

Aplikacja BUWing - testy

Aleksandra Falkiewicz, Maciej Kozłowski,
Joanna Ostrowska, Kacper Rokicki

14 maja 2019

1 Testy funkcjonalne

Aplikacja mobilna wykonuje zadania dwóch typów: wyświetla użytkownikowi aktualne dane lub umożliwia mu wykonanie pewnej akcji tj. logowanie, wylogowanie, zajęcie miejsca, zaproszenie, dodanie lub usunięcie znajomych, zmiana danych użytkownika. W przypadku pierwszego typu testy będą polegały na sprawdzaniu poprawnego umiejscowienia pól na ekranie oraz poprawność wyświetlanych danych. W drugim przypadku testy będą polegały na zasymulowaniu akcji z poprawnymi i niepoprawnymi danymi, a następnie sprawdzeniu czy aplikacja wykonała oczekiwaną przez nas akcję np. wyświetliła odpowiednią wiadomość użytkownikowi lub uruchomiła nową aktywność. Testy zrealizujemy automatycznie przy pomocy jednego z dostępnych frameworków.

Przypadki testowe:

- sprawdzenie poprawności wyświetlanych danych
- sprawdzenie poprawności wyświetlania danych (umiejscowienie pól na ekranie)
- sprawdzenie poprawności reakcji aplikacji na różne dane podane w możliwych dla użytkownika akcji

2 Testy interfejsu użytkownika

Sprawdzenie wyglądu ekranów, reagowania aplikacji na naciśnięcie przycisku, otwierania wirtualnej klawiatury itp. zostanie przetestowane ręcznie.

3 Testy bazy danych

Komunikacja aplikacji mobilnej z bazą danych jest realizowana na zasadzie web service'u. Aplikacja wysyła żądanie do serwera zawierające dane uwierzytelniania użytkownika, natomiast serwer weryfikuje ich poprawność i w razie sukcesu odsyła odpowiednie informacje. Technicznie jest to realizowane za pomocą skryptów PHP - login i hasło użytkownika przesyłane są w argumentach POST, następnie serwer realizuje odpowiednie zapytania do bazy danych PostgreSQL, a ich wyniki odsyła zapisane w formacie JSON. Umożliwia to łatwe testowanie poprawności zachowania serwera oraz bazy danych - wystarczy sprawdzić, czy po wywołaniu skryptu dostaniemy poprawne dane.

Przypadki testowe:

- próba dostępu do danych przez nieistniejącego/nieaktywnego użytkownika
- próba wykonania SQL injection, np. podanie spreparowanego loginu przy rejestracji
- normalne użytkowanie: sprawdzenie czy po wywołaniu skryptu do zajmowania/zwolnienia miejsca wynik wywołania skryptu do sprawdzenia zajętości będzie
- przypadki szczególne: próba jednoczesnego zajęcia tego samego miejsca przez dwie osoby
- sprawdzenie, czy po żądaniu usunięcia konta wszystkie informacje dotyczącego danego użytkownika zostały usunięte
- testy obciążeniowe: wiele jednoczesnych zapytań do bazy wymagających modyfikowania tych samych tabel (symulacja akcji "BUW dla sów")
- sprawdzenie liczby kolizji zahashowanych haseł użytkowników i tokenów generowanych przy istotnych zmianach w koncie
- sprawdzenie (szybkości) zachowania bazy dla dużej liczby użytkowników i znajomości pomiędzy nimi