

Dokumentacja testów automatycznych

1. Registration.py

Skrypt odpowiedzialny za rejestrację nowego użytkownika. W celu zabezpieczenia przed rejestracją dwa razy tego samego użytkownika (niepozwolone) użyto funkcji *random.randrange(100)* przy nowej nazwie email. Sprawdzono upload pliku img, a następnie po zarejestrowaniu i zalogowaniu zweryfikowano, czy plik dodał się poprawnie.

Screenshots:

- screenshots/register/newuser_before.png
- screenshots/register/newuser_after.png
- screenshots/register/edit_user_avatar.png

	Rejestracja użytkownika
Opis:	Testowanie procesu rejestracji nowego użytkownika
Tester:	MN
Data:	20.05.2019
Dane wejściowe:	Email, hasło, avatar (zdjęcie)
Dane wyjściowe:	Strona główna aplikacji
Wynik:	zaliczony
Uwagi:	brak

2. Login_csv.py

Skrypt odpowiedzialny za logowanie użytkowników. Dane do logowania pobierane z zewnętrznego pliku w formacie CSV (lokalizacja: csv/logins.csv). Skrypt sprawdza zarówno zwykłego użytkownika jak również admina.

Screenshots:

- screenshots/admin-login.png
- screenshots/login/user-loginEMAIL.png

	Logowanie użytkownika
Opis:	Testowanie procesu logowania użytkownika
Tester:	MN
Data:	20.05.2019
Dane wejściowe:	Email, hasło
Dane wyjściowe:	Strona główna aplikacji
Wynik:	zaliczony
Uwagi:	brak

3. Products.py

Skrypt odpowiedzialny na dodanie nowego produktu przez użytkownika Admin. Po dodaniu nowego produktu sprawdzenie, czy wyświetlił się na liście produktów, a następnie sprawdzenie, czy jest dostępny na liście składników do wyboru. Po sprawdzeniu usunięcie produktu z listy.

Screenshots:

- screenshots/products/new_productNAME

	Nowy produkt
Opis:	<i>Testowanie procesu dodawania nowego produktu przez administratora</i>
Tester:	<i>MN</i>
Data:	<i>20.05.2019</i>
Dane wejściowe:	Zalogowany jako administrator, nazwa nowego produktu
Dane wyjściowe:	Produkt wyświetlony na liście dostępnych produktów
Wynik:	zaliczony
Uwagi:	brak

4. New_ingredient.py

Skrypt odpowiedzialny za dodanie nowego składnika z listy dostępnych produktów. W celach testowych wybrano produkt "Avocado". Wybranie produktu z listy, wybranie ilości 1, data ważności 27-08-2019, sprawdzenie czy składnik dodał się do tabeli składników użytkownika, następnie usunięcie go z tej listy.

Screenshots:

- screenshots/ingredients/new_ingredientPRODUCT.png

	Nowy składnik
Opis:	<i>Testowanie procesu dodawania nowego składnika z listy dostępnych produktów</i>
Tester:	<i>MN</i>
Data:	<i>20.05.2019</i>
Dane wejściowe:	Zalogowany użytkownik, nazwa składnika, ilość, jednostka, data
Dane wyjściowe:	Składnik dodany do tabeli składników
Wynik:	zaliczony
Uwagi:	brak

5. Generate_recipes.py

Skrypt odpowiedzialny za sprawdzenie, czy generowana jest lista przepisów z wybranych składników z tabeli. Skrypt zaznacza pierwszy składnik, generuje przepis, odznacza składnik i zaznacza następny aż skończą się składniki w tabeli.

Screenshots:

- screenshots/API/recipesCOUNTER.png

6. Edit_product.py

Skrypt odpowiedzialny za edycję produktu. Po wejściu na stronę edycji produktu pobierana jest wartość pola input i następnie do niej dodawana jest wartość "test", zapis i sprawdzenie, czy nazwa zmieniła się.

Screenshots:

- screenshots/products/edit_productNAME.png

	Edycja produktu
Opis:	Testowanie procesu edycji produktu przez administratora
Tester:	MN
Data:	20.05.2019
Dane wejściowe:	Dostępny produkt na liście produktów
Dane wyjściowe:	Zmieniona nazwa produktu
Wynik:	zaliczony
Uwagi:	brak

7. Edit_ingredient.py

Skrypt odpowiedzialny za edycję składnika. Najpierw pobierane są aktualne wartości (nazwa, data ważności, ilość, jednostka) dla pierwszego składnika w tabeli. Następnie przechodzimy do strony edycji, zmieniamy wartości, zapisujemy i porównujemy, czy w tabeli są inne wartości niż pierwotne.

Screenshots:

- screenshots/ingredients/edited_ingredientNAME.png

	Edycja składnika
Opis:	Testowanie procesu edycji składnika przez użytkownika
Tester:	MN
Data:	20.05.2019
Dane wejściowe:	Dostępny składnik w tabeli składników

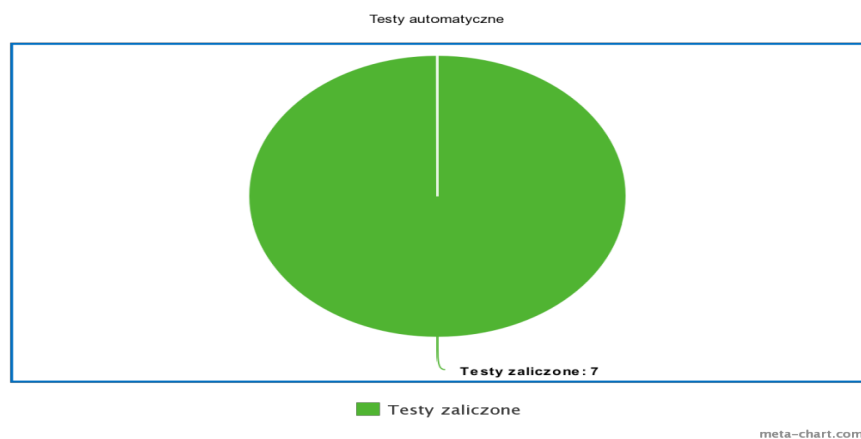
Dane wyjściowe:	Zmieniona nazwa, ilość, jednostka i data składnika
Wynik:	zaliczony
Uwagi:	brak

8. Login_func.py

Skrypt zawierający funkcję "logging" odpowiedzialną za logowanie do aplikacji

9. Sandbox.py

Skrypt do testowania kodu, sprawdzania poprawności danych itp. Bez znaczenia dla testów aplikacji.



Struktura katalogów:

