

# COLLECT众测平台使用指南

Group: 191250166\_outlast

## COLLECT众测平台使用指南

Group: 191250166\_outlast

### 1. 部署介绍

1. jenkins部署详情
2. jenkins脚本
3. webhook实现截图
4. Jacoco生成覆盖率报告

### 2. 网页功能及页面跳转介绍

1. 首页
1. 登录界面
  - 2.1. 登录成功
  - 2.2. 登录失败
    - 2.2.1. 用户名不存在
    - 2.2.2. 密码输入错误
    - 2.2.3. 未填写用户名
3. 注册界面
  - 3.1. 注册成功
  - 3.2. 注册失败
    - 3.2.1. 用户名已被使用
    - 3.2.2. 未填写用户名
    - 3.2.3. 未填写密码
4. 任务广场（众包工人版）
  - 4.1. 任务详情界面
  - 4.2. 退出登录
5. 接受任务
  - 5.1. 接受任务
    - 5.1.1. 接受任务成功
6. 已接受任务界面（众包工人版）
  - 6.1. 下载待测试应用和待测文件
  - 6.2. 填写报告
  - 6.3. 报告提交成功
  - 6.2. 报告提交失败
    - 6.2.1. 未填写缺陷情况描述
    - 6.2.2. 未填写缺陷复现步骤
    - 6.2.3. 未填写测试设备信息
7. 任务广场（发包方版）
  - 7.1. 任务详情界面
  - 7.2. 退出登录
8. 创建新任务界面（发包方版）
9. 已发布任务界面（发包方版）

## 1. 部署介绍

### 1. jenkins部署详情

Jenkins上创建了两条流水线，分别负责前端和后端的构建部署。前端的步骤：从git上获取代码、npm构建、使用nginx部署、将结果返回git。后端的步骤：从git上获取代码、重置数据库、maven构建jar包并将jacoco的覆盖率报告通过nginx部署、将jar包部署、将结果返回git。

## 2. Jenkins脚本

前端：

```
pipeline {
    agent any

    stages {
        stage('Git') {
            steps {
                git credentialsId: 'ed08ca8b-f78c-40d4-8170-0eaf29cd6b7b', url:
'https://git.nju.edu.cn/191250194/frontend-outlast.git'
            }
        }
        stage('Build') {
            steps {
                sh "npm install"
                sh "npm run build"
            }
        }
        stage('Deploy') {
            steps {
                sh "rm -rf /home/molgypeter/frontend/*"
                sh "cp -r dist/* /home/molgypeter/frontend"
            }
        }
    }

    post {
        success{
            updateGitlabCommitStatus name: 'build', state: 'success'
        }
        failure {
            updateGitlabCommitStatus name: 'build', state: 'failed'
        }
    }
}
```

后端：

```
pipeline {
    agent any

    stages {
        stage('Git') {
            steps {
                git credentialsId: 'ed08ca8b-f78c-40d4-8170-0eaf29cd6b7b', url:
'https://git.nju.edu.cn/191250194/backend-outlast.git'
            }
        }
    }
}
```

```

stage('SQL') {
    steps {
        sh "mysql -uroot -p123456 -h localhost <
/home/molgypeter/.jenkins/workspace/backend-outlast/sql/collect.sql "
    }
}
stage('Build') {
    steps {
        sh "mvn clean package"
        sh "rm -rf /home/molgypeter/jacoco/* && cp -r
target/site/jacoco/* /home/molgypeter/jacoco"
    }
}
stage('Deploy') {
    steps {
        //sh "cd /home/molgypeter/backend/ && ./run_backend.sh"
        sh ''
        SOURCE_DIR=/home/molgypeter/.jenkins/workspace/backend-
outlast/target
        TARGET_DIR=/home/molgypeter/backend

        JAR_NAME=`ls $SOURCE_DIR | grep '\.*\\.jar$'`
        if [ -n "$JAR_NAME" ];then
            echo '\删除旧版本'
            rm $TARGET_DIR/*.jar && cp $SOURCE_DIR/$JAR_NAME $TARGET_DIR
        fi
        echo '\检查8081端口是否被占用...'
        pid_blog=`netstat -tunpl | grep 8081|awk '{printf $7}'|cut -d/
-f1`  

        if [ "$pid_blog" != "" ];
        then
            echo '\8081端口被占用'
            kill -9 "$pid_blog"
            echo $pid_blog '\进程已被杀死'
        else
            echo "端口未被占用"
        fi
        echo '\后台运行jar包...'
        fileName=`date +log-%Y-%m-%dT%H%M%S.txt`  

        BUILD_ID=DONTKILLME
        nohup java -jar $TARGET_DIR/$JAR_NAME --server.port=8081
>$TARGET_DIR/logs/$fileName 2>&1 &
        echo '\jar包部署结束'
        ''
    }
}
}

post {
    success{
        updateGitlabCommitStatus name: 'build', state: 'success'
    }
    failure {
        updateGitlabCommitStatus name: 'build', state: 'failed'
    }
}
}

```

### 3. webhook实现截图

git push到172.29.4.49上的仓库时通过Mirroring repositories推送到git.nju.edu.cn的对应仓库。

Mirrored repositories	Direction	Last update attempt	Last successful update	
<a href="https://****:****@git.nju.edu.cn/191250194/backend-outlast.git">https://****:****@git.nju.edu.cn/191250194/backend-outlast.git</a>	Push	16 hours ago	16 hours ago	

git.nju.edu.cn的仓库接收到push之后使用webhook触发jenkins的构建。

Project Hooks (1)

<http://114.212.243.24:8082/project/backend-outlast> Test Edit Delete

Push Events SSL Verification: enabled

jenkins项目接受到后开始构建。

**构建触发器**

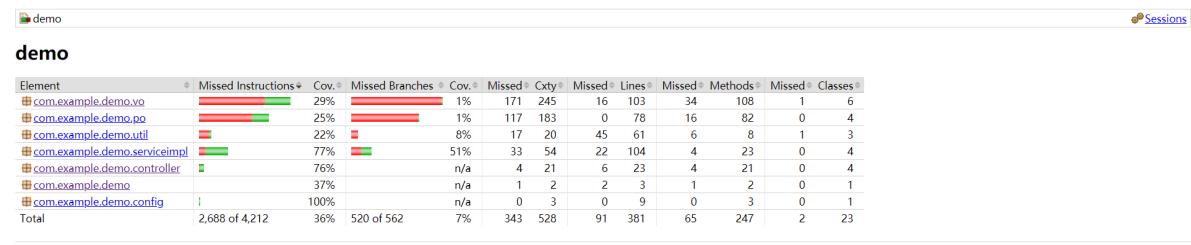
Build after other projects are built ?  
 Build periodically ?  
 Build when a change is pushed to GitLab. GitLab webhook URL: <http://114.212.243.24:8082/project/backend-outlast> ?

