### U Test User Story 1:

Priority: Major

As a Product Owner

I want that while using the KiwiSaver Retirement Calculator all fields in the calculator have got the information icon present

So that

The user is able to get a clear understanding of what needs to be entered in the field .

**Acceptance Criteria**

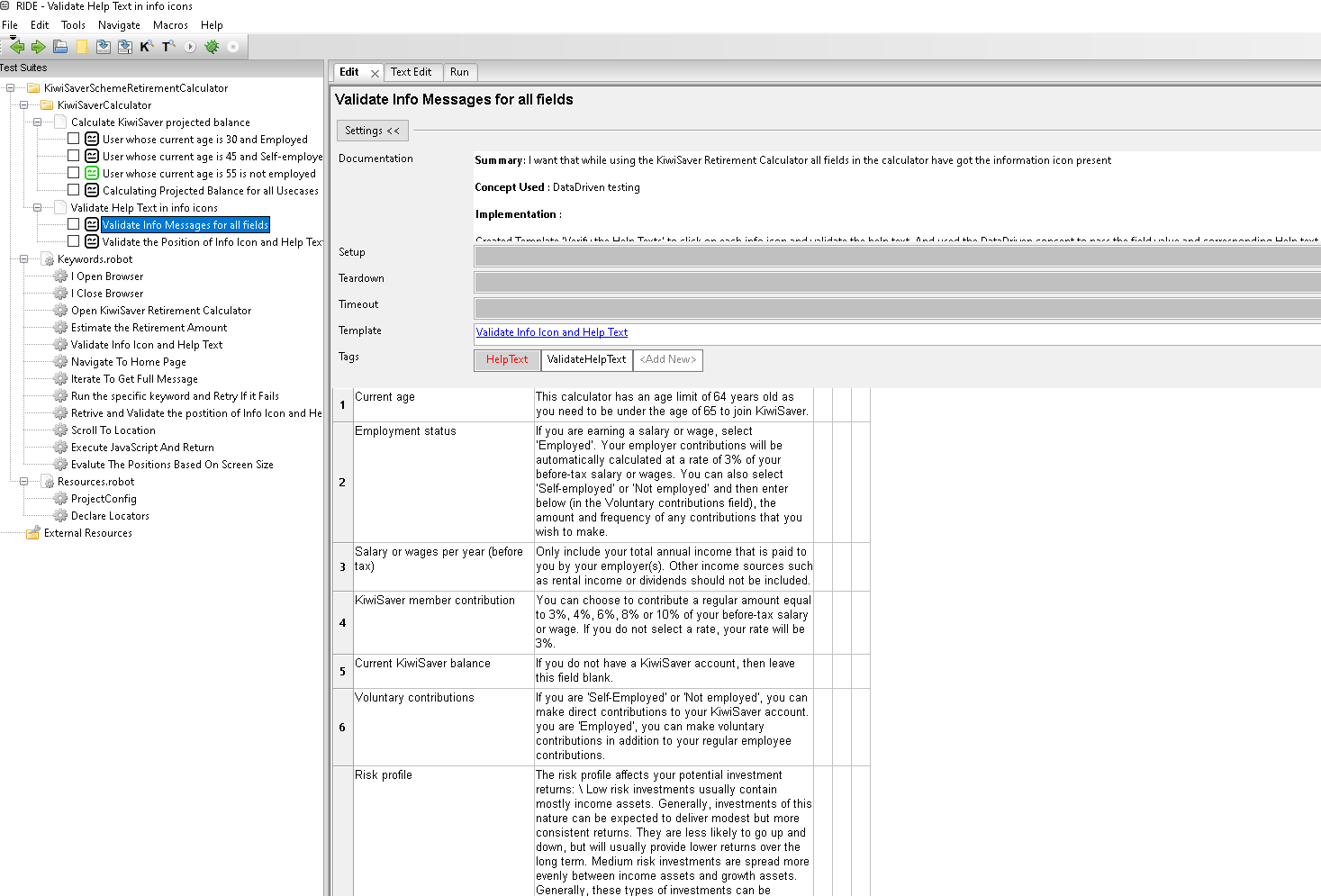
Scenario 1

Given User Clicks information icon besides Current age the message “This calculator has an age limit of 64 years old as you need to be under the age of 65 to join KiwiSaver.” is displayed below the current age field.

