# Projekt KONCOWY

Agnieszka Mazurowska ZDtestpol93 [AgMz\_Projekt\_końcowy]

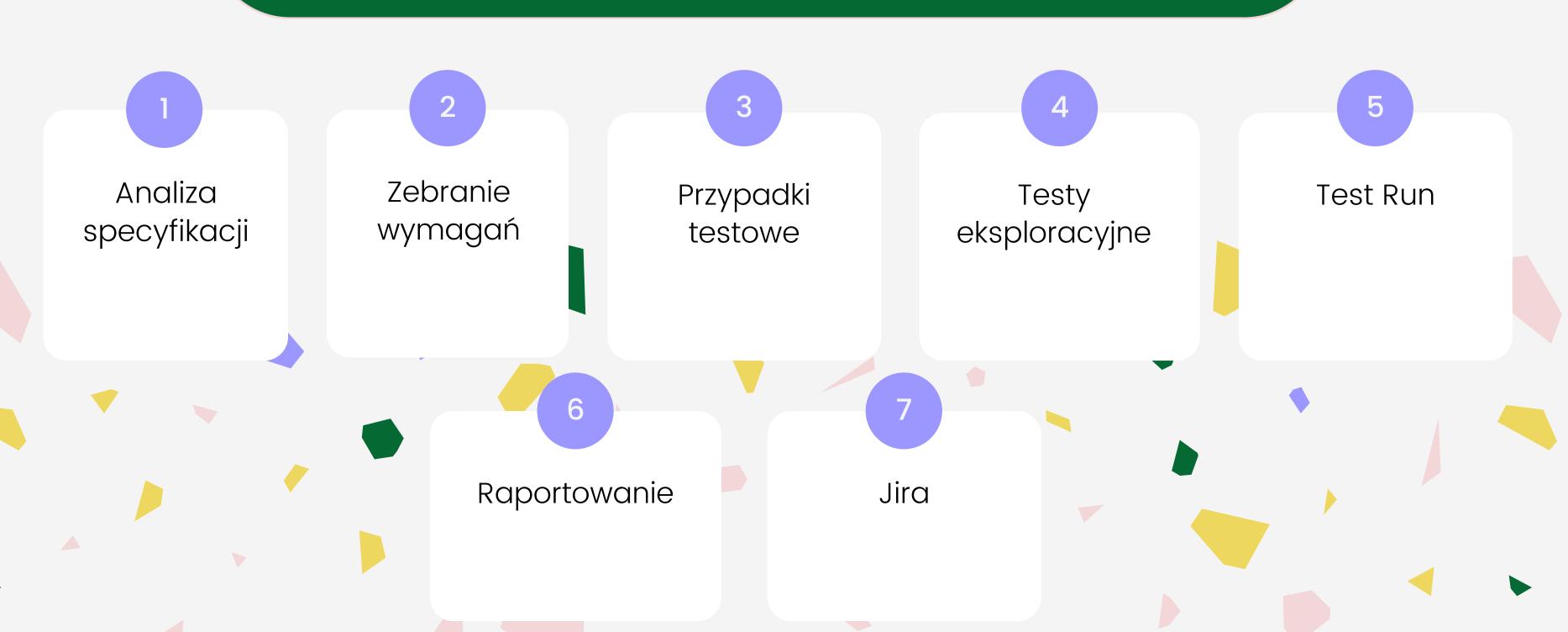
### AgMz\_Projekt\_końcowy

Niniejszy projekt końcowy zamieszczony jest w moim repozytorium na Github:

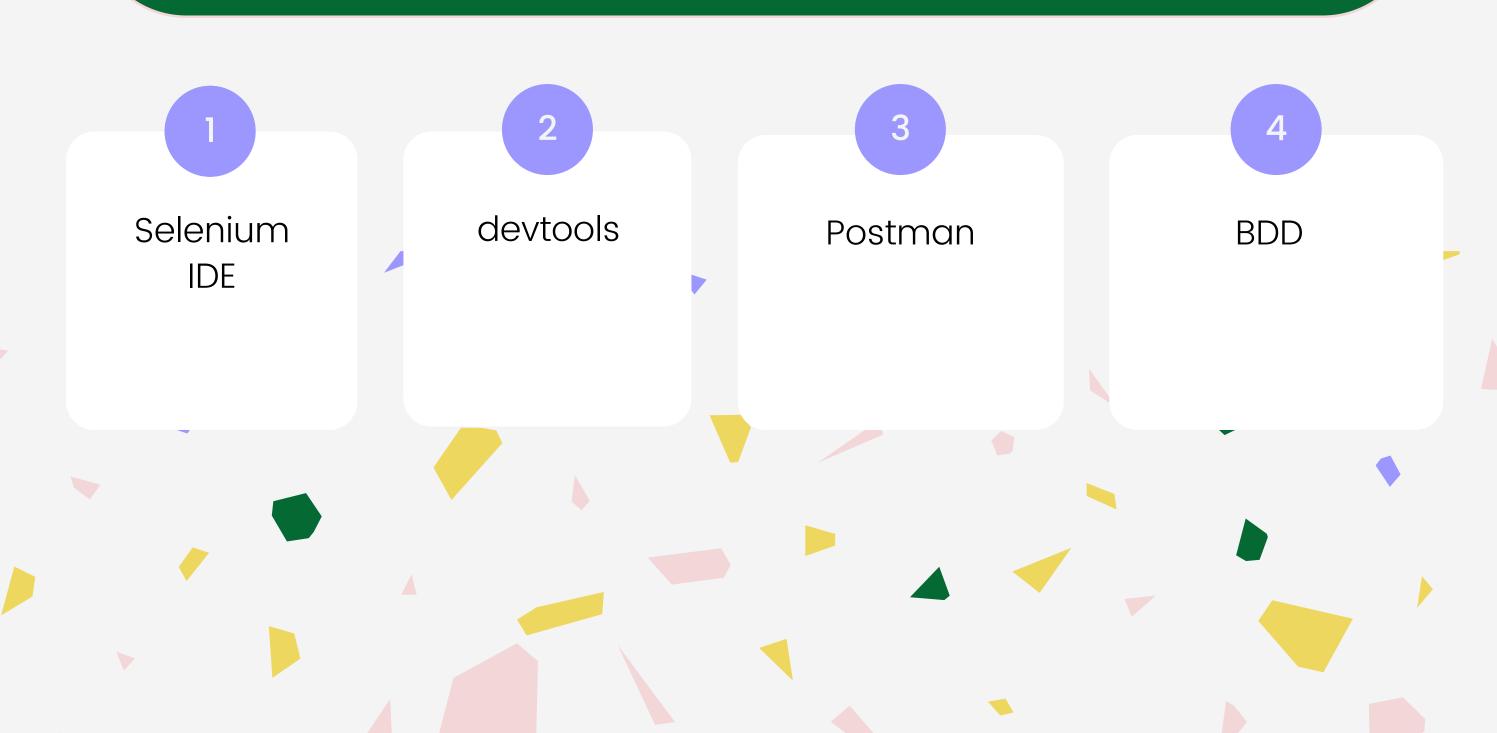
https://github.com/AgnieszkaMz/AgMz\_Projekt\_koncowy