**Enabled GitLab triggers**

Push Events  
 Push Events in case of branch delete  
 Opened Merge Request Events  
 Build only if new commits were pushed to Merge Request ?  
 Accepted Merge Request Events  
 Closed Merge Request Events

Rebuild open Merge Requests

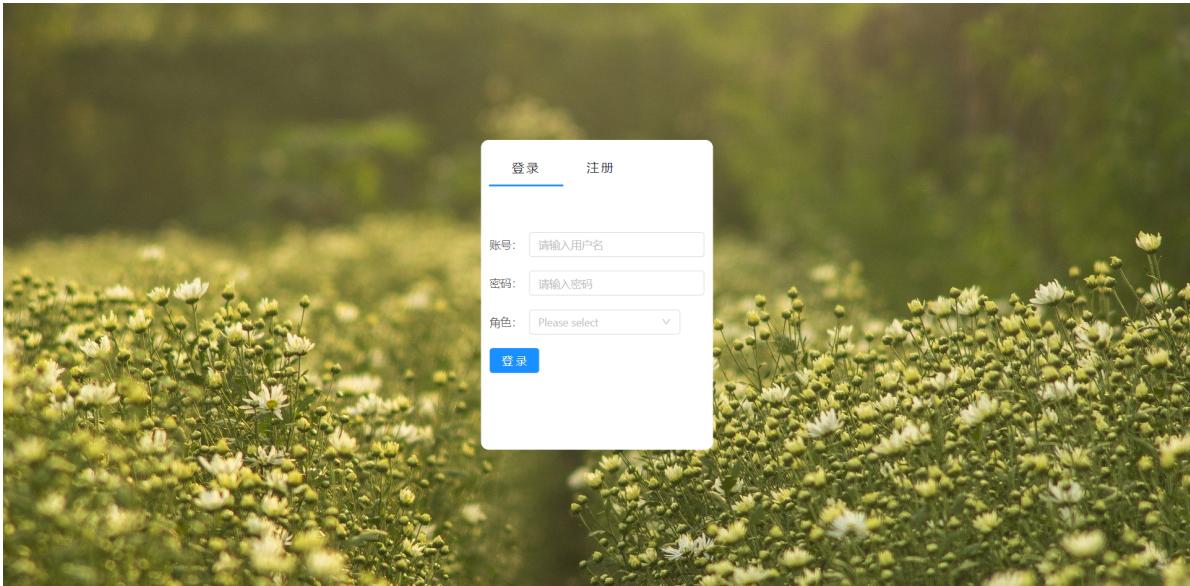
### 4. Jacoco生成覆盖率报告



## 2. 网页功能及页面跳转介绍

### 1. 首页

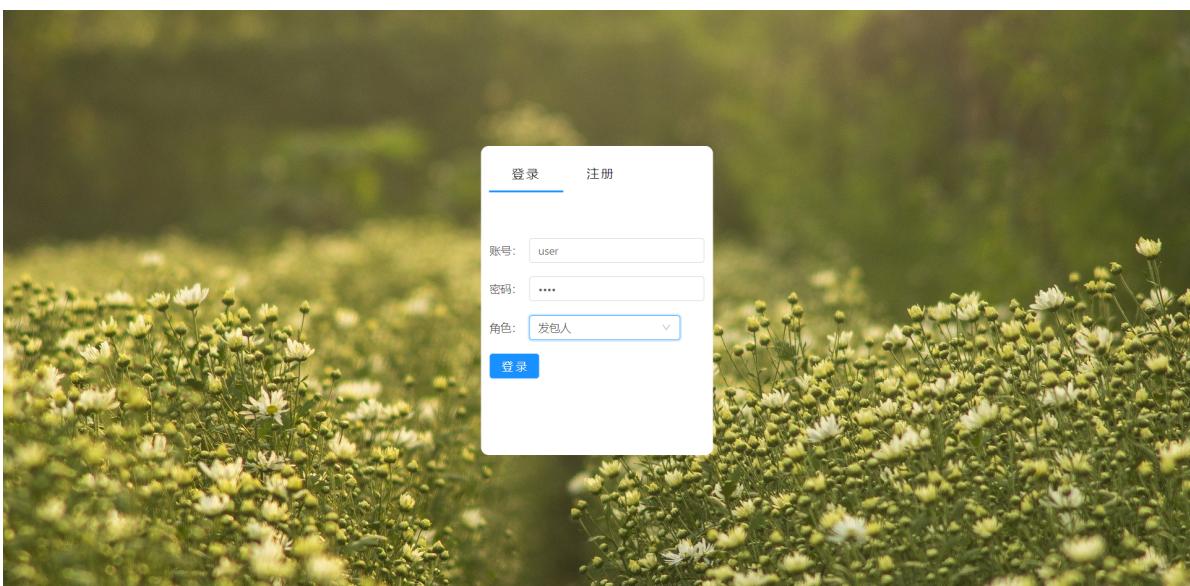
打开网址<http://114.212.243.24:80>, 进入登录界面。 (暂时部署在机房电脑上, 所以需要在校园网环境内, 在晚上十点前访问)



## 1. 登录界面

### 2.1. 登录成功

输入正确的用户名与密码，点击“登录”即可成功登录，进入任务广场界面



TaskSquire

任务广场

已发布任务

创建新任务

sadasdas  
截止日期: 2022-03-18  
任务描述: sss  
[任务详情](#) [接受任务](#)

12  
截止日期: 2022-03-16  
任务描述: 12  
[任务详情](#) [接受任务](#)

1234  
截止日期: 2022-03-11  
任务描述: 123  
[任务详情](#) [接受任务](#)

1  
截止日期: 2022-03-06  
任务描述: 1  
[任务详情](#) [接受任务](#)

1  
截止日期: 2022-03-06  
任务描述: 1  
[任务详情](#) [接受任务](#)

