Projekt Relacyjnej Bazy danych

System zarządzania miejscami w akademiku

Filip Maciejak (260375)

Piotr Grygoruk (260299)

Jakub Szwedowicz (243416)

Spis treści

[1. Podsumowanie projektu 3](#_Toc120136818)

[1.1. Analiza projektu 3](#_Toc120136819)

[1.1.1. Początkowe założenia 3](#_Toc120136820)

[1.1.2. Początkowe ograniczenia 3](#_Toc120136821)

[1.2. Kierunki rozwoju 3](#_Toc120136822)

# Podsumowanie projektu

## Analiza projektu

### Mocne strony projektu

Pierwotnie ustalone założenia projektu zostały częściowo zrealizowane. Podstawowe funkcjonalności systemu takie jak przeglądanie wniosków, wolnych i zajętych pokojów, a także filtrowanie wyników bazy danych zostały wykonane poprawnie. System znajduje się na działającym całodobowo serwerze i udostępnia możliwość zdalnej obsługi akademików. Ponad to zgodnie z początkowym założeniem został wykonany w czasie 2 miesięcy.

O wydajności systemu można powiedzieć, że część zapytań skierowanych do bazy danych została zoptymalizowana, a zaprojektowana struktura bazy danych pozwala na szybki dostęp do przechowywanych informacji.

### Słabe strony projektu

Do niezrealizowanych początkowych założeń można zaliczyć dostęp do systemu przez stronę internetową. Choć serwer udostępnia zdalną możliwość połączenia się z bazą danych to wymaga to specjalistycznej wiedzy. Podobnie sama interakcja z bazą danych wymaga znajomości wiersza poleceń. Jest to zatem interfejs nieprzystosowany do odbiorcy końcowego (internauty, zarządcy budynków, uczelni).

Należy również zaznaczyć, że niespełnione zostały niektóre wymagania funkcjonalne przewidziane dla użytkowników. Takie funkcjonalności jak logowanie na konto a także podział użytkowników systemu na role (administrator, pracownik, lokator) nie zostały zrealizowane. Dostęp do bazy na ten moment odbywa się przy pomocy jednego konta z pełnymi uprawnieniami, co ze względów bezpieczeństwa uniemożliwia udostępnienie dostępu jakiemukolwiek użytkownikowi końcowemu.

Ostatnią istotną wadą systemu jest wysoki czas realizacji niektórych zapytań, których nie udało się bardziej zoptymalizować. W przypadku wzrostu liczby jednoczesnych zapytań do systemu może to doprowadzić do jego przeciążenia, a więc do problemów, a nawet do braku możliwości użytkowania.

### Szanse na rozwój projektu

Do tego momentu wszystkie nasze działania miały na celu zaimplementowanie potrzebnych funkcjonalności. Choć w projekcie można wskazać niezrealizowane części, to pierwotna wizja systemu oraz jego realizacja są na dobrej drodze. Dotychczas wykonane działania pozwalają w dalszym ciągu rozwijać system oraz rozszerzyć jego możliwości o resztę funkcjonalności zakładanych w wizji.

Takie funkcjonalności jak dostęp przez stronę internetową oraz utworzenie ról użytkowników bazy danych wciąż mogą zostać dodane. Zapytania mogą również być dalej optymalizowane, a z racji na dotychczasowe oszczędne podejście do nadmiarowych danych – redundancję systemu wciąż można zwiększyć w celu skrócenia czasu realizacji zapytań.

### Zagrożenia dla projektu

Główne zagrożenia jakie można wymienić to przede wszystkim obecność już istniejących, dojrzałych rozwiązań na rynku oraz dość wąskie grono podmiotów, które mogłyby być zainteresowane wprowadzeniem systemu.

Dodatkowo, potencjalny klient musiałby zainwestować we własną infrastrukturę serwerową, która ze względu na przeznaczenie systemu, jakim jest obsługa dziesiątek tysięcy użytkowników jednocześnie, oznacza wysoki nakład początkowy.

## Kierunki rozwoju

Tak, jak zostało wspomniane wcześniej, rozwinięcie projektu będzie polegać na zaimplementowaniu reszty funkcji opisanych w wizji – głównie strony internetowej. Będzie to interfejs udostępniony użytkownikom końcowym. System logowania będzie czuwał nad tym, żeby jedynymi zmianami w bazie wywoływanymi przez użytkowników były te, na które pozwalają im ich uprawnienia. Skończona strona internetowa będzie służyła użytkownikom do wszystkich operacji jakie powinni być w stanie wykonać, czyli użytkownikom końcowym nie będzie potrzebny dostęp do bazy w żaden inny sposób. Innym kierunkiem rozwoju będzie dalsza optymalizacja zapytań. Zostanie to osiągnięte wprowadzając dodatkową denormalizację tam, gdzie warte jest to podjęcia ryzyka niespójności danych. Kolejnym kierunkiem rozwoju będzie obsługiwanie komunikacji między systemami – na przykład, gdy dwie uczelnie mają swoje systemy zarządzania akademikami, jest duża szansa na to, że będą chciały kooperować i udostępniać niektóre informacje drugiemu systemowi. Zostanie to zrealizowane za pomocą wprowadzenia API (z zasadami projektowania REST) z którego będą mogły korzystać inne systemy naszych innych klientów.

### Wprowadzona modyfikacja systemu

W początkowej wersji projektu przechowywaliśmy wszelkie statusy (wynajmu, należności, płatności itd.) jako ciągi znaków – np. „Aktywny”. Rozwiązanie to oczywiście działa, ale wprowadziliśmy dodatkowe wymagania, które sprawiły, że musieliśmy to zmienić. Wymaganiem było to, żeby dane w bazie nie były powielane, co ma związek z normalizacją do postaci normalnych. W obliczu takiego wymagania, wprowadziliśmy od każdego statusu enumerację wszystkich możliwych wartości w osobnej encji. Pomaga to nie tylko w uniknięciu dublowania danych, ale także ułatwia późniejsze wprowadzanie większej ilości różnych statusów.