Został on podzielony na dwa etapy ze względu na testowane aplikacje.

## Etapl



## Etap II





W moim projekcie wykorzystałam następujące aplikacje:

- MrBuggy3 1.2.0.146
- SwagLabs https://www.saucedemo.com/
- API https://regres.in











#### MrBuggy3 1.2.0.146

Jest to aplikacja stworzona na potrzeby TestingCup 2015.

Zawiera ona defekty, które mają na celu zweryfikować wiedzę i umiejętności testerów oprogramowania.

Udostępniona publicznie specyfikacja aplikacji daje możliwość nauki testowania już od samego początku.



#### SwagLabs

Jest to aplikacja typu ecommerce, zaprojektowana i utrzymywana przez firmę Sauce Labs.

Strona pozwala na korzystanie ze sklepu, poprzez konta 4 różnych, zdeklarowanych z góry użytkowników.

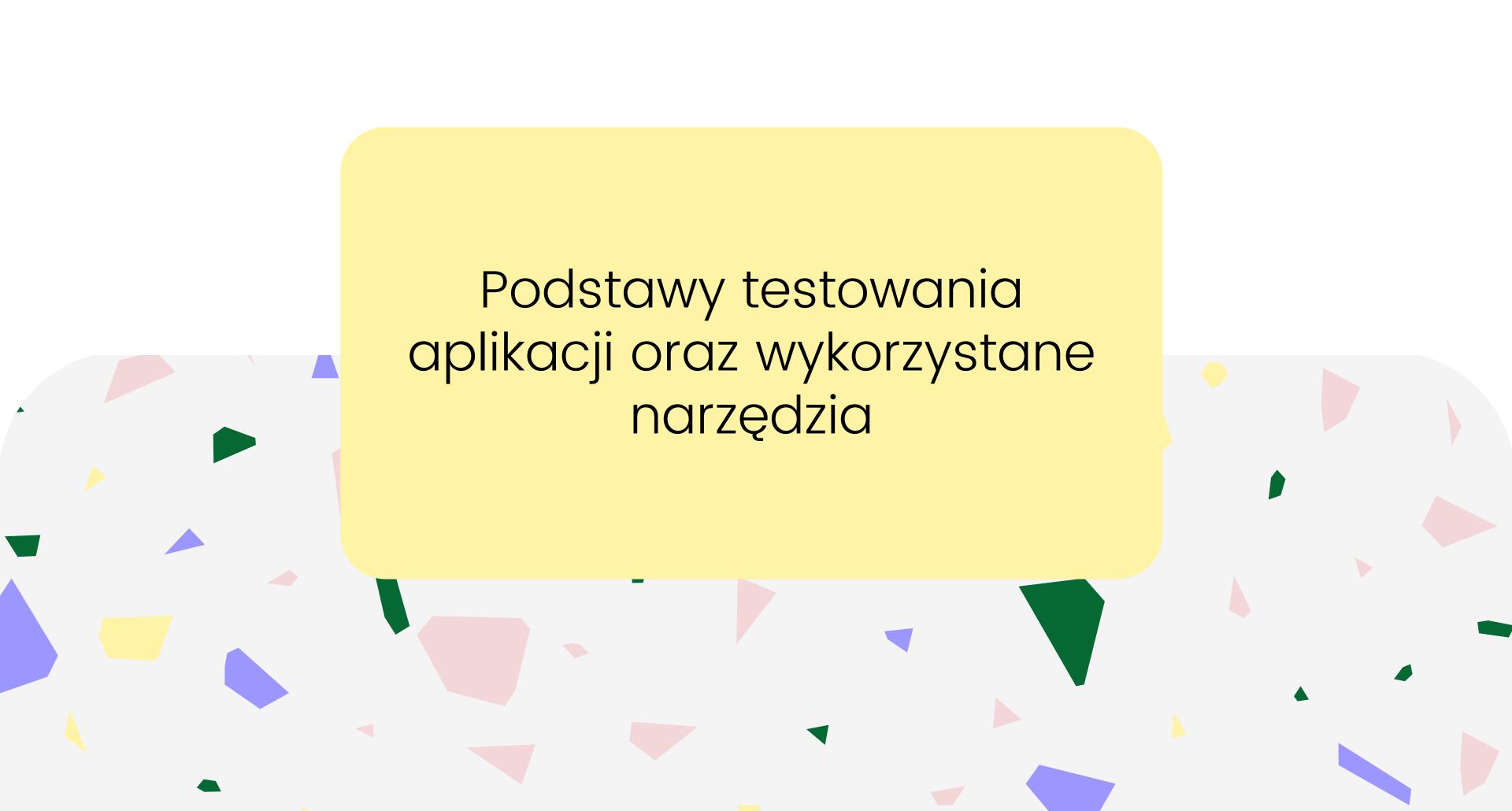


#### API regres.in

Jest to publicznie dostępne API.