**Automation Test Cases**

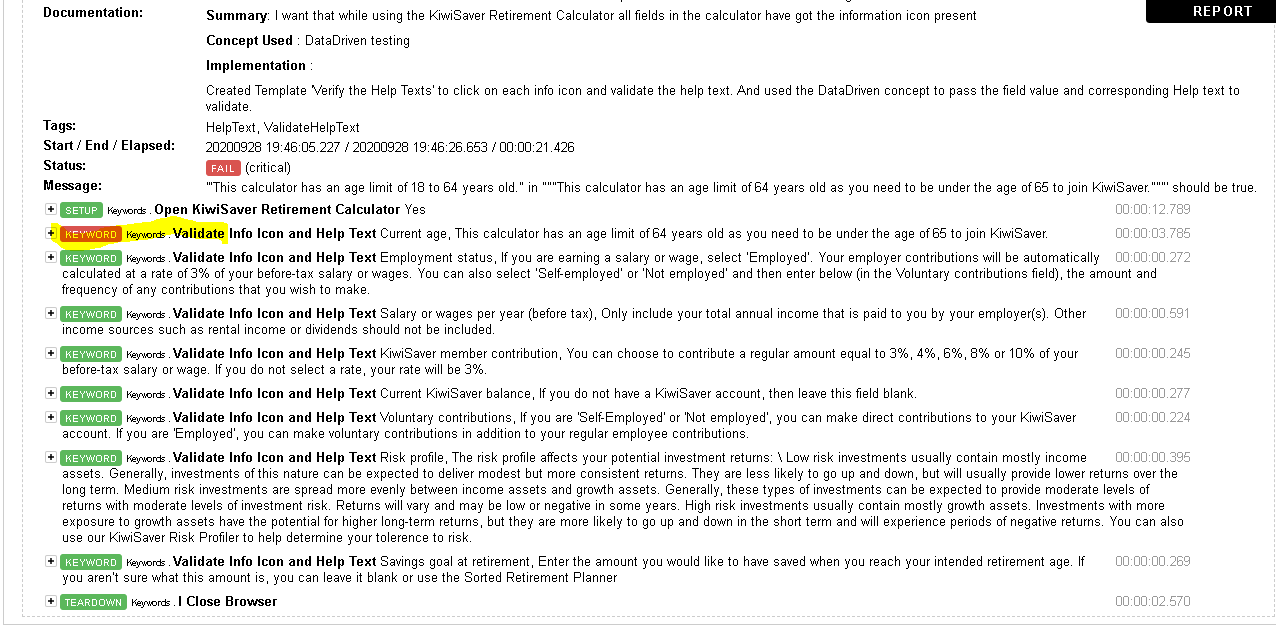
I split the user story into two different test cases.

