**TRƯỜNG ĐẠI HỌC SƯ PHẠM TPHCM**

**Khoa Công Nghệ Thông Tin**



|  |
| --- |
|  |
| **BÁO CÁO**  **MÔN HỌC : KIỂM THỬ PHẦN MỀM CƠ BẢN**  **ĐỀ TÀI : TÌM HIỂU CÔNG CỤ KIỂM THỬ PHẦN MỀM KOBITON** |



Giáo viên hướng dẫn: Trần Sơn Hải

Nhóm 12:

Trịnh Kim Chi – 44.01.104.064

Hồ Thị Kim Hà – 44.01.104.077

Mục Lục

[Phần 1: Giới thiệu về Kobiton 4](#_Toc55658083)

[1. Kobiton là gì? 4](#_Toc55658084)

[2. Các đặc điểm của Kobiton 4](#_Toc55658085)

[3. Các lợi ích của Kobiton 5](#_Toc55658086)

[4. Thành tích của Kobiton 5](#_Toc55658087)

[Phần 2: Tính năng của Kobiton 6](#_Toc55658088)

[1. Giao diện của phần mềm 6](#_Toc55658089)

[Để vào Kobiton chúng ta cần tạo tài khoản ở trang này: https://portal.kobiton.com/register 6](#_Toc55658090)

[Tạo test tự động đầu tiên 8](#_Toc55658091)

[Toolbar 20](#_Toc55658092)

[Icon 20](#_Toc55658093)

[Description 20](#_Toc55658094)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2014%3A58%3A59.png?version=1&modificationDate=1498723139000&api=v2 20](#_Toc55658095)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A0%3A23.png?version=1&modificationDate=1498723224000&api=v2 20](#_Toc55658096)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A0%3A55.png?version=1&modificationDate=1498723255000&api=v2 20](#_Toc55658097)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A1%3A27.png?version=1&modificationDate=1498723287000&api=v2 20](#_Toc55658098)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A2%3A0.png?version=1&modificationDate=1498723320000&api=v2 20](#_Toc55658099)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A2%3A58.png?version=1&modificationDate=1498723379000&api=v2 20](#_Toc55658100)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A3%3A21.png?version=1&modificationDate=1498723401000&api=v2 20](#_Toc55658101)

[http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A3%3A58.png?version=1&modificationDate=1498723438000&api=v2 21](#_Toc55658102)

[Tests Explorer View 21](#_Toc55658103)

[Keywords Browser View 22](#_Toc55658104)

[Editors 23](#_Toc55658105)

[**Test Case Editor** 23](#_Toc55658106)

[Manual tab 23](#_Toc55658107)

[Script tab 24](#_Toc55658108)

[Variables tab 25](#_Toc55658109)

[Integration tab 25](#_Toc55658110)

[Properties tab 25](#_Toc55658111)

[**Test Object Editor** 26](#_Toc55658112)

[**Web Service Editor** 27](#_Toc55658113)

[*RESTful Request Object Editor* 27](#_Toc55658114)

[*SOAP Request Object Editor* 27](#_Toc55658115)

[**Test Suite Editor** 28](#_Toc55658116)

[*Main tab* 28](#_Toc55658117)

[*Integration tab* 28](#_Toc55658118)

[**Test Suite Collection Editor** 29](#_Toc55658119)

[**Data File Editor** 29](#_Toc55658120)

[**Checkpoint Editor** 31](#_Toc55658121)

[**Keyword Editor** 33](#_Toc55658122)

[Global Variables View 33](#_Toc55658123)

[Job Progress View 34](#_Toc55658124)

[Problems View 34](#_Toc55658125)

[Console View 34](#_Toc55658126)

[Log Viewer View 35](#_Toc55658127)

[Search View 35](#_Toc55658128)

[Report View 36](#_Toc55658129)

[Test Suite Collection Report view 36](#_Toc55658130)

# Phần 1: Giới thiệu về Kobiton

# Kobiton là gì?

* **Kobiton** là một phần mềm được sáng lập và phát triển bởi một đội ngũ kĩ sư chuyên nghiệp ở Việt Nam. Đây là một ứng dụng điện toán đám mây cung cấp giải pháp cho các doanh nghiệp quản lý, truy cập và kiểm thử ứng dụng trên thiết bị thực. Từ test bằng tay cho đến test tự động Kobiton dễ dàng sử dụng để thực hiện test trên Mobile, Web.
* Kobiton cho phép người dùng kiểm soát toàn bộ các thiết bị di động trong quá trình kiểm tra thủ công với thời gian thực tế, định hướng và mô phỏng GPS. Ngoài ra, còn điều khiển loa và camera và quản lý các kết nối thiết bị.
* Kobion được ra mắt vào năm 2016, đến năm 2018 Kobiton nhận được $3 tỷ đầu tư của công ty Kinetic Ventures. Với toàn bộ 8 tỷ đô la Mỹ từ 2 vòng tài trợ, Kobiton đang tập trung phát triển, hỗ trợ khách hàng nhằm nâng tầm sản phẩm công nghệ Việt Nam trên thị trường ứng dụng Mỹ.
* Người dùng có thể mua số phút thử nghiệm trả trước mà không bao giờ hết hạn chỉ với 10 đô la. Kobiton là nền tảng thiết bị di động sáng tạo nhất trên thị trường.
* Kobiton hiện vẫn đang bổ sung các tính năng mới, như là “Record-Playback” cho phép tạo trước các đoạn scripts thử nghiệm bằng cách tương tác với ứng dụng và sử dụng trí tuệ nhân tạo (AI) để phát lại đoạn scripts.

# Các đặc điểm của Kobiton

* Là một công cụ test vì vậy Kobiton có 5 đặc trưng sau đây:
* **Check real equipment**: Kiểm tra thủ công và tự động trên thiết bị thực. Truy cập vào các thiết bị vật lý trên đám mây với hỗ trợ đầy đủ cho thao tác, cử chỉ, điều hướng,… Kiểm tra tình trạng thiết bị của bạn ngay lập tức và xem phần bị ảnh hưởng
* **Natural gestures:** Thực hiện các các cử chỉ để kiểm tra thiết bị như: chạm, vuốt và cuộn, thao tác đơn giản và tự nhiên. Hiệu suất và khả năng phản hồi tốt cho phép bạn kiểm tra như thể bạn đang có thiết bị trong tay.
* **Root cause analysis:** Trong quá trình manual hoặc automation test, Kobiton đều lưu lại nhật kí chi tiết bao gồm video, screenshots, performed gestures và chỉ số hệ thống như bộ nhớ và hiệu suất pin. Dễ dàng phát lại các đoạn scripts trong quá trình thực hiện nhằm phân tích các nguyên nhân thực sự để giải quyết vấn đề.
* **Manual and automatic inspection**: Thực thi song song các lệnh test scripts. Tích hợp nhanh chóng vào quy trình CI/CD và cho ra kết quả gần như ngay lập tức**.**
* **Debug ADB:** Truy cập các thiết bị đám mây trực tiếp từ máy trạm dành cho nhà phát triển và truy cập root vào thiết bị trực tiếp từ bên trong IDE. ADB remote debugging giúp kiểm soát và vận chuyển ứng dụng nhanh hơn.
* Ngoài ra Kobiton còn có một số đặc điểm sau:
* **Agile Test Enabler:** Kiểm tra tự động hóa thiết bị với các khuôn khổ Appium và Selenium mới nhất. Hỗ trợ thực thi tự động hóa với các công cụ kiểm tra di động.
* **Optimized Efficiency:** Tải lên và truy cập ứng dụng mà bạn cần kiểm tra dễ dàng. Thời gian khởi động nhanh hơn và cho phép trải nghiệm chỉ trong vài phút.

# Các lợi ích của Kobiton

* Có lịch sử test và nhật kí test giúp dễ quản lí hơn
* Quản lý các thiết bị nội bộ
* Hỗ trợ Appium 1.6.4
* Trải nghiệm người dùng đơn giản
* 100% thiết bị thực
* Giá cả phải chăng
* Ngăn chặn: Catch cross-platform trong code và test trên thiết bị ngay trong IDE. Nâng cao chất lượng code và tránh các lỗi “có thế mắc phải”.

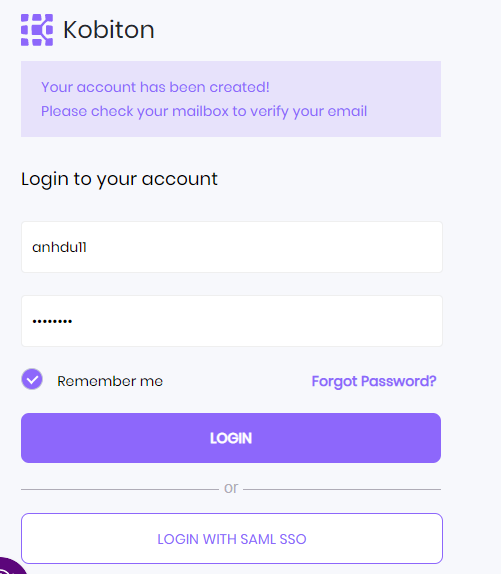
# Thành tích của Kobiton

Một ứng dụng phần mềm do đội ngũ kỹ sư Việt Nam phát triển đã huy động thành công vòng gọi vốn A (Series A) với 5,2 triệu đô la Mỹ từ BIP Capital.

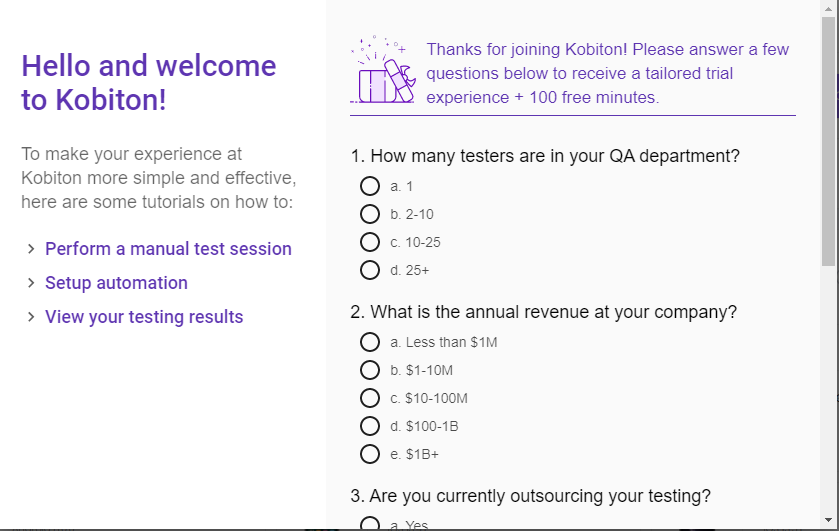
# Phần 2: Tính năng của Kobiton

# Giao diện của phần mềm

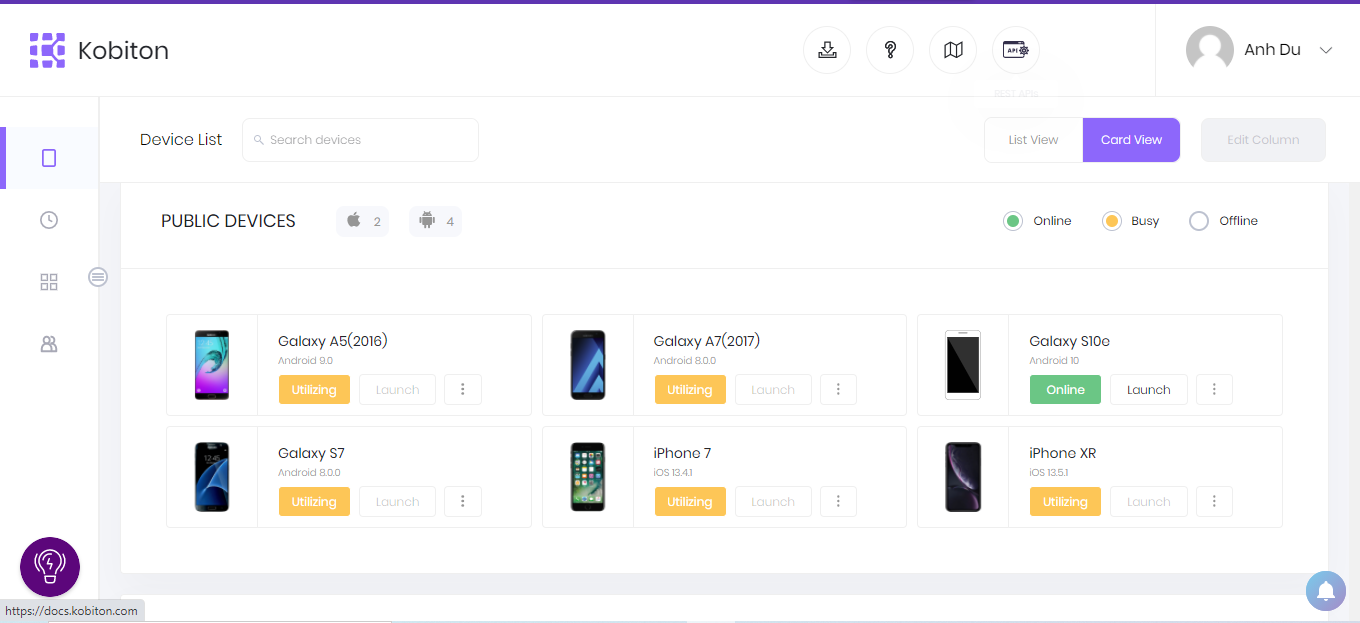
# Để vào Kobiton chúng ta cần tạo tài khoản ở trang này: https://portal.kobiton.com/register

Sau khi tạo account chúng ta login bằng account vừa tạo.

Sau khi đăng nhập, hệ thống sẽ hiển thị bảng câu hỏi trắc nghiệm nhằm khảo sát mục đích của người sử dụng, có thể bỏ qua bước này.



