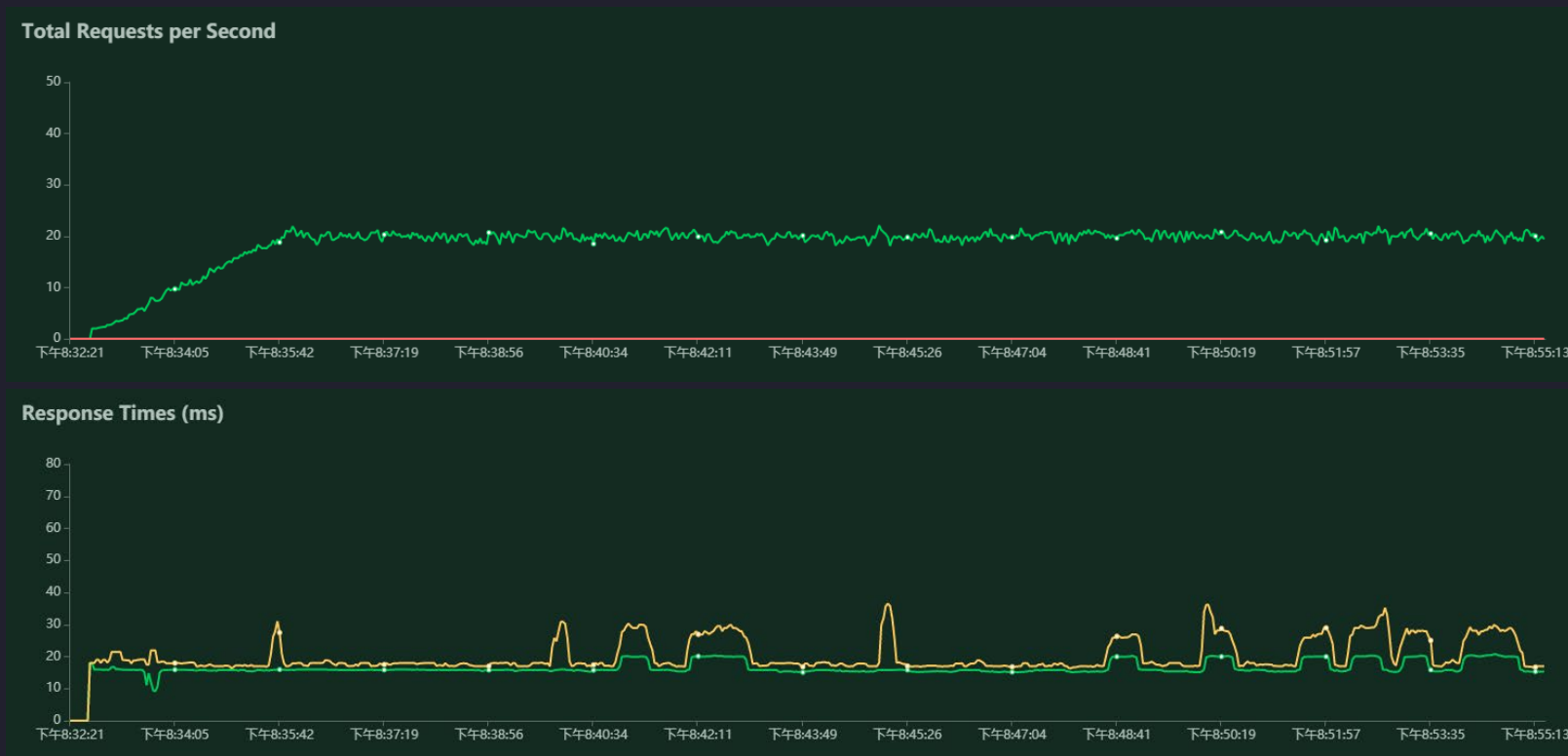
class WebsiteUser(HttpLocust):
    task_set = WebsiteTasks
    wait_time = between(5, 15)
```





# 性能测试结果

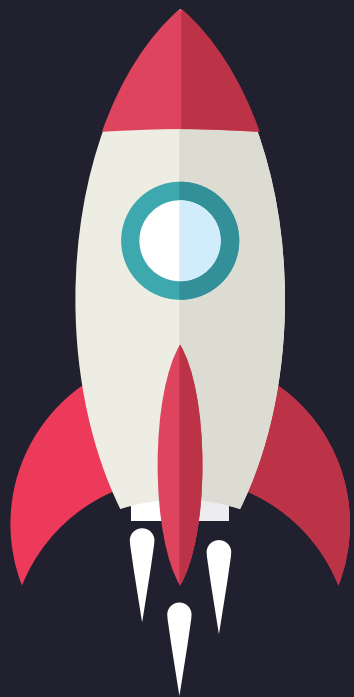
## Locust Chart





## 性能测试总结

测试开始日期	2019.12.30
测试结束日期	2020.01.01
提交缺陷数	1
延期/未解决缺陷数	0
拒绝缺陷数	0
总测试场景	8个独立场景+5个混合场景
执行场景数	8个独立场景+5个混合场景
场景执行率	100%
缺陷率（%）	7.7%（十三分之一）
投入人数	2
系统上线建议	部分页面随加压响应时间上升需改进



# 项目演示

**谢谢大家！**