**Specyfikacja wymagań  
*System Tutor (Tutor.pl)***

**Mateusz Markiewicz**

**Bartosz Sobocki**

***Czwartek, 8:15***

Wprowadźmy następujące aliasy nazw poszczególnych użytkowników:

* Uczeń - użytkownik chcący pobierać konsultacje
* Tutor - użytkownik chcący udzielać konsultacji
* Użytkownik - zarówno Uczeń, jak i Tutor

**1. Wymagania funkcjonalne**

* użytkownik może założyć nowe konto, wybierając czy chce być uczniem, czy tutorem, podając wszystkie dane wymagane oraz wybrane dane opcjonalne
* uczeń może dodać nowe ogłoszenie o chęci pobierania konsultacji z danej dziedziny, w danym terminie, sprecyzować preferencje dotyczące ceny, lokalizacji, tutora (np. wiek, płeć)
* uczeń może przejrzeć sugerowanych tutorów, dopasowanych przez naszą aplikację do danego ogłoszenia ucznia
* uczeń może przejrzeć listę wszystkich tutorów, filtrując ich po cenie, terminie, lokalizacji, dziedzinie oraz pozycjonować ogłoszenia po cenie, ocenie tutora, odległości od jego lokalizacji
* uczeń może zarezerwować spotkanie z danym tutorem (wybranym z ww. list) w wybranym terminie, jeśli rezerwacja zostanie potwierdzona przez tutora uczeń może zobaczyć numer telefonu, mail oraz inne sposoby komunikacji z tutorem
* uczeń może zaakceptować propozycję konsultacji od danego tutora, po czym zobaczyć numer telefonu, mail oraz inne sposoby komunikacji z tutorem
* uczeń może ocenić tutora po przekroczeniu daty spotkania, które zaaranżował w aplikacji
* uczeń może dodać nową notatkę
* uczeń może przejrzeć listę wszystkich swoich notatek, sortując je po dziedzinie
* uczeń może przejrzeć listę pomocy naukowych, sortując je po dziedzinie oraz pozycjonując po czasie dodawania oraz ocenach użytkowników
* tutor może dodać nowe ogłoszenie o chęci udzielania konsultacji z danej dziedziny, w danym terminie, sprecyzować wymagania odnośnie ceny oraz lokalizacji
* tutor może przejrzeć sugerowanych uczniów, dopasowanych przez naszą aplikację do danego ogłoszenia tutora
* tutor może przejrzeć listę wszystkich uczniów, filtrując ich po cenie, terminie, lokalizacji, dziedzinie oraz pozycjonować ogłoszenia po cenie, odległości od jego lokalizacji
* tutor może zarezerwować spotkanie z danym uczniem (wybranym z ww. list) w wybranym terminie, jeśli rezerwacja zostanie potwierdzona przez ucznia tutor może zobaczyć numer telefonu, mail oraz inne sposoby komunikacji z uczniem
* tutor może zaakceptować propozycję konsultacji od danego ucznia, po czym zobaczyć numer telefonu, mail oraz inne sposoby komunikacji z uczniem
* tutor może dodać nową pomoc naukową i sprecyzować jej dziedzinę
* użytkownik może przeglądać swój terminarz, w którym widzi zaproponowane spotkania, spotkania które zostały mu zaproponowane oraz zaaranżowane spotkania
* użytkownik może dokupić dodatkowe możliwości proponowania spotkania

**2. Wymagania niefunkcjonalne**

* *szybki czas działania*

System powinien być zoptymalizowany pod względem złożoności obliczeniowej algorytmów i zastosowanych rozwiązań, aby korzystanie z niego nie wiązało się z długim czasem oczekiwania.

* *intuicyjny interfejs wersji mobilnej aplikacji oraz webowej*  
  Użytkownik powinien z łatwością posługiwać się aplikacją, szybko znajdować interesującego go funkcjonalności
* *bezpieczeństwo i anonimowość*  
  Uczeń i Student pozostają anonimowi   
  Ich dane osobowe nie są w żaden sposób przetwarzane ani rozpowszechniane, użytkownicy posiadają tylko i wyłącznie identyfikatory.
* *bezawaryjność*System musi działać poprawnie niezależnie od czasu i okoliczności. Niedopuszczalna jest sytuacja, kiedy np. podczas płatności ucznia lub organizacji nauki wystąpi błąd. Może to wiązać się z konsekwencjami.
* *migracja systemu na inny serwer musi być możliwa.* Instrukcja musi zawierać opis przebiegu takiej migracji.
* *spójność treści pomiędzy aplikacją mobilną i webową*Użytkownicy, którzy dobrze już zapoznali się z jedną z wersji obsługi systemu powinni z łatwością odnaleźć się w drugiej. Obsługa obu wersji nie powinna być różna.