Giao diện màn hình chính:

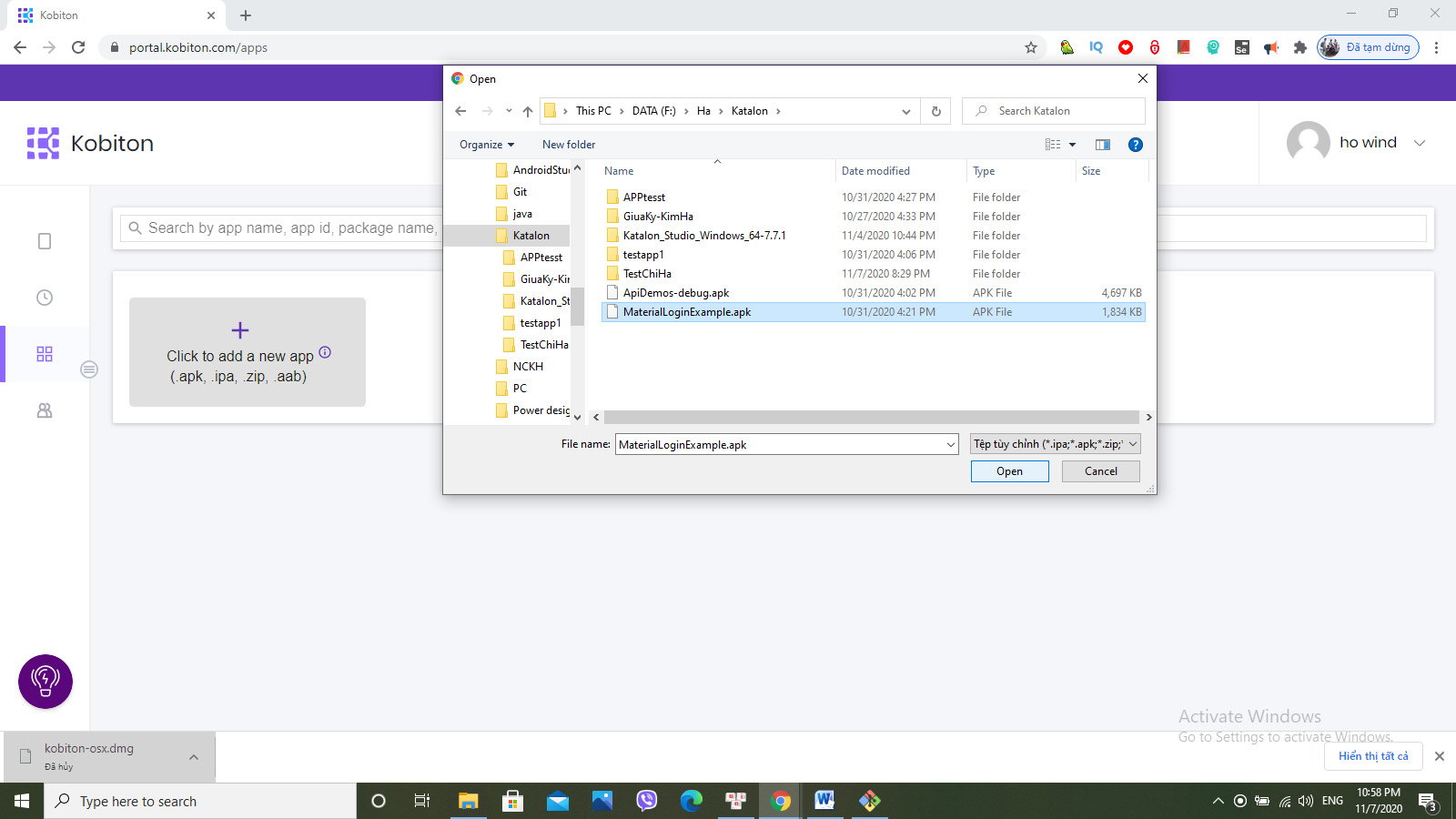


1. Manual Testing

Kobiton cho phép bạn truy cập từ xa đến các thiết bị di động thực. Kobiton sẽ tự động ghi lại những thao tác kiểm tra thủ công và lưu trữ chi tiết.

Bước 1: Click vào tab Devices, các thiết bị di động sẽ hiển thị ra. Di chuyển chuột tới thiết bị cần test, click vào Launch.

Bước 2: Click tab Apps sau đó click vào add a new app để thêm app cần test vào.



Ta có thể đặt tên, thêm mô tả,… cho phiên test: click vào Session Settings. Thay đổi tên, mô tả để dễ tìm kiếm.

-Các thao tác trên thiết bị:

-Chụp màn hình thiết bị và xem các ảnh chụp:

1.Chụp màn hình thiết bị  
2.Thao tác chạm trên thiết bị  
3.Thao tác zoom trên thiết bị  
4.Thiết lập vị trí của thiết bị  
5.Thiết lập múi giờ  
6. Xoay hướng thiết bị ngang hoặc dọc  
7.Nút nguồn của thiết bị   
8.Nút home của thiết bị   
9.Nút xem ứng dụng gần đây  
10.Nút quay trở lại.

Bước 3: Click vào tab Sesions để xem các các ghi chép của các phiên test.

1. Automation Testing

Bạn có thể sử dụng kobiton để tạo automation test trên thiết bị thực. Kobiton sẽ tạo ra một test report bao gồm detailed logs, HTTP commands, and metadata.

Kobiton hỗ trợ appium test automation.

Bước 1: Đăng nhập kobiton

Bước 2: Click tab Devices, sau đó di chuyển chuột đến thiết bị cần test click vào Show automation settings.

Bước 3: Thiết lập cấu hình trong kịch bản kiểm thử tự động Appium của bạn

Bước 4: Chạy test của bạn

Bước 5: Xem lại ghi chép quá trình kiểm thử.

1. Device management

Để test trên thiết bị thực trước tiên cần cài đặt Kobiton Desktop App:

Hướng dẫn cài đặt

Sau khi cài đặt:

Bước 1: Đăng nhập bằng email và mật khẩu đăng kí kobiton

Bước 2: Sau khi hệ thống các thực thành công, cắm thiết bị di động của bạn vào.

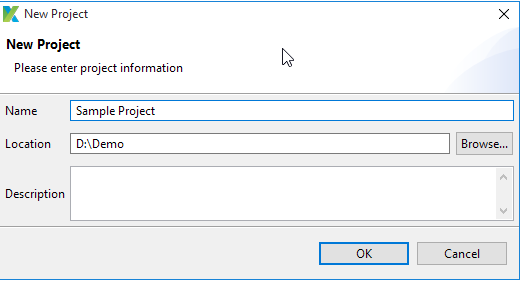
Bước 3: Kết nối điện thoại vào danh sách thiết bị

Bước 4: Click Activity và bắt đầu kiểm thử.

1. Organization Management

Tạo một test project

1. Chọn **File**> **New**> **Project**, cửa sổ New Project sẽ xuất hiện.



2. Điền trường **Name** và chọn đường dẫn để lưu project, sau đó click **OK**.

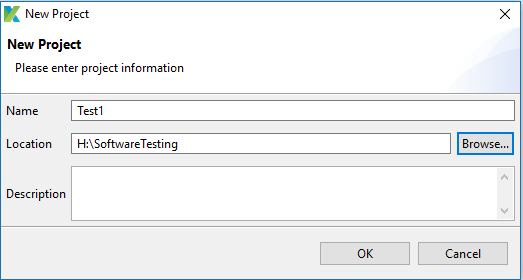
# Tạo test tự động đầu tiên

*1.Record- Playback*

Record-Playback là một ứng dụng cơ bản giúp bạn tạo và chạy các test scripts một cách nhanh chóng. Nó sẽ giúp chúng ta hiểu code và thực thi các test case, Ứng dụng đặc biệt của chức năng này là nó sẽ tự động tương thích với chức năng debug của các browser sau đây như IE,FireFox,Chrome

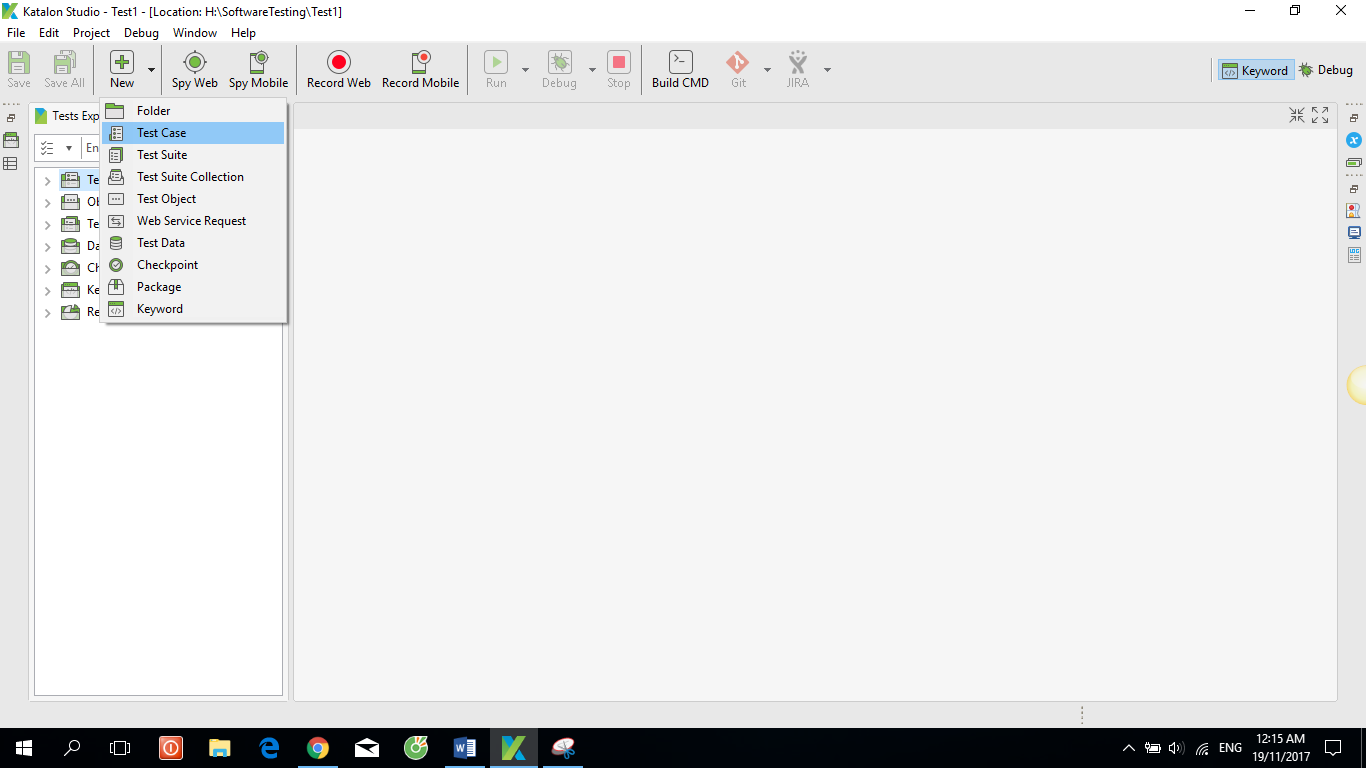
Chạy test đầu tiên:

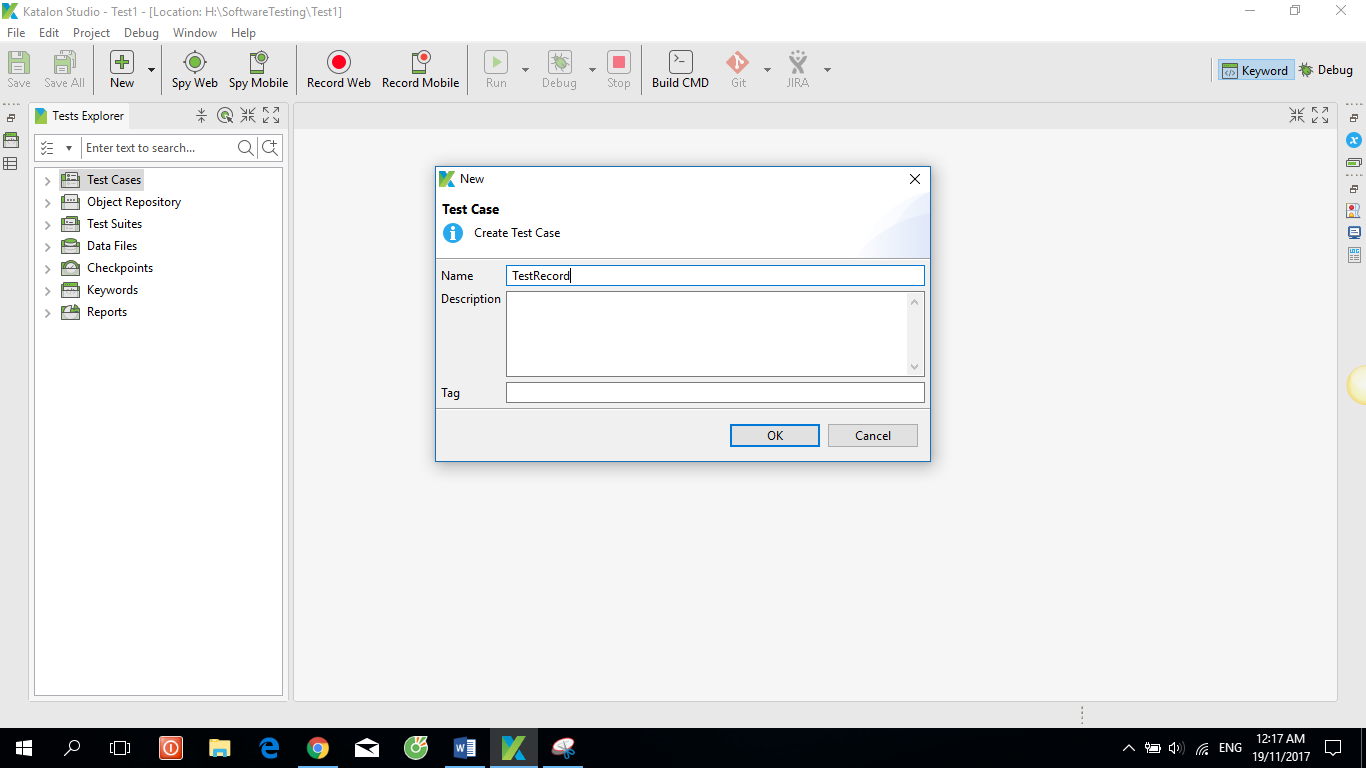
Record log in: <https://www.katalon.com/>