发包人

No Data

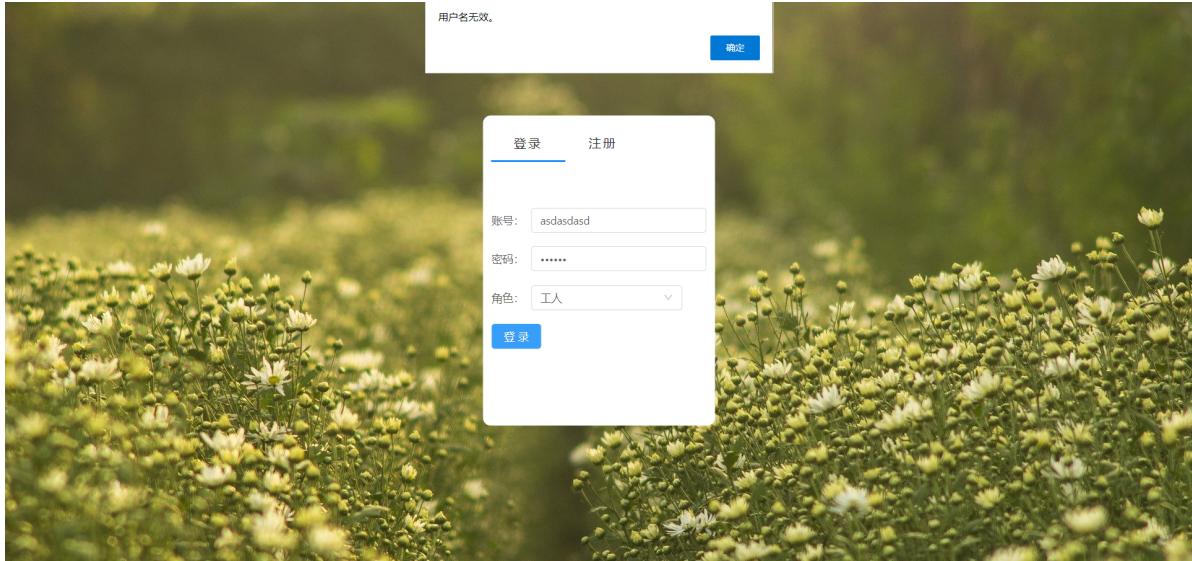
< 1 >

A screenshot of the TaskSquire task management platform. The top navigation bar shows 'TaskSquire' and '发包人' (Task Sender). On the left, there's a sidebar with '任务广场' (Task Square) and buttons for '已发布任务' (Published Tasks) and '创建新任务' (Create New Task). The main area displays five task cards. Each card contains a task ID, deadline, description, and two buttons: '任务详情' (Task Details) and '接受任务' (Accept Task). The first three cards have their details filled in, while the last two only show the ID and deadline. At the bottom right, there's a message icon and the text 'No Data' with navigation arrows (<, 1, >).