**TestCase 1** – To Click on Info Icon and validate the Help Text

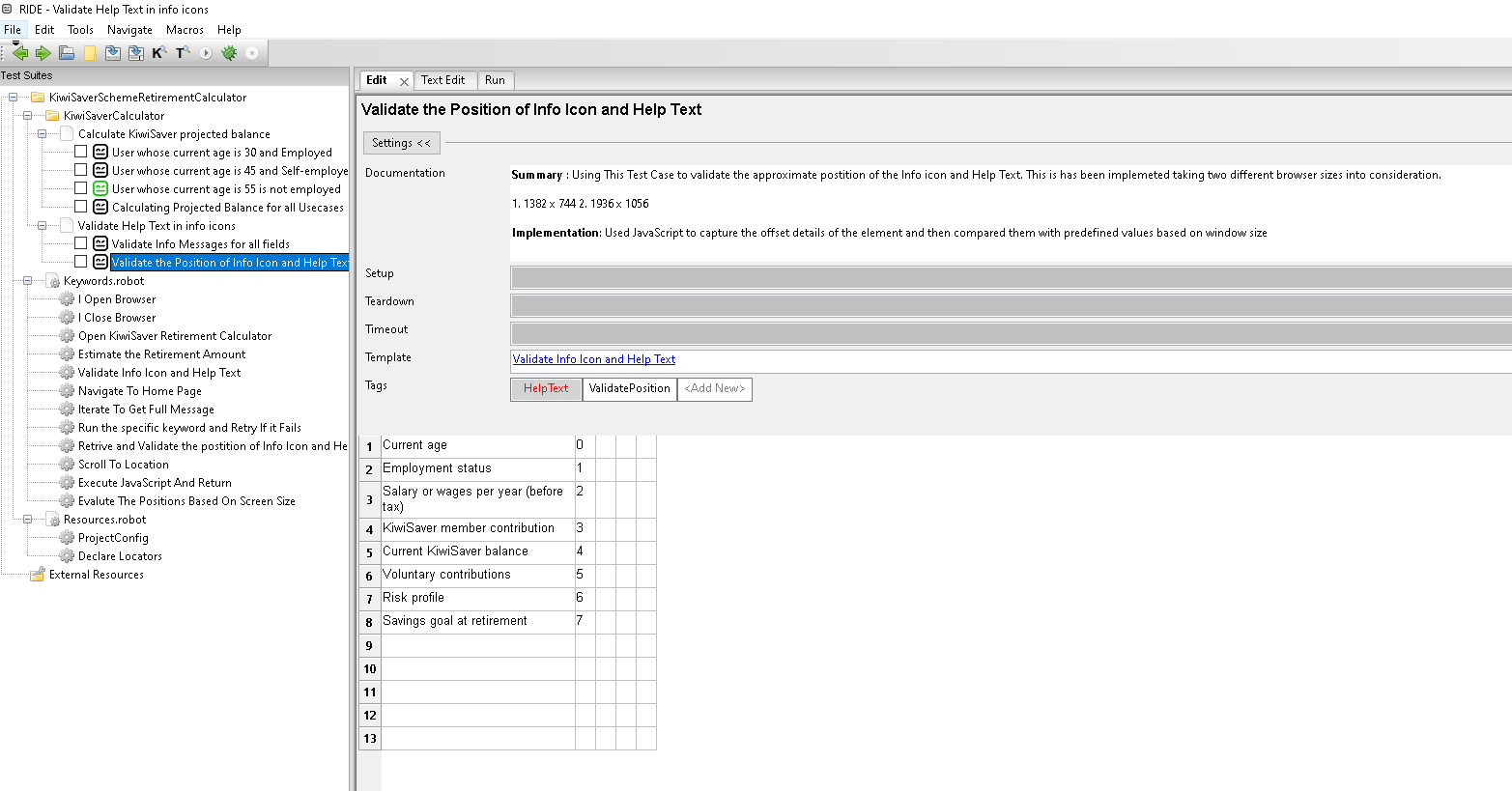


Framework design:

Used the DataDriven concept to pass the label and expected help text to the Template ‘Validate Info Icon and Help Text’. This template contains the logic to click on the icon next to each label, capture the Help text and validate it with the expected help text. The benefit of this approach is each set of arguments are treated as separate test case. And if one of the help texts fails on validation, the execution will not stop there, it will continue until all the tests are executed. Logs will be displayed as below. As you see in the logs Help text validation passed for all the labels, except for current age, as the help text for Current age is not as per the acceptance criteria

