

Rapport de l'exécution des tests

Table des matières

1- Objectif du document :.....	2
2- Aperçu de l'application :.....	2
3- Portée du test :.....	2
a- Portée :.....	2
b- Hors de portée :.....	3
4- Métriques :.....	3
5- Types de tests effectués :.....	3
a- Test unitaire :.....	3
b- Test fonctionnel :.....	3
c- Test d'intégration :.....	3
6- Environnement de tests et outils :.....	4
7- Meilleures pratiques :.....	4
8- Critères de sortie :.....	4
9- Conclusion :.....	4
10- Définitions, acronymes et abréviations.....	5



PostgreSQL



heroku



Selenium
WebDriver

1- Objectif du document :

Ce document explique les différentes activités réalisées dans le cadre des tests de l'application «pureBeurre»

2- Aperçu de l'application :

PureBeurre est une application web, basé sur le framework python : [django](#).

L'objectif est de permettre à l'utilisateur de rechercher un substitut alimentaire. Il existe plusieurs modules tels que la recherche, la création du compte utilisateur, l'authentification, l'enregistrement des substituts afin de remplir l'objectif.

3- Portée du test :

a- Portée :

Les tests fonctionnels pour les modules suivants sont dans la portée des tests

- [homepage](#)
- [spaceUser](#)
- [products](#)

Il y a également un [test de performance](#).

Coverage report: 94%

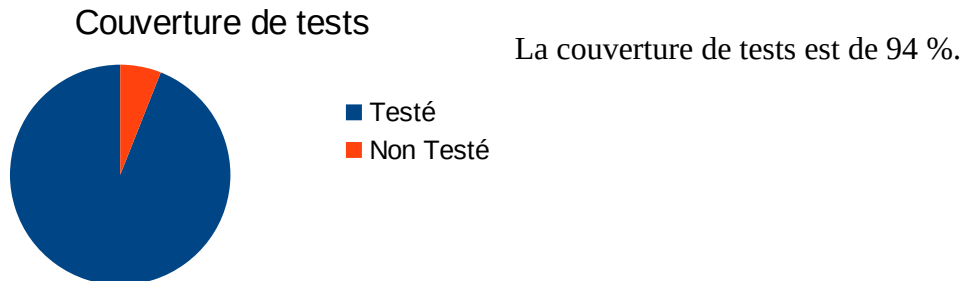
Module ↑	statements	missing	excluded	coverage
homepage__init__.py	0	0	0	100%
homepage\admin.py	1	0	0	100%
homepage\apps.py	4	0	0	100%
homepage\forms.py	3	0	0	100%
homepage\models.py	1	0	0	100%
homepage\urls.py	3	0	0	100%
homepage\views.py	18	4	0	78%
local_setting.py	4	4	0	0%
products__init__.py	0	0	0	100%
products\admin.py	14	0	0	100%
products\algoSubstitution.py	40	0	0	100%
products\management__init__.py	0	0	0	100%
products\migrations\0001_initial.py	5	0	0	100%
products\migrations\0002_auto_20210929_1719.py	5	0	0	100%
products\migrations\0003_favorites.py	6	0	0	100%
products\migrations\0004_auto_20211014_1124.py	4	0	0	100%
products\migrations\0005_auto_20211014_1126.py	4	0	0	100%
products\migrations\0006_alter_favorites_unique_together.py	5	0	0	100%
products\migrations__init__.py	0	0	0	100%
products\models.py	23	0	0	100%
products\urls.py	3	0	0	100%
products\views.py	55	4	0	93%
spaceUser__init__.py	0	0	0	100%
spaceUser\admin.py	4	0	0	100%
spaceUser\apps.py	4	0	0	100%
spaceUser\form.py	17	0	0	100%
spaceUser\migrations\0001_initial.py	8	0	0	100%
spaceUser\migrations__init__.py	0	0	0	100%
spaceUser\models.py	5	0	0	100%
spaceUser\urls.py	3	0	0	100%
spaceUser\views.py	41	7	0	83%
tests_functional__init__.py	0	0	0	100%
tests_functional\test_identification.py	24	0	0	100%
Total	304	19	0	94%

coverage.py v6.0.2, created at 2021-11-01 10:15 +0100

b- Hors de portée :

Les commandes django pour remplir la base de données n'ont pas été testées.

4- Métriques :



5- Types de tests effectués :

a- Test unitaire :

L'application a été créée sous le modèle [MVT](#) (Modèle Views Templates) avec le framework [Django](#), qui reprend le patron de conception [MVC](#) (Modèle Vue Contrôleur).

Chaque partie [MVT](#) de chaque modules (spaceUser, products, homepage) ont été testé de façon unitaires.






b- Test fonctionnel :

La [création du compte utilisateur](#) a été testé avec [selenium](#).

c- Test d'intégration :

Il s'agit du [test](#) effectué sur l'application testée, pour vérifier que l'utilisateur qui s'enregistre peut également s'identifier sur le site, conformément aux exigences du cahier des charges.

6- Environnement de tests et outils :

Framework développement		Django 3.2.7
Framework de tests		Pytest 6.2.5
Module de couverture de tests		Pytest-cov 3.0.0
Module d'intégration		Selenium 3.141.0
langage		Python 3.9
Base de données		PostgreSQL 13.1
Application url		https://purebeurre1.herokuapp.com/
Plateforme de déploiement		Heroku

7- Meilleures pratiques :

Une tâche répétitive effectuée manuellement à chaque fois prenait du temps. Cette tâche a été automatisée en créant des scripts et exécutée à chaque fois, ce qui a permis de gagner du temps et des ressources.

8- Critères de sortie :

Tous les cas de test doivent être exécutés et être au vert - **Oui**

9- Conclusion :

Étant donné que les critères de sortie ont été satisfaits et satisfaits comme indiqué dans la section 8, cette application peut être mise en production.

10- Définitions, acronymes et abréviations

framework	un framework désigne un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes d'un logiciel (architecture).	
Test unitaire	est une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel .	
Test fonctionnel	consiste à vérifier que la solution cible, a été développée conformément aux attentes exprimées par le Client.	
Test intégration	est une phase de tests , précédée par les tests unitaires , vérifiant le bon fonctionnement d'une partie précise d'un logiciel; dans le test d'intégration, chacun des modules indépendants du logiciel est assemblé et testé dans l'ensemble.	
Test de performance	a pour objectif de mesurer les temps de réponse d'un système applicatif en fonction de sa sollicitation.	
Couverture de tests	est une mesure utilisée pour décrire le taux de code source exécuté d'un programme quand une suite de tests est lancée. Un programme avec une haute couverture de code, mesurée en pourcentage, a davantage de code exécuté durant les tests ce qui laisse à penser qu'il a moins de chance de contenir de bugs logiciels non détectés.	
MVC	Modèle-vue-contrôleur est un motif d' architecture logicielle destiné aux interfaces graphiques . Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs. - Un modèle (Model) contient les données à afficher. - Une vue (View) contient la présentation de l'interface graphique. - Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur.	
MVT	Est presque équivalent au modèle MVC	
	MVT	MVC
	Modèle	Modèle
	Vue	Controleur
	Template	Vue
exigences	Une exigence est un besoin, une nécessité, une attente auquel un produit ou un service doit répondre ou une contrainte qu'il doit satisfaire.	