# Scenariusz testowy – Rejestracja

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id | Nazwa | Opis | Typ | Czynności przygotowawcze | Czynności końcowe |
| 2 | Rejestracja | Sprawdzenie poprawności działania endpointów do operacji na komentarzach | Testy jednostkowe | * Uruchomić lokalnie backend oraz serwer bazy danych * Uruchomić postmana | - |

## Przypadki testowe – **Rejestracja**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Nazwa | Warunki wstępne | Kroki wykonania | Oczekiwany rezultat |
| 1 | Próba zarejestrowania, jeśli nie podano wszystkich wymaganych danych. | * Nie uzupełniono wszyskich wymaganych danych. | * Przy uzupełnianiu danych pozostawić jedno miejsce z wymaganymi danymi puste * Wysłać żądanie POST na dopowiedni endpoint odpowiadający za rejestrację | Użytkownik nie został zarejestrowany do systemu, wysłanie odpowiedzi informującej o nieudanym logowaniu |
| 2 | Próba zarejestrowania użytkownika wykorzystując adres email przypisany do aktywnego konta. | * Poprawne uzupełnienie pól rejestracji bez adresu email. * Podanie emailu przypisanego do aktywnego konta | * Wysłać żądanie POST na dopowiedni endpoint odpowiadający za rejestrację | Użytkownik nie został zarejestrowany do systemu, wysłanie odpowiedzi o statusie informującym o wykorzystaniu tego adresu email przez innego użytkownika |
| 3 | Próba zarejestrowania użytkownika wykorzystując adres numer telefonu przypisany do aktywnego konta. | * Poprawne uzupełnienie pól rejestracji bez numeru telefonu. * Podanie numeru telefonu przypisanego do aktynego konta | * Wysłać żądanie POST na dopowiedni endpoint odpowiadający za rejestrację | Użytkownik nie został zarejestrowany do systemu, wysłanie odpowiedzi o statusie informującym o wykorzystaniu tego numeru telefonu przez innego użytkownika |
| 4 | Próba zarejestrowania użytkownika przy podaniu poprawnych danych. | * Uzupełnione wszystkie wymagane pola | * Wysłać żądanie POST na dopowiedni endpoint odpowiadający za rejestrację | Utworzenie konta użytkownika |

# Scenariusz testowy – **Logowanie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id | Nazwa | Opis | Typ | Czynności przygotowawcze | Czynności końcowe |
| 1 | Logowanie | Sprawdzenie poprawności logowania użytkownika. | Testy jednostkowe | * Uruchomić lokalnie backend oraz serwer bazy danych * Uruchomić postmana | - |

## Przypadki testowe – Logowanie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Nazwa | Warunki wstępne | Kroki wykonania | Oczekiwany rezultat |
| 1 | Próba zalogowania przy wpisaniu poprawnego loginu i hasła | Istnieje użytkownik zarejestrowany w systemie | * Wprowadzenie poprawnego loginu i hasła * Wysłanie żądania POST | Użytkownik został zalogowany o systemu |
| 2 | Próba zalogowania przy wpisaniu niepoprawnego loginu i poprawnego hasła | Istnieje użytkownik z podanym hasłem z w systemie | * Wprowadzenie niepoprawnego loginu i poprawnego hasła * Wysłanie żądania POST | Użytkownik nie został zalogowany do systemu, wysłanie odpowiedzi informującej o nieudanym logowaniu |
| 3 | Próba zalogowania przy wpisaniu niepoprawnego hasła i poprawnego loginu | Istnieje użytkownik z podanym loginem z w systemie | * Wprowadzenie niepoprawnego hasła i poprawnego loginu * Wysłanie żądania POST | Użytkownik nie został zalogowany do systemu, wysłanie odpowiedzi informującej o nieudanym logowaniu |

# Scenariusz testowy – **Wylogowanie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id | Nazwa | Opis | Typ | Czynności przygotowawcze | Czynności końcowe |
| 3 | Wylogowanie | Sprawdzenie poprawności działania endpointów do operacji wylogowania | Testy jednostkowe | * Uruchomić lokalnie backend oraz serwer bazy danych * Uruchomić postmana | - |

## Przypadki testowe – **Wylogowanie**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Nazwa | Warunki wstępne | Kroki wykonania | Oczekiwany rezultat |
| 1 | Sprawdzenie poprawności procesu wylogowania użytkownika z aktywnej sesji | * Sesja musi być aktywna * Użytkownik musi być zalogowany | * Wysłanie żądania GET na endpoint odpowiadający za wylogowanie | Zwrócono informację o poprawnym wylogowaniu |
| 2 | Próba wylogowania użytkownika z nieaktywnej sesji | * Sesja musi być nieaktywna * Użytkownik musi być zalogowany | * Wysłanie żądania GET na endpoint odpowiadający za wylogowanie | Odpowiedź informująca o potrzebie odświeżenia strony |

# Scenariusz testowy – **Zmiana danych użytkownika**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id | Nazwa | Opis | Typ | Czynności przygotowawcze | Czynności końcowe |
| 4 | Zmiana danych użytkownika | Sprawdzenie poprawności działania endpointów do operacji na danych z profilu użytkownika | Testy jednostkowe | * Uruchomić lokalnie backend oraz serwer bazy danych * Uruchomić postmana | - |

## Przypadki testowe – **Zmiana danych użytkownika**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Nazwa | Warunki wstępne | Kroki wykonania | Oczekiwany rezultat |
| 1 | Sprawdzanie poprawności procesu zmiany danych profilu użytkownika | * Sesja musi być aktywna * Użytkownik musi być zalogowany * Muszą zostać wprowadzone poprawne dane | * Wysłanie żądania PUT na endpoint odpowiadający za zmianę danych profilu użytkownika | Zaktualizowanie bazy danych i zwrócenie odpowiedzi informującej o poprawnym zaktualizowaniu profilu |
| 2 | Próba wprowadzenia niepoprawnych danych do profilu użytkownika | * Sesja musi być aktywna * Użytkownik musi być zalogowany * Muszą zostać wprowadzone niepoprawne dane | * Wysłanie żądania PUT na endpoint odpowiadający za zmianę danych profilu użytkownika | Zwróconie odpowiedzi informującej o niepoprawności wprowadzonych danych |
| 3 | Próba zapisania profilu użytkownika bez wprowadzenia wymaganych danych | * Sesja musi być aktywna * Użytkownik musi być zalogowany * Przynajmniej jedno pole z wymaganymi danymi musi pozostać puste | * Wysłanie żądania PUT na endpoint odpowiadający za zmianę danych profilu użytkownika | Zwróconie odpowiedzi informującej o braku wymaganych danych |