## Test approach/ design

A heuristic is a rule-of-thumb, or a guide toward what behavior is appropriate for a certain situation. Heuristics are also known as “mental shortcuts”. For testing, we have some heuristic to apply when they are designing, creating tests, and performing. Then I choose SFDIPOT (San Francisco Depot) – Test Strategy Heuristics by [James Bach](http://www.satisfice.com/blog/) for my testing approach and design.

The scope of this heuristic is applied to the product, however I could use some keywords from this heuristic to test Registration at Step 1 on uTest page

##### 

* Function: Registration at Step 1
  + Success
  + UnSuccess
* Data: since the input method is entering to the text box, then we have 3 types of test data
  + Text (e.g: latin, non latin)
  + Number
  + Special character
  + Combine them all
  + Specially, Email textbox contains special character then we will have correct format and incorrect format
* Interface: we are able to view on GUI
* Flatform: This function is built in web base, we have some tools to access, so I will do the test on these browsers
  + Chrome
  + Firefox
  + IE
  + Edge
  + Safari
* Time (Performance test - manually): measure by Development tool on browser. Normally, the average of loading time is less than 5 seconds.
  + Landing on the home page
  + After hit to Next Location button after complete form at Step 1

## Automation

* Test case design for RegisterOnUTestPage.feature
* Location: TrialSeleniumFW/src/test/resources/RegisterOnUTestPage.feature

### Prerequisites

* Java version: 1.8
* Gradle version: 6.8.3
* Platform: Windows 10, Mac OS X
* Support browser:
  + Chrome: version 88
  + Firefox: version 86