Ogólny interfejs API zgodny z zasadami REST i akceptujący typ zawartości application/json

Pozwala przetestować podstawowe metody HTTP, czyli GET, POST, PUT, DELETE.





#### MrBuggy3 1.2.0.146

Do opracowania etapu I projektu wykorzystałam aplikację MrBuggy3 1.2.0.146, która częściowo omówiona była podczas zajęć "Projekt końcowy".

Aplikacja testowana była na podstawie analizy specyfikacji. Została sporządzona lista nieścisłości oraz wymagań aplikacji. Przygotowane zostały prawdopodobne przypadki testowe.

Przeprowadzona została sesja eksploracyjna oraz przygotowane zostały raporty.



#### MrBuggy3 1.2.0.146

Przy pracy nad aplikacją korzystałam z narzędzi TestRail oraz JIRA.

W TestRail przygotowane zostały potencjalne przypadki testowe oraz dopisane nowe po przeprowadzeniu sesji eksploracyjnej. Wygenerowane zostały Test Run oraz raporty Summary Defects, Comparsion for Results, Runs Summary.

W Jira został zgłoszony defekt na podstawie nieudanego przypadku testowego przeprowadzonego w narzędziu TestRail



#### SwagLabs

Do opracowania etapu II projektu wykorzystałam aplikacje internetową SwagLabs.

W oparciu o wiedzę i umiejętności odnaleziony został błąd, na podstawie którego przygotowany został przypadek testowy nagrany przy pomocy narzędzia Selenium IDE.

Przy pomocy devtools wykonałam przykładowe zadanie gdzie prezentuje jak można pomieszać w frondendzie aplikacji lub sprawdzić interesujące nas informacje.



#### API regres.in

W etapie II projektu wykorzystałam również API, dzięki któremu mogłam wykorzystać narzędzie Postman.

Przetestowałam wszystkie najbardziej podstawowe metody HTTP oraz przeprowadziłam na ich podstawie testy.