## 2.2. 登录失败

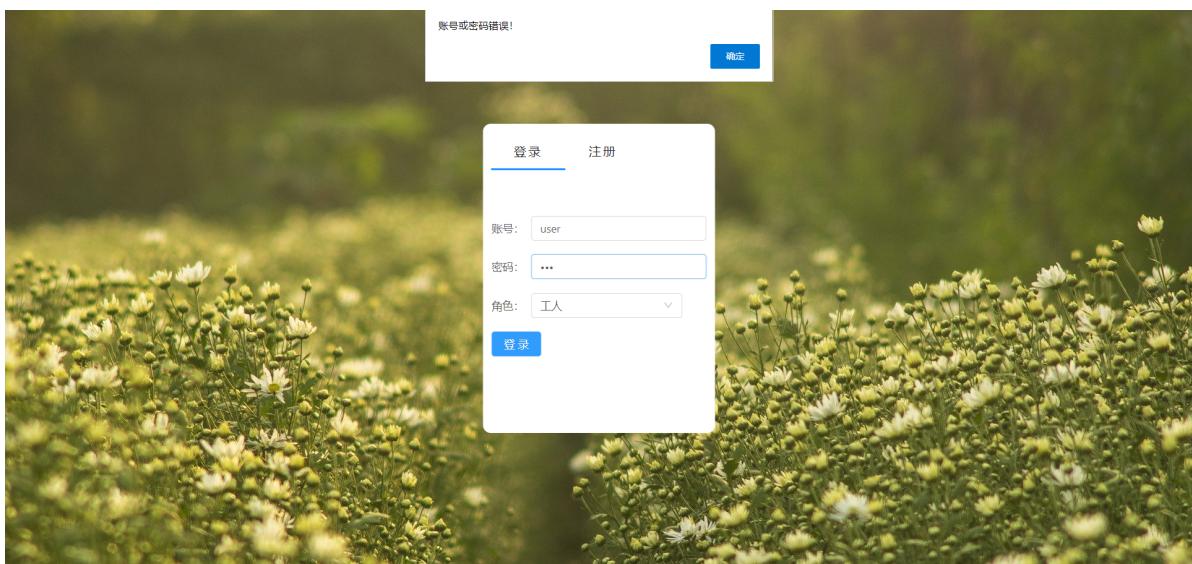
### 2.2.1. 用户名不存在

若该用户名在数据库中不存在，即尚未注册，则弹窗提示用户名无效。



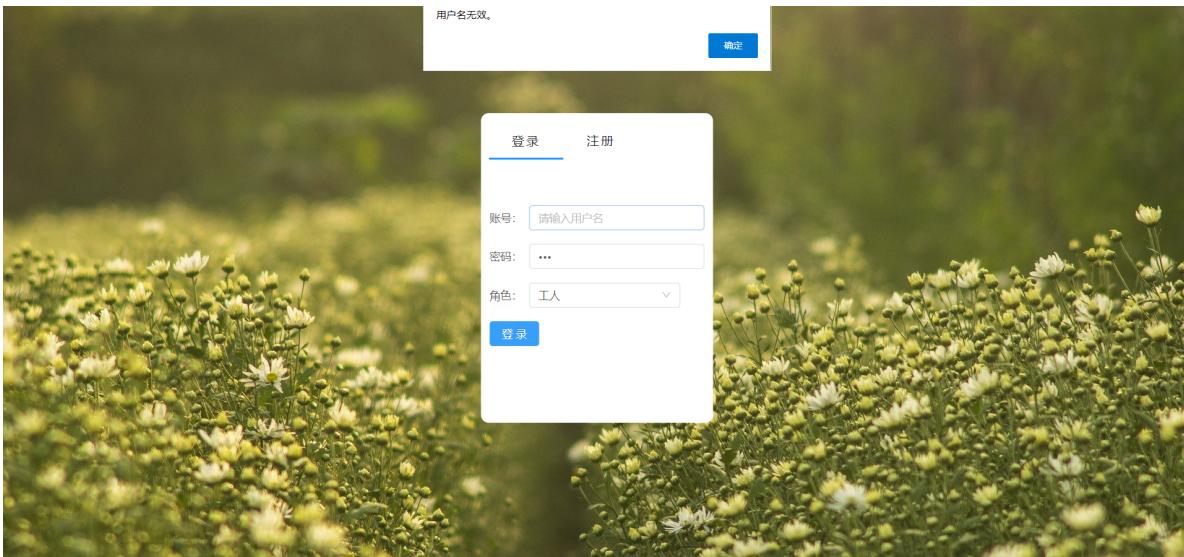
### 2.2.2. 密码输入错误

若用户名在数据库中存在，即已被注册，但密码与用户名不匹配，则弹窗提示账号或密码输入错误。



### 2.2.3. 未填写用户名

若未填写用户名，则弹窗提示用户名无效。

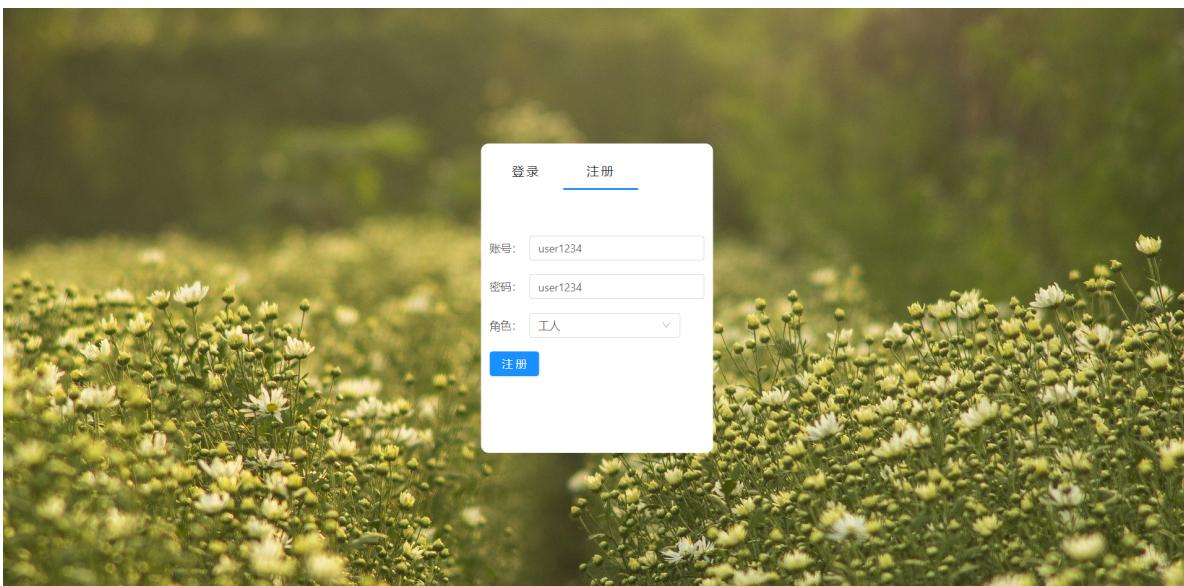


### 3. 注册界面

进入注册界面。

#### 3.1. 注册成功

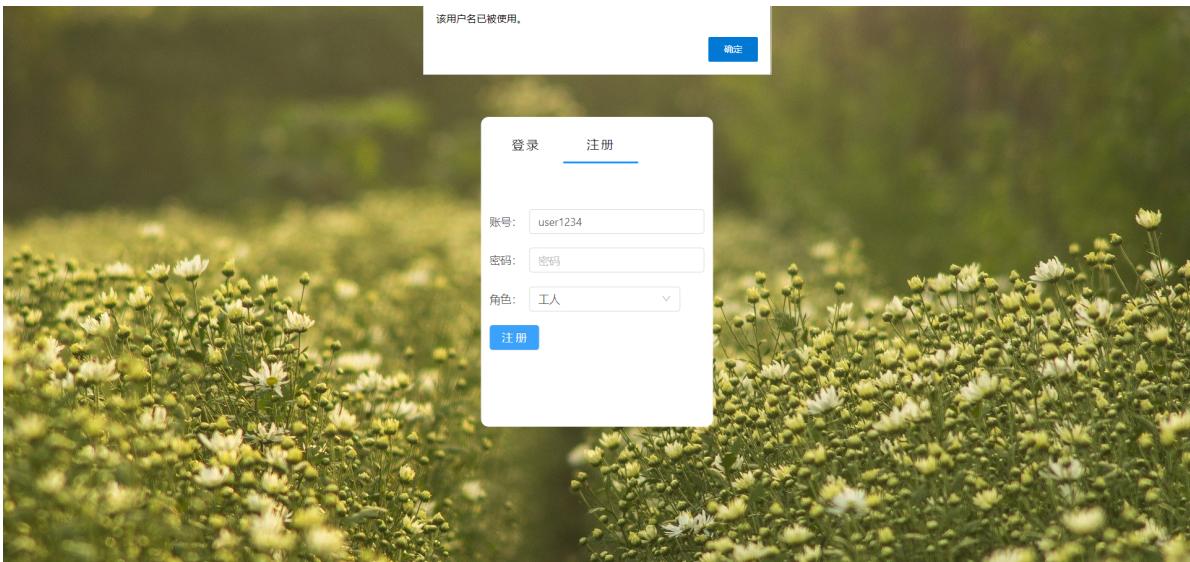
输入用户名、密码，并选择身份，点击“注册”即可成功注册，自动跳转至任务广场界面。