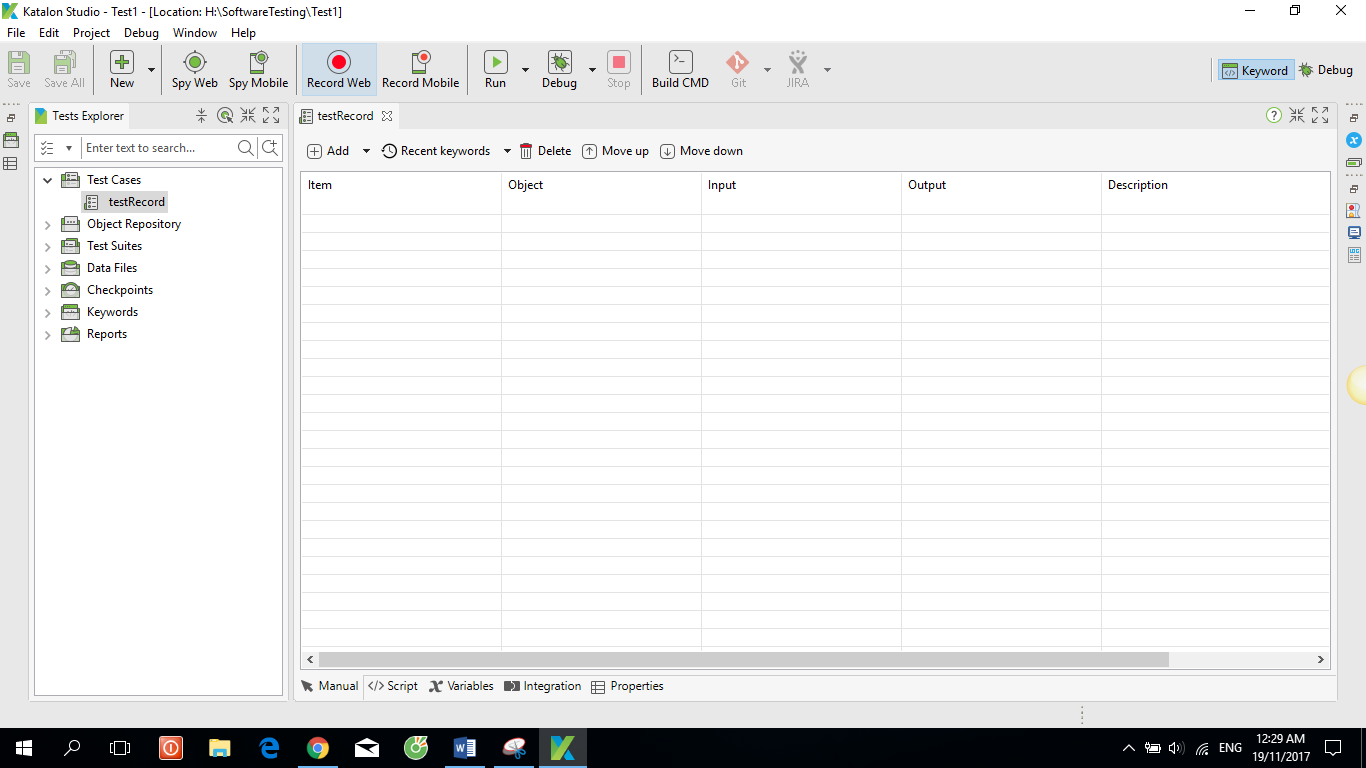
*1. Tạo project mới* 

*2. Tạo test mới*

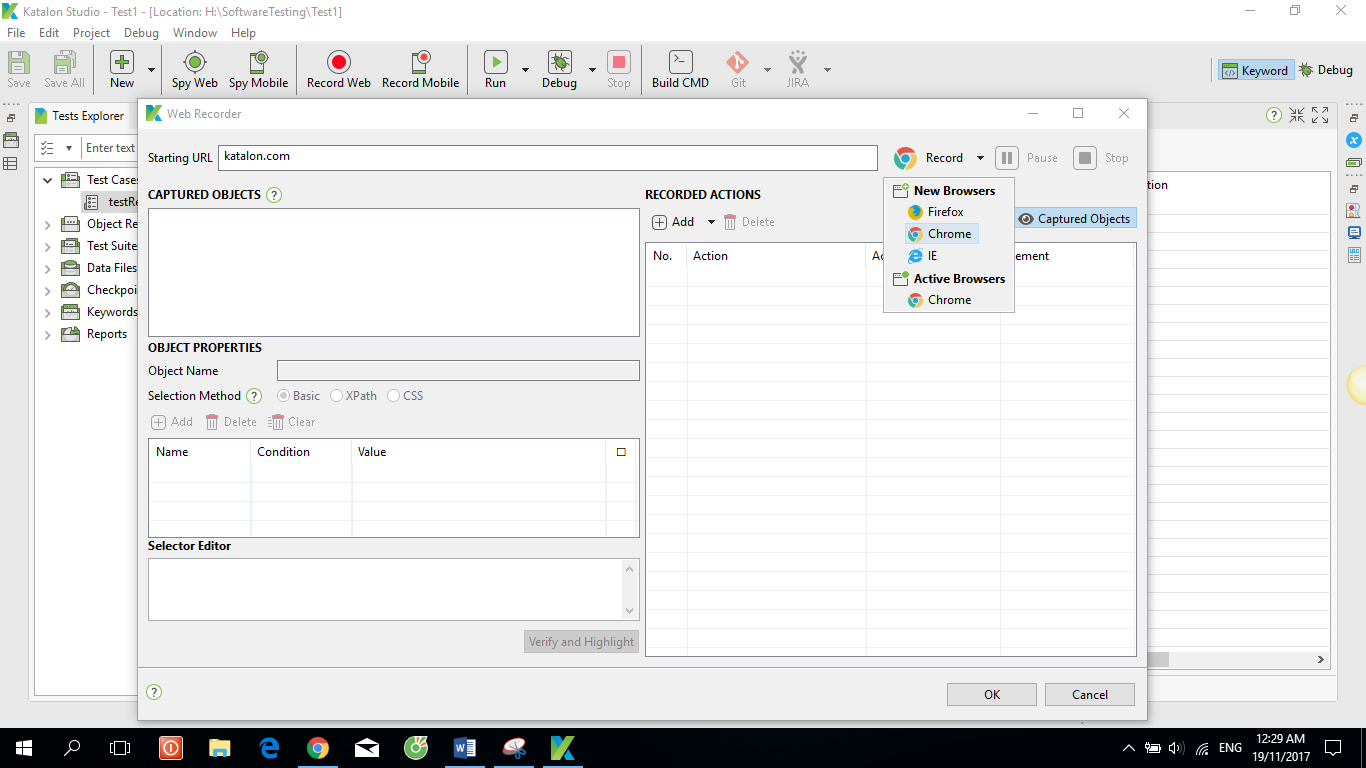
Chọn **New** > **Test case**

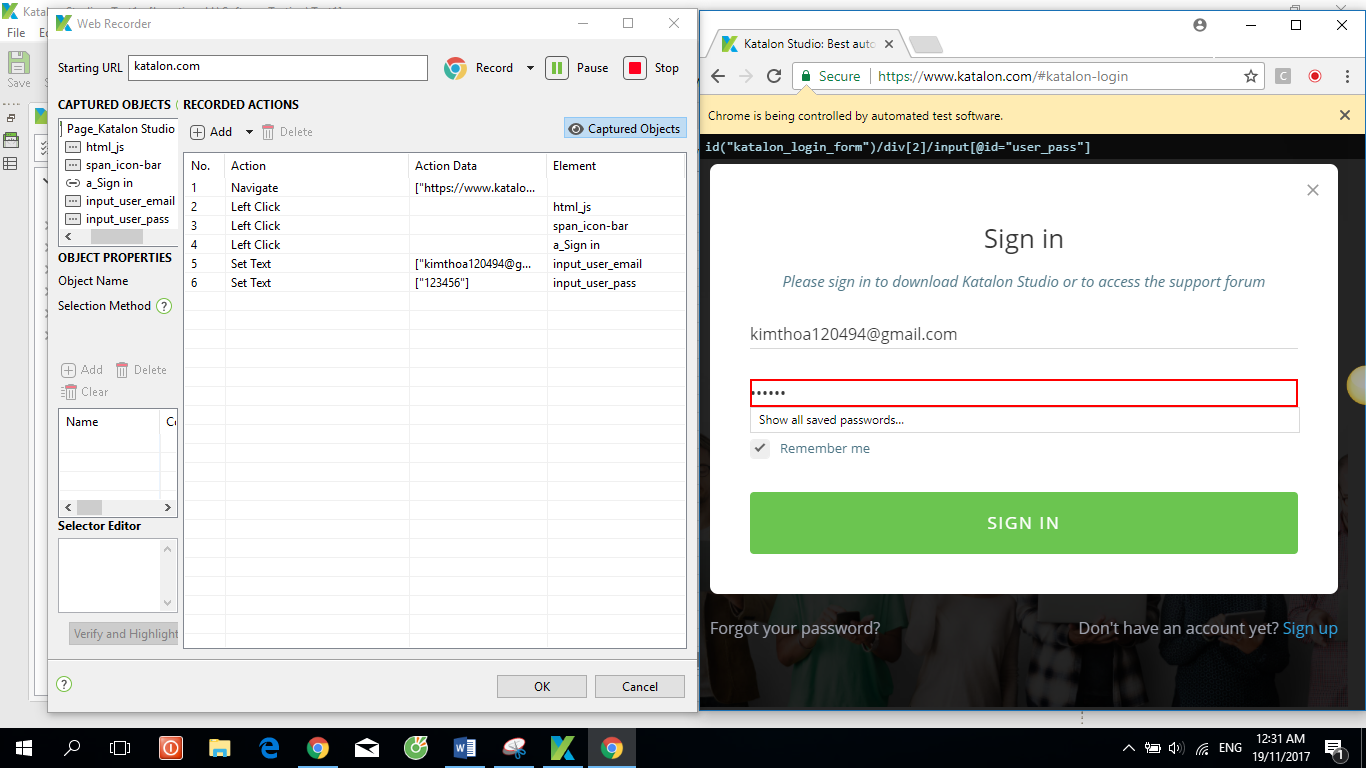




*3*. *Chọn Record Web ở main tool bar*. 

*4. Nhập url và chọn browser tương thích. Tiếp tục bằng việc tương tác với browser (test chức năng login).*

