

基于 python+Djano+Easyui 实现的 UI 自动化测试框架 V1.0

by:授客 QQ: 1033553122

博客: <https://www.cnblogs.com/shouke/>

欢迎加入软件测试交流 QQ 群: 7156436

目录

1、 开发环境.....	
2、 框架功能简介	
3、 服务端部署	
4、 大致操作流程	
a) 数据库配置	
b) 日志配置	
c) 新增项目	
d) 系统配置	
e) 新增页面元素	
f) 新增用例	
g) 新增 UI 计划	
h) 关联 UI 测试用例	
i) 新增并执行“运行计划”	
j) 查看测试报告	
5、 页面元素填写规范	
6、 用例设计约定	
7、 调试	
① 项目级别的调试	
② 用例(套件)级别的调试	
8、 测试步骤数据信息同步	
9、 数据备份	

1、 开发环境

服务端:

win7 64\Windows Server 2008 R2 x64

数据库

MariaDB 5.5.45, MySQL Server 5.7

Django-1.11.4.tar.gz

下载地址:

<https://pan.baidu.com/s/1hsc1V5y>

官方下载地址:

<https://www.djangoproject.com/download/>

Easyui 1.5.3

下载地址:

<https://pan.baidu.com/s/1i6E597R>

下载地址:

<http://www.jeasyui.com/download/v153.php>

Jquery-3.2.1.min.js

下载地址:

<https://pan.baidu.com/s/1bqP7jeZ>

下载地址:

<http://www.jq22.com/jquery/jquery-3.2.1.zip>

jquery-easyui-datagrid-dnd

下载地址:

<http://www.jeasyui.net/demo/193.html>

下载地址:

<https://pan.baidu.com/s/1EtiFA1W3-u3T5USCLNE0eQ>

mysqlclient-1.3.6-cp34-none-win_amd64.whl

下载地址:

<https://pan.baidu.com/s/1o9YDMrg>

官方下载地址: <https://pypi.python.org/pypi/mysqlclient/1.3.4>

参考链接:

<https://docs.djangoproject.com/en/1.11/ref/databases/>

httpd-2.4.23-win64.zip

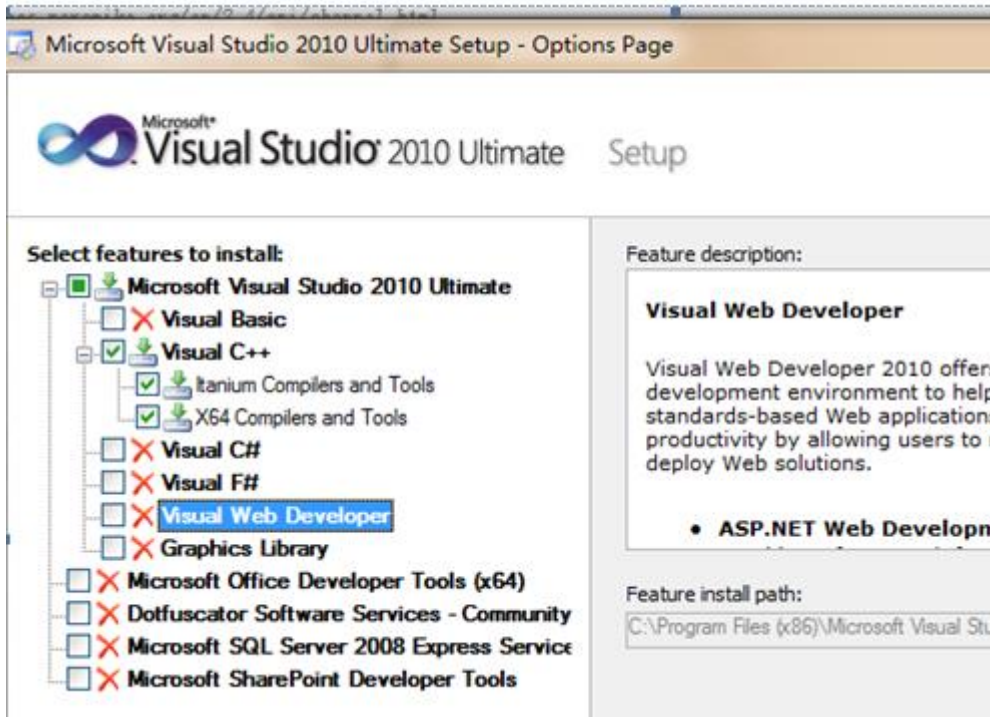
下载地址 1: <https://www.apachelounge.com/download/VC10/>

下载地址 2: <https://pan.baidu.com/s/1hsc1V5y>

Apache24-win64-VC10|mod_wsgi-py34-VC10.so

下载地址: <https://pan.baidu.com/s/1boEF7VD>(链接包含不同版本的集合包)

