Raport 2

Mikołaj Zawada 259431

Tomasz Płuciennik

Projekt wykonany został przy pomocy Visual Studio Code 2022 MAC, APS.NET CORE 6.

**Pierwsze kroki wykonania projektu są podobne do tych z raportu 1. Należy:**

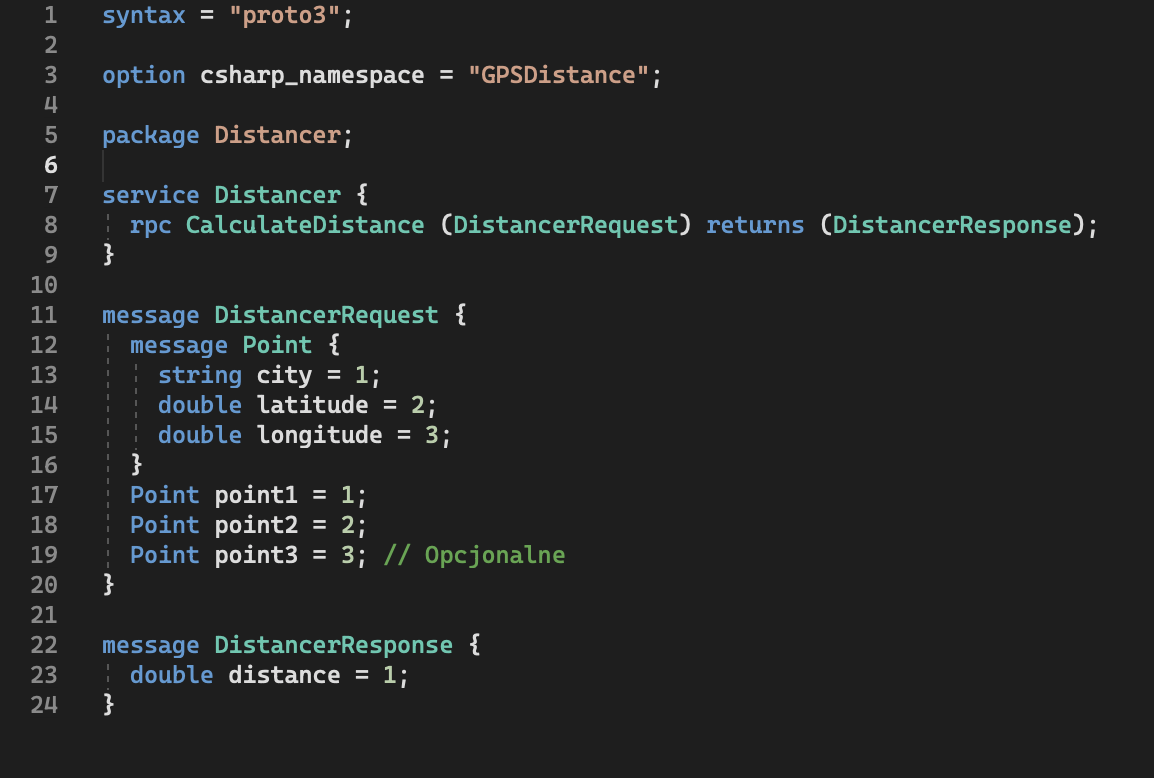
1. Zainstalowanie wymaganego IDE oraz ASP.NET CORE 6.
2. Uruchomić Visual Studio dla komputerów Mac.
3. Stworzyć nowy projekt w zakładce Web and Console > App > gRPC service.
4. Wybrać platformę .NET 6.0 dla platformy docelowej i kliknąć kontynuuj
5. Nazwać projekt GPSDistance
6. Aby program działał, jak należy, wymagane jest zaufanie certyfikatom przy pomocy komendy: dotnet dev-certs https –trust.
7. W aplikacji GPSDistance należy stworzyć nowy projekt o nazwie GPSDistanceClient wykonując wcześniej podane kroki.

**Kolejne kroki dotyczą już konkretnie implementacji wymogów zadania:**

1. W projekcie GPSDistance i GPSDistanceClient stworzyć klasę MyData oraz ustawić wywołanie jej statycznej metody .info() jako pierwszą operację w plikach Program.cs dla obu projektów.

