

## 1.1 Troubleshooting

Trong quá trình xây dựng kịch bản kiểm thử, người dùng có thể gặp một số vấn đề khác nhau liên quan đến tính logic, thuật toán... dẫn đến việc kịch bản kiểm thử được thực thi không đúng theo yêu cầu nghiệp vụ. Một số vấn đề thường gặp có thể kể đến như:

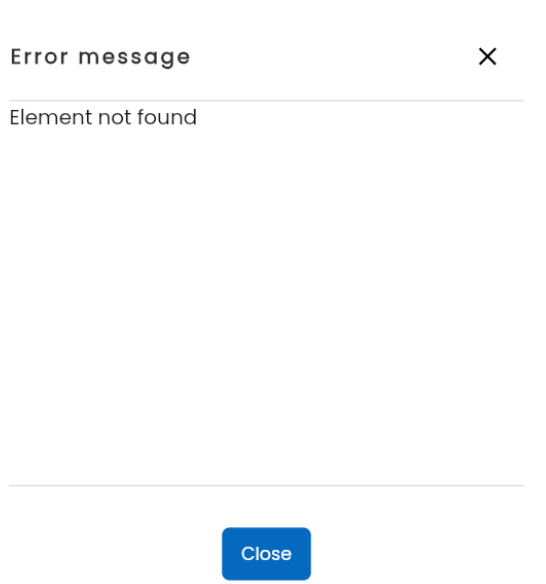
- Lỗi cấu hình kịch bản kiểm thử
- Vấn đề khi thực thi kịch bản kiểm thử
- Vấn đề liên quan đến Agent...

Khi cấu hình kịch bản kiểm thử, người dùng có thể mắc các lỗi về logic, kiểu dữ liệu... dẫn đến fail, thậm chí error khi đang debug hoặc thực thi kịch bản kiểm thử.

Trong phần này, LLQ QA Platform sẽ chỉ ra các lỗi thường gặp khi cấu hình kịch bản kiểm thử, hỗ trợ khách hàng khắc phục tình trạng gặp phải.

### 1.1.1 Lỗi cấu hình kịch bản kiểm thử

#### 1.1.1.1 Element not found



*Thông báo lỗi “Element not found” khi play keyword/ debug list action*

**TEST RUN RESULT**

|                 |              |
|-----------------|--------------|
| Expected result | Pass         |
| Actual result   | Fail         |
| Final status    | Fail         |
| Report bug      | Not reported |

**Actual result description** ⓘ  
Element not found

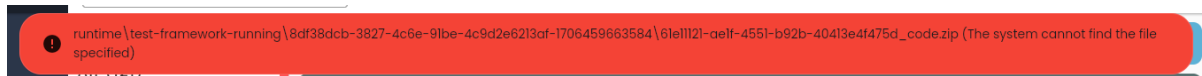
*Mô tả lỗi “Element not found” trong kết quả Run test script*

Element not found: Là lỗi không tìm thấy phần tử cần tương tác trên giao diện người dùng.  
Để dẫn đến sự cố “Element not found”, có một số nguyên nhân như sau:

| Nguyên nhân                                         | Mô tả                                                                                                                                                                                                                                                                                                                  | Cách khắc phục                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Địa chỉ xpath của element sai hoặc chưa đủ chi tiết | Khi sai địa chỉ xpath, hệ thống không thể tìm thấy element trên giao diện người dùng.                                                                                                                                                                                                                                  | Ưu tiên sử dụng kiểu dữ liệu Repository Element để định nghĩa locator cho các phần tử. Cần phân tích giao diện ứng dụng nhằm phân loại các loại element và xác định rõ các thuộc tính đặc trưng, đặc biệt đối với những phần tử có tính chất đặc biệt. Nên cấu hình locator sử dụng từ hai thuộc tính trở lên trong công thức XPath để đảm bảo khả năng phân biệt chính xác giữa các phần tử. Với các element đặc biệt, cần viết XPath thủ công theo đúng cấu trúc chuẩn: <code>//Element[@tag attribute='value']</code> . |
| Chọn sai element cho action                         | Khi sử dụng Element Repository hoặc công cụ Find element để xác định giá trị XPath của phần tử cần tương tác, người dùng có thể chọn sai loại element so với hành động mong muốn. Ví dụ, áp dụng action "select" cho một element có type là "input". Điều này dẫn đến lỗi hoặc hành vi không như kỳ vọng khi thực thi. | Kiểm tra lại định dạng của element và action đã lựa chọn để tương tác trong kịch bản kiểm thử.                                                                                                                                                                                                                                                                                                                                                                                                                             |