软件包依赖 VS2010



客户端:

Win7 64 位

Python 3.4.0 Windows x86-64 MSI installer

下载地址 1: <https://www.python.org/ftp/python/3.4.0/python-3.4.0.amd64.msi>

mysql-connector-python-2.1.7-py3.4-windows-x86-64bit.msi

下载地址 1:

官方下载地址: <http://dev.mysql.com/downloads/connector/python/>

Selenium-3.7.0.tar.gz

下载地址 1:

<https://pypi.python.org/pypi/selenium#downloads>

Firefox 57.0 (64 位)

geckodriver-v0.11.1-win64.zip

下载地址 1: <http://pan.baidu.com/s/1jI0SHTG>

官方下载地址: <https://github.com/mozilla/geckodriver/releases>

IE 11

IEDriverServer-win64.zip

下载地址:

[https://pan.baidu.com/s/1kTjG0Z1#list/path=%2F%E5%85%B1%E4%BA%AB%E8%BD%AF%E4%BB%B6%E8%87%AA%E5%8A%A8%E5%8C%96%E6%B5%8B%E8%AF%95-%E5%AE%A2%E6%88%B7%E7%AB%AF&parentPath=%2F%E5%85%B1%E4%BA%AB%E8%BD%AF%E4%BB%B6](https://pan.baidu.com/s/1kTjG0Z1#list/path=%2F%E5%85%B1%E4%BA%AB%E8%BD%AF%E4%BB%B6%2FWeb%E8%87%AA%E5%8A%A8%E5%8C%96%E6%B5%8B%E8%AF%95-%E5%AE%A2%E6%88%B7%E7%AB%AF&parentPath=%2F%E5%85%B1%E4%BA%AB%E8%BD%AF%E4%BB%B6)

Chrome 浏览器

版本 1: 56.0.2924.87

下载地址: <http://pan.baidu.com/s/1kU0reMr>

版本 2: 59.0.3053.3

chromedriver_win32 2.2.5

下载地址 1: <http://chromedriver.storage.googleapis.com/index.html?path=2.25/>

下载地址 2: <http://pan.baidu.com/s/1slGn5hV>

2、 框架功能简介

- 1) 采用 WEB 页面, 可灵活管理测试项目, 测试计划, 测试用例(支持模块层级化组织用例, 支持自由拖拽分组), 运行计划
- 2) 通过项目关联计划, 计划关联用例的设计方式, 支持一次运行单个、多个测试计划
- 3) 支持一键复制单个测试用例, 也支持一键复制用例步骤, 禁用、启用用例步骤等(禁用的步骤将不被运行)
- 4) 纯界面化, 无须写代码(特殊需求除外)
- 5) 支持文件、控制台的日志打印, 可分别控制开关
- 6) 可集成 Jenkins 自动运行脚本

3、 服务端部署

参考文档“基于 python+Django+Easyui 实现的接口自动化测试框架 V1.0.doc”

4、 大致操作流程

客户端

a) 数据库配置

编辑 conf/db.conf 文件

```
[TESTPLATFORM]
```

```
host = 192.168.1.102
```

```
port = 3306
```

```
user = testacc
```

```
passwd = test1234
```

```
db = testplatform
charset = utf8
```

说明: 这里配置的数据库, 即为部署服务端时使用的数据库。

b) 日志配置

编辑 conf/log.conf 文件

```
logconfig.conf x
[LOGGING]
log_file = F:\project\interface_project\logs\log.txt ← 日志文件所在路径
max_bytes_each = 51200 ← 单个日志文件大小
backup_count = 10 ← 同一时刻, 可存留的日志文件数
fmt = |(asctime)s |(filename)s[line: |(lineno)d] |(levelname)s: |(message)s
logger_name = test_logger ← 日志打印器名称
log_level_in_console = 10 ← 日志级别
log_level_in_logfile = 20 ← 日志级别
console_log_on = 1 ← 日志打印开关
logfile_log_on = 1 ← 日志打印开关

[README]
log_level = '日志级别: CRITICAL = 50 ERROR = 40 WARNING = 30 INFO = 20 DEBUG = 10 NOTSET = 0'
log_on = 'console_log_on = 1 开启控制台日志, 0则关闭, logfile_log_on = 1 开启文件日志, 0则关闭'
```

服务端:

c) 新增项目

项目管理-UI 项目配置

d) 系统配置

浏览器配置

操作配置(如果没有初始化配置的话)

断言配置(如果没有初始化配置的话)

函数配置

e) 新增页面元素

页面元素管理-页面元素管理

f) 新增用例

测试用例管理-UI 测试用例管理, 支持嵌套管理

g) 新增 UI 计划

测试计划管理-UI 测试计划

h) 关联 UI 测试用例

可先关联要最先运行的用例，关联后也可做适当的调整。

i) 新增并执行“运行计划”