**Implementacja serwera:**

1. Należy stworzyć plik Distancer.proto. Na wzór



1. Aby program odpowiednio działał w pliku GPSDistance.csproj należy podać:



1. Następnie należy stworzyć DistancerService.cs dokonujący obliczeń przy pomocy funkcji haversine.

**Implementacja klient:**

1. Należy stworzyć plik Proto w programie klienta na wzór pliku Distancer.proto w projekcie GPSDistance.
2. Należy stworzyć program w pliku Program.cs który udostępni nam interfejs do komunikacji z serwerem.

Text

Description automatically generated

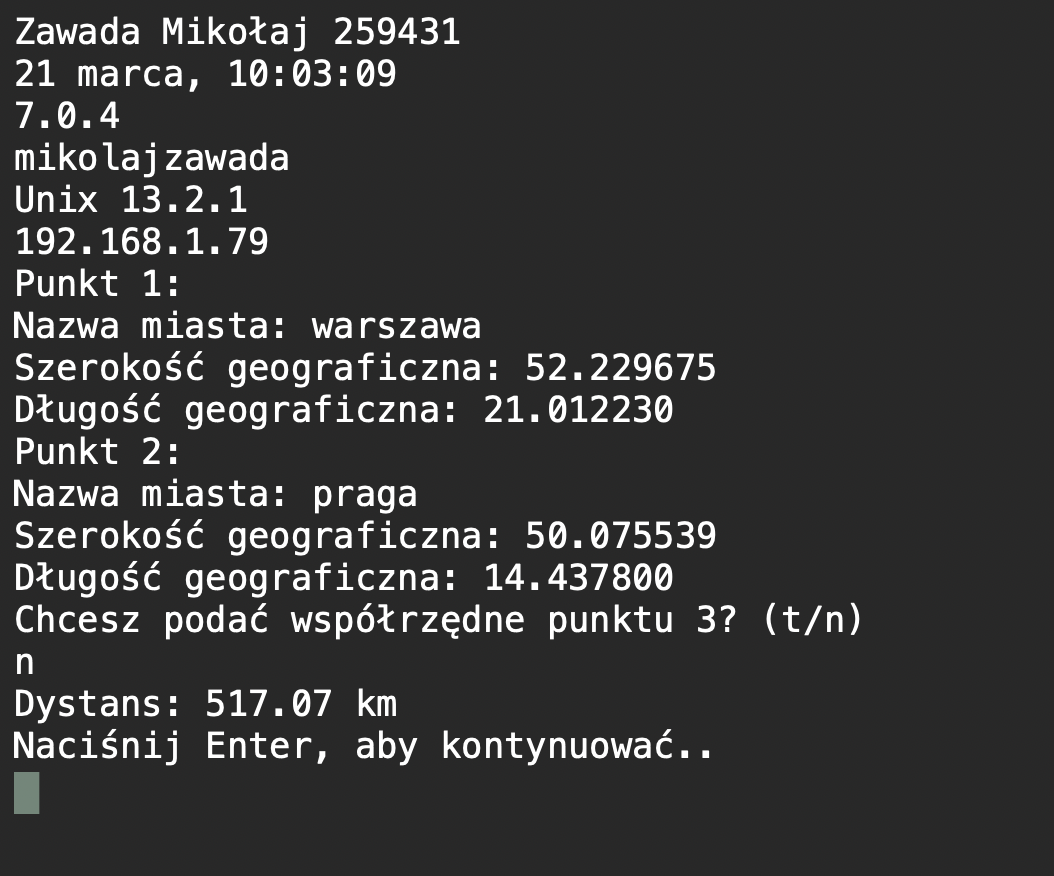
1. W celu wyzbycia się redundancji stworzyć należy osobną metodę sczytującą dane dla konkretnego punktu.

Text

Description automatically generated

**Wyniki działania programu:**

W wersji dla dwóch punktów:



W wersji dla 3 punktów:

Text

Description automatically generated