| Nguyên nhân                                           | Mô tả                                                                                                                                                                                                                                                                                                                                           | Cách khắc phục                                                                                                                                                  |
|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Giá trị truyền vào không tồn tại (các keyword select) | Khi dùng keyword Select cho element có định dạng là combobox (select by value, select by visible text), giá trị của tham số truyền vào không tồn tại trong combobox.                                                                                                                                                                            | Đảm bảo giá trị truyền vào chính xác (ký tự viết hoa, viết thường và dấu cách).                                                                                 |
| Element không tồn tại                                 | Do trình tự các bước trong kịch bản kiểm thử không chính xác hoặc điều kiện để xuất hiện element này không xảy ra.                                                                                                                                                                                                                              | Kiểm tra lại kịch bản kiểm thử và xây dựng các bước đảm bảo chính xác và đầy đủ logic cho các keyword If-else.                                                  |
| Không thể tương tác với element                       | <ul style="list-style-type: none"> <li>- Element nằm trong một hoặc một số iframe, hệ thống sẽ không thể tương tác với element đó nếu chưa có action Switch To Frame.</li> <li>- Element nằm ở một tab khác của trình duyệt, hệ thống sẽ không thể tương tác đến nếu chưa có action switch to window/switch to next/previous window.</li> </ul> | Xác định vị trí của element để thực hiện các action switch to frame/window. Khi thao tác xong các bước ở trong iframe, cần thực hiện Switch To Default Content. |
| Thời gian chờ quá lâu                                 | Khi thực hiện một action nào đó, nếu hiệu năng của hệ thống kiểm thử không đạt dẫn đến loading quá lâu, khi quá thời gian chờ mặc định vẫn không tìm thấy element cần tương tác.                                                                                                                                                                | Cài đặt thời gian sleep hoặc sử dụng wait element exist/disappear để tăng timeout cho action.                                                                   |

### 1.1.1.2 The system cannot find the file specified



Thông báo lỗi “The system cannot find the file specified” khi debug list action

*The system cannot find the file specified:* Là lỗi hệ thống không tìm thấy file được chỉ định

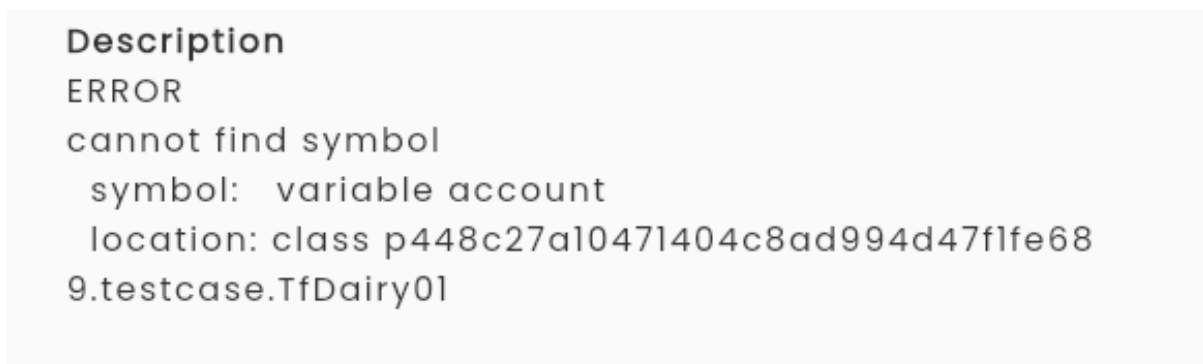
Nguyên nhân: Sự cố này phát sinh khi trong test script có chứa keyword Variable/ các keyword If-else, while, foreach... chưa được điền thông tin cho các thuộc tính của keyword

Cách khắc phục: Kiểm soát toàn bộ keyword trong script, đảm bảo không có keyword bị rỗng thông tin

### 1.1.1.3 Sự cố “Cannot find Symbol”



Thông báo sự cố “cannot find symbol” khi debug list action



Mô tả lỗi “cannot find symbol” khi run script

Cannot find symbol: là sự cố không tìm thấy tham số của action, shared steps hoặc test script được gọi ra không ở trạng thái active

Mô tả lỗi:

## Description

ERROR

cannot find symbol

1

symbol: **class RepositoryElement**892a1ce1716349cf

992a3ed6a34f64f2

2

location: class pa2180c92dcb34706a11b92f323b9781  
2.testcase.TfTestOfLoginIf01

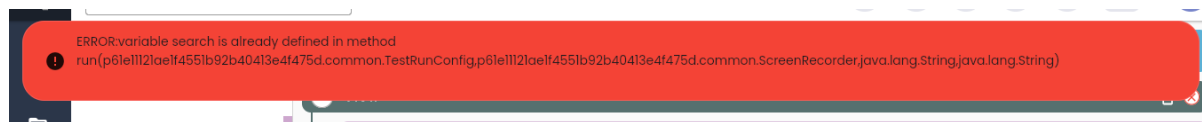
- 1: symbol không tìm thấy của action, trong ví dụ này là thẻ Element Repository
- 2: Vị trí phát sinh lỗi