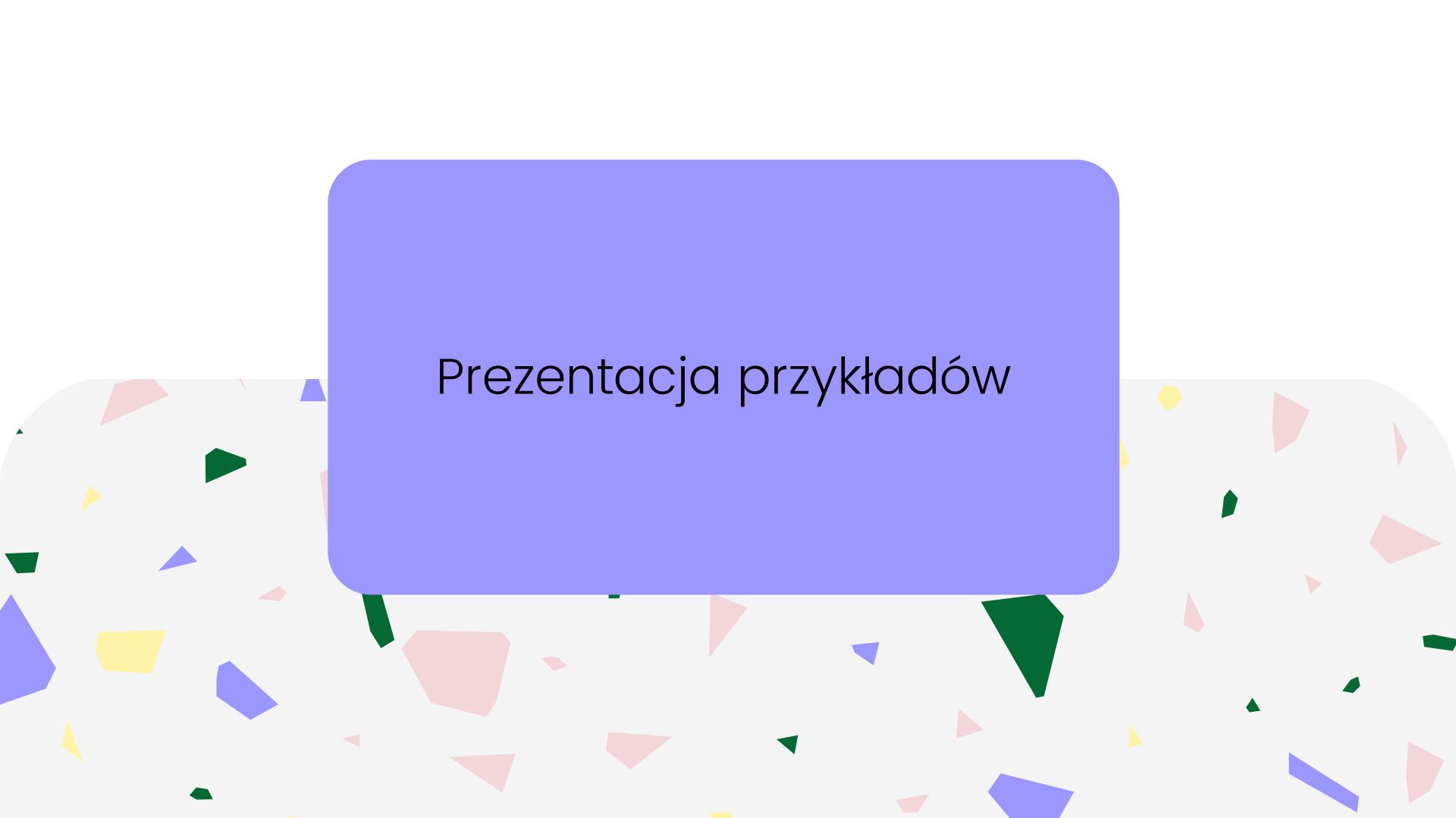


#### MrBuggy3 i SwagLabs

W II etapie zaprezentowałam techniki Behavior Driven Development. W tym celu wykorzystałam wcześniej opracowywane przypadki testowe z aplikacji MrBuggy3 oraz aplikacji internetowej

Opisałam dwa przypadki testowe na podstawie Scenario

Rozpisałam również Scenario Outline z Examples dla konkretnej funkcjonalności



### Przykładowe wymagania

1	Wymagania środowiskowe do części klienckiej: Wlndows 7, Wlndows 8/8.1				
2	W oknie aplikacji musi się znaleźć ikona MrBuggy 3				
3	wymóg logowania przy użyciu loginu 2 do 8 znaków, login nie może zawierać białych znaków ani polskich znaków dialektycznych				
4	Wersja demo posiada dwa zadania				
5	Trzy wersje aplikacji - demo, koncowa, publiczna				
6	Aplikacja MrBuggy 3 składa się z dwóch części - kliencka i serwerowa				
7	Wymagania środowiskowe do części serwerowej: WIndows 7				
8	część kliencka aplikacji w rozdzielczości 1024x 768				
9	W aplikacji dostępnych ma być nie mniej niż 15 zadań (problemów)				
10	Wszystkie komunikaty, informacje oraz nazwy klawiszy w interfejsie graficznym muszą być w języku polskim.				

### Przykładowe przypadki testowe

TCID	Tytuł	Warunki wstępne	ID Kroku	Opis kroku	Oczekiwany rezultat
1	Sprawdzenie zgodności aplikacji z systemem operacyjnym Windows 7	komputer z systemem operacyjnym Windows 7, rozpakowane archiwum z aplikacją	1	uruchom aplikację klikając w plik wykonywalny	Aplikacja została poprawnie uruchomiona
2	Sprawdzenie możliwości zarejestrowania się w aplikacji MrBuggy przy użyciu loginu o konkretnych wymaganiach	Użytkownik nie znajduje się w bazie, login musi mieć od 2 do 8 znaków, bez białych znaków, bez polskich znaków	1	Uruchom aplikację	
			2	kliknij pole zarejestruj się	
			3	wypełnij pole login	
			4	wypełnij pole adres e-mail	
			5	wypełnij pole hasło	
			6	wypełnij pole powtórz hasło	
			7	kliknij pole zarejestruj sie	Powiodła się rejestracja, jest możliwe logowanie
3	Logowanie do aplikacji MrBuggy	Użytkownik jest zarejestrowany w bazie, login musi mieć od 2 do 8 znaków, bez białych znaków, bez polskich znaków	1	Uruchom aplikację	
			2	Kliknij pole zaloguj się	
			3	wypełnij pole login, zgodnie z wymaganiami z warunków wstępnych	
			4	wypełnij pole adres e-mail	
			5	wypełnij pole hasło	
			6	kliknij pole zaloguj się	Logowanie do aplikacji powiodło się
4	Ikona MrBuggy 3 widoczna w oknie aplikacji	Uruchomienie aplikacji	1	uruchom aplikację	
			2	kliknij pole zaloguj się	
			3	wypełnij pole adres e-mail	
			4	wypełnij pole hasło	
			5	kliknij pole zaloguj się	
			6	w oknie aplikacji użytkownik widzi ikonę MrBuggy3	Ikona MrBuggy 3 widoczna w lewym górynym rogu okna aplikacji (Weryfikacja zgodności z makietami)

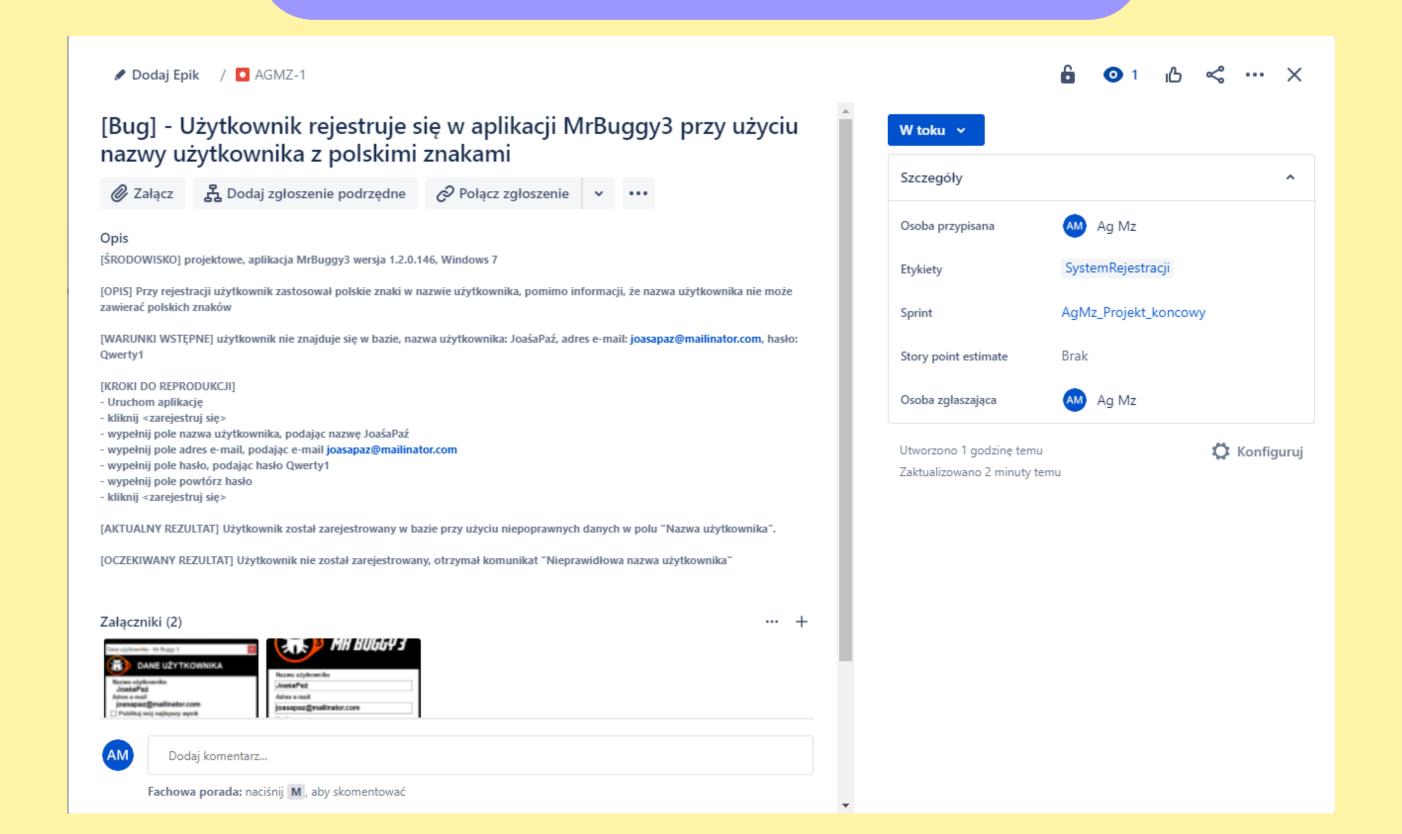
### Przykładowy przypadek testowy – TestRail

OVERVIEW MILESTONES **TEST RUNS & RESULTS** TEST CASES REPORTS TODO Sprawdzenie możliwości zarejestrowania się w aplikacji MrBuggy przy użyciu nazwy użytkownika z polskimi znakami Test Cases Type Priority Estimate References Other Low None None **Automation Type** None Preconditions Użytkownik nie znajduje się w bazie, login musi mieć od 2 do 8 znaków Steps Uruchom aplikację kliknij pole zarejestruj się wypełnij pole nazwa użytkownika, podając nazwę z polskimi znakami wypełnij pole adres e-mail wypełnij pole hasło wypełnij pole powtórz hasło kliknij pole zarejestruj się **Expected Result** rejestracja się nie powiodła, otrzymuję komunikat o nieprawidłowej nazwie użytkownika