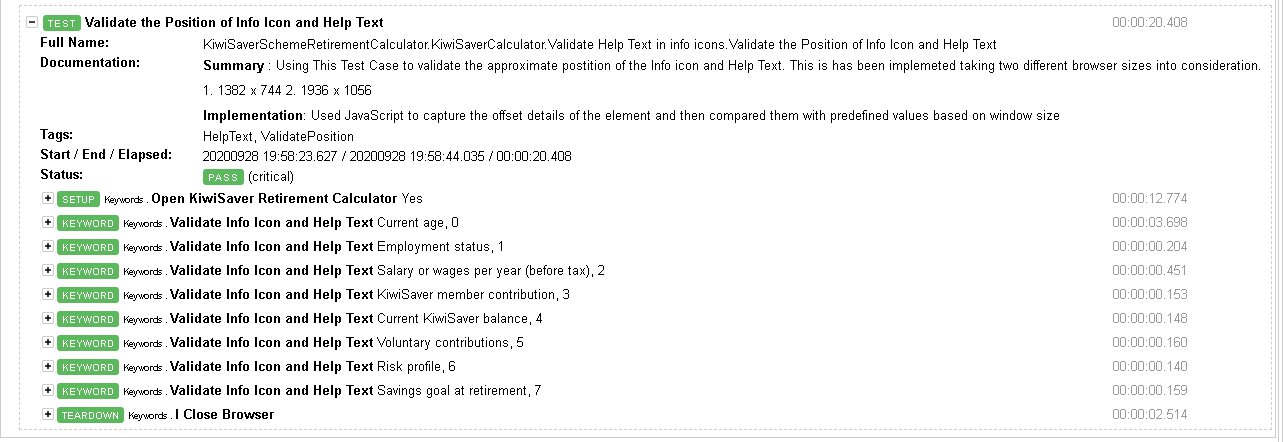


**TestCase2 -** To validate the position of Info Icon and Help Text (As there are no coordinates defined, I used the existing UI as a base)



Framework design:

Used the same DataDriven logic. Passed the arguments label and index value to the Template ‘Validate Info Icon and Help Text’. In the template I am using JavaScript to capture the position of each Icon and Help Text. Captured values are then compared with the predefined (Used current UI as base line) values. As the position if the icon changes based on browser size, I considered two different screen sizes (1382 x 744, 1936 x 1056) and implemented it.



### Test User Story 2:

Priority: Major

As a Product Owner

I want that the KiwiSaver Retirement Calculator users are able to calculate what their KiwiSaver projected balance would be at retirement

So that

The users are able to plan their investments better.