#### 3.2. 注册失败

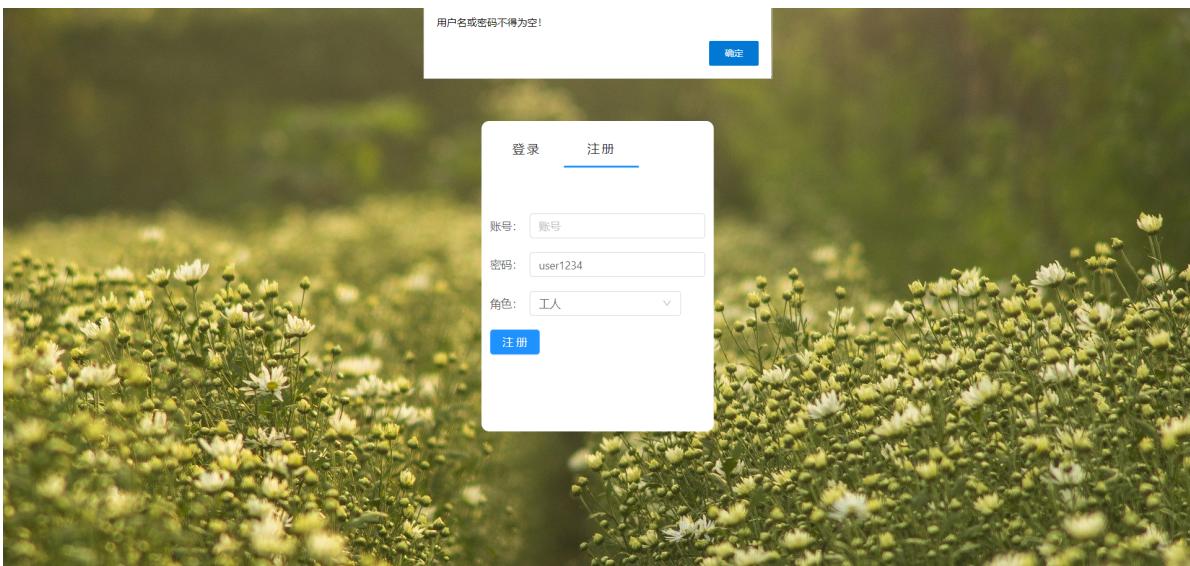
##### 3.2.1. 用户名已被使用

若该用户名在数据库中存在，即已被注册，则弹窗提示用户名已被使用。



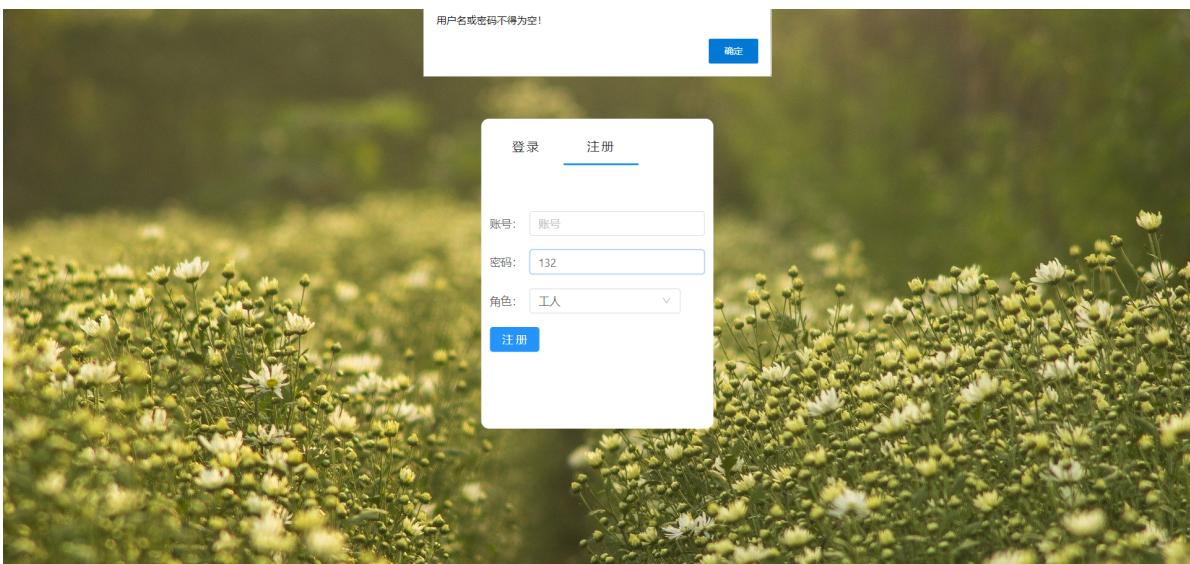
### 3.2.2. 未填写用户名

若未填写用户名，则弹窗提示用户名或密码不得为空。



### 3.2.3. 未填写密码

若未填写密码，则弹窗提示用户名或密码不得为空。



## 4. 任务广场 (众包工人版)

进入众包工人的任务广场，可浏览任务广场、历史任务、已接收任务界面。

The screenshot shows the TaskSquare interface for workers. At the top, there's a blue header bar with the text "TaskSquare" on the left and "工人" (Worker) on the right. Below the header is a sidebar on the left labeled "任务广场" (Task Square) with two sections: "历史任务" (Past Tasks) and "已接收任务" (Received Tasks). The main area displays a grid of five task cards. Each card contains a task ID, deadline, description, and two buttons: "任务详情" (Task Details) and "接受任务" (Accept Task). The first card has the ID "sadasdas", deadline "2022-03-18", and description "sss". The second card has the ID "12", deadline "2022-03-16", and description "12". The third card has the ID "1234", deadline "2022-03-11", and description "123". The fourth card has the ID "1", deadline "2022-03-06", and description "1". The fifth card has the same details as the fourth. At the bottom right of the main area, there's a "No Data" message with a computer icon and a "1" button.

### 4.1. 任务详情界面

可点击任意任务的任务详情右侧边栏即可显示详细信息界面。

This screenshot shows the TaskSquare interface with a detailed view of a task. The layout is similar to the previous one, with a sidebar on the left and a main task list. The task list shows five tasks. The third task, with ID "1234" and deadline "2022-03-11", is selected. A detailed sidebar on the right provides specific information for this task: "描述: sss", "截止时间: 2022-03-18T00:00:00", "类型: sss", "工作人数: 30", and "其他: 132". The bottom of the screen features a horizontal scrollbar.

### 4.2. 退出登录

可点击右上角下拉框中的“退出”，退出登录，回到首页。

The screenshot shows the TaskSquire platform's task acceptance interface. On the left, there is a sidebar with '历史任务' (Past Tasks) and '已接收任务' (Accepted Tasks). The main area displays a grid of tasks. Each task card contains a task ID, deadline, description, and two buttons: '任务详情' (Task Details) and '接受任务' (Accept Task). A modal window titled 'localhost:8080 显示' (Display) is open, asking '确定' (Confirm) if the user wants to accept the task. The right side of the screen shows a worker profile with basic information like name, description, and statistics.