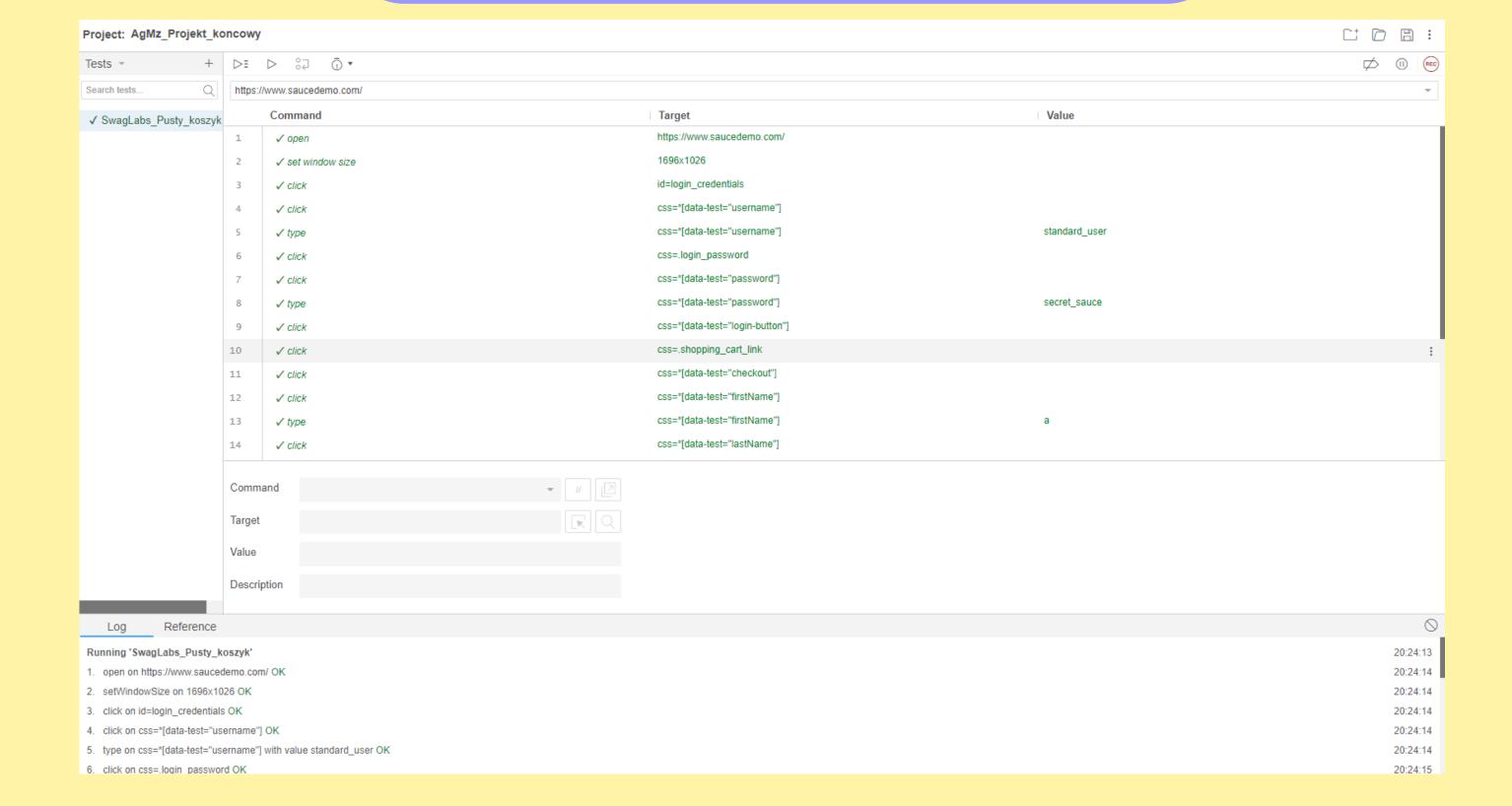
### Sesja eksploracyjna

ID Sesji: TSE001								
Tester	Ag Mz	Data	30/05/2022					
Czas Rozpoczęcia	09:28	Czas zakończenia	11:19					
Cel	Przetestowanie funkcjonalności rejestracji i logowania. Aplikacja MrBuggy3							
Znaloziono błody	Rejestracja i logowanie ma następować przy pomocy loginu o określonych warunkach (2-8 znaków, bez białych znaków i bez polskich znaków), jednak w okienku rejestracji i w okienku logowania nie widnieje pole o nazwie "login". Przy rejestracji widnieje pole o nazwie "nazwa użytkownika", natomiast przy logowaniu nie podaje sie ani loginu ani nazwy użytkownika (do wypełnienia sa pola "adres e-mail" i "hasło"). Możliwa jest rejestracja przy użyciu w nazwie użytkownika polskich znaków oraz nazwy z większą ilością znaków niż zadeklarowane 8. Po wprowadzeniu wszystkich danych potrzebnych do rejestracji i kliknięciu "zaloguj się" wyskakuje kominukat "Nieoczekiwany błąd. Spróbować ponownie?". Jeśli klikniemy "tak" pojawia się komunikat "Podany adres e-mail istnieje już w bazie danych." W tym momencie nic więcej się nie dzieje. Więc nie wiadomo czy zostaliśmy zarejestrowani czy nie.							
Dalsza analiza	Dalsza analiza Przy rejestracji należy zmienić pole "nazwa użytkownika" na pole "login", przy logowaniu zmienić pole "adres e-mail na pole "login" oraz wprowadzić wymogi dotyczące loginu, tj. 2-8 znaków bez biały i polskich znaków. Należy wyeliminować błędy przy rejestracji z komunikatami "Nieoczekiwany błąd" oraz "Podany adres -email istnieje już w bazie danych". Należy poinformowac użytkownika czy została lub nie został zarejestrowany w bazie aplikacji.							

