

QA - Structure Projet avec RobotFramework

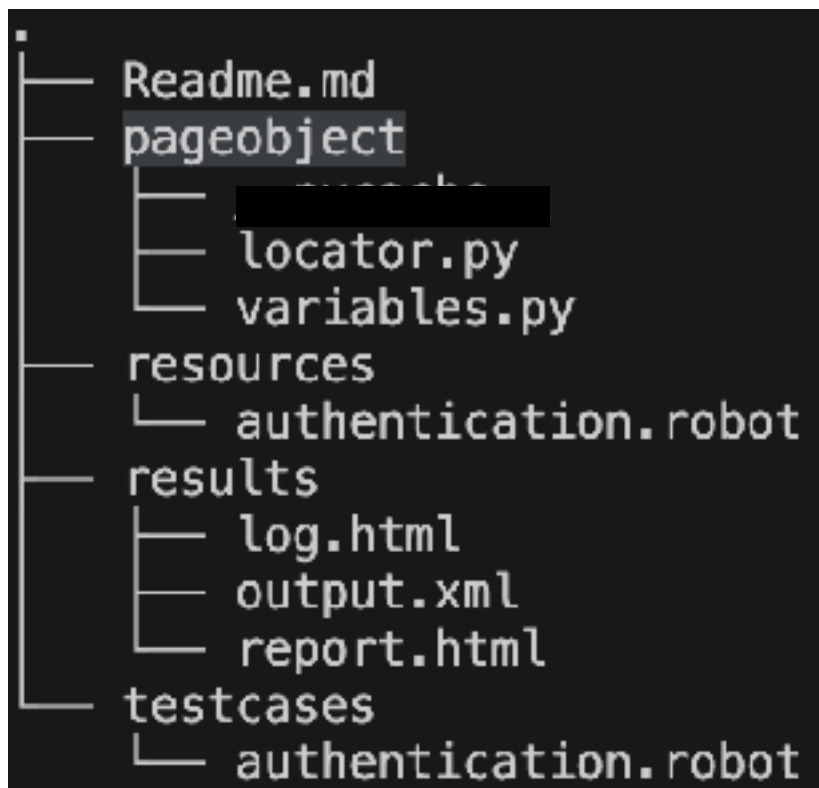
Objectif

Automatiser des scénarios de test Web avec Robot Framework et SeleniumLibrary

- ➔ Mettre en place des bonnes pratiques de structuration d'un projet avec RF
- ➔ Écrire des tests automatisés lisibles
- ➔ Réutiliser les variables et les mots-clés
- ➔ Générer un rapport de test
- ➔ Assurer la maintenabilité

Création de la Structure du projet

Rq: ne pas créer le dossier « **__pycache__** », il est géré automatique.



1. Création du fichier « **authentication.robot** » dans le dossier « **testscases** » pour déclarer les cas de Tests
2. Création du fichier « **authentication.robot** » dans le dossier « **resources** » pour déclarer les « keywords »
3. Création du fichier « **locator.py** » dans le dossier « **pageobject** » pour déclarer les référence aux objets.

```
id_textbox_username= "id=j_username"
```

4. Création du fichier « **variables.py** » dans le dossier « **pageobject** » pour déclarer les variables: **URL** et **BROWSER**

```
URL= "http://18.201.83.167:8080"
```

5. Importer les deux fichiers « **locator.py** » et « **variables.py** » comme variable dans le fichier « **resources/authentication.robot** » .

```
Variables ../pageobject/locator.py
```

6. Importer le fichier keywords « **resources/authentication.robot** » dans le fichier des cas de test « **testcases/authentication.robot** »:

```
Resource ../resources/authentication.robot
```

7. Mettre à jour les variables dans le fichier des keywords « **resources/authentication.robot** » en utilisant les variables déclarées aux étapes (4) et (5) .

```
Input Text    ${id_textbox_username}    admin
```

Exécution des test

```
robot -d results ./testcases/authentication.robot
```

8. Gestion des TAGS

Ajouter les TAG suivants au niveau des cas de tests:

- ***Cas1-Login correcte et mot de passe incorrect***

```
[Tags]    smoke    auth
```

- ***Cas2- Login incorrecte et mot de passe incorrecte***

```
[Tags]    sanity    auth
```

9. (BONUS) Utilisation des TAGS

1. Execution des cas de tests contenant le TAG « sanity »
2. Execution des cas de tests contenant le TAG « auth »

```
robot -i auth -d results ./testcases/authentication.robot
```