新增运行计划后，可直接点击对应运行计划的“运行按钮”，

也可以直接客户端传参运行

```
cd /d E:\interface_project_for_dev
python main.py rop 1517487657
```

说明：

- 1、这里的数字 1517487657 即为运行计划编码
- 2、rop -run one project,运行单个运行计划

j) 查看测试报告

所在模块：测试报告-UI 测试报告

5、 页面元素填写规范

元素选择器 1 必填，选择器 2 选填，元素选择器格式：

adapter = selector

形如：xpath=/html/body/header/div/div[3]/a[1]

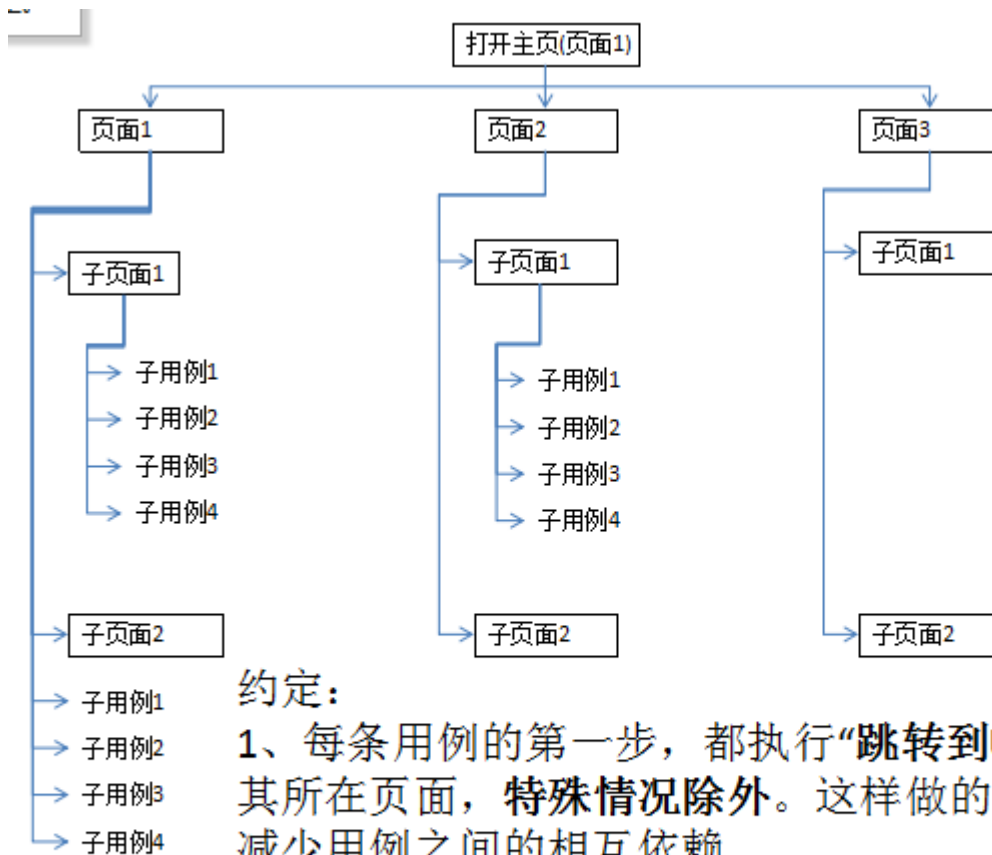
adapter 支持：

id	元素 ID
xpath	元素 xpath
name	元素名称
link_text	元素链接文本内容
partial_link_text	元素链接文本内容的部分文本
css	元素 css
tag_name	元素标签名
class_name	元素 class 名称

样例：

登录页面 ×			
新增 修改 删除			
	<input type="checkbox"/>	元素ID 元素名称	选择器1 选择器2
1	<input type="checkbox"/>	33 跳转等录页按钮	xpath=/html/body/header/div/div[3]/a[1]
2	<input type="checkbox"/>	32 登录按钮	xpath=//*[@id="app"]/div/div/div[1]/div/button
3	<input type="checkbox"/>	31 验证码输入框	xpath=//*[@id="app"]/div/div/div[1]/form/div[3]/div/div[1]/input
4	<input type="checkbox"/>	2 密码输入框	xpath=//*[@id="app"]/div/div/div[1]/form/div[2]/div/div[1]/input
5	<input type="checkbox"/>	1 账号输入框	xpath=//*[@id="userName"] xpath=//*[@id="app"]/div/div/div[1]/form/div[1]/div/div[1]/input

6、 用例设计约定



7、调试

① 项目级别的调试

```
python main.py rop 1517487657 debug
```

说明:

- 1、这里 1517487657 为运行计划编码 2
- 2、开启调试模式，不会往数据库记录相关数据

② 用例(套件)级别的调试

```
python main.py debug 项目 ID, 形如: python main.py debug 1
```

程序运行后，按要求输入会要求测试用例、用例套件的 ID，

如果输入正确，会要求继续输入要运行的浏览器，输入对应浏览器对应的数字，回车即可开始运行

```
main
"D:\Program Files\python332\python.exe" E:/mygit/AutoTestPlatform/UIAutotest/main.py debug 2
当前控制台生效的日志级别为: 30
2018-02-09 14:47:56,447 main.py[line: 26] INFO: 执行当前脚本的Python版本: 3.3.2
2018-02-09 14:47:56,447 main.py[line: 28] INFO: 当前运行模式为: debug
2018-02-09 14:47:56,448 main.py[line: 31] INFO: 当前运行计划编号为: 2
2018-02-09 14:47:56,448 main.py[line: 75] INFO: 调试模式
请输入用例ID、套件ID: 5
请选择要运行的浏览器(输入数字,回车): 1、谷歌 2、IE 3、火狐: 2
2018-02-09 14:48:06,189 debug.py[line: 23] INFO: 正在查询项目[ID: 2]相关信息
2018-02-09 14:48:06,191 mydb.py[line: 96] INFO: query: SELECT home_page, environment, valid_flag FROM `website_ui_pro
2018-02-09 14:48:06,191 mydb.py[line: 99] INFO: 执行的查询语句为: SELECT home_page, environment, valid_flag FROM `web
2018-02-09 14:48:06,199 debug.py[line: 29] INFO: 正在查询与项目关联的数据库信息
2018-02-09 14:48:06,199 mydb.py[line: 124] INFO: query: SELECT db_type, db_alias, db_name, db_host, db_port, db_user,
```

UI测试用例管理 x

用例管理

下拉选择项目

数据灯塔-官网

ID1-数据灯塔官网

ID12-nodename

nodename

ID11-nodename

ID9-nodename

ID6-nodename

ID8-nodename

ID7-nodename

ID5-登录页面

ID2-正常登录

ID10-nodename

用例详情

ID2-正常登录 x

新增

修改

删除

	<input type="checkbox"/>	步骤ID	步序	对象类型	所在页面	操作对象
1	<input type="checkbox"/>	10	8	页面元素	数据灯塔-官网->登录页面	登录按钮
2	<input type="checkbox"/>	1	7	页面元素	数据灯塔-官网->登录页面	验证码输入框
3	<input type="checkbox"/>	9	6	页面元素	数据灯塔-官网->登录页面	验证码输入框
4	<input type="checkbox"/>	5	5	页面元素	数据灯塔-官网->登录页面	密码输入框
5	<input type="checkbox"/>	8	4	页面元素	数据灯塔-官网->登录页面	密码输入框
6	<input type="checkbox"/>	6	3	页面元素	数据灯塔-官网->登录页面	账号输入框
7	<input type="checkbox"/>	7	2	页面元素	数据灯塔-官网->登录页面	账号输入框
8	<input type="checkbox"/>	11	1	页面元素	数据灯塔-官网->登录页面	跳转等录页按钮

注意:

- 1、用例套件的调试,不支持目录层级的嵌套,如果套件下包含非用例目录,会被自动跳过。
- 2、套件下用例的运行顺序: 从下往上,顺序执行。
- 3、调试之前,需要开启并配置好项目相关信息

8、测试步骤数据信息同步

```
cd /d E:\mygit\AutoTestPlatform\DataSyncAssistant
python data_sync_assisant.py
```

说明: 程序主要用于同步测试步骤表,操作的用例、页面元素所在页面信息,操作用例名称,可以写成批处理,定时同步

9、数据备份

冷备份:

```
net stop mysql2
xcopy /E /Y D:\MySQL57\data\testplatform\*
```



```
D:\MySQL57\backup\%date:~0,4%%date:~5,2%%date:~8,2%\
net start mysql2
```

命令说明:

```
net stop mysql2
```

停止 mysql 服务

```
xcopy /E /Y D:\MySQL57\data\testplatform\*
D:\MySQL57\backup\%date:~0,4%%date:~5,2%%date:~8,2%\
```

`xcopy /E /Y` 源文件路径 目标目录路径

`D:\MySQL57\data\testplatform*` 表示以 `testplatform` 命名数据库的所有数据文件

`D:\MySQL57\backup\%date:~0,4%%date:~5,2%%date:~8,2\` 表示要复制到的目标目录, `%date:~0,4%%date:~5,2%%date:~8,2%` 表示当前日期, 格式形如 `20180315`

`/E` 表示按目录结构循环复制, 包括空目录

`/Y` 表示如果文件存在, 则直接静默覆盖, 不需要询问

```
net start mysql2
```

重启 mysql 服务