## 5. 接受任务

### 5.1. 接受任务

#### 5.1.1. 接受任务成功

用户任务广场任意任务的“接受任务”，系统弹窗提示是否确认接受该任务，点击确认即可接受任务，系统提示任务接受成功。

This screenshot shows the same interface as the previous one, but with a different outcome. The modal window now says '任务接受成功' (Task accepted successfully) and has a '确定' (Confirm) button. The worker profile on the right is identical to the previous screenshot.

## 6. 已接受任务界面 (众包工人版)

进入已接受任务界面，可以查看任务，发包人，发布时间并下载可执行文件和文档，填写任务报告。

TaskSquire

工人

任务广场	任务	发布人	发布时间	下载可执行文件	下载文档	填写报告
历史任务	sadasdas	adcd	2022-03-05	<a href="#">点击下载</a>	<a href="#">点击下载</a>	<a href="#">点击填写</a>
已接收任务	1	123	2022-03-05	<a href="#">点击下载</a>	<a href="#">点击下载</a>	<a href="#">点击填写</a>
	1	123	2022-03-05	<a href="#">点击下载</a>	<a href="#">点击下载</a>	<a href="#">点击填写</a>
	12	user2	2022-03-05	<a href="#">点击下载</a>	<a href="#">点击下载</a>	<a href="#">点击填写</a>

< 1 >

## 6.1. 下载待测试应用和待测文件

可点击“下载可执行文件”和“下载文档”来下载文件。

TaskSquire

工人

任务广场	任务	发布人	发布时间	下载可执行文件
历史任务	sadasdas	adcd	2022-03-05	<a href="#">点击下载</a>
已接收任务				

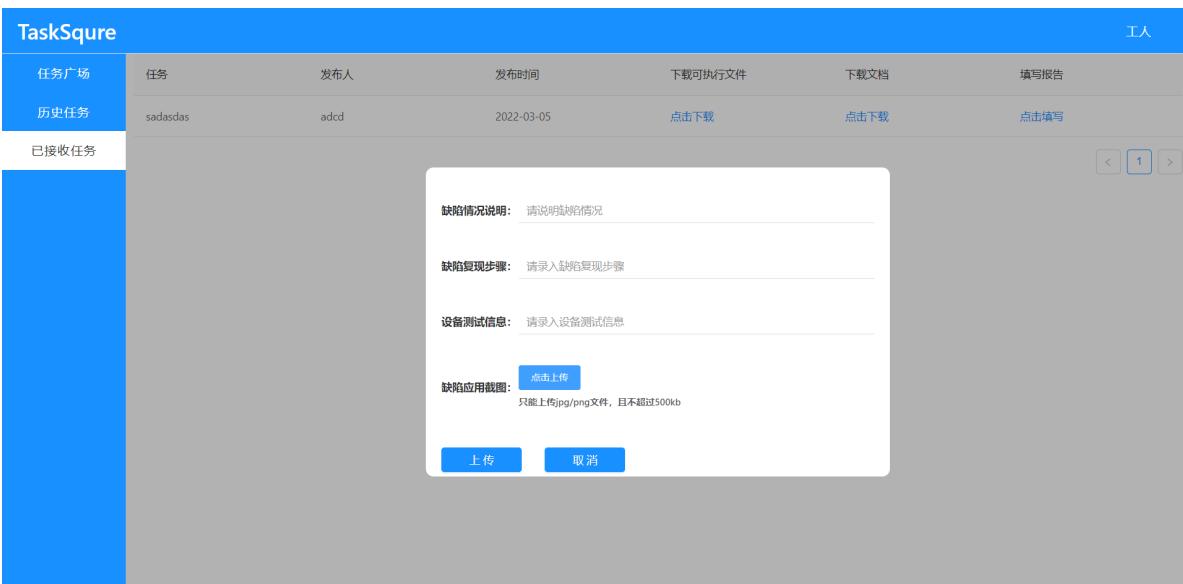
下载

- 5-doc 打开文件
- 5-executable 已删除
- 5-executable 已删除
- 5-doc 已删除
- 5-executable 已删除

