

Wizualizacja danych sensorycznych - projekt

1. Temat projektu:

Wizualizacja pogody w Japonii.

2. Wykonawca:

Filip Malinowski 209193

3. Opis projektu:

Projektowany program będzie pobierać pogodę z internetowych serwisów pogodowych. Dane będzie aktualizował dynamicznie co okres czasu wybrany przez użytkownika, np. co 1 minutę, 1 godzinę, 2 godziny. Informacje o pogodzie będą wyświetlane na mapie.

Interaktywność mapy będzie polegać na tym, że domyślnie dla każdej stolicy prefektury będzie wyświetlana temperatura i ikona zachmurzenia. Po najechaniu kursorem na nazwę miasta rozwinie się szczegółowa pogoda zawierająca: temperaturę, szansę opadów, ciśnienie, wilgotność, zachmurzenie, prędkość i kierunek wiatru oraz poziom promieniowania UV. Mapę będzie można również przesuwac oraz zmieniać jej powiększenie.

W programie będzie również druga zakładka, w której użytkownik będzie mógł wyszukać miasto i wyświetlić dla niego szczegółowe informacje na temat pogody. Wyświetlona zostanie prognoza godzinowa wszystkich czynników podanych wcześniej do wyświetlania na mapie oraz odpowiednie wykresy.

Ma być też możliwość dodania nowych zakładek za pomocą zakładki +. Wtedy jej konfiguracja będzie przebiegać tak samo jak dla drugiej zakładki. Schemat jej zawartości będzie taki sam jak w przypadku drugiej zakładki. Dodatkowe zakładki będą umożliwiać skasowanie się.

4. Źródła pogodowe: Dane pogodowe są pobierane z serwisu developer.forecast.io. Serwis ten pozwala na 1000 darmowych zapytań dziennie ale przy tym udostępniając bardzo dużo danych pogodowych. Dane z tego serwisu pochodzą z: USA NOAA's NEXRAD system, UK Met Office's NIMROD system, The USA NOAA's LAMP system, UK Met Office's Datapoint API, Norwegian Meteorological Institute's meteorological forecast API, USA NOAA's Global Forecast System, USA NOAA's Integrated Surface Database, USA NOAA's Public Alert system, UK Met Office's Severe Weather Warning system, Environment Canada's Canadian Meteorological Center ensemble model, US Navy's Fleet Numerical Meteorology and Oceanography Ensemble Forecast System, USA NOAA and Environment Canada's North American Ensemble Forecast System, USA NOAA's North American Mesoscale Model, USA NOAA's Rapid Refresh Model, Norwegian Meteorological Institute's GRIB file forecast for Central Europe, Norwegian Meteorological Institute's GRIB file forecast for Northern Europe, Worldwide METAR weather reports, USA NOAA/NCEP's Short-Range Ensemble Forecast, USA NOAA/NCEP's Real-Time Mesoscale Analysis model, USA NOAA/ESRL's Meteorological Assimilation Data Ingest System.

Dane ze wszystkich powyżej wymienionych serwisów są następnie uśredniane i przesyłane do użytkownika na zapytanie wysłane do `developer.forecast.io`. Serwis ten korzysta z bezpiecznego połączenia szyfrowanego protokołem SSLv2.

5. Zrealizowane funkcjonalności:

- Widżet Mapa z mapą Japonii jako tło;
- Obiekt Miasto pobierający i przechowujący dane pogodowe.

6. Funkcjonalności do zrealizowania:

- Chmurki umieszczone na mapie wyświetlające ikonę pogody oraz temperaturę;
- Zakładka przechowująca obiekt klasy Miasto i wyświetlająca wykresy;
- Belka z ustawieniami dająca możliwość ustawienia okresu odświeżania danych.

7. Opis zaimplementowanych funkcjonalności:

- Widżet Mapa Przechowuje plik png z mapą Japonii oraz wskaźnik na obiekt klasy QLabel. Wykorzystuje obiekt QLabel w celu ustawienia swojego tła jako mapę Japonii.

Docelowo widżet ten ma być skalowalny i przesuwany, tj. posiadać dwa przyciski: plus, minus do powiększania i pomniejszania mapy; oraz dwie belki z suwakami, które będą przesuwać powiększoną mapę. Jeśli mapa mieści się w całości w oknie to belki z suwakami mają zniknąć. Przy przeskalowywaniu okna programu mapa również ma się przeskalowywać. Na mapie mają być wyświetlane chmurki przy większych miastach Japonii pokazujące temperaturę i ikonę zachmurzenia w tych miejscach. Po najechnięciu na ikonę miasta chmurka ma się rozwinąć przedstawiając szczegółowe informacje nt tego miejsca.

- Obiekt Miasto Obiekt ten modeluje jedno miasto. Przechowuje koordynaty miasta oraz dane pogodowe na temat tego miasta w formacie json. Łączy