上海交通大学在线教学平台Canvas

测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <01/06/2023> | <1.0> | Web GUI测试报告 | 董云鹏，罗世才，冯逸飞，谷金龙 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 定义、首字母缩写词和缩略语 4

1.3 参考资料：《软件测试》 4

1.4 概述 4

2. 测试过程 4

2.1 测试场景1：登录 4

2.2 测试场景2：下载文件 6

2.3 测试场景3：上传作业 7

3. 测试结论 8

测试报告

# 简介

本测试报告是针对上海交通大学在线教学平台Canvas进行的测试。Canvas作为交大人最常使用的网站，有建立课程班组，上传下载资料，收发作业，传递公告等功能。本次测试为Web GUI测试，分为一共 3个功能性使用场景涵盖了6 种页面元素和至少两级以上页面。本次测试了采用selenium作为测试框架，python作为脚本语言，配合geckodriver驱动，用pytest断言作为测试工具。

## 目的

本测试报告展示了针对上海交通大学在线教学平台Canvas的Web GUI测试过程明细与测试结论，在测试内容上，我们的测试根据功能性应用场景进行划分，分别有登录，下载文件和交作业三个场景，覆盖了按钮、输入框、下拉框、多选框、模态框、文件上传等6种不同元素。在测试方法上，我们结合测试框架selenium中的多种元素定位方法，进行综合测试。

## 定义、首字母缩写词和缩略语

|  |  |
| --- | --- |
| 缩写词 | 含义 |
| Canvas | 上海交通大学在线教学平台 |
| Web GUI测试 | 针对基于浏览器的网页网站测试 |
| Selenium | 本次测试使用python语言的测试框架 |

## 参考资料：《软件测试》

## 概述

本次测试报告包含针对测试用例设计的说明，测试结果，以及对被测试软件改进的建议。

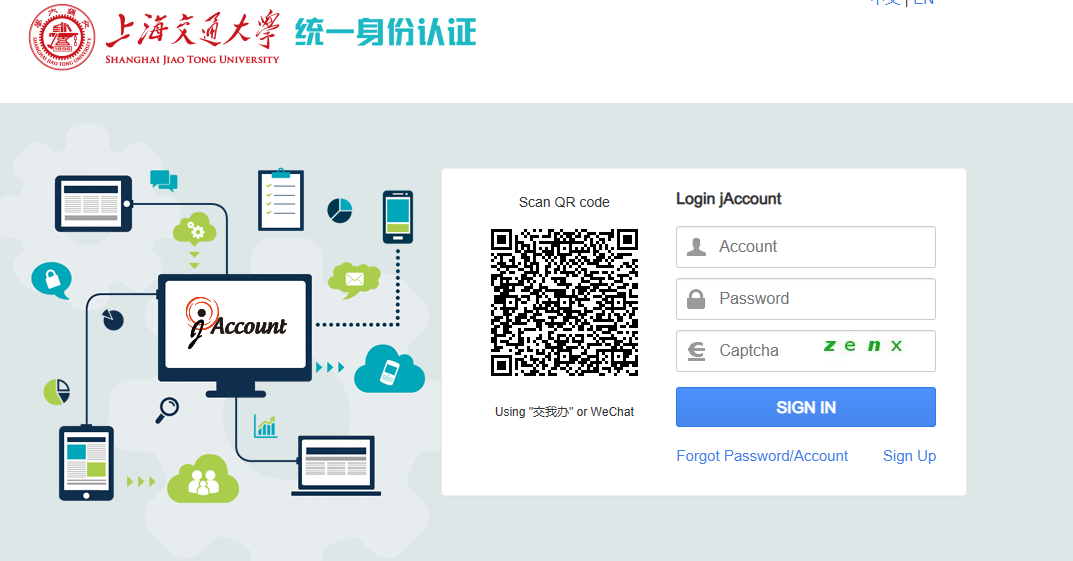
# 测试过程

# 测试场景1：登录

在登录场景中，我们选择Canvas两层登录面板的各个元素进行测试，其中包含了输入框、文本、按钮、链接的控件。同时，测试框架selenium具有多种定位html元素的方法（id, css, xpath等），我们针对不同种类的控件，采用不同的定位方法进行测试。具体分类和测试情况如下表所示。



首先是登录页面，我们选择了driver.find\_element方法的By.ID，By.CSS\_SELECTOR和By.XPATH三种方法进行查找元素，并用assert断言进行测试。