**Acceptance Criteria**

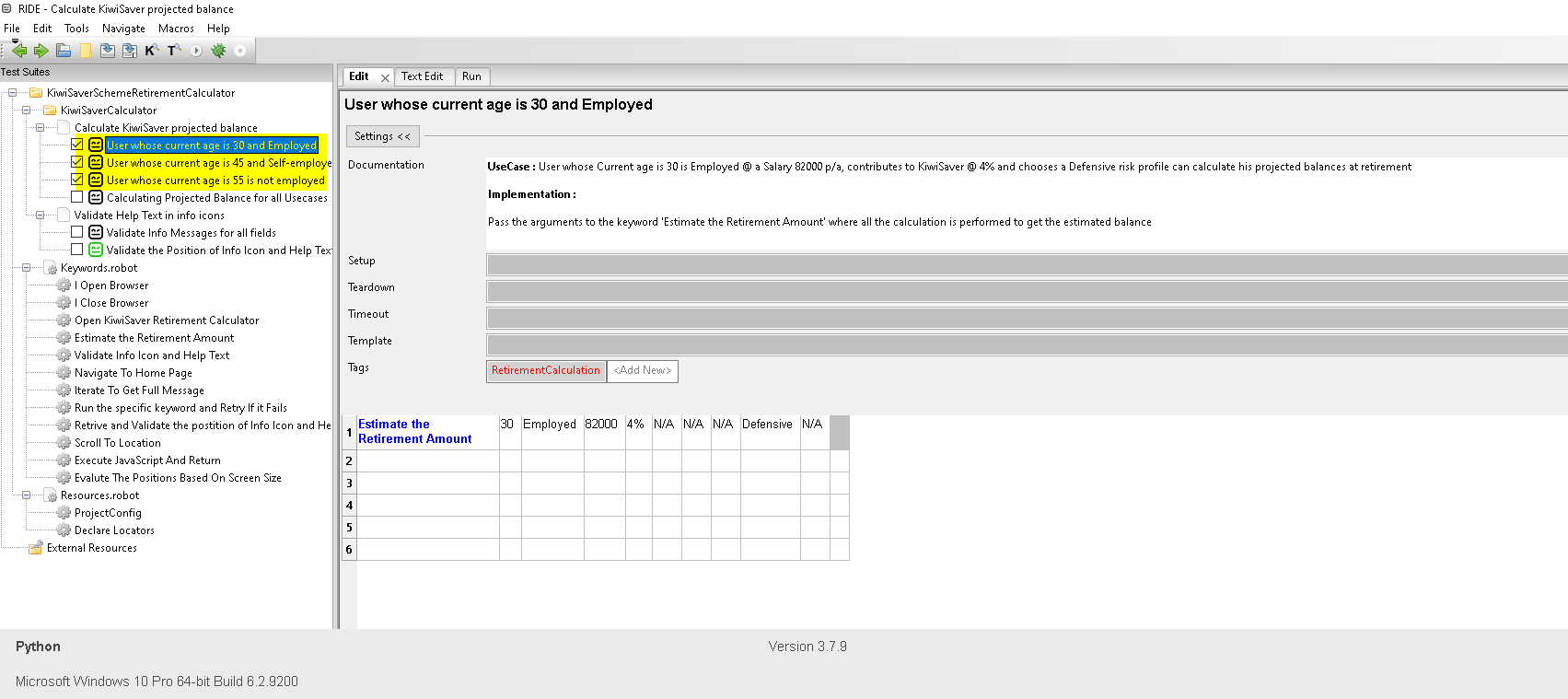
* User whose Current age is 30 is Employed @ a Salary 82000 p/a, contributes to KiwiSaver @ 4% and chooses a Defensive risk profile can calculate his projected balances at retirement.
* User whose current aged 45 is Self-employed, current KiwiSaver balance is $100000, voluntary contributes $90 fortnightly and chooses Conservative risk profile with saving goals requirement of $290000 can calculate his projected balances at retirement.
* User whose current aged 55 is not employed, current KiwiSaver balance is $140000, voluntary contributes $10 annually and chooses Balanced risk profile with saving goals requirement of $200000 is able to calculate his projected balances at retirement.

**Automation Test Cases**

Implemented two design for this use story. Both are effective and easy to maintain