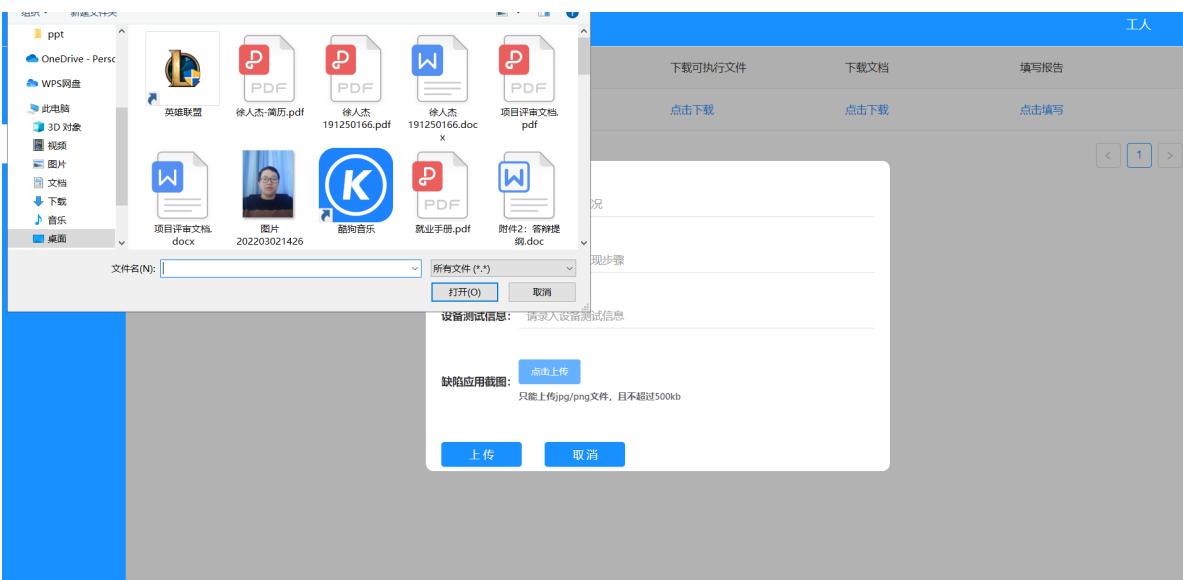
查看更多

## 6.2. 填写报告

可点击“填写报告”进入填写报告界面。

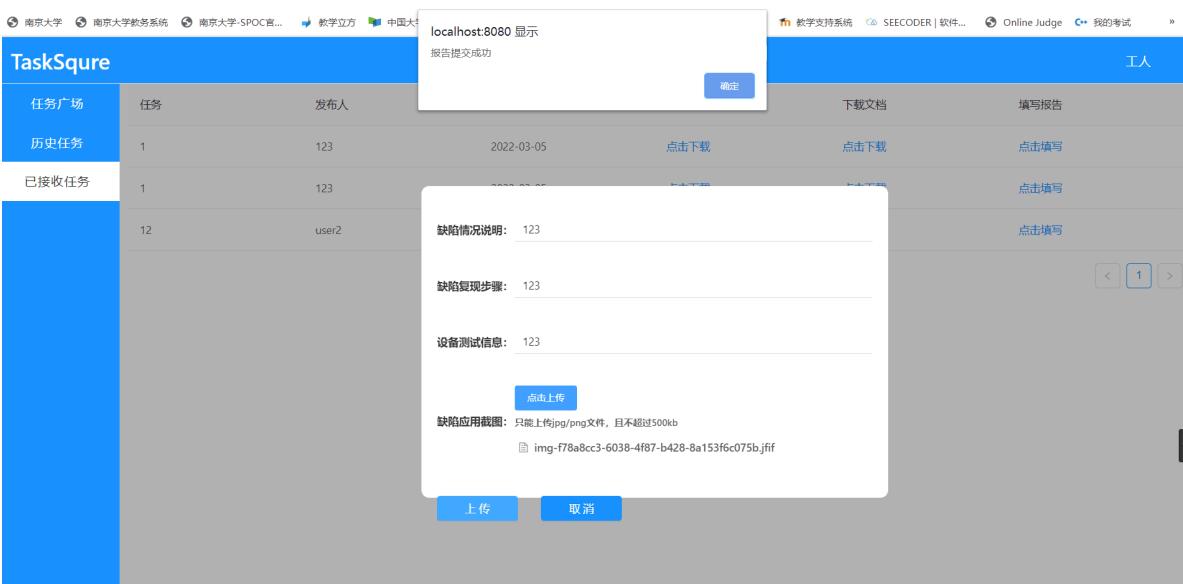


点击上传缺陷应用截图可以上传相关缺陷应用截图。



## 6.3. 报告提交成功

若正确填写报告名称、出现缺陷的应用截图、缺陷情况说明、缺陷复现步骤和测试设备信息，则可点击“确认提交”成功提交报告。



## 6.2. 报告提交失败

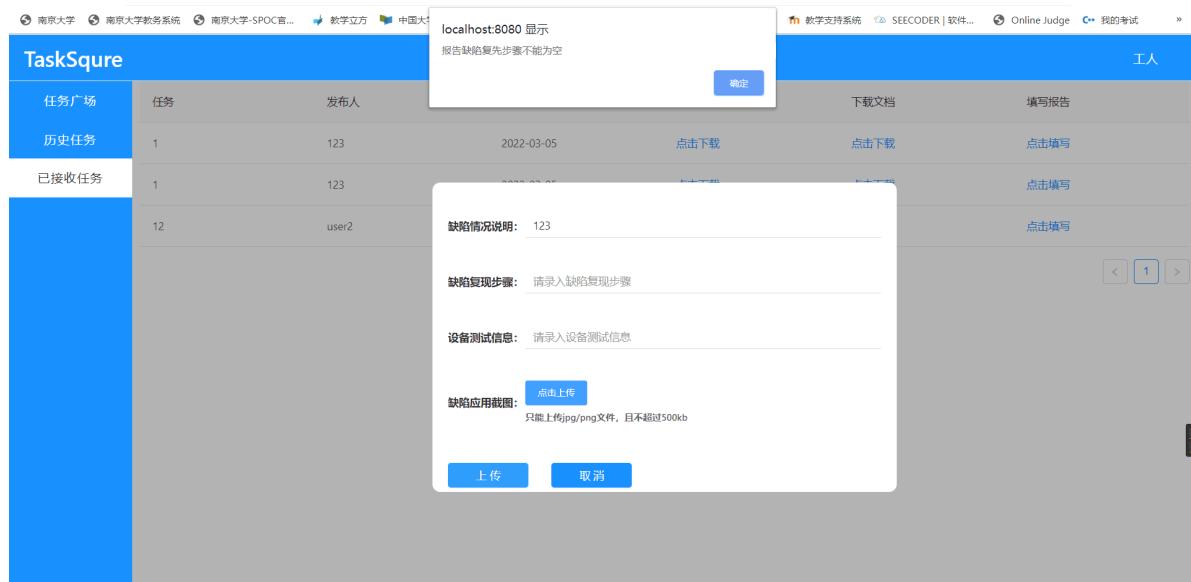
### 6.2.1. 未填写缺陷情况描述

若尚未填写缺陷情况描述，则弹窗提示尚未填写缺陷情况描述。



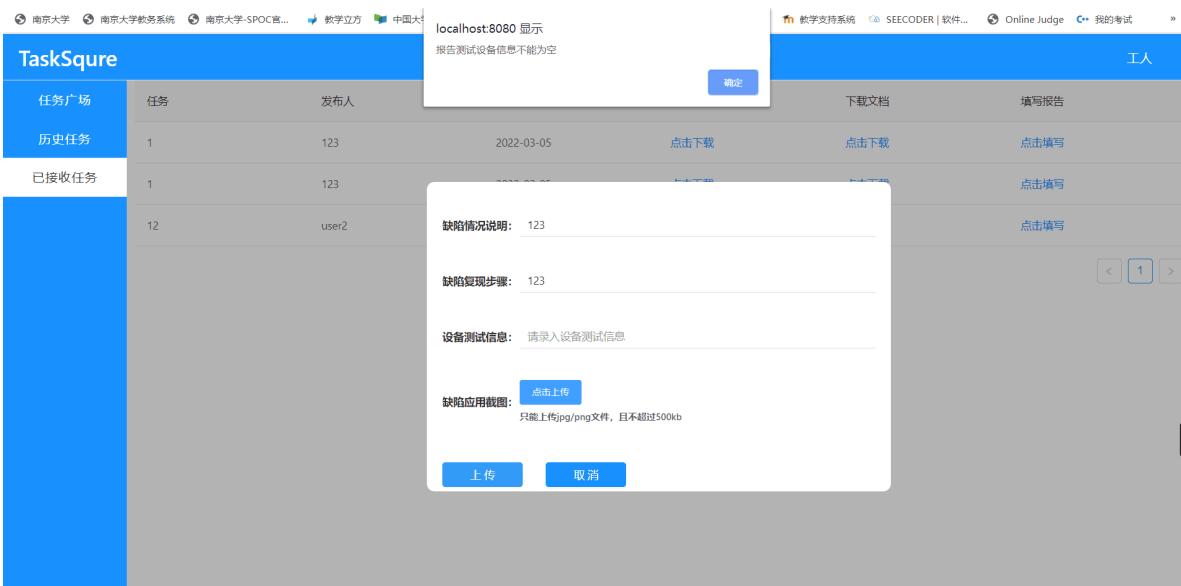
### 6.2.2. 未填写缺陷复现步骤

若未填写缺陷复现步骤，则弹窗提示缺陷复现步骤。



### 6.2.3. 未填写测试设备信息

若未填写测试设备信息，则弹窗提示未填写测试设备信息。



查看历史任务界面与已接受任务界面类似

## 7. 任务广场 (发包方版)

进入发包方的任务广场，可浏览正在招募工人的任务。

ID	截止日期	任务描述	操作