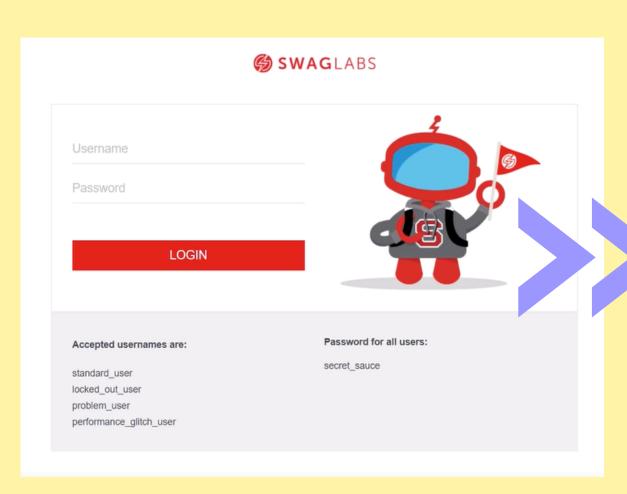
### Przykładowy raport z awarii – JIRA

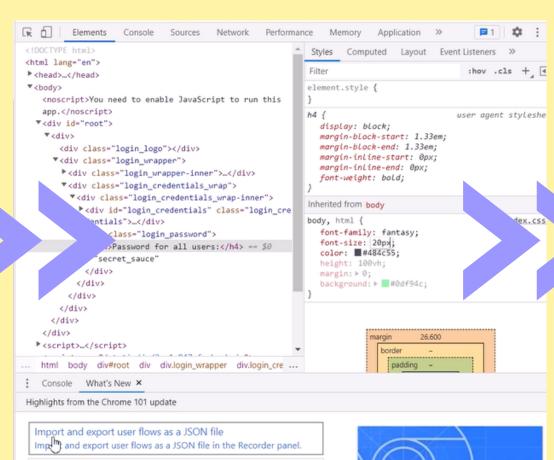


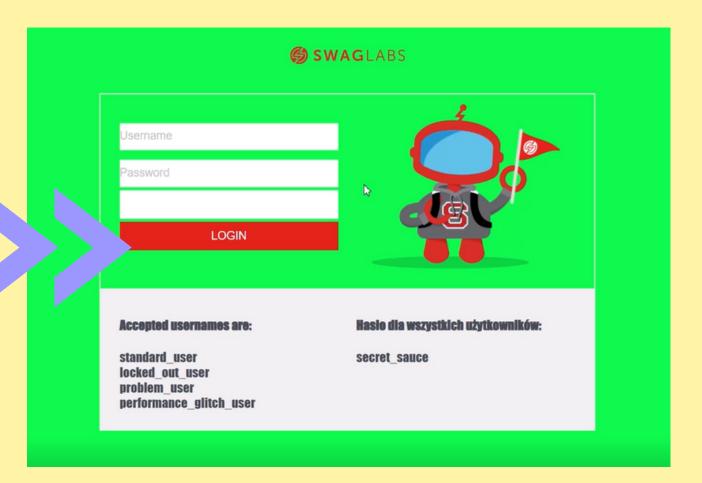
### Nagrywanie testów -Selenium IDE



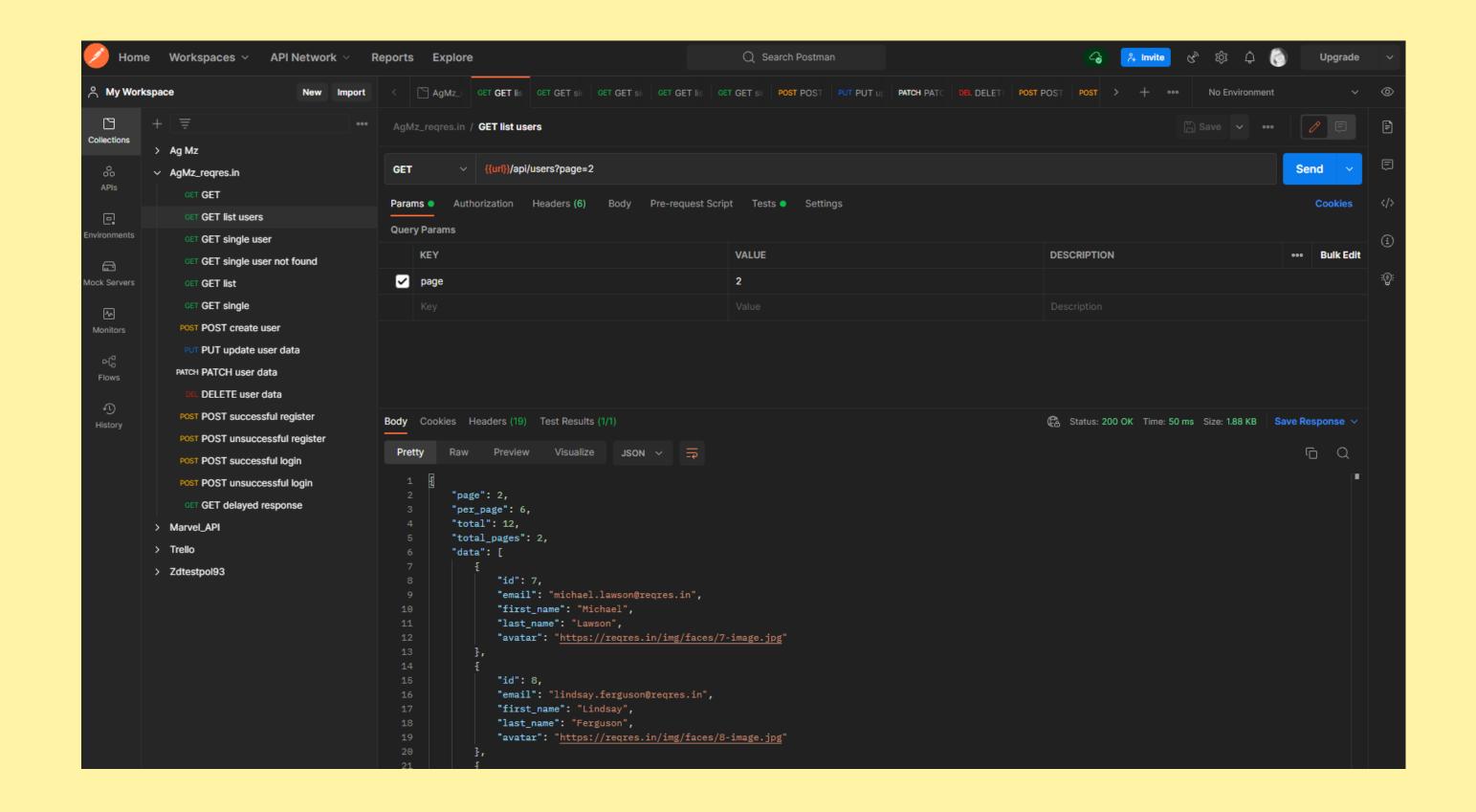
### devtools



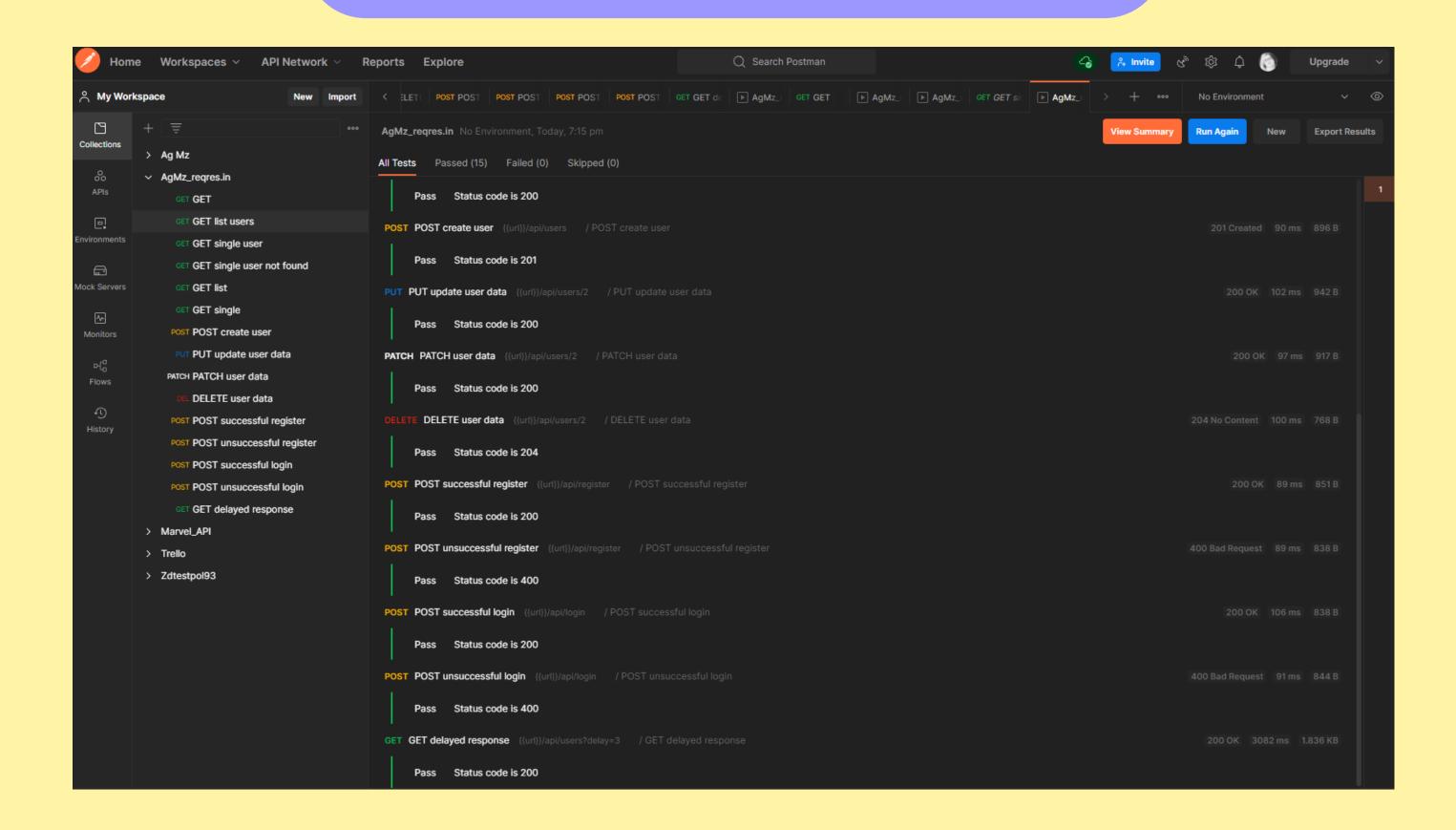




#### Postman



### Test Run - Postman



### Behavior Driven Development

```
Feature: Finalizacja zakupu pustego koszyka w aplikacji internetowej https://www.saucedemo.com/
   Scenario: Możliwość sfinalizowania transakcji zakupu pustego koszyka
        Given Użytkownik nie wybiera żadnego produktu na stronie
        And Użytkownik przechodzi to pustego koszyka
        When Użytkownik potwierdza puste zamówienie
        And Użytkownik podaje wszystkie dane potrzebne do wysłyki
        Then Transakcja się powiodła
        But Pojawił się komunikat o przyjęciu zamówienia
11
12
13 Scenario: Brak możliwości sfinalizowania transakcji zakupu pustego koszyka
        Given Użytkownik nie wybiera żadnego produktu na stronie
15
        And Użytkownik przechodzi to pustego koszyka
16
        When Użytkownik potwierdza puste zamówienie
17
        Then Transakcja się nie powiodła
        But Pojawił się komunikat "Twój koszyk jest pusty"