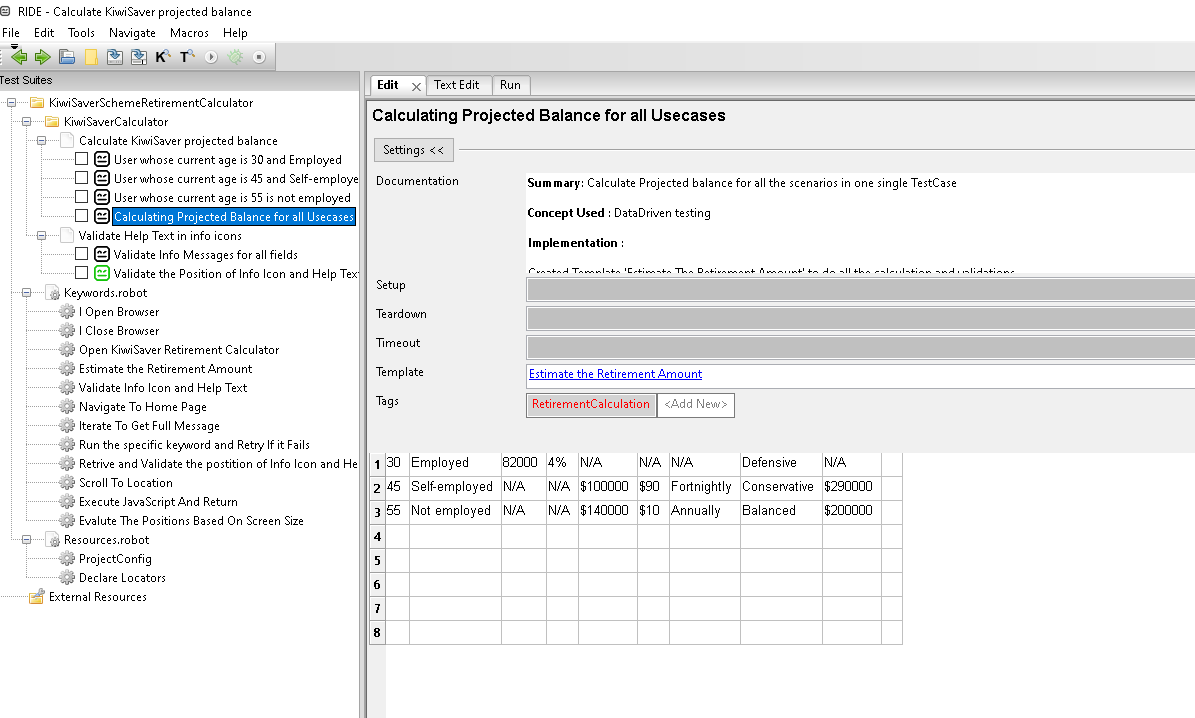
Design1

Automated the above three scenarios in three test cases. All the three test cases pass the acceptance criteria as arguments to the keyword ‘Estimate the Retirement Amount’ and the enters the details on the form and validates that Retirement balance is being generated.



**Design 2**

Used the DataDriven logic like UserStory1, to run all the three scenarios from one single location.

