

Cucumber to narzędzie oparte na frameworku Behavior Driven Development (BDD). BDD to metodologia umożliwiająca zrozumienie funkcjonalności aplikacji w postaci prostej reprezentacji tekstowej.

Narzędzie Cucumber jest zwykle używane w czasie rzeczywistym do pisania testów akceptacyjnych aplikacji. Narzędzie Cucumber zapewnia obsługę wielu języków programowania, takich jak Java, Ruby, .Net itp. Może być zintegrowane z wieloma narzędziami, takimi jak Selenium, Capybara itp.

Gherkin to język używany przez narzędzie Cucumber. Jest to prosta angielska reprezentacja zachowania aplikacji. Cucumber używa koncepcji plików funkcji do celów dokumentacyjnych. Treść plików funkcji jest napisana w języku Gherkin.

Framework Cucumber Gherkin używa pewnych słów kluczowych, które są niezbędne do napisania pliku funkcji.

W plikach funkcji najczęściej używane są następujące terminy:

1) Funkcja:

Plik funkcji musi zawierać opis wysokiego poziomu testowanej aplikacji (AUT).

Pierwsza linia pliku funkcji musi zaczynać się słowem kluczowym „Funkcja”. Zgodnie ze standardami określonymi przez Cucumber, plik funkcji musi zawierać następujące trzy elementy jako pierwszy wiersz.

- **Słowo kluczowe funkcji**
- **Nazwa funkcji**
- **Opis funkcji (opcjonalnie)**

Po słowie kluczowym funkcji musi następować nazwa funkcji. Może zawierać opcjonalną sekcję opisu, która może obejmować wiele wierszy pliku funkcji. Plik funkcji ma rozszerzenie .feature.

2) Scenariusz:

Scenariusz to testowa specyfikacja testowanej funkcjonalności. W idealnym przypadku plik elementu może zawierać jeden lub więcej scenariuszy jako część elementu. Scenariusz obejmuje wiele kroków testowych. Zgodnie ze standardami ogórkowymi scenariusz musi obejmować 3-5 kroków testowych, ponieważ długie scenariusze mają tendencję do tracenia swojej mocy ekspresyjnej, gdy liczba kroków wzrasta.

Scenariusz może obejmować następujące kroki:

- **Czynność do wykonania przez użytkownika.**
- **Oczekiwane wyniki działania.**

W języku Gherkin scenariusz musi zawierać następujące słowa kluczowe:

- **Dany**
- **Gdy**
- **Następnie**
- **I**

Dany:

Słowo kluczowe Given jest używane do określenia warunków wstępnych wykonania określonego scenariusza. Scenariusz może zawierać więcej niż jedno określone stwierdzenie lub może nie być podanych instrukcji dla scenariusza.

Gdy:

To słowo kluczowe służy do określenia akcji lub zdarzenia wykonywanego przez użytkownika, takiego jak kliknięcie przycisku, wprowadzenie danych do pola tekstowego itp. W jednym scenariuszu może być wiele instrukcji when.

Następnie:

Następnie słowo kluczowe służy do określenia spodziewanego wyniku akcji wykonanej przez użytkownika. W idealnym przypadku po słowie kluczowym When musi występować słowo kluczowe Then, aby zrozumieć oczekiwany wynik działań użytkownika.

I:

Słowo kluczowe jest używane jako słowo kluczowe w połączeniu do łączenia wielu instrukcji. Na przykład, wiele stwierdzeń Given i When w scenariuszu można łączyć za pomocą słowa kluczowego „And”.

3) Zarys scenariusza:

Zarys scenariusza jest sposobem parametryzacji scenariuszy.

Jest to idealnie używane, gdy ten sam scenariusz musi zostać wykonany dla wielu zestawów danych, ale kroki testowe pozostają takie same. Po konspekcie scenariusza należy umieścić słowo kluczowe „Przykłady”, które określa zestaw wartości dla każdego parametru.

Przykład:

Zarys scenariusza: Prześlij plik

Dany że użytkownik jest na ekranie przesyłania plików.

Gdy użytkownik klika przycisk Przeglądaj.

I użytkownik wchodzi do pola tekstowego przesyłania.

I użytkownik klika przycisk Enter.

Następnie sprawdza, czy przesłanie pliku się powiodło.

Przykłady:

| nazwa pliku |

| plik1 |

| plik2 |

Parametry w zarysie scenariusza należy określić za pomocą znaków „”. Lista wartości danych dla parametru musi być określona za pomocą symbolu Pipe (|).

4) Tło:

Słowo kluczowe w tle służy do grupowania wielu podanych instrukcji w jedną grupę.

Jest to zwykle używane, gdy ten sam zestaw danych instrukcji jest powtarzany w każdym scenariuszu pliku funkcji. Zamiast wielokrotnie podawać dane stwierdzenia dla każdego scenariusza, można je określić za pomocą słowa kluczowego „Tło” przed pierwszym scenariuszem.

Przykład:

Tło:

Podany użytkownik znajduje się na stronie logowania do aplikacji

Zalety Cucumber Gherkin Framework

- Cucumber to narzędzie typu open source.
- Reprezentacja zwykłego tekstu ułatwia użytkownikom nietechnicznym zrozumienie scenariuszy.
- Wypełnia lukę komunikacyjną między różnymi interesariuszami projektu, takimi jak analitycy biznesowi, programiści i personel ds. Zapewnienia jakości.
- Przypadki testowe automatyzacji opracowane przy użyciu narzędzia Cucumber są łatwiejsze do utrzymania i zrozumienia.
- Łatwa integracja z innymi narzędziami, takimi jak selen i kapibara.