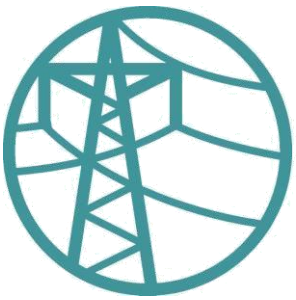

WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI
POLITECHNIKI RZESZOWSKIEJ

Post Mortem

Bramka SMS

Techniki multimedialne

Stefan Dyjak

Rzeszów, 2018

Spis treści

1. Podsumowanie projektu

Realizację projektu można uznać za zakończoną sukcesem mimo braku funkcjonalności, które pojawiły się w określonych założeniach. Zabrakło także niektórych pomysłów które zostały wymyślone już w trakcie rozwoju aplikacji. Udało się zrealizować podstawowe założenia projektu, takie jak: aplikacja mobilna działająca w tle i wysyłająca smsy, aplikacja serwerowa wraz z bazą danych do przechowywania smsów do wysłania.

2. Elementy, które się udały

Elementy, które zostały przyjęte w założeniach i udało się je zaimplementować w wyznaczonym czasie.

2.1. Architektura

Aplikacja mobilna została napisana z zastosowaniem wzorca projektowego MVVM, który pozwala na łatwe rozwijanie aplikacji. Nowe elementy dodane do aplikacji nie będą wymagały zmiany już istniejących elementów lub tylko zmiany w minimalnym stopniu tak aby połączyć te elementy.

Aplikacja serwerowa została napisana w oparciu o wzorzec projektowy MVC z pominięciem części odpowiadającej za prezentacje ponieważ w tym projekcie nie jest ona istotna. Zastosowanie tego wzorca również ułatwia dalszy rozwój aplikacji. Zastosowano także różne interfejsy, które są implementowane przez odpowiednie klasy. Powaliło to na zastosowanie mechanizmów refleksji oraz wstrzykiwania zależności, co również ułatwia dalszy rozwój aplikacji.

2.2. Baza danych

W projekcie jako bazę danych wykorzystano MongoDB, która jest nierelacyjną bazą danych. Dzięki temu jest możliwość przechowywania smsów dodanych przez użytkownika. Użytkownik przekazuje dane do bazy danych poprzez aplikacje serwerową.

2.3. REST

Do komunikacji między aplikacją mobilną i aplikacją serwerową zaimplementowano styl architektoniczny o nazwie Representational State Transfer (REST). Wykorzystuje on m. in. jednorodny interfejs, bezstanową komunikację, zasoby, reprezentacje. Implementacja tego stylu sprawia że klient może pobierać lub modyfikować zasoby na serwerze korzystając ze ścieżek oferowanych przez serwer. Klient wysyła zdefiniowane żądanie HTTP, które określa typ akcji. Natomiast serwer odbierając to żądanie, przetwarza je oraz wysyła zdefiniowaną odpowiedź. Zastosowanie tego stylu pozwala na oddzielenie aplikacji klienta od serwera. Zmiana jednego z nich nie musi się wiązać ze zmianą drugiego.

2.4. Wysyłanie smsów w tle z aplikacji mobilnej

W obecnym stanie aplikacja pozwala na dodawanie i pobieranie smsów z serwera. Pobieranie smsów odbywa się w oddzielnym wątku aplikacji który działa w tle. Po odebraniu smsów aplikacja również w tle wysyła te smsy, a następnie wysyła żądanie na serwer aby usunąć sms który został już wysłany.

Pobieranie smsów w aplikacji z serwera jest możliwe dzięki zastosowanego stylu RESTowego. Na początku kłopoty sprawiało połączenie się z emulatora lub urządzenia mobilnego do serwera zainstalowanego lokalnie na komputerze. Problem wynikał z tego że serwer IIS Express na którym działała aplikacja serwerowa, domyślnie sprawiał że była ona widoczna tylko lokalnie. Rozwiązaniem problemu okazało się zainstalowanie rozszerzenia do Visual Studio 2017 Community o nazwie Conveyor, który konfiguruje serwer tak że aplikacja jest widoczna z innych urządzeń wewnątrz tej samej sieci.

Z wysyłaniem smsów w tle również początkowo były problemy. Było to spowodowane tym że pomimo dodaniu do pliku konfiguracyjnego AndroidManifest.xml, wymaganych uprawnień jakich potrzebuje aplikacja, nie były one przyznawane. Przyczyną tego było to że od wersji systemu Android 6.0 (SDK Level 23) uprawnienia które są uznawane za niebezpieczne są przyznawane przez użytkownika w trakcie działania aplikacji. W tym projekcie uprawnieniem niebezpiecznym jest uprawnienie do wysyłania smsów. Rozwiązaniem tego problemu było dopisanie do programu kodu, który przy uruchomieniu aplikacji pytał użytkownika o przyznanie uprawnień.

3. Elementy, które się nie udały

Elementy, które były ujęte w założeniach projektu ale nie udało się ich zaimplementować ze względu na brak czasu czy też wiedzy.

3.1. Więcej możliwości interfejsu użytkownika

W obecnej aplikacji użytkownik może tylko dodawać smsy, wyświetlać je oraz wysyłać. Natomiast brakuje aby mógł usunąć lub edytować wybrane smsy zanim zostaną one wysłane. Także brakuje możliwości wyłączenia wysyłania smsów w tle, co sprawia że jeśli chcemy przestać wysyłać smsy to trzeba wyłączyć aplikacje, nie wystarczy jej tylko zminimalizować.

3.2. Dodanie możliwości wysłania smsów o konkretnej godzinie

Jednym z przyjętych założeń było to że użytkownik przy tworzeniu smsa podaje datę kiedy ma zostać on wysłany, lub najwcześniej jak tylko to możliwe po tej dacie. Nawet w bazie danych została dodana

kolumna aby przechowywać ten rodzaj danych. Ale nie zostało to zaimplementowane w aplikacji mobilnej ani w aplikacji serwerowej.

4. Dalszy rozwój aplikacji

Zastosowane mechanizmy oraz wzorce projektowe ułatwiają dalsze rozwijanie aplikacji. Pierwszym krokiem w rozwoju aplikacji mogłoby być zwiększenie możliwości kontroli nad smsami dla użytkownika. Na przykład dodanie możliwości usuwania czy edytowania smsów oraz możliwość wyłączenia wysyłania smsów w tle. Kolejnym krokiem mogłoby być dodanie możliwości wysyłania smsów o ustalonej godzinie.