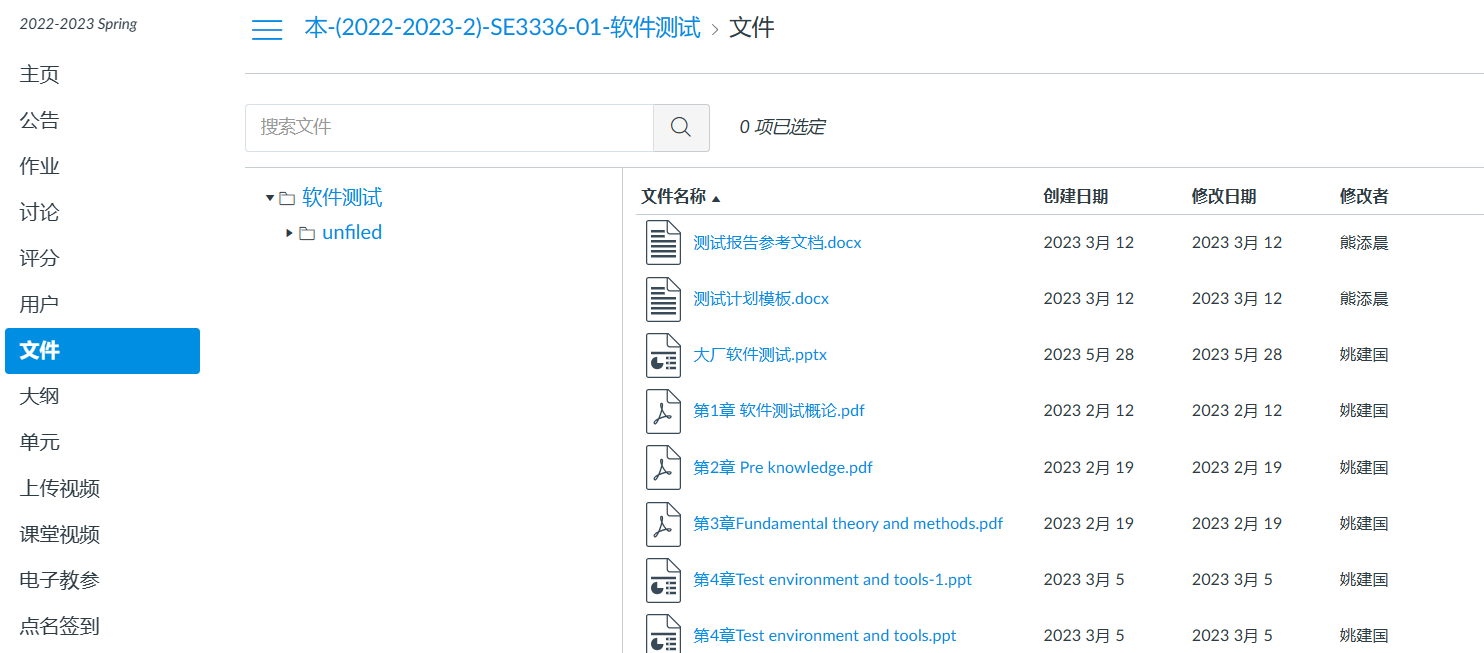


之后跳转到登录二级页面，填充进个人的账号和密码，手动输入验证码，成功登录。



进入到预先设置好的网址，登录成功。

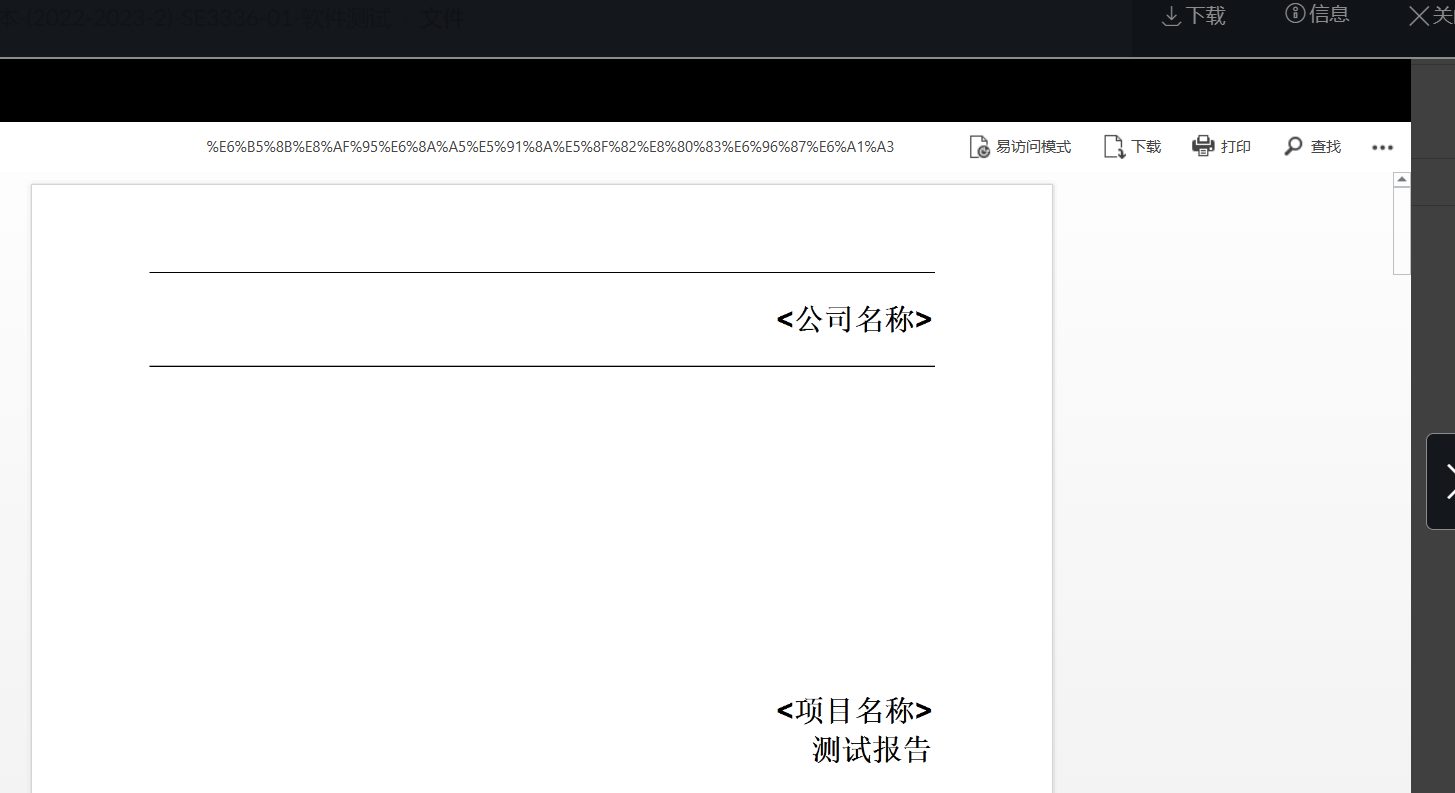
# 测试场景2：下载文件



本课程的ID是52932，我们跳转https://oc.sjtu.edu.cn/courses/52932/files，成功进入本课程的文档界面。

使用driver.find\_element(By.CLASS\_NAME, "ic-Input")定位搜索框，并且搜索我们要下载的文档名字。

这里搜索得到的是一个list，我们首先判定非空，之后打开其中的第一位，这是一个模态框。然后就可以进行下载。



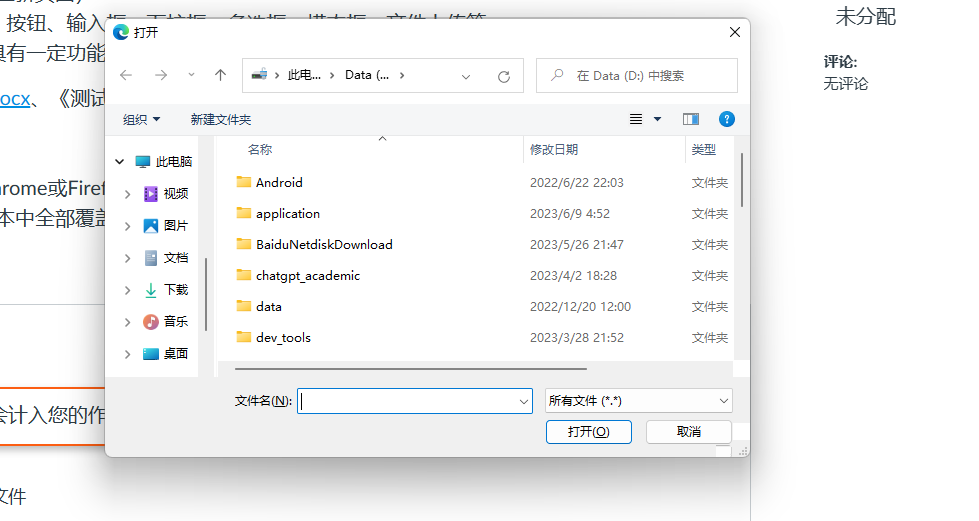
我们在脚本中设置好下载路径，download\_filepath = '/home/os/Downloads/'，点击下载即可完成下载。之后使用断言assert len(download\_files) > 0来判定是否下载成功。

# 测试场景3：上传作业



首先打开提交作业的界面，选择上传文件。

这里使用了文件窗口，我们使用新的库pywinauto来进行自动化处理



首先使用dlg = app["文件上传"]来定位窗口，之后跳转到具体的路径，并选择正确的文件名。

使用判定if not os.path.exists("D:/config"):来判断路径下是否存在这一文件名的文件，若存在则上传，不存在则报错。

# 测试结论

在 1.1 中描述的针对Canvas的4个使用场景和6种页面元素的功能性测试全部通过，因此可以认为Canvas能够符合用户的基本使用需求。

本次测试使用了 python 语言编写脚本，使用 selenium 作为框架，使用火狐浏览器最新版和 对应的geckodriver驱动配合，在关键节点处进行assert断言测试，成功完成了GUI 测试。

在分工上，我们每个人负责一个使用场景和多个页面元素，贡献度分别是：董云鹏25%，冯逸飞25%，罗世才25%，谷金龙25%。