19
20
    Feature: Możliwości rejestracji w aplikacji MrBuggy3
22
    #Login (2-8 znaków, bez białych i bez polskich znaków), email, hasło
24
    Scenario Outline: Weryfikacja rejestracji
        Given Użytkownik ma uruchomiony formularz rejestracji
26
        When Użytkownik podaje login <login>
27
28
        And Użytkownik podaje e-mail <e-mail>
        And Użytkownik podaje hasło <hasło>
        And Użytkownik klika przycisk potwierdzający rejestrację
31
        Then <Odpowiedź systemu>
32
33 Examples:
    llogin
                                                            |Odpowiedź systemu
34
                          le-mail
                                              |hasło
                                              |poprawneHasło|Rejestracja się powiodła
    |poprawnyLogin
                          |prawidłowy@email
    |loginZPolskimiZnakami|prawidłowy@email
                                              |poprawneHasło|Rejestracja się nie powiodła, komunikat "Błędny login. Login nie może zawierać polskich znaków"|
```

## Dziękuję!

#### DOSTĘPY:

Github: https://github.com/AgnieszkaMz/AgMz\_Projekt\_koncowy

TestRail: https://agmz.testrail.io/

Jira: https://agmz19.atlassian.net/jira/software/projects/AGMZ/boards/1