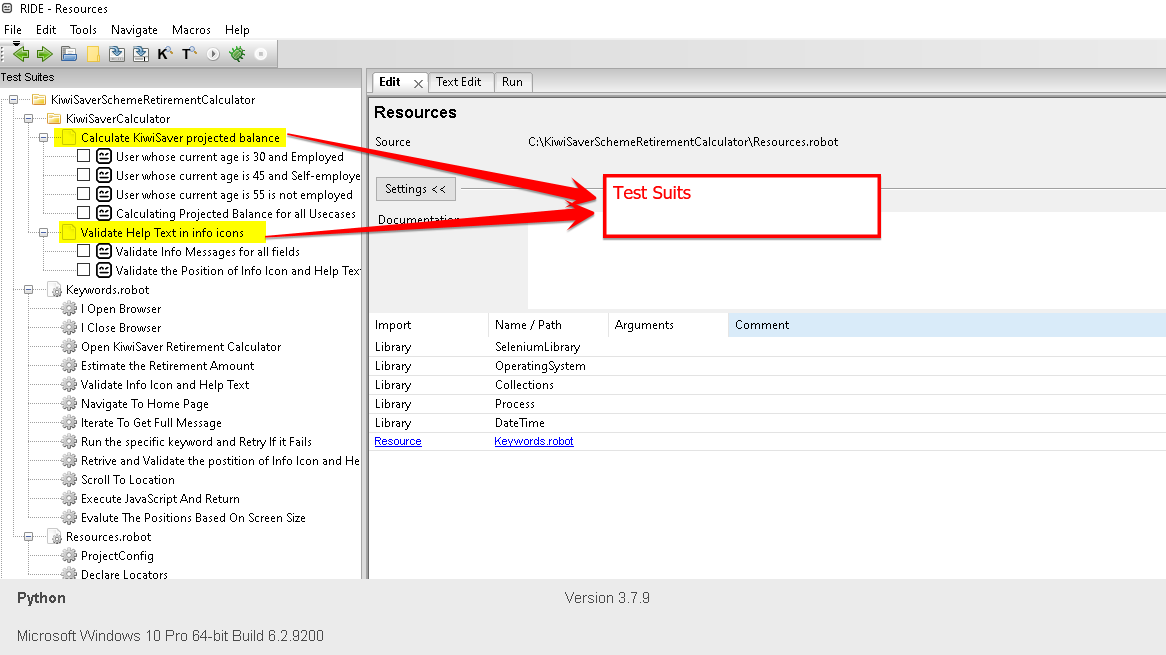


**Project Structure**

Created two test suits for the given stories

Test Suit - Validate\_Help Text\_in\_info\_icons (User Story1)

Test Suit - Calculate\_KiwiSaver\_projected\_balance (User Story2)



Validate\_Help Text\_in\_info\_icons (User Story1), contains two test cases

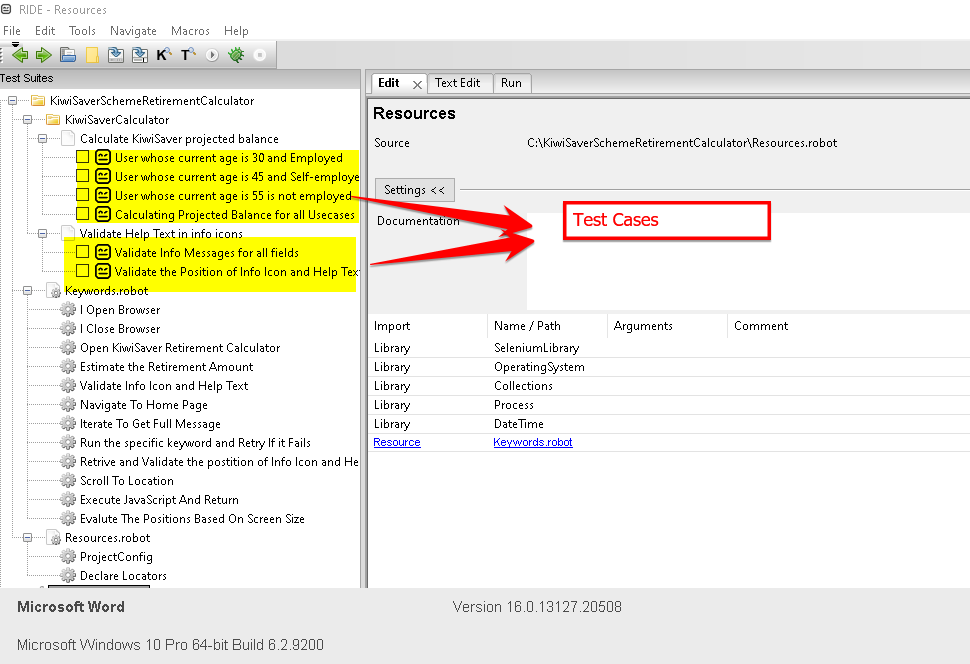
Calculate\_KiwiSaver\_projected\_balance (User Story2), contains 4 test cases

Desing1

1. User whose current age is 30 and Employed – For Acceptance criteria 1
2. User whose current age is 45 and Self-employed – For Acceptance criteria 1
3. User whose current age is 55 is not employed – For Acceptance criteria 1

Design2

1. Calculating Projected Balance for all Usecases – Run all three scenarios using DataDriven concept



**Keywords.robot**

file contains all the pageobjects or re-usable keywords. These keywords carry the logic behind the execution. Open RIDE and click on any keyword and read the documentation for details

**Resources.robot**

Declare Locators – All the xpaths, global variables, Text are declared here.

ProjectConfig - When we run any test case, this is the first file that gets executed, so it contains URL,BROWSER details and also keyword Declare Locators is triggered from here.

