**Wybrany algorytm:**

* Transit node routing

Uzasadnienie:

Prosty koncepcyjnie algorytm będący „nadbudówką” nad dowolnym algorytmem pozwalających na szukanie tras w grafie. Jest złotym środkiem pod kątem zużycia pamięci pomiędzy Hub Labellingiem i HPA\*.

**Wybrane narzędzia:**

* JGraphT
* Spark (preprocessing itd.)

Uzasadnienie:  
Wg. decyzji podjętym na poprzednim spotkaniu algorytm zostanie zaimplementowany jako rozszerzenie do biblioteki JGraphT będącej interfejsem pomiędzy formatem danych specyficznym dla rozważanego problemu a grupą algorytmów operujących na tych danych. Pozwoli odseparować główny projekt od implementacji tego algorytmu a jednocześnie w prosty sposób pozwoli na finalne zintegrowanie go z algorytmem. Zaproponowana biblioteka Spark niekoniecznie zostanie wykorzystana ale ze względu na fakt dużego rozmiaru grafów jest na to duże prawdopodobieństwo.

**Uruchomione narzędzia:**

Uruchomiłem przykłady wykorzystania JGraphT w celu głębszego zapoznania się z biblioteką.

**Zakres prac:**

* Implementacja algorytmu transit node routing dla generycznego grafu przedstawionego przez odpowiednie interfejsy biblioteki JGraphT
* Wyeksportowanie algorytmu jako dodatek do biblioteki JGraphT
* Implementacja interfejsów dla reprezentacji danych problemu wymaganych do wykorzystania JGraphT
* Testy poprawności, oraz ewentualne poprawki wydajnościowe