**Selenium Project SDLC**

**Requirements**:

Functional:

* System must load website
* System must locate and select a specific product
* System must select add category
* System must select a category to add

Non-Functional:

* Action must be done in reasonable time

**Design**:

* Sequence Diagram :
  + Run program
  + Web driver is set up through file path and the loads set URL.
  + Load target URL
  + Locate product
  + Click product and load web page for product details
  + Find Add Category
  + Click on Add Category and proceed to add category page
  + Find Category
  + Click Add to add Category to product, proceeds to confirm
  + Find Assign Button
  + Click Assign button to confirm category to add to product.
  + Quit Driver

**Implementation**:

* Code written in python utilizing selenium.
* Code written to automate inventory management system testing
* Code loads web page, finds product, finds add category, find category of index position specified, add category to product.
* Wrote notes for code
* **GITHUB REPO**: <https://github.com/rockcm/SeleniumTesting>

**Testing**:

* Test Cases:
  + Correct Page URL Load
    - Steps – Set page\_url, run script and check if browser opens correctly.
      * Pass – Loads correctly as long as URL is correct, URL hardcoded. <https://localhost:7095/> browser opens
  + Correct product was clicked with ID of 1
    - Steps – Code driver to find element that matches href="Product/Details/1". Run program.
    - Happy Path : Product with href="Product/Details/1" (Product 1) is clicked. Navigates to details page for product 1
    - Unhappy Path : Product with href="Product/Details/1" Product 10 is selected and moves to the details page.
      * Can add additional matching criteria to remedy in next update/release.
  + Click on Add Category link
    - Steps – set code up to find element that matches Add Category link text. Run Program. Verify Add Category Page is accessed for correct product.
    - Pass Loads Add Category page correctly.
  + Click Add button for Add Category
    - Steps – set up code to find elements with the link text “Add” and select the element at index position 2.
    - Pass – Add button for Category at index position 2 is clicked, proceeds to confirm screen.
  + Click on Assign Button – found solution on stack overflow
    - Steps - set code to wait for element to be clickable, and then find and click element.
    - Happy Path : If category with index of 2 is not assigned yet, it will be assigned to the product.
    - Unhappy Path : If category with index of 2 is already assigned, program will close and give error message about it being our of range
      * Will add code to handle this error more gracefully in next iteration/release
  + Code Ran in reasonable time
    - Steps – ensure that previous tests work even if unhappy path is taken.
    - Pass – Code executes extremely fast (surprised me how effective selenium is)

**Release**:

* Uploaded the final code to GitHub for public access and review.
* Will maintain versions/branches of code that serve a specific function, such as selecting products, altering products, removing products etc. These will come via updates/maintenance which would complete the SDLC.
* Added Read Me for a brief explanation of repo contents.

**Maintenance and Updates to follow in weeks to come**

**GITHUB REPO**: <https://github.com/rockcm/SeleniumTesting>