|  |  |  |
| --- | --- | --- |
| Imie i nazwisko:  Paweł Szmytke | Rok I st. II, Gr IV | Ocena: |
| **WIMiIP ROK I st. II - INFORMATYKA STOSOWANA** | | Data: |
| **Systemy Równoległe i Rozproszone Projekt – Web Service pobierający dane personalne z SkOs AGH** | | Prowadzący:  Mgr. Inż. Kazimierz Michalik |

1. Początkowy opis projektu

Projekt jest wstępną implementacją jednego z modułów przyszłej pracy magisterskiej która będzie miała za zadanie rozpoznawać twarz osoby i wyszukiwać informacje ta jej temat na stronie SkOs AGH.

Moduł który został zaimplementowany ma za zadanie wyszukiwać informacje o danej osobie na podstawie imienia i nazwiska z wykorzystaniem WS wystawionego na serwer internetowy.

1. Opis realizacji poszczególnych etapów

WS został napisany w języku C# i wystawiony na 2 niezależne darmowe serwery :

<http://webserviceskosminer.gear.host/WebService.asmx>

<http://skosminer-001-site1.mywindowshosting.com/WebService.asmx>

Po zapytaniu serwer sprawdza czy w ciągu ostatniej minuty była wyszukiwana taka osoba i jeśli tak to zwraca natychmiast wynik, jeśli nie to przeszukuje SkOs i zwraca odpowiedni wynik, ewentualnie zwraca wynik „Not found” jeśli nie odnajdzie takiej osoby.

Klient na urządzenia z systemem Android posiada zabezpieczenie przed wysłaniem pustego imienia i nazwiska, oraz zabezpieczenie przed błędami serwera w postaci odpytania serwera awaryjnego. W razie błędów wyświetla odpowiedni komunikat.

Przeszukiwanie SkOs polega na sprawdzeniu pierwszej litery nazwiska i odpytaniu stronę o listę osób których nazwiska rozpoczynają się na tą literę. Następnie strona jest rozparsowana, podzielona na sekcje i przeszukiwana w celu odnalezienia odnośnika do informacji o danej osobie. Jeśli odnośnik zostanie odnaleziony zostaje zwrócone źródło strony z informacjami personalnymi. Następnie to źródło jest rozparsowane, podzielone na odpowiednie sekcje i przeszukiwane w poszukiwaniu poszczególnych informacji. Finalnie informacje są zbierane do jednej zmiennej i zwracane.

Dokumentacja została wygenerowana dla obu programów w formacie Doxygen.

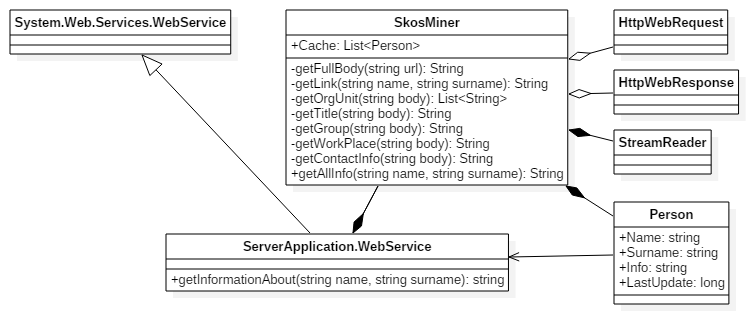
Całość kodów i dokumentacji znajduje się na serwerach GutHub.com, projekty były wykonywane przy użyciu Systemu Kontroli Wersji Git :

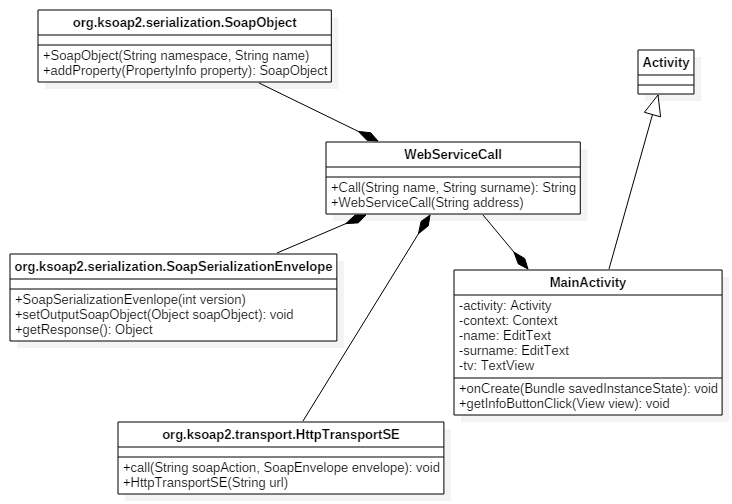
<https://github.com/Szmytek1991/SkosMinerWebServiceServerGit>

<https://github.com/Szmytek1991/SkosMinerWebServiceClientAndroid>

Diagramy UML Klas dla aplikacji serwerowej, klas dla aplikacji klienckiej oraz przypadków użycia zostały wykonane przy użyciu narzędzia StarUML 2.3 i załączone do dokumentacji.

1. Cechy rozproszone systemu :
2. Współbieżność – **Sprawdzono działanie systemu na 2 urządzeniach jednocześnie**, nie spowodowało to spowolnienia wykonywania operacji
3. Współdzielenie danych – W przypadku częstego odpytywania o tą samą osobę nie ma sensu przeszukiwać za każdym razem strony internetowej ponieważ zapytania trwają dość długo dlatego informacje są zbierane na listę i jeśli są wystarczająco aktualne są zwracane natychmiast. **System współdzieli listę osób tymczasowych**
4. Tolerowanie awarii – w razie błędu podczas przetwarzania danych **zwracany jest błąd, system działa dalej**, w razie awarii serwera odpytywany jest **serwer alternatywny.**
5. Skalowalność - ? należałoby sprawdzić działanie dla większej ilości urządzeń jednocześnie
6. Diagramy
7. Diagram klas – Serwer



1. Diagram klas – Klient
2. Diagram przypadków użycia