sadasdas	2022-03-18	sss	任务详情 接受任务
12	2022-03-16	12	任务详情 接受任务
1234	2022-03-11	123	任务详情 接受任务
1	2022-03-06	1	任务详情 接受任务
1	2022-03-06	1	任务详情 接受任务

功能与众包工人类似

### 7.1. 任务详情界面

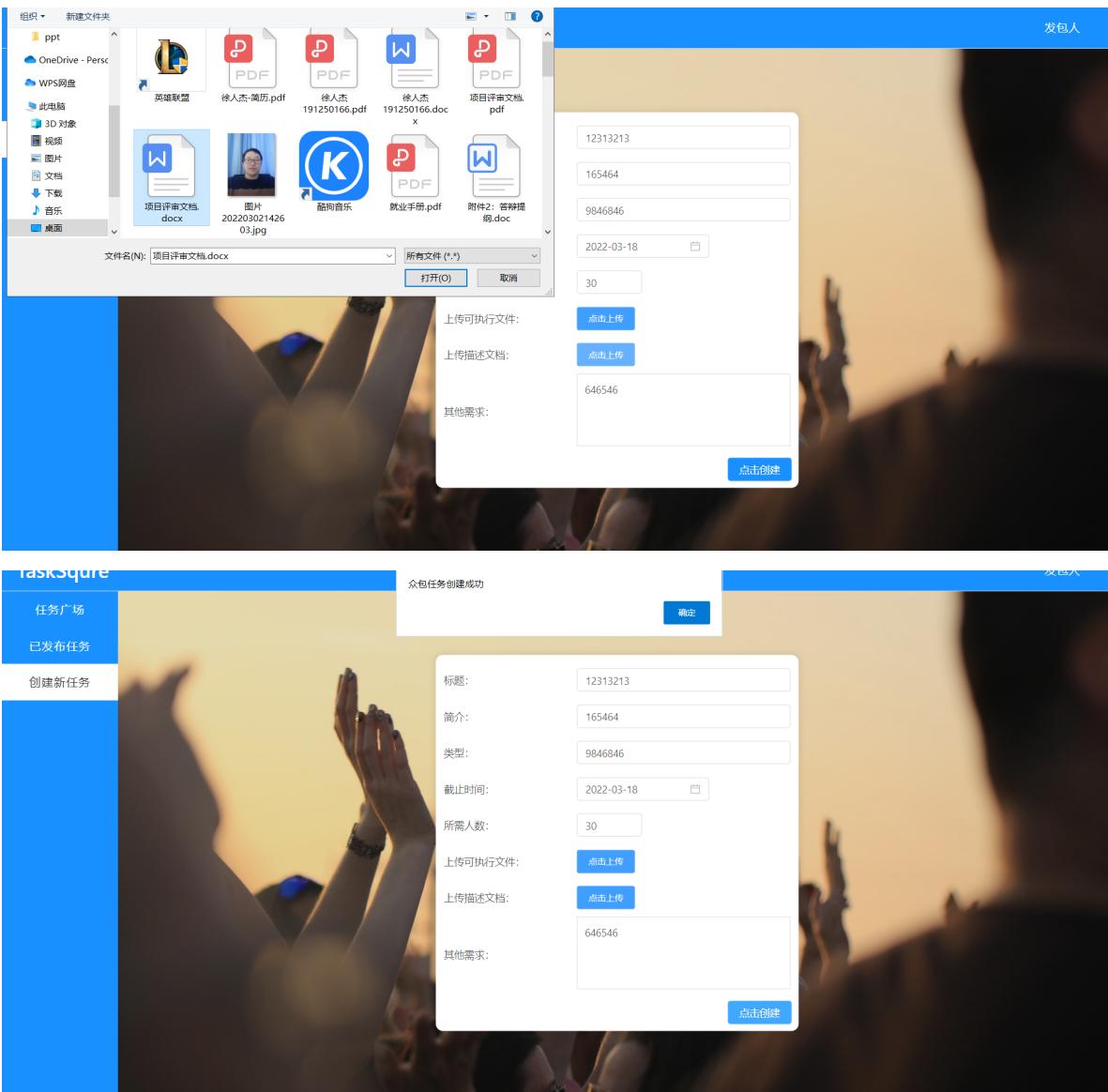
可点击任意任务的“任务详情”按钮，进入任务的详细信息界面。

### 7.2. 退出登录

可点击发包方位置的下拉框中的“退出登录”，退出登录，回到登录界面。

## 8. 创建新任务界面 (发包方版)

进入创建新任务界面，可以填写任务相关的信息，并上传可执行文件和描述文档。



## 9. 已发布任务界面 (发包方版)

进入已发布任务界面，可查看任务名称、截止时间、任务类型以及查看已经填写的报告。

TaskSquire					发包人
任务广场	任务名称	截止时间	任务类型	查看详情	
已发布任务	12313213	2022-03-18T00:00:00.000+00:00	9846846	<a href="#">点击查看</a>	
<a href="#">创建新任务</a>					