Các nguyên nhân dẫn đến sự cố Cannot find symbol:

| Nguyên nhân                    | Mô tả                                                                                                                                                                                                                                 | Cách khắc phục                                                                                                                                                                                                                                                                                           |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Xoá parameter                  | Trong quá trình chỉnh sửa script/ suite, người dùng xoá các parameter không cần thiết đã xoá nhầm tham số còn cần sử dụng hoặc xoá parameter nhưng không xoá action                                                                   | Xác định vị trí lỗi theo tên symbol mà hệ thống không tìm thấy trong mô tả. <ul style="list-style-type: none"><li>- Trong trường hợp còn sử dụng action: Bổ sung parameter cho action bằng click vào action đó và click save</li><li>- Không còn sử dụng action đó trong script: xoá bỏ action</li></ul> |
| Không tạo tham số khi cấu hình | Khi sử dụng các action có thuộc tính returnType (ví dụ get text...), lựa chọn type là Variable và nhập tên tham số chưa có trong danh sách variable của test script/ suite dẫn đến hệ thống không tìm thấy tham số khi debug hoặc run | Đối với các action có thuộc tính returnType: <ul style="list-style-type: none"><li>- Người dùng chọn type là NewVariable nếu tên tham số chưa có trong danh sách variable của script</li><li>- Click “Find variable” để chọn 1 variable đã có nếu chọn type là Variable, hạn</li></ul>                   |

| Nguyên nhân                               | Mô tả                                                                                                                                                                                                                                                                                                | Cách khắc phục                                                |
|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
|                                           |                                                                                                                                                                                                                                                                                                      | chế nhập thủ công tránh nhập sai tên của tham số.             |
| Duplicate test script từ một project khác | Trong test script có sử dụng element repository hoặc shared steps, khi được nhân bản sang dự án khác, các tài nguyên đang sử dụng (repository, shared steps) không tồn tại, khi tiến hành cấu hình test script, người dùng không rà soát hết các vị trí bị null trong flow để thay thế hoặc loại bỏ. | Rà soát lại test script, thay thế hoặc loại bỏ các vị trí lỗi |
| Xoá repository                            | Trong script tồn tại action sử dụng element repository bị xoá khỏi hệ thống                                                                                                                                                                                                                          | Cập nhật xpath cho element đó                                 |
| Test script/Shared steps chưa active      | Khi cấu hình thực hiện gọi test script/shared steps nhưng các case này không ở trạng thái active                                                                                                                                                                                                     | Đổi Status của test script/shared steps thành active          |

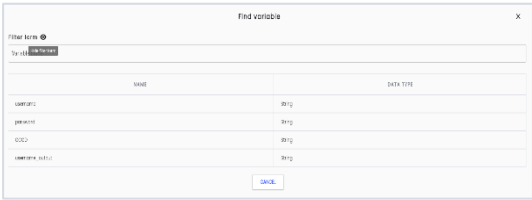
#### 4.9.1.4 Variable ... is already defined in method



Thông báo lỗi “*variable search is already defined in method*” khi debug list action

***Variable ... is already defined in method*** là lỗi một tham số nào đó đã được định nghĩa trong script.

Trong mỗi test của hệ thống kiểm thử tự động không cho phép đặt trùng tên của các variable. Khi người dùng tạo một tham số nhiều hơn 1 lần thì sẽ xảy ra lỗi này. Các nguyên nhân dẫn đến sự cố này:

| Nguyên nhân                                                      | Mô tả                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Cách khắc phục                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tạo 1 parameter 2 lần (bao gồm cả parameter cho input và output) | Trong khi sử dụng, người dùng vô tình tạo 2 parameter trùng tên do thao tác copy – drag keyword hoặc không kiểm soát được tên các parameter                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Sửa tên của parameter trong keyword và lưu, đồng thời thực hiện xoá parameter trong metadata của script                                                                                                                                                                                                                                                                                                                                                            |
| Sử dụng variable trùng tên với parameter                         | <p>Với các action có thuộc tính returnType, variable của action đó không phải là parameter. Khi action bị xoá hoặc người dùng sửa tên của variable thì tên cũ của variable vẫn tồn tại trong list variable.</p>  <ul style="list-style-type: none"> <li>- Khi returnType có type là variable và người dùng click “Find variable”, click chọn variable trùng tên với parameter đã có.</li> <li>- ReturnType với type là newVariable: Nhập tên trùng với 1 variable đã tồn tại (có thể là parameter hoặc không)</li> </ul> <p><b>Lưu ý:</b> Nếu variable trùng tên với parameter nhưng không sử dụng thì không phát sinh lỗi</p> | <p>Danh sách variable được sắp xếp theo thứ tự từ trên xuống dưới như sau:</p> <ul style="list-style-type: none"> <li>- Variable (tạo ra khi returnType)</li> <li>- Parameter (input)</li> <li>- Output</li> </ul> <p>Trong variable (Input/Output) hệ thống đều sắp xếp theo thứ tự từ cũ nhất đến mới nhất. Dựa theo đó mà lựa chọn đúng variable cho keyword. Trường hợp không dùng đến parameter (input/output) thì thực hiện xoá trong metadata của test.</p> |

### 1.1.2 Sự cố khi Run

Thông báo lỗi “Node not exists” khi thực thi kiểm thử



*Node not exists*: Là lỗi không tìm thấy agent nào đã start có profile phù hợp với profile đã chọn

