**# jako klient aplikacji chcę widzieć oferty pracy dla Junior Java Developera**

1. korzystamy ze zdalnego serwera HTTP (skrypt który pobiera oferty ze stron WWW)
2. klient musi używać tokena, żeby zobaczyć oferty
3. klient może się zarejestrować
4. aktualizacja ofert w bazie danych jest co 3 godziny (wtedy odpytujemy zdalny serwer z pkt. 1)
5. oferty w bazie nie mogą się powtarzać (decyduje url oferty)
6. klient może pobrać jedną ofertę pracy poprzez unikalne Id
7. klient może pobrać wszystkie dostępne oferty **kiedy jest zautoryzowany**
8. jeśli klient w ciągu 60 minut robi więcej niż jedno zapytanie do dane powinny pobierać się z cache (ponieważ niepobieranie z bazy danych kosztuję)
9. klient może ręcznie dodać ofertę pracy
10. każda oferta pracy ma (link do oferty, nazwę stanowiska, nazwę firmy, zarobki (mogą być widełki))

#klient chce pobrać dostępne oferty ale musi być zalogowany

1.nie ma ofert na serwerze  
2.apka odpytuje zewnętrzną bazę i dodaje 0 ofert  
3.użytkownik próbuje się zalogować i otrzymuje brak autoryzacji 401  
4.użytkownik próbuje pobrać oferty i otrzymuje brak autoryzacji 401  
5.użytkownik nie posiada konta i chce się zarejestrować  
6.użytkownik wypełnia formularz rejestracji i go wysyła status 200  
7.użytkownik próbuje się zalogować , jeśli logowanie jest poprawne otrzymuje token status 200  
8.użytkownik próbuje pobrać oferty z poprawnym tokenem w bazie nie ma ofert otrzumuje 0 ofert   
9.w zewnętrznej bazie są 2 nowe oferty

10.apka odpytuje zewnętrzny serwer i dodaje nowe oferty  
11.Użytkownik próbuje pobrać nieistniejącą ofertę – otrzymuje 404 – not found   
12.Użytkownik próbuje pobrać istniejącą ofertę – otrzymuje ją z kodem 200  
13.użytkownik wysyła zapytanie o oferty i otrzymuje oferty z kodem 200  
14.użytkownik dodaje ofertę i otrzymuję ją w odpowiedzi z nadanym numerem id oraz otrzymuje status 201

15.uzytkownik usuwa dodaną ofertę i otrzymuje status 204 no content oraz oferta zostaje usunięta.  
16.wylogowanie ręczne lub auto.

TOKEN AUTORIZATOR

OFFER FACADE

USER LOGIN AND REGISTERING

USER