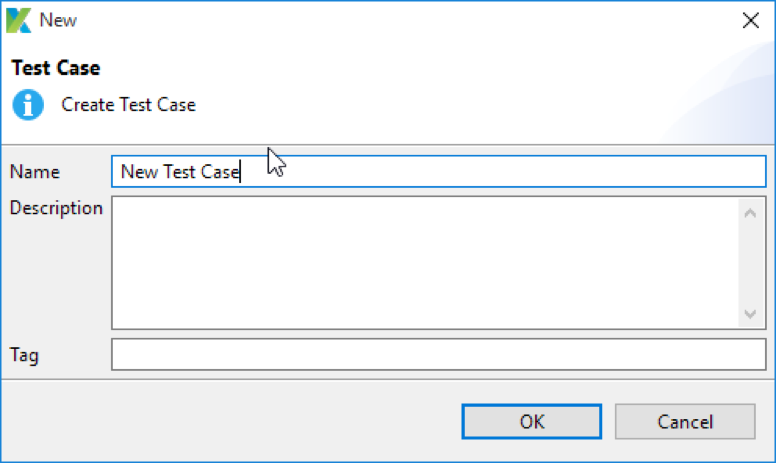


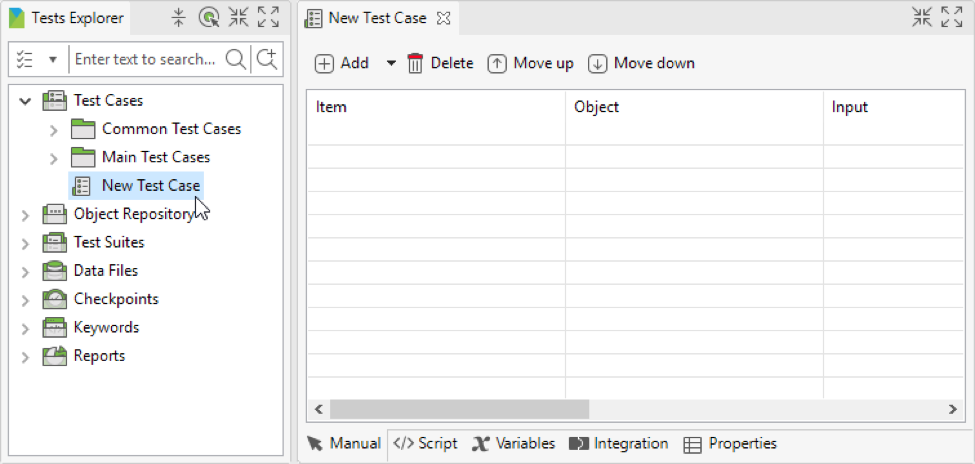
***Kết quả:*** null 

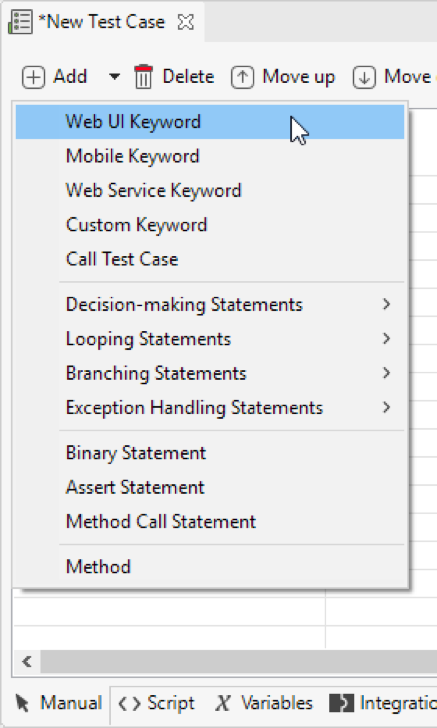
*2.Chế độ Manual*

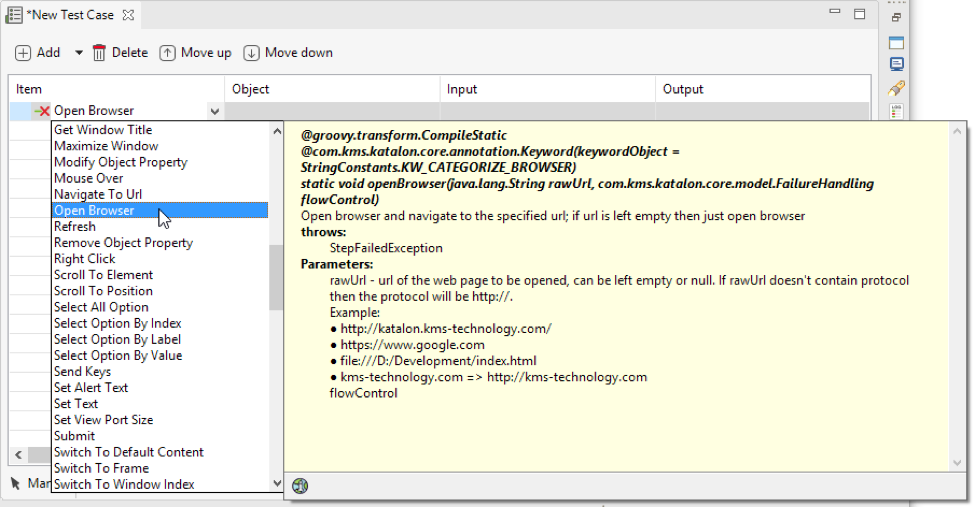
**Chế độ Manual**is là một chức năng giúp người người dùng tạo các test case thủ công trong Katalon Studio, sử dụng phương pháp testing theo keyword.Điều này cho phép người dùng không quen với việc lâp trình có thể tự tạo ra các test case cho riêng mình một cách dễ dàng hơn,tạo một test case tự đọng bằng chế độ manual.

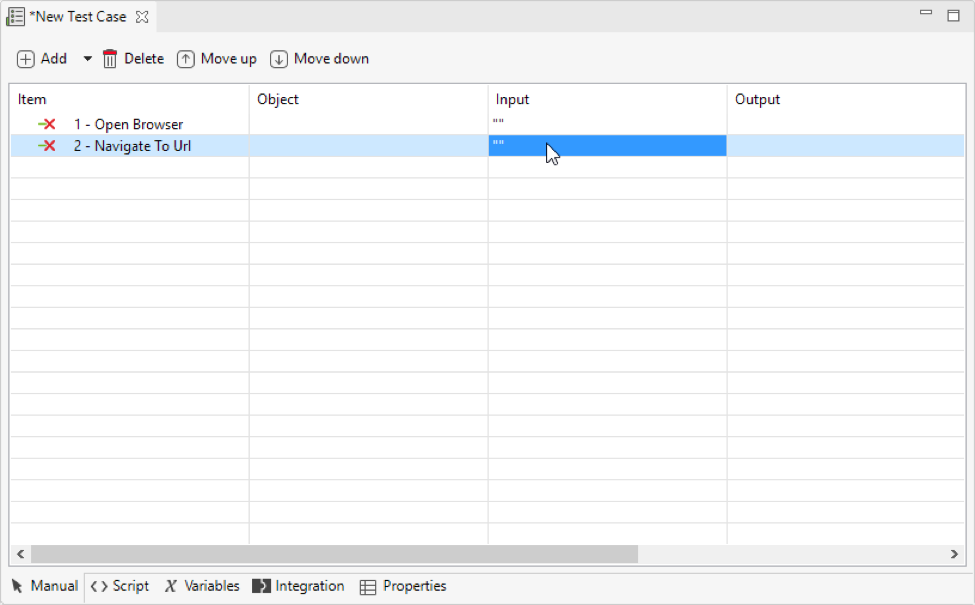
Thực hiện các bước sau đây để tạo ra test case manual:

*1.chọn****File > New > Test Case****từ menu chính để tạo một test case. Đặt tên cho test case, rồi click****OK****.* 

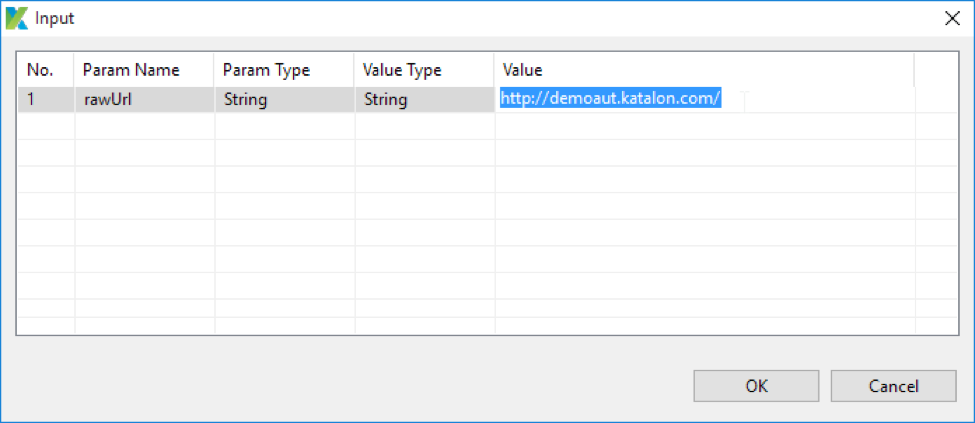
Test case khi đã được tạo:

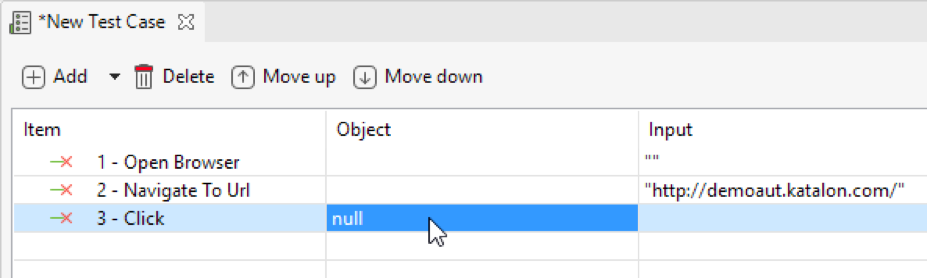
*2. Chọn****Add > Web UI Keyword****từ command toolbar.* 

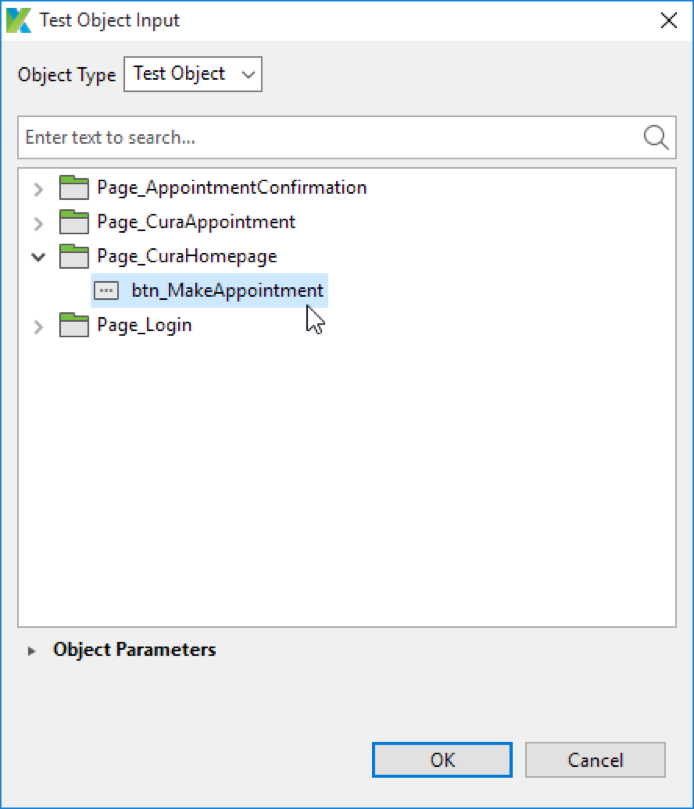
*3. Chọn* ***Open Browser****keyword:* 

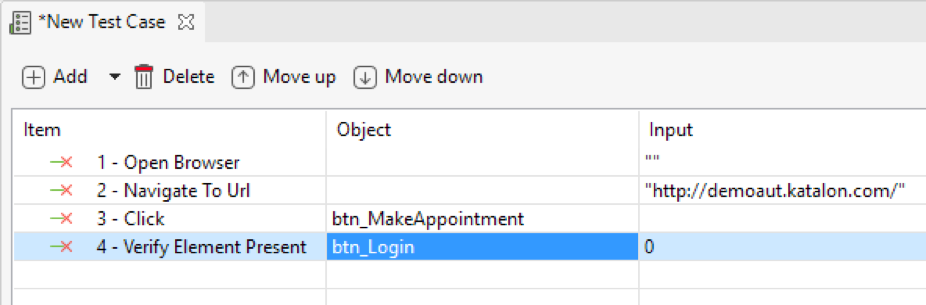
*4.Thêm keyword cho* ***Navigate To Url****. Keyword này sẽ điều hướng đến một URL cụ thể. Double click mục****Input****để cung cấp data bổ sung (parameters) cho keyword.* 

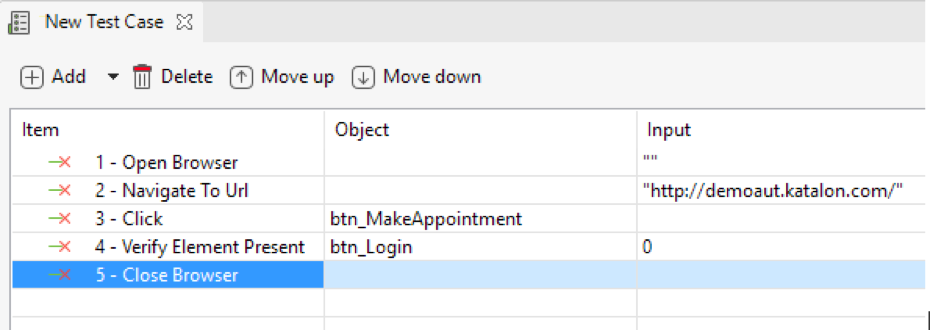
**Input** dialog xuất hiện giống như trong hình. Mỗi dòng cho thấy một parameter.



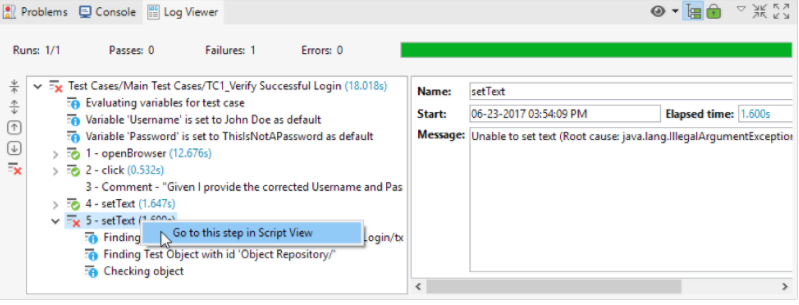
*5.Nhập URL của Katalon demo AUT vào cột****Value****rồi click****OK*** *->* *thêm keyword cho nút****Click****. This keyword thể hiện một đối với object của nó.Bạn phải tạo mối liên kết giữa object và nút. Double click vào* ***Object****để mở****Test Object Input****dialog* 

*6. Trong Test Object Input dialog, chọn object rồi nhấn* ***OK****.* 

*7.Thêm keyword cho* ***Verify Element Presen****t*

*8. Thêm keyword* ***Close Browser*** *sau đó save test case ­->* *Click****Run****ở main toolbar* 

***Kết quả:***



*3.Chế Script*

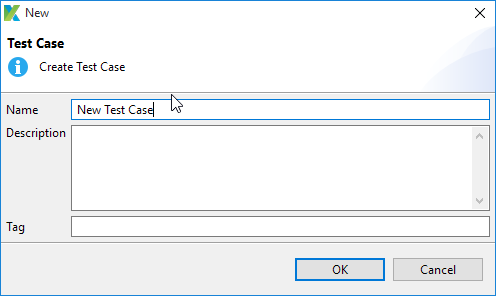
ở mục [Manual view](https://docs.katalon.com/display/KD/Test+Case+Manual+View), Katalon Studio cho phép các lâp trình viên chuyên nghiệp có thể viết các test tự động ở phần Script view của test cases.Người dùng với kiến thức về Groovy/java có thể tùy ý điều chình test case nếu cần.

Tạo một test case mẫu với ví dụ sau đây:

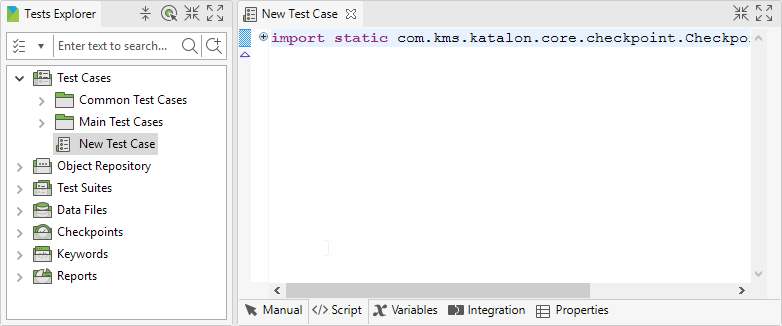
* Mở browser
* Chạy một website
* Click vào một mục nào đó
* Kiểm tra xem mục đó có tồn tại trong trang web không
* Đóng browser

Làm theo các bước sau đây để tạo ra một test case mẫu theo bối cảnh như trên **Script view**:

1. Chọn **File > New > Test Case** từ main menu.Mục **New Test Case** dialog sẽ xuất hiện. Cung cấp tên cho test case mới, click **OK**.



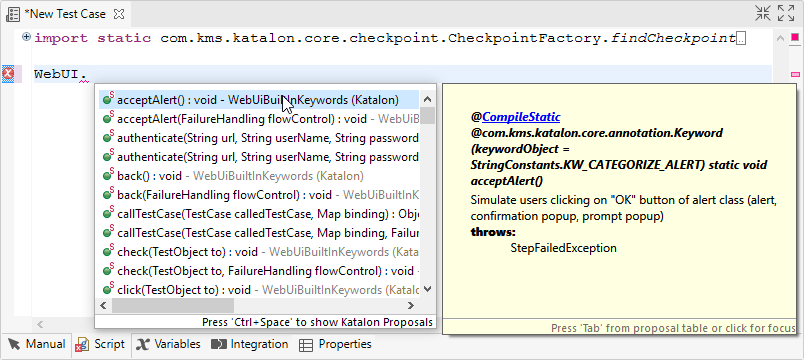
1. Khi test case mới đã được tạo,bạn có thể chuyển sang chế độ **Script view** bằng cách bấm vào thẻ Script ở phần dưới của test case editor và quan sát các test ở **Manual view** được biên dịch Groovy script của **Script view**.



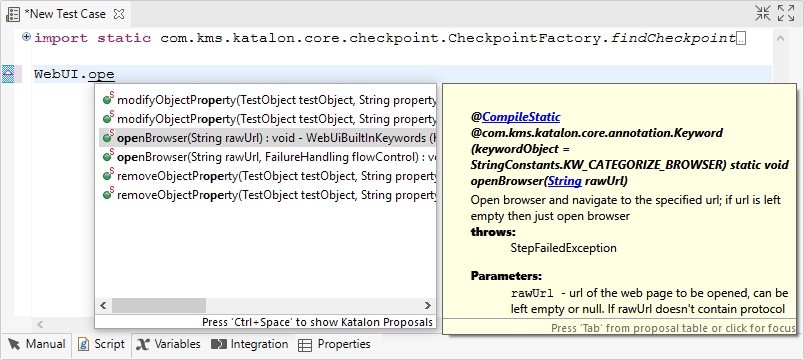
1. Để sử dụng các keyword được tích hợp sẵn trong **WebUI**, nhập các từ khóa sau đây vào mục editor.



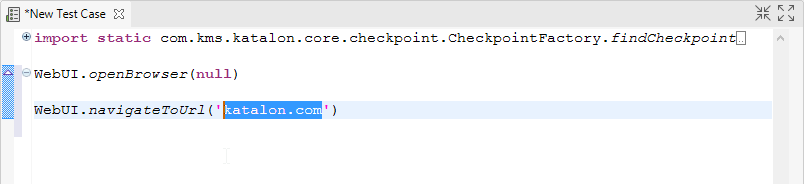
**Hàm Content Assist** sẽ được gọi sau dấu chấm. **Content Assist** sẽ cung cấp người dùng với suggestion để hoàn thiện code. Cho nên, các keyword được tích hợp cho WebUI testing sẽ được thể hiện giống như dưới đây:



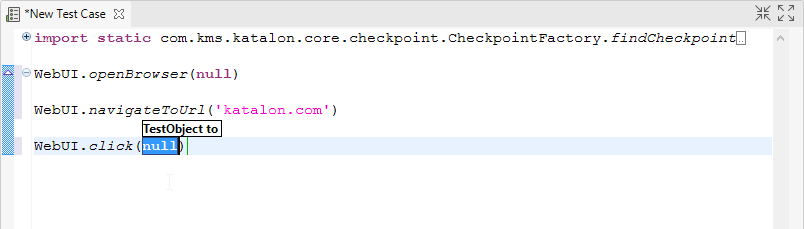
1. Chọn [**Open Browser**](https://docs.katalon.com/display/KD/%5BWebUI%5D+Open+Browser) keyword.Keyword này sẽ điều hướng đến URL được cung cấp.Keywords sẽ có các giá trị tương ứng với nó



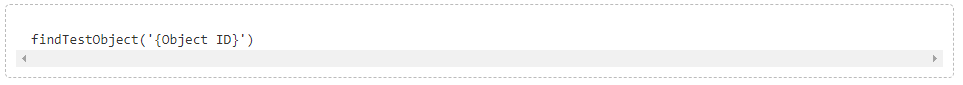
1. Nhập keyword cho [**Navigate To Url**](https://docs.katalon.com/display/KD/%5BWebUI%5D+Navigate+to+Url).Keyword này sẽ điều hướng đên URL cụ thể. Bây giờ, Nhập URL của Katalon Studio ([katalon.com](http://katalon.com/)) như một giá trị input.



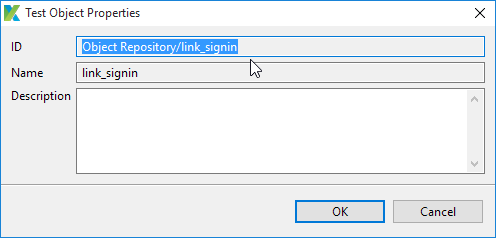
1. Nhập keyword [**Click**](https://docs.katalon.com/display/KD/%5BWebUI%5D+Click).Keyword này thể hiện hành động đối với object được cho.Ta cần phải chỉ ra một object cụ thể cho hành động.



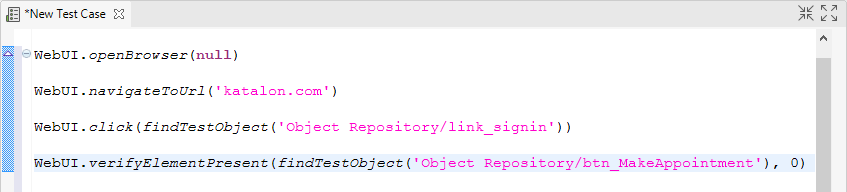
1. Sử dụng các cú pháp sau đây để chỉ ra object trong **Object Repository** (tương tự, ta có thể kéo thả các object vào mục test case editor để biên dịch ra cá cú pháp)



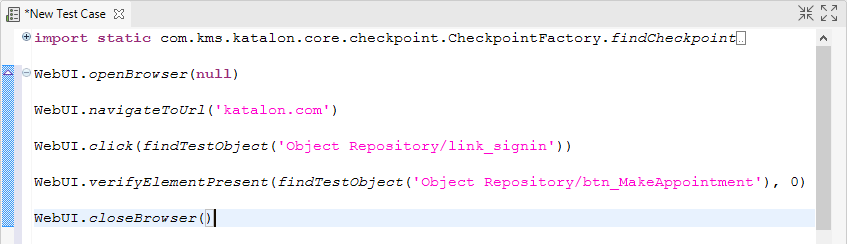
**Object ID** chính là Id của object đó trong Katalon Studio.



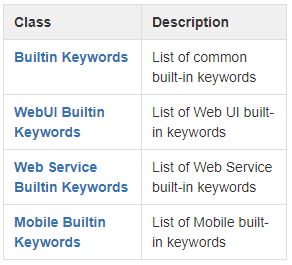
1. Nhập các keyword cho [**Verify Element Present**](https://docs.katalon.com/display/KD/%5BWebUI%5D+Verify+Element+Present).Keyword này sẽ xác nhận liệ một object có được show ra trong browser. Tương tự như bước vừa rồi,chứng ta cần phải xác định cụ thể object được sử dụng với keyword sau đây.



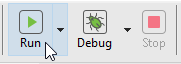
1. Thêm keyword cho [**Close Browser**](https://docs.katalon.com/display/KD/%5BWebUI%5D+Close+Browser) và lưu test case



các API docs có thể hữu dụng trong Script view



1. Click **Run** ở toolbar để thực hiện test case.



Katalon Studio có thể thục hiện tất cả bước của test key mẫu.

Graphical User Interface overview

Toolbar

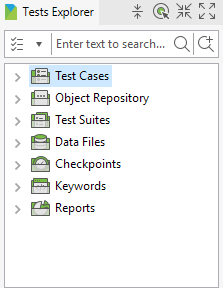
**Toolbar** cung cấp cức chức năng chính mà chúng ta thường sử dụng (e.g. tạo ra test resources hoặc thực thi test case).



|  |  |
| --- | --- |
| Icon | Description |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2014%3A58%3A59.png?version=1&modificationDate=1498723139000&api=v2 | Lưu lại tiến trình đang mở. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A0%3A23.png?version=1&modificationDate=1498723224000&api=v2 | Lưu lại tất cả tiến trình đang mở. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A0%3A55.png?version=1&modificationDate=1498723255000&api=v2 | Tạo một test mới. Các chức năng sau đây được lưu trong dropdownlist:   * Folder * Test Case * Test Suite * Test Suite Collection * Test Object * Web Service Request * Test Data * Checkpoint * Package * Keyword |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A1%3A27.png?version=1&modificationDate=1498723287000&api=v2 | Mở **Object Spy** dialog để dò các giá trị trong website. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A2%3A0.png?version=1&modificationDate=1498723320000&api=v2 | Mở **Mobile Object Spy** dialog để dò các giá trị trong mobile applications. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A2%3A58.png?version=1&modificationDate=1498723379000&api=v2 | Mở **Record** dialog để ghi lại WebUI test cases. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A3%3A21.png?version=1&modificationDate=1498723401000&api=v2 | Mở **Mobile Recorder** dialog để lưu lại Mobile test cases. |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A3%3A58.png?version=1&modificationDate=1498723438000&api=v2 | Chạy test case đang mở. Bạn có thể chọn các browser sau đây từ mục dropdownlist:   * Chrome * Firefox * IE * Safari * Edge * Remote * Headless * Android * iOS (on macOS) * Custom |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A4%3A31.png?version=1&modificationDate=1498723471000&api=v2 | Debug test case đang mở. Bạn có thể chọn các browser sau đây từ mục dropdownlist :   * Chrome * Firefox * IE * Safari * Edge * Remote * Headless * Android * iOS (on macOS) * Custom |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A5%3A16.png?version=1&modificationDate=1498723516000&api=v2 | Dừng thực thi |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A5%3A45.png?version=1&modificationDate=1498723545000&api=v2 | Mở **Command Builder** để chạy các command trong mục CMD |
| http://docs.katalon.com/download/attachments/786558/image2017-6-29%2015%3A6%3A29.png?version=1&modificationDate=1498723590000&api=v2 | Các lệnh cho Git. Bạn có thể chọn các lựa chọn sau đây từ dropdownlist (after  [enabling Git](https://docs.katalon.com/display/KD/Git+Integration)):   * Clone Project * Share Project * Show History * Manage Branches * Commit * Push * Pull * Fetch |

# Tests Explorer View

**Tests Explorer** view cho phép ta xem lại cấu trúc của project mình và truy cập vào các mục mong muốn một cách nhanh chóng. Sử dụng menu ở the view, bạn có thể tạo một tiến trình mới, xắp xếp lại các item trong mục view hoặc kéo thả chúng vào view thích hợp.

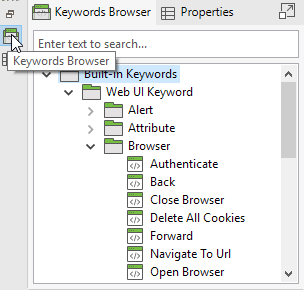


Where:

|  |  |
| --- | --- |
| **Group** | **Description** |
| Test Cases | Liệt kê ra tất cả các [test cases](https://docs.katalon.com/display/KD/Test+Case+Manual+View) trong project |
| Object Repository | Liệt kê ra tất cả các [test objects](https://docs.katalon.com/display/KD/Manage+Test+Object) trong project |
| Test Suites | Liệt kê ra tất cả các [test suites](https://docs.katalon.com/display/KD/Execute+a+test+suite) và [test suite collections](https://docs.katalon.com/display/KD/Execute+a+test+suite+collection) trong project |
| Data Files | Liệt kê ra tất cả các [test data](https://docs.katalon.com/display/KD/Manage+Test+Data) trong project |
| Checkpoints | Liệt kê ra tất cả các [checkpoints](https://docs.katalon.com/display/KD/Manage+Checkpoint) trong project |
| Keywords | Liệt kê ra tất cả các [custom keywords](https://docs.katalon.com/display/KD/Define+custom+keywords) trong project |
| Reports | Liệt kê ra tất cả các [reports](https://docs.katalon.com/display/KD/Test+Report) trong project |

# Keywords Browser View

**Keywords Browser** hiện ra các keyword được hỗ trợ bởi Katalon Studio. Bạn có thể kéo thả các keywords vào mục Test Case editor.



# Editors

The editor được sử dụng để thay đổi thông tin của object. Mỗi thực thi có editor riêng.

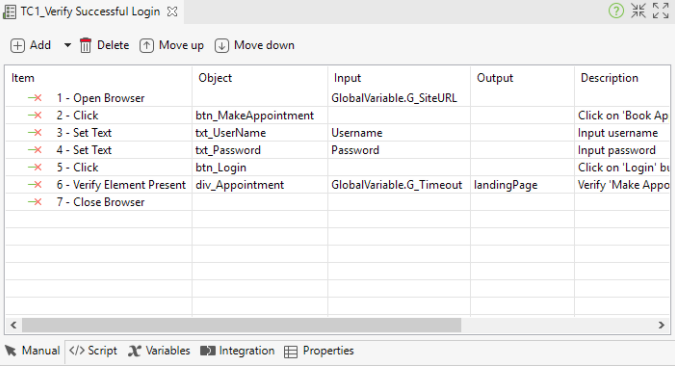
## **Test Case Editor**

Khi mở một test case, thông tin chi tiết của được thể hiện ở editor chức các tabs sâu đây:

* Manual tab
* Script tab
* Variables tab
* Integration tab
* Properties tab

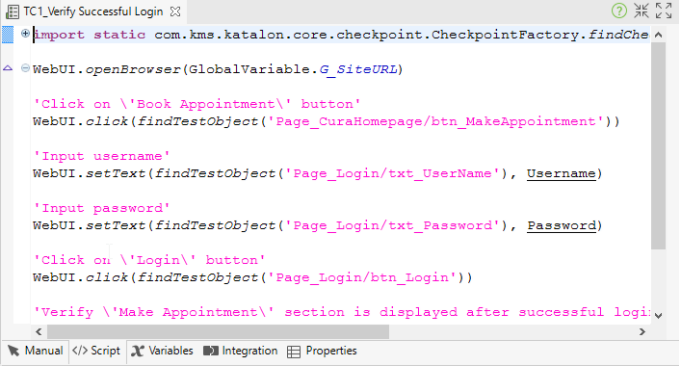
### Manual tab

Manual tab thể hiện Manual view nơi các keyword căn bản cho phép users các test case một cách dễ dàng. Đọc  [Test Case Manual View](http://docs.katalon.com/display/KD/Test+Case+Manual+View) để biết thêm chi tiết.



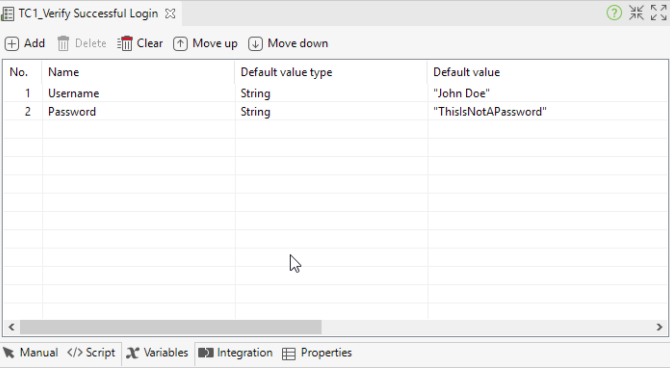
### Script tab

Tab này thể hiẹn Script view nơi các lập trình viên kinh nghiệm với groovy/java có thể tùy chỉnh test case. Đọc [Test Case Script View](http://docs.katalon.com/display/KD/Test+Case+Script+View) để hiểu rõ hơn.



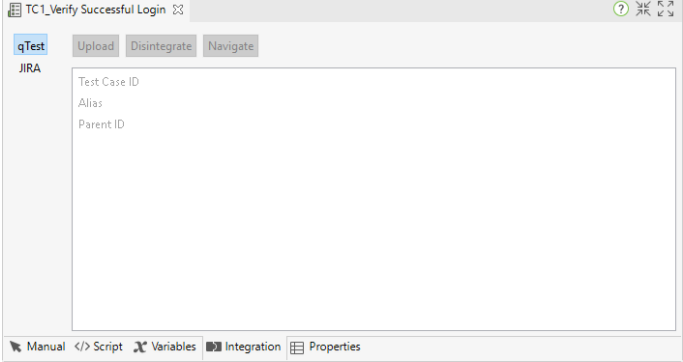
### Variables tab

The Variables tab cho biết tất cả các biến của test case. Đọc [Public Variables](http://docs.katalon.com/display/KD/Variable+Types#VariableTypes-Publicvariables) để hiểu rõ hơn.



### Integration tab

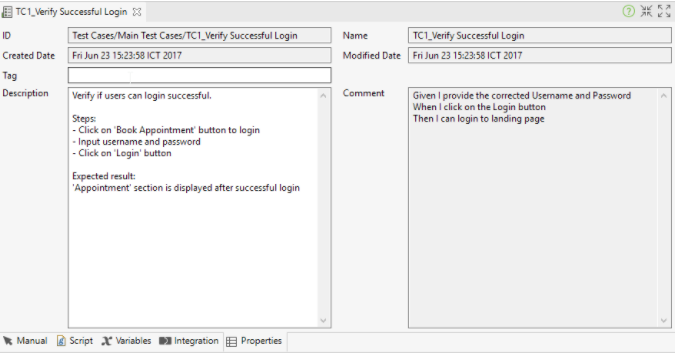
Các thông tin liên quan đến sự tương thích của test case với qTest được thể hiện ở tab này. Đọc [Integrate test case](http://docs.katalon.com/display/KD/Integrate+test+case) để hiểu rõ hơn.



### Properties tab

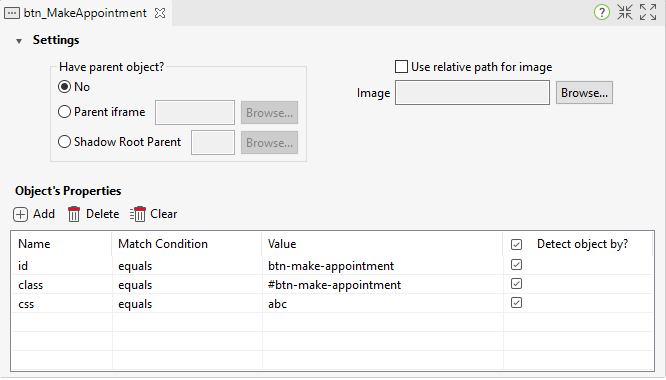
Các thông tin chung về test case được thể hiện ở tab này.

* **Description:** Người dùng có thể tùy chỉnh mục này để cung cấp thông tin chi tiết cho test case
* **Comment:**mục này là **read-only**. Nội dung được giải nén và được nhân lên nhìu lần từ keyword ở [Comment](https://docs.katalon.com/display/KD/%5BWebUI%5D+Comment) trong test case. Ví dụ dưới đây diễn tả một test case xác nhận login thành công và comment diễn tả bối cảnh được viết bằng BDD's Gherkin syntax.



## **Test Object Editor**

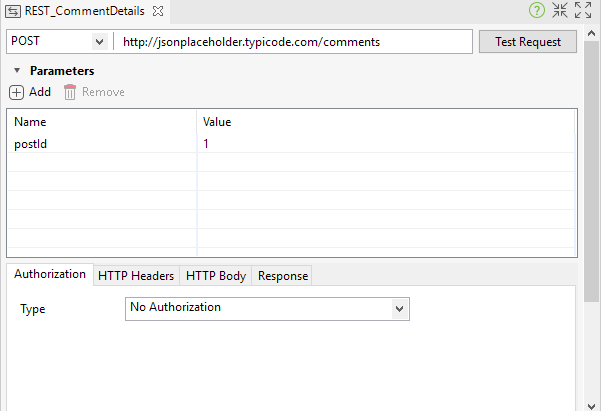
Khi mở một test object, chi tiết của nó bao gồm including properties và cơ chế xác định object được thể hiện ở Test Object editor. Đọc  [Spy Object](http://docs.katalon.com/display/KD/Spy+Object) để hiểu rõ hơn.



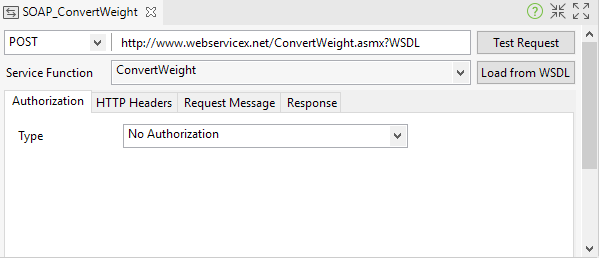
## **Web Service Editor**

### *RESTful Request Object Editor*

Khi bạn mở một RESTful request object, chi tiết của nó bao gồm URL, phương pháp yêu cầu, các thông số... được hiện ra the editor. Đọc [Web Services](http://docs.katalon.com/display/KD/Web+Services) để hiểu rõ hơn.



### *SOAP Request Object Editor*

Khi bạn mở SOAP request object, thông tin của nó bao gồm URL, phương pháp yêu cầu, các thông số... được thể hiện ở mục the editor. Đọc [Web Services](http://docs.katalon.com/display/KD/Web+Services) để biết thêm chi tiết.

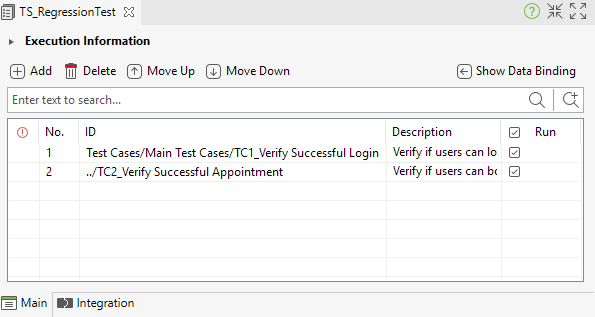
## **Test Suite Editor**

Khi bạn mở một test suite, các thông tin của nó được thể hiện ở Test Suite editor chứa các tab sau đây:

* Main tab
* Integration tab

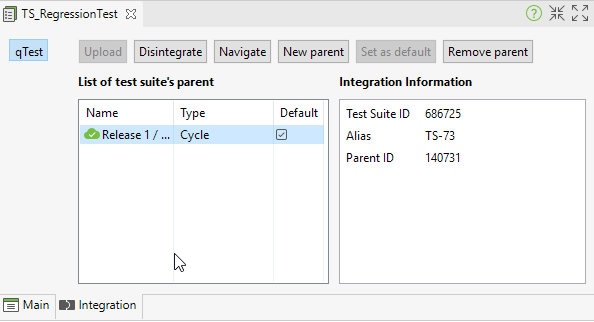
### *Main tab*

The main tab thể hiện các thông tin cơ bản về test suite ví dự như test cases được thực thi, Cơ chế thực thi và dữ liệu của nó. Đọc [Execute a test suite](http://docs.katalon.com/display/KD/Execute+a+test+suite) để biết thêm chi tiết.



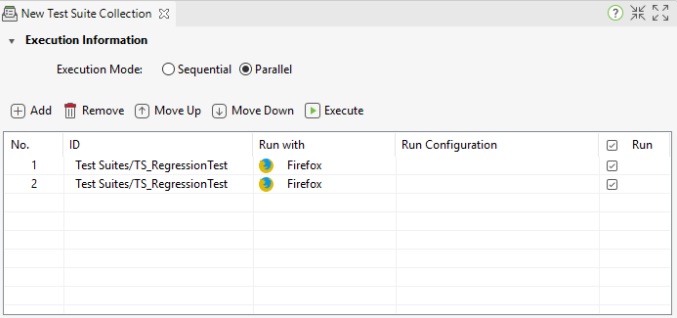
### *Integration tab*

Các thông tin liên quan đến test suite kết hợp với qTest được thể hiện ở tab này. Đọc [Integrate test suite](http://docs.katalon.com/display/KD/Integrate+test+suite) để biết them chi tiết.



## **Test Suite Collection Editor**

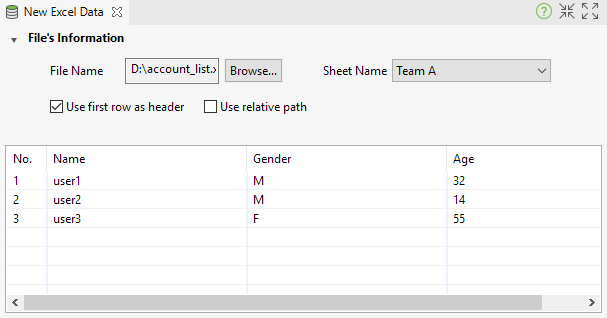
Test Suite Collection chứa một bộ test suite cho phép người dùng thực thi nhiều test suite với nhau. Đọc [Execute a test suite collection](http://docs.katalon.com/display/KD/Execute+a+test+suite+collection) để hiểu rõ hơn.



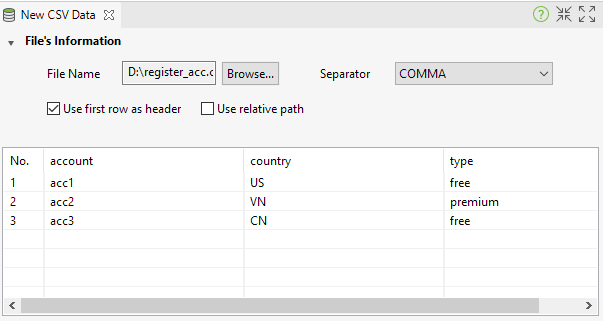
## **Data File Editor**

Khi bạn mở một data file, các thông tin của nó bao gồm data source và preview data set... được thể hiện ở the editor. Đọc [Manage Test Data](http://docs.katalon.com/display/KD/Manage+Test+Data) để biết thêm chi tiết.

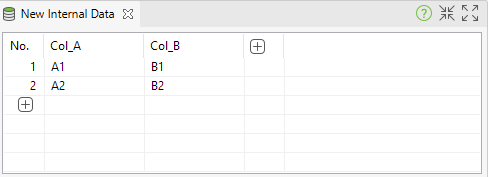
* Nếu dữ liệu từ Excel file:



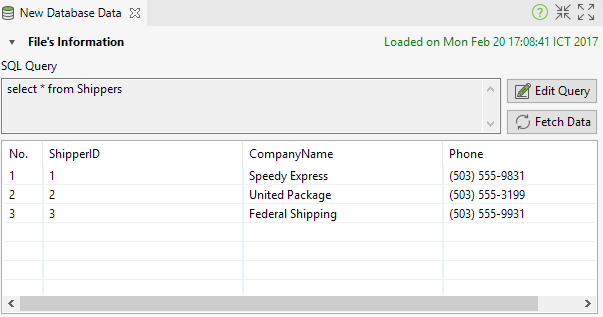
* Nếu dữ liệu test từ CSV file:



* Nếu dữ liệu test ở trong test project:

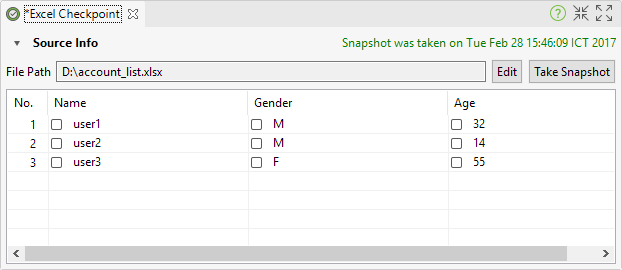
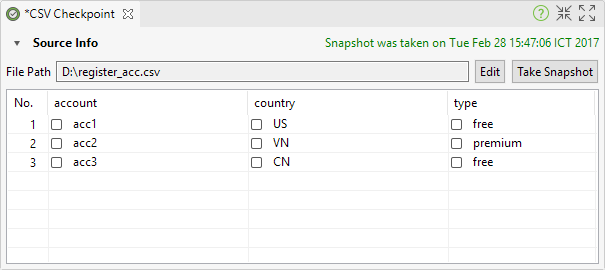
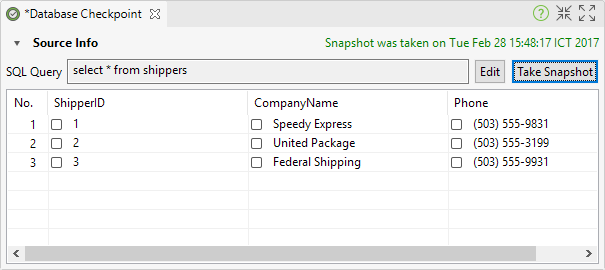
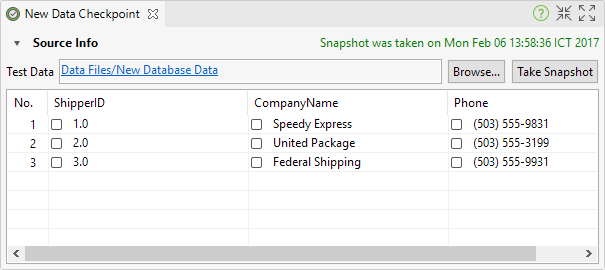


* Nếu data test từ một chuỗi SQL:

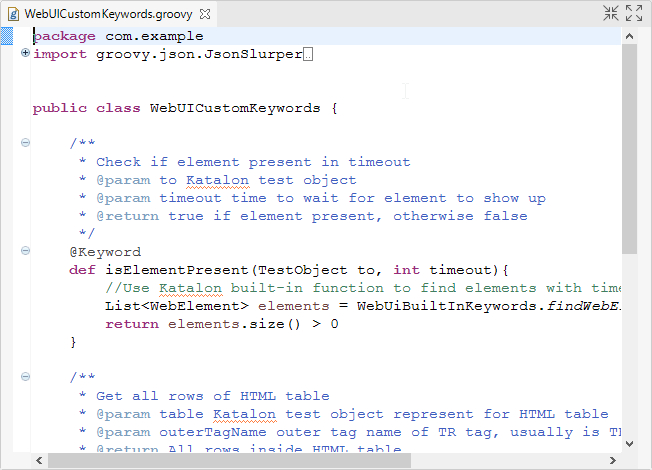


## **Checkpoint Editor**

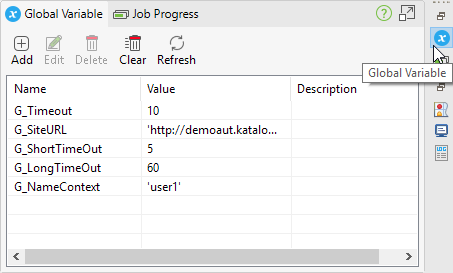
Khi bạn mở một checkpoint, các thông tin của nó bao gồm data source... được thể hiện the editor. Đọc [Manage Checkpoint](http://docs.katalon.com/display/KD/Manage+Checkpoint) để hiểu rõ hơn.

* Nếu như data từ một file Excel: 
* Nếu như source data từ một file CSV: 
* Nếu như source data từ database: 
* Nếu như nguồn data đã có sẵn: 

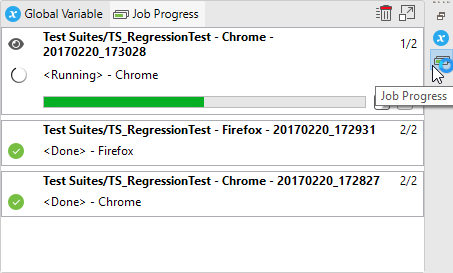
## **Keyword Editor**

Khi bạn mở một custom keyword, các thông tin của nó được thể hiện ở scripting editor. Scripting editor tưa như Script view của test cases nơi bạn có thể định nghĩa những custom keywords dễ dàng bằng Groovy hoặc Java. Đọc [Define custom keywords](http://docs.katalon.com/display/KD/Define+custom+keywords) để hiểu rõ hơn. 

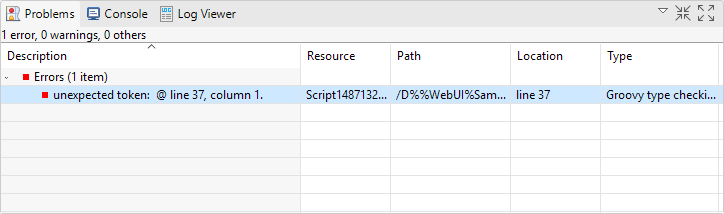
# Global Variables View

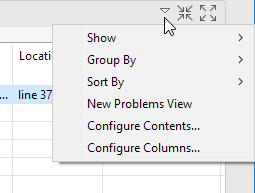
The Global Variables view cho phép bạn duyệt qua danh sách các biến toàn cục được định nghĩa trong project của bạn. Đọc [Global Variables](http://docs.katalon.com/display/KD/Variable+Types#VariableTypes-Globalvariables) để hiểu rõ hơn. 

# Job Progress View

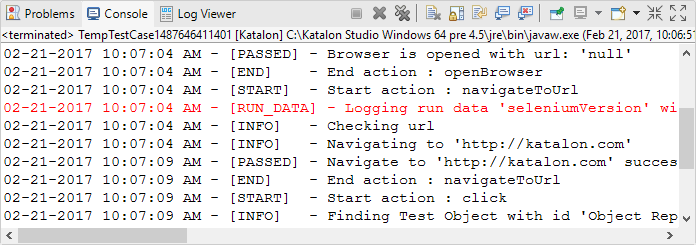
The Job Progress view cho phép bạn xem quá trình thực thi của test cases và test suites.  


# Problems View

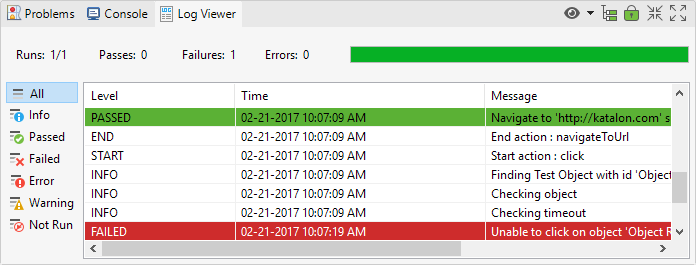
The Problems view cho phép xem lỗi và nhưng cảnh báo khi bạn thiết kế một project hay test case, test suite, test object hoặc test data. 

Thông báo từ the Problems view có thể được cấu hình thông qua dropdown-menu của view này. 

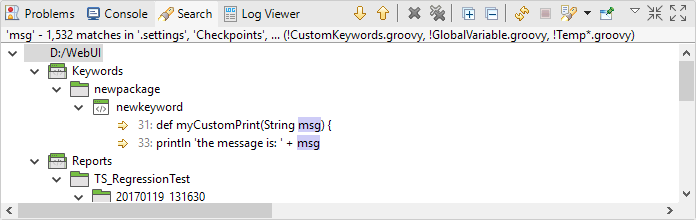
# Console View

The Console view thể the system logs của tất cả những tiến trình đang chạy khi test case đang được thực thi. Console output được tạo ra từ from test scripts cũng được thể hiện ở đây. 

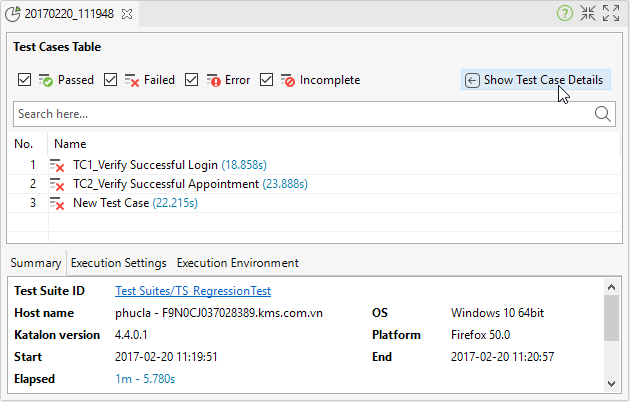
# Log Viewer View

The Log Viewer view cho biết các báo cáo report/log của một test thực thi. Đọc [View Execution Log](http://docs.katalon.com/display/KD/View+Execution+Log) để hiểu rõ hơn. 

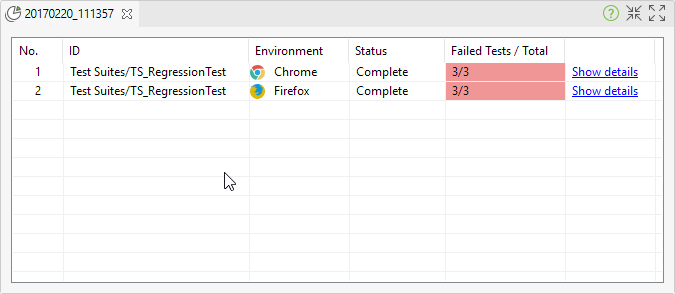
# Search View

The Search view cho biết các kết quả từ hàm search. Bạn có thể double-click vào trường search để định vị đến vị trí tương ứng trong editor. 

# Report View

The Report view cho phép ta xem các thông tin tiết của một test case hoàn thiện đói với một test suite. Đọc [Test Suite Report](http://docs.katalon.com/display/KD/Test+Suite+Report) để hiểu rõ hơn. 

# Test Suite Collection Report view

The Test Suite Collection Report view cho phép ta xem các thông tin chi tiết của những test case hoàn thiện đối với những test. Đọc [Test Suite Collection Report](http://docs.katalon.com/display/KD/Test+Suite+Collection+Report) để hiểu rõ hơn. 

Tham khảo

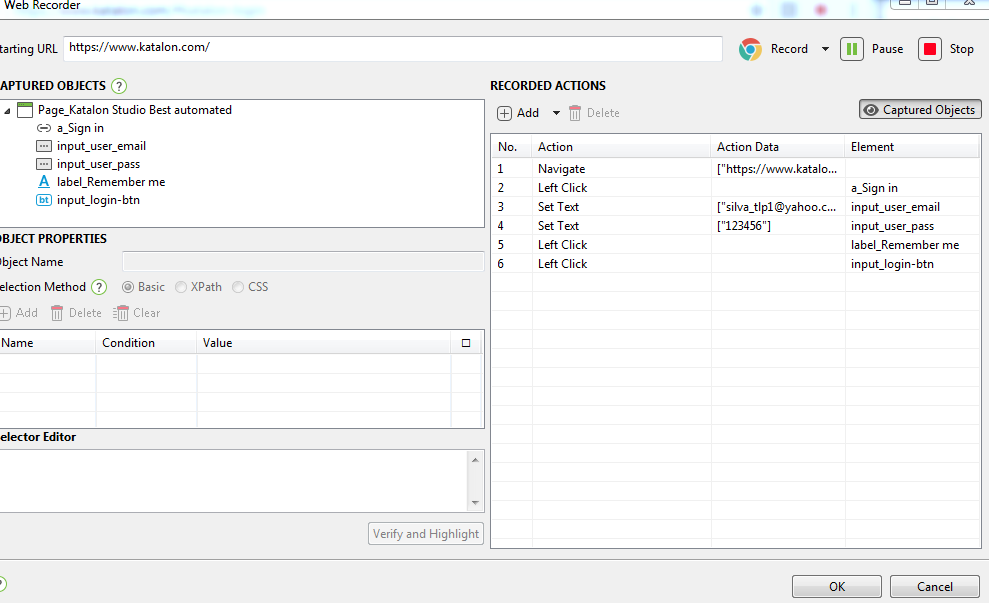
Katalon Studio homepage – [https://katalon.com](https://katalon.com/).

Katalon Studio documentation - [https://docs.katalon.com](https://docs.katalon.com/).

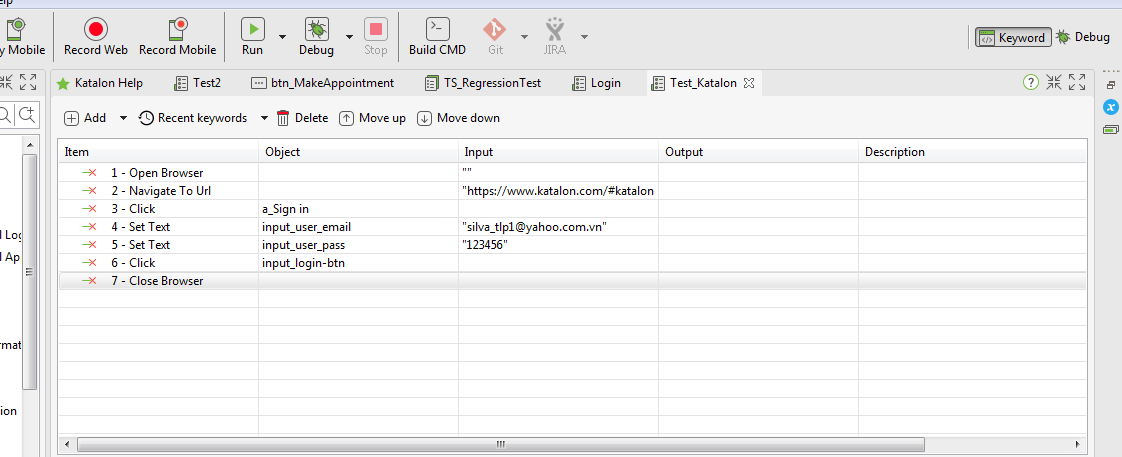
DEMO KATALON STUDIO

TESTING WEB

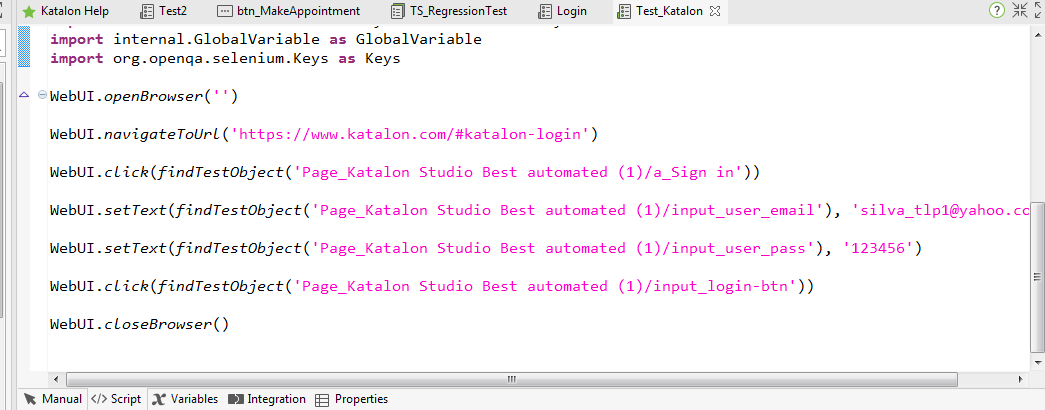
1. Tạo ra một web Test case và Record Web
2. Chonj URL và Browser để ghi lại
3. Khi ghi lại, mọi hành động sẽ được liệt kê ở RECORDED ACTIONS
4. Script được lưu lại ở created Test case để sử dụng sau này

1/ Katalon home page

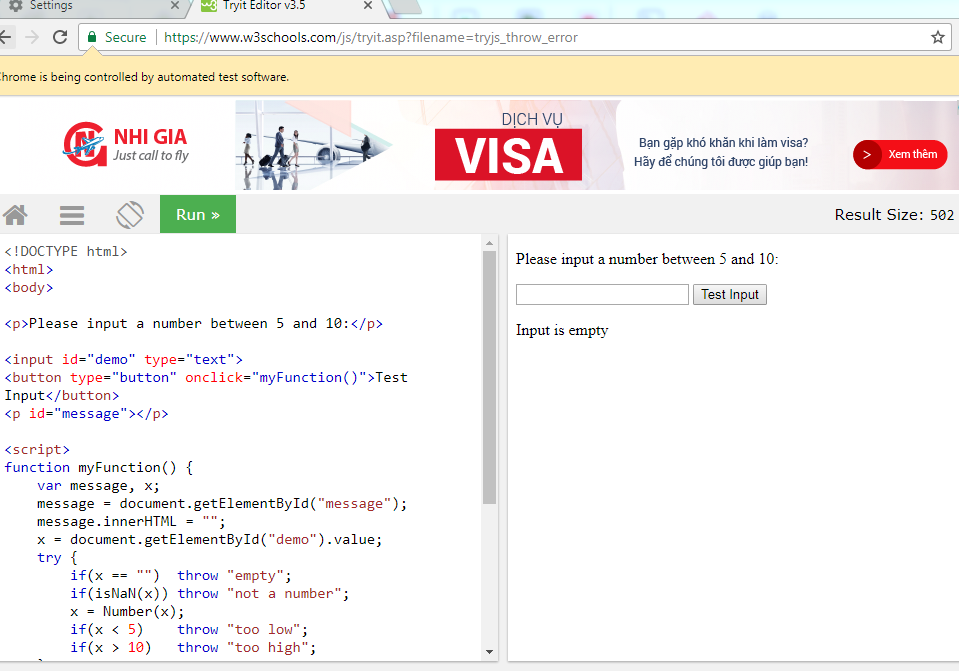
Record Menu



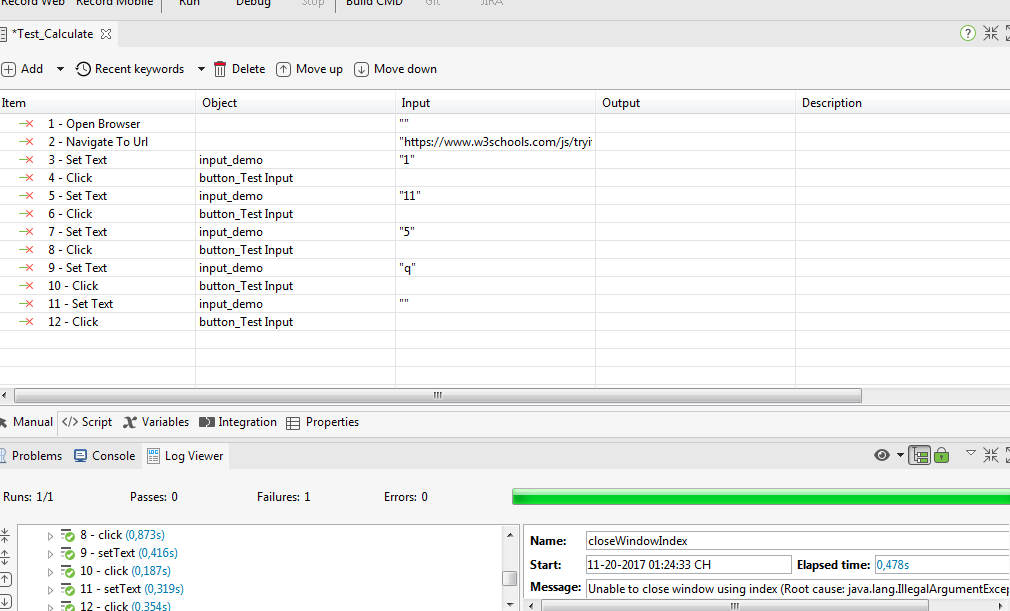
Manual mode



Script mode

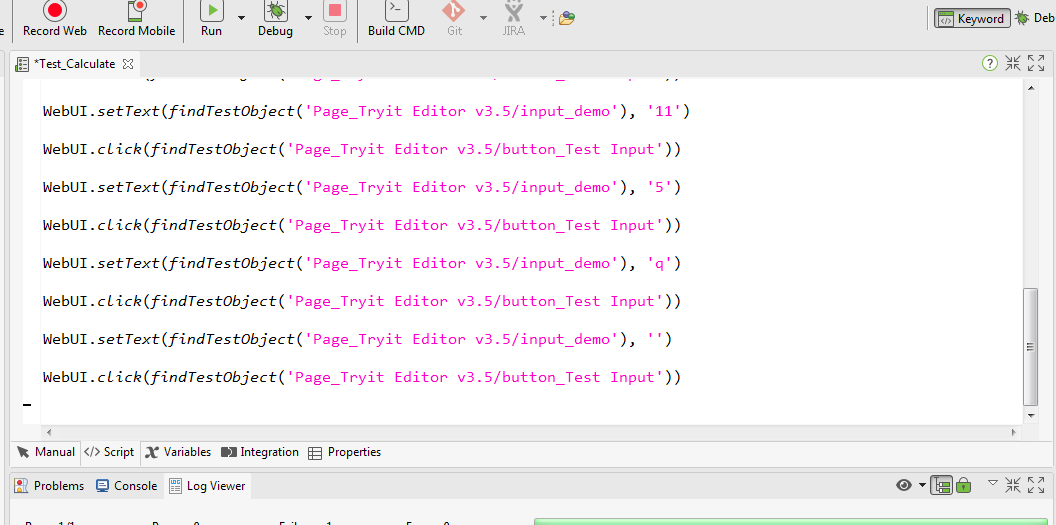
2/ Simple HTML-Checking right number

Test screen from w3schools

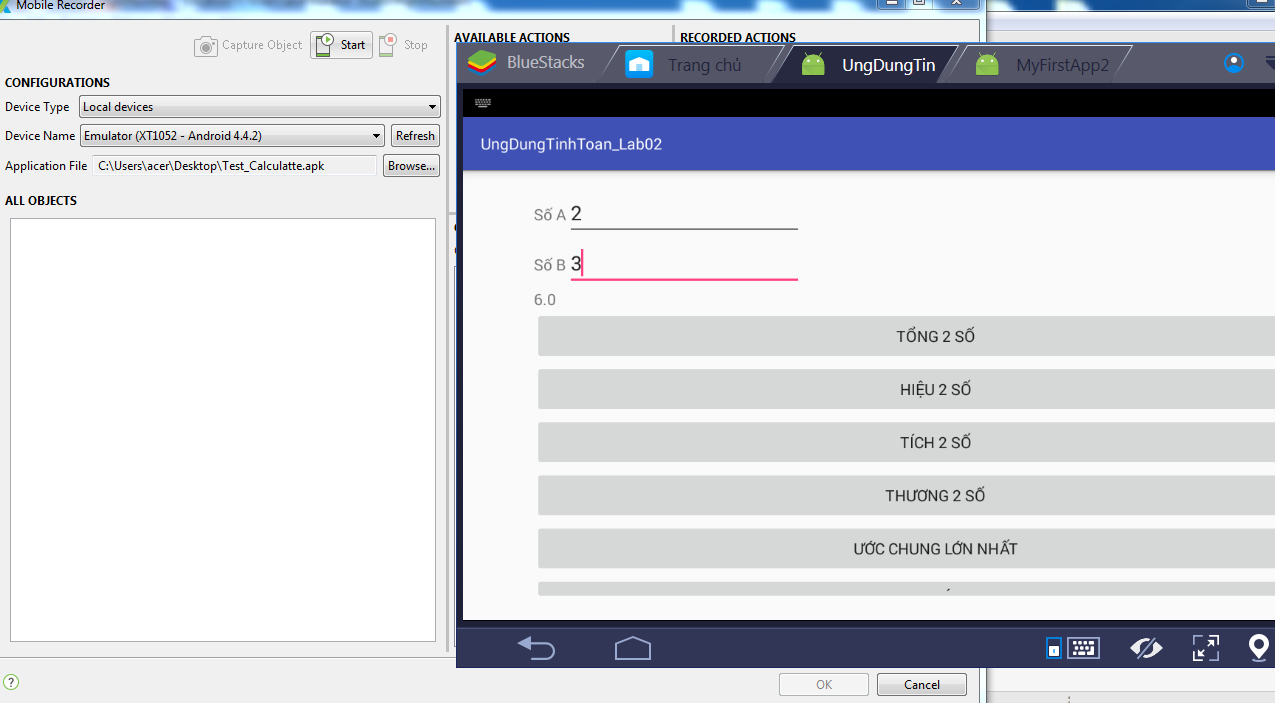


Manual mode

Script mode

TESTING MOBILE APPLICATION

1. Mở Mobile Emulator và app cần test
2. Tạo ra một mobile Test case và Record Mobile
3. Chọn file apk của bạn
4. Khi ghi lại, mọi hành động đều được liệt kê trong RECORDED ACTIONS
5. Script được lưu lại ở Test case để sử dụng sau này



Evaluate

Để làm kiểm thử tự động với ứng dụng Web và Mobile, thường mọi người sẽ chọn Selenium/Appium và tiến hành thiết kế các framework theo nhu cầu của từng dự án. Nhưng nhìn chung , việc này yêu cầu kỹ sư kiểm thử tự động phải có một khả năng lập trình nhất định. Katalon ra đời giúp chúng ta giải quyết hầu hết các vấn đề cần có của một framework như hướng từ khóa, dữ liệu và tạo các báo cáo kiểm thử. Mặc dù, trong từng dự án cụ thể, việc viết thêm các từ khóa mới là điều chắc chắn, nhưng Katalon giúp chúng ta giảm thiểu việc này. Vấn đề viết thêm từ khóa này chỉ cần một người Senior trong nhóm là đủ.Mặc dù Katalon Studio mới xuất hiện nhưng cũng đã cung cấp những giải pháp cơ bản cho những dự án kiểm thử tự động. Với một tương lai phát triển, Katalon Studio hứa hẹn sẽ là một công cụ kiểm thử phổ biến trong cộng đồng kiểm thử phần mềm chúng ta.