Nguyên nhân: Người dùng lựa chọn sai profile của client, cụ thể

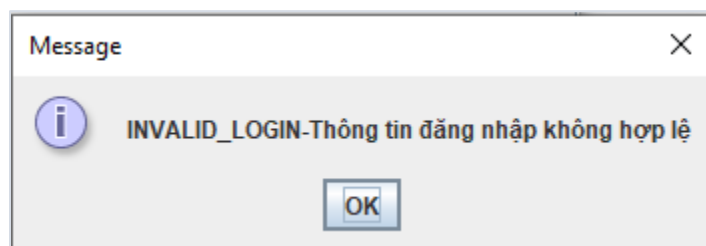
- Đối với Agent private: Chọn sai hệ điều hành thiết bị của user, trình duyệt không được chọn trong Agent của người dùng hoặc chọn sai phiên bản của trình duyệt đang sử dụng trên thiết bị.
- Đối với Agent public/ any Agent: Không tìm thấy Agent active trên hệ thống có profile giống client profile đã chọn.

### **1.1.3 Sự cố liên quan đến AGENT**

#### **1.1.3.1 Không hiển thị màn hình Test Framework Launcher**

- Nguyên nhân: Server liên kết với Agent chỉ cho phép truy cập khi sử dụng đúng mạng internet bảo mật của dự án
- Khắc phục: Kết nối đúng internet cho phép truy cập server

#### **1.1.3.2 Invalid Login**



INVALID\_LOGIN: là lỗi đăng nhập không hợp lệ

Nguyên nhân: Username chưa được thêm vào hệ thống hoặc sai mật khẩu

Cách khắc phục: Nhập đúng username và password của user đã tồn tại trên server

### **1.1.4 Các kỹ thuật kiểm thử bổ trợ**

#### **1.1.4.1 Sử dụng tham số đầu vào là random value**

Random Value là sử dụng dữ liệu ngẫu nhiên được tạo tự động khi thực thi kiểm thử. Để cấu hình tham số đầu vào là Random Value, hãy làm như sau:

**Bước 1:** Tại giao diện cấu hình keyword, tìm block **value**, chọn giá trị **Random value**

value

Value Variable Agent Parameter Device Parameter Version Parameter Environment Parameter

RandomValue username

Find value

**Bước 2:** Nhấn **Done** để lưu lại thông tin vừa cấu hình.

**Bước 3:** Nhấn biểu tượng **Setting** ở góc trên bên phải trên giao diện cấu hình

VCB Digibank Dashboard All Automation Test Manual Test Batch Testing Test Data Type

Keywords Hist

Platform All 222 >

All 1 >

Client 99 >

Action 10 >

Event 14 >

Control 42 >

Variable 29 >

Operator 5 >

Assert 15 >

File 4 >

Security 3 >

Other

Demol

Flow

Android(default\_client) Client

Clear And Send Keys

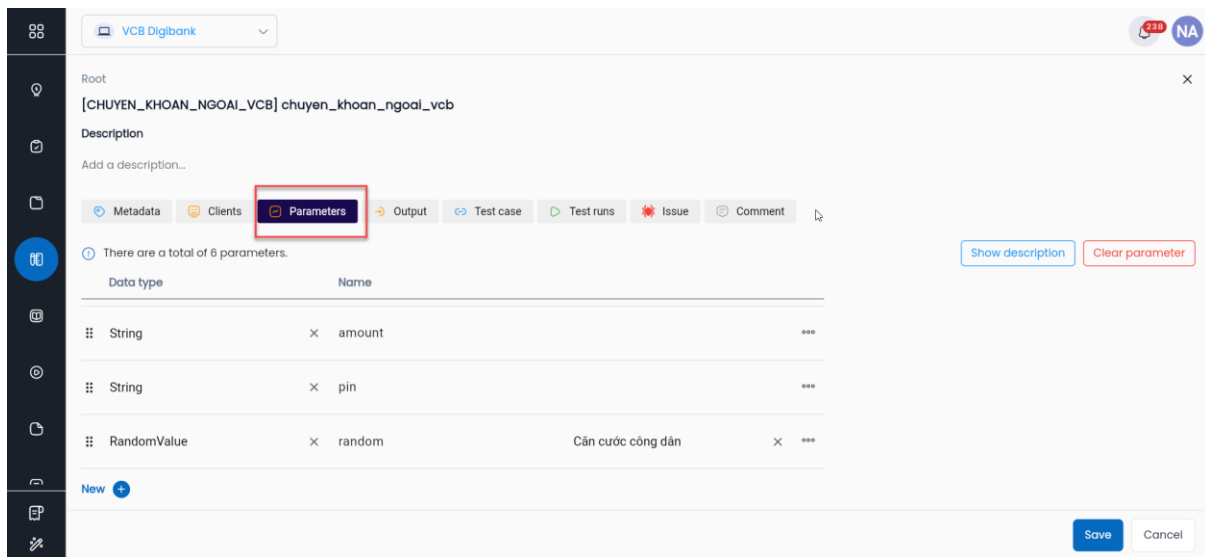
End Client

End Flow

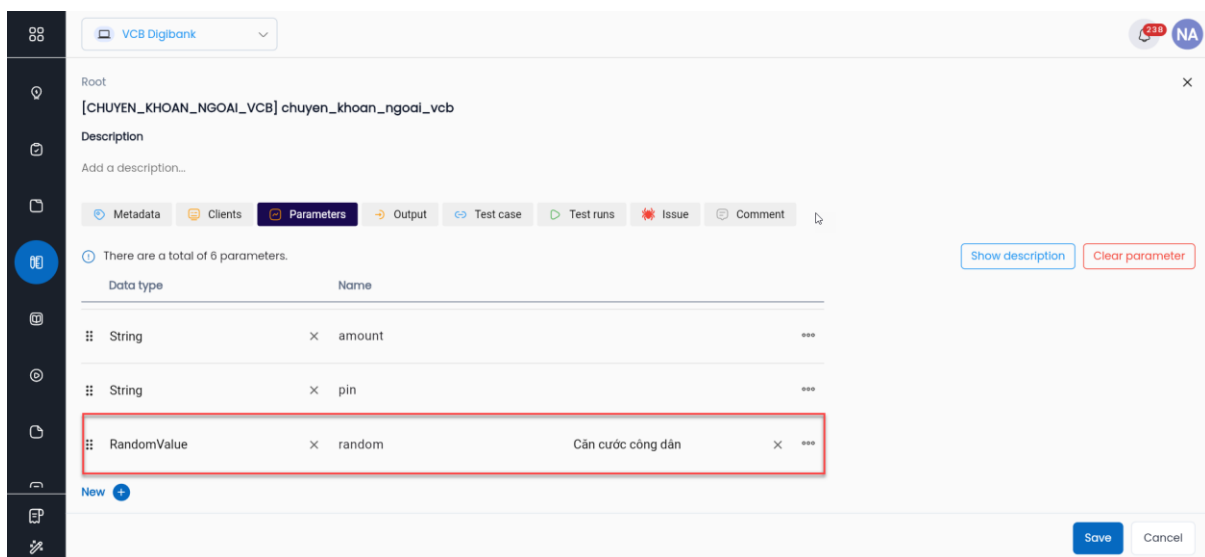
End Test Script

Back Delete Save Save as Start Debug Run

**Bước 4:** Chọn tab **Parameters**



## Bước 5: Chọn **Choose a random type** phù hợp với tham số của người dùng



**Lưu ý:** Random type là danh sách random type ở trạng thái active đã được cấu hình sẵn.

[Tìm hiểu thêm về Random Type](#)

### 1.1.4.2 Output

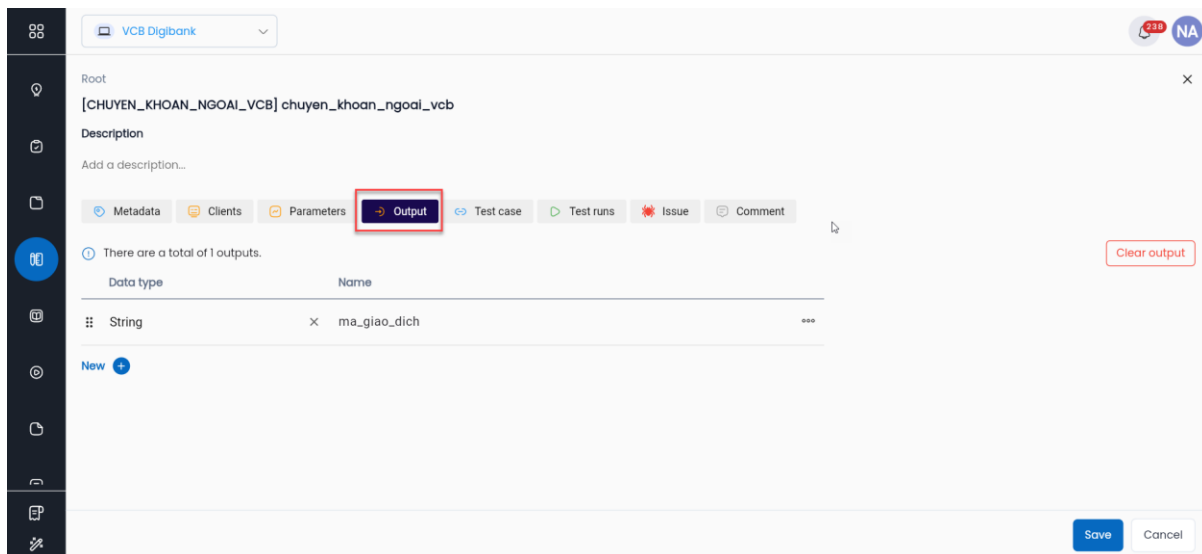
Output là kết quả đầu ra của một kịch bản kiểm thử, là giá trị trả về tương ứng với *actual result* (kết quả thực tế). Trường hợp sử dụng: Cấu hình lấy output khi người dùng muốn lấy thông tin hiển thị trên màn hình làm tham số đầu ra sau khi kết thúc thực thi kiểm thử.

Đặc biệt trong trường hợp cấu hình test suite: một luồng đi qua nhiều test script khác nhau và kết quả đầu ra của test script này sẽ là đầu vào của một test script khác.

Ví dụ: có 2 test script là *Đặt lệnh* và *Duyệt lệnh*. Sau khi *Đặt lệnh* xong sẽ có mã giao dịch, cấu hình lấy output là mã giao dịch, sau đó sử dụng mã giao dịch đó làm dữ liệu đầu vào cho test script *Duyệt lệnh*.

#### 1.1.4.2.1 Thêm mới một output

**Bước 1:** Trên tab Output của màn hình meta-data kịch bản kiểm thử, nhấn + **New**



**Bước 2:** Chọn Data type và đặt tên cho output, Nhấn **Save** để lưu.

#### 1.1.4.2.2 Output hệ thống tạo tự động

Trong quá trình cấu hình, đối với những Keyword có block Output, nếu thực hiện chọn Type Output là Variable, hệ thống kiểm tra biến này chưa tồn tại trong phần cấu hình của kịch bản thì hệ thống sẽ tự động tạo ra output mới.

**Bước 1:** Trên giao diện cấu hình, chọn 1 keyword có block thông tin Output

Ví dụ: Cấu hình Output cho keyword Gettext

**Bước 2:** Chọn thông tin Type là Variable, đặt tên cho biến với tên chưa tồn tại trên hệ thống

Get Text property
×

Element

Value

Variable

Agent Parameter

Device Parameter

Version Parameter

Environment Parameter

Repository Element

Element Repository

Output

Type

variable

×

Data type

String

×

Name

Enter a name

Find variable

Cancel

Done

Nhập các trường thông tin:

| Trường thông tin | Bắt buộc | Định dạng | Ràng buộc                    | Mô tả                                                                                                                                    |
|------------------|----------|-----------|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| Type             | Có       | Combobox  |                              | Thực hiện chọn Variable                                                                                                                  |
| Data Type        | Có       | Combobox  |                              | Hiển thị kiểu dữ liệu của 1 parameter. Người dùng được phép chọn 1 giá trị. Hệ thống khi khởi tạo parameter mới mặc định để là “String”. |
| Name             | Có       | Text      | Là duy nhất, không trùng với | Nhập tên cho output                                                                                                                      |

|  |  |  |                                                                                                              |                                                                                                                         |
|--|--|--|--------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
|  |  |  | bất kỳ parameter/variable nào đã được khởi tạo trong kịch bản<br>Chỉ bao gồm các ký tự chữ thường, số và “_” | Nên đặt tên theo mục đích sử dụng, đặc tính của element trả giá trị cho output... để thuận tiện trong quá trình sử dụng |
|--|--|--|--------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|

**Bước 3:** Nhấn **Done** để lưu

### 1.1.5 Sử dụng output

Vì giá trị của output là kết quả đầu ra được trả ra sau khi thực thi kịch bản kiểm thử nên giá trị của output không thể được nhập vào bởi người dùng.

Thông thường, giá trị của output sẽ được lấy từ một keyword có thuộc tính “Output” hoặc gán giá trị theo một *parameter* (tham số) của kịch bản kiểm thử theo một điều kiện nào đó xảy ra trong quá trình thực thi.

#### 1.1.5.1 Lấy giá trị cho output từ một keyword có thuộc tính “Output”

Sử dụng khi người dùng cần hiển thị kết quả của một hành động nào đó trong kịch bản kiểm thử

Các keyword có thuộc tính “Output” thích hợp sử dụng output làm variable: get Text, get attribute, get URL, Child Element get text, Child Element get attribute, get alert text, get window handle... Đây đều là các keyword lấy giá trị theo thuộc tính của phần tử, phổ biến là Get text, get alert text (lấy chuỗi text của phần tử).

**Bước 1:** Tạo biến output cho kịch bản kiểm thử theo hướng dẫn tại mục [Thêm mới một output](#).

**Bước 2:** Sau khi tạo output thành công, tại màn hình cấu hình kịch bản kiểm thử, chọn keyword có thuộc tính “Output”

**Bước 3:** Hệ thống hiển thị màn hình keyword property, chọn type của thuộc tính “Output” là variable và nhập tên cho variable là tên của một output đã được tạo trước đó của kịch bản kiểm thử.

Get Text property

×

Element

ValueVariableAgent ParameterDevice ParameterVersion ParameterEnvironment Parameter

Repository Element

Element Repository

Output

Typevariable×

DatatypeString×

username\_output

Find variable

Cancel

Done

DataType là kiểu dữ liệu của output, thường là String.

Để hạn chế phát sinh sai sót trong quá trình nhập liệu, người dùng nên sử dụng **Find Variable** để lựa chọn output

## Sử dụng Variable

Get Text property

×

Element

ValueVariableAgent ParameterDevice ParameterVersion ParameterEnvironment Parameter

Repository Element

Element Repository

Output

Typevariable×

DatatypeString×

username\_output

Find variable

Cancel

Done

Nhấn “Find Variable”, hệ thống hiển thị danh sách variable của kịch bản kiểm thử, bao gồm biến, tham số đầu vào và output.

Danh sách variable được sắp xếp từ trên xuống theo trình tự:

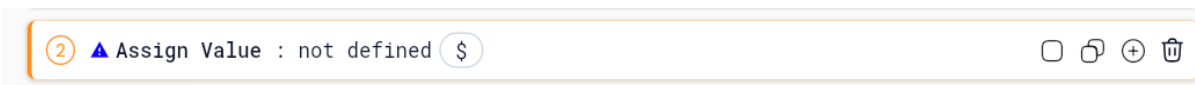
- Variable (output newVariable), parameter, output.
- Tham số tạo trước ở trên

Nhấn chọn output mong muốn, đảm bảo output đã được nhập các thông tin một cách chính xác và hợp lệ, nhấn **Done** để lưu.

#### 1.1.5.2 Gán giá trị cho output theo điều kiện xảy ra trong kịch bản kiểm thử

Tính năng này thường sử dụng cho các trường hợp kết quả đầu ra bị thay đổi tùy thuộc vào điều kiện cụ thể trong kịch bản kiểm thử.

Để gán giá trị cho output, có thể sử dụng keyword *Assign Value*



Thao tác gán giá trị như sau:

Nhấn chọn vào bước cấu hình cần gán

Trên màn hình Property Value của keyword, nhập đầy đủ thông tin cho keyword, trong đó name là tên của output đã khởi tạo

Assign Value property

Step comment

Gán output

Data type

String

Name

username\_output

Find variable

Nhấn **Done** để hoàn tất gán giá trị cho output

Kết quả cấu hình đúng cho việc gán giá trị “username” cho output “username\_output”:



5 Assign Value : username\_output \$  
Gán output

### 1.1.5.3 Xem giá trị của output khi thực thi kịch bản kiểm thử

Kết quả output được trả ra sẽ hiển thị tại màn hình Test Run data.

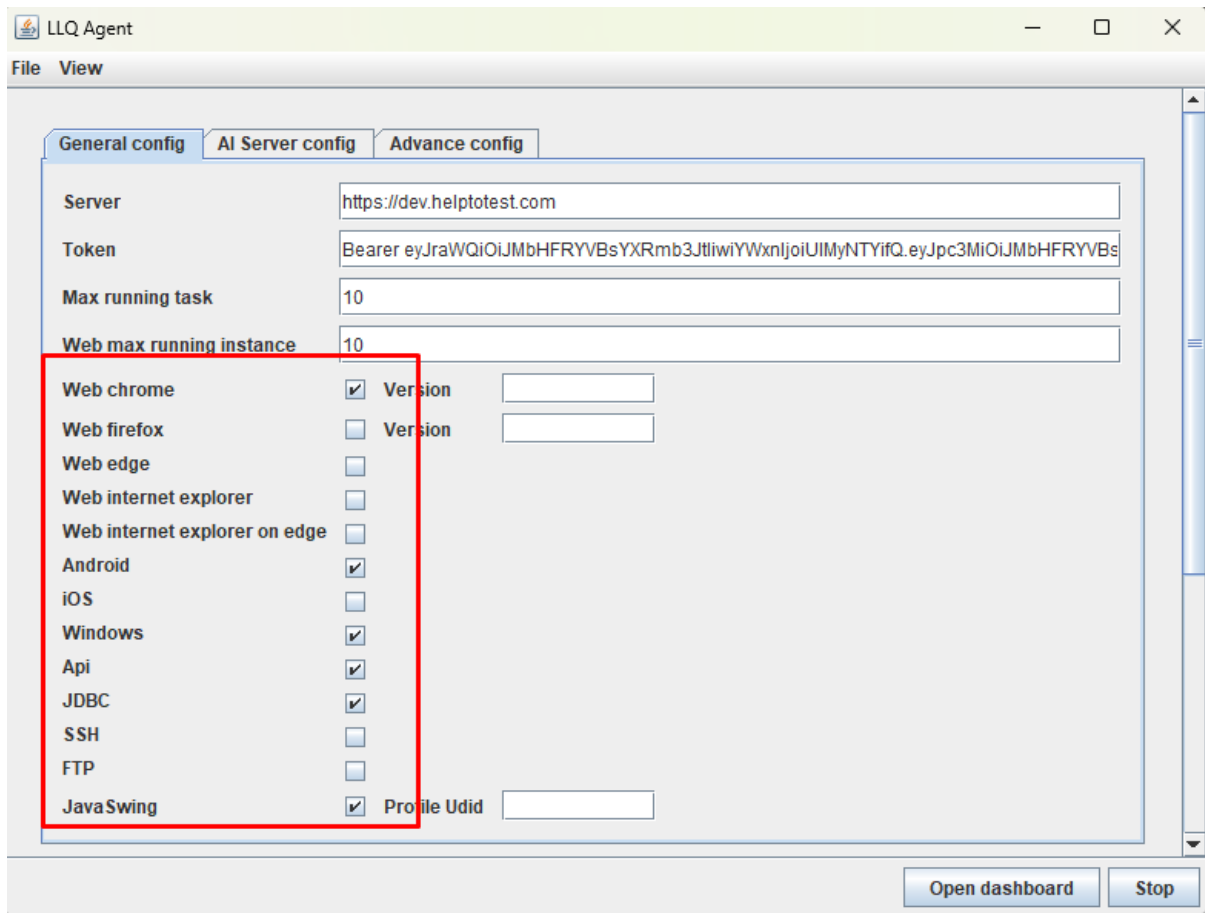
The screenshot displays the 'Test Run data' interface in the LLQ QA Platform. The top navigation bar includes 'Dashboard', 'Automation Test', 'Manual Test', 'Batch Testing', 'Test Plan', and 'Issue'. The main content area is titled 'Data 1 of [T-44] IOS-Chuyển tiền trong nước demo'. It features three tabs: 'Information', 'Output File', and 'Test Run Data'. The 'Test Run Data' tab is active, showing test run details and results. The 'Test Run Result' section indicates 'Expected result: Pass', 'Actual result: Pass', and 'Final status: Pass'. The 'Test Run Data Information' section lists details such as Agent, Start run, End run, Run time, Version, and Environment. The 'Test Run Data' section shows Task group name, Test case code, Test Run Expected Result, Severity, and Priority. A red box highlights the 'Test run output' section, which contains the following data:

| Test run output                                         |
|---------------------------------------------------------|
| ten_nguoi_nhan: NGUY V VI                               |
| so_tien_chuyen_ck_long: 10000                           |
| hinh_thuc_chuyen_xn: (Mười nghìn đồng)                  |
| tai_khoan_nguon_xn: Tài khoản nguồn                     |
| tai_khoan_nhan_xn: Tài khoản nhận                       |
| ten_nguoi_nhan_xn: Tên người nhận                       |
| ngan_hang_nhan_xn: Ngân hàng TMCP Ngoại Thương Việt Nam |

### 1.1.6 Kiểm thử đa môi trường

LLQ QA Platform hỗ trợ người dùng kiểm thử đa môi trường và đa nền tảng trong một kịch bản kiểm thử duy nhất. Để cấu hình hoặc thực thi một kịch bản kiểm thử đa môi

trường, các nền tảng môi trường phải được thiết lập trong Agent trước khi [Start Agent](#).



+**Keyword Client** xác định các bước cấu hình bên trong sẽ được thực thi trên môi trường hoặc nền tảng nào. Vì vậy, để kiểm thử bao nhiêu môi trường, bao nhiêu nền tảng người dùng sẽ phải tạo từng ấy client, mỗi client tương ứng với một môi trường hoặc nền tảng.

Để cấu hình đa môi trường đa nền tảng trên một kịch bản kiểm thử, thực hiện như sau:

**Bước 1:** Kéo thả keyword client từ danh sách keyword vào vị trí phù hợp.

**Bước 2:** Chọn platform đã khởi tạo và đặt tên cho client.

**Client property** ×

Client

Web-default\_client ×   

Step comment

Enter a description

Cancel

Done

**Bước 3:** Hệ thống hiển thị popup thêm mới như sau:

**Add client** ×

Platform

Choose a platform ▼

Name

Save

Cancel

Nhập các trường thông tin cho client:

- Platform: Chọn một nền tảng sẽ sử dụng trong danh sách (Web, Android, IOS, ...)
- Name: Nhập tên định danh cho client. Yêu cầu: Tên client là duy nhất trong kịch bản kiểm thử và phải bắt đầu bằng chữ cái thường, chỉ được chứa chữ cái thường, số và “\_”

**Bước 4:** Sau khi nhập đầy đủ thông tin, nhấn **Save** để tạo mới client.

**Bước 5:** Nhấn **Done** để lưu lại thông tin cấu hình.

**Client property** ×

Client

Android-clientname ×   +

Step comment

Enter a description

Cancel Done

**Bước 6:** Tạo xong client, thực hiện thiết kế các bước cấu hình phù hợp với từng loại client theo kịch bản kiểm thử.

**Lưu ý:** Kịch bản kiểm thử có bao nhiêu client thì khi thực thi sẽ phải chọn bấy nhiêu loại môi trường phù hợp với từng client.

Ví dụ: Hình ảnh ở dưới là ví dụ cho một trường hợp trong kịch bản kiểm thử có 2 client là Web và Android nên khi thực thi sẽ phải chọn profile cho cả 2 client

The screenshot shows the testapi interface with the following configuration:

- testapi** (header)
- Navigation:** Dashboard, All, **Automation Test**, Manual Test, Test Data Type
- Actions:** Old data, Choose file, Upload Excel, Download Excel
- Select profile and device:**
  - Client default\_client: API
  - Client default\_client2: android\_hanh
  - Device: Any
- Agent type:** Any Agent
- Test Run Type:** Parallel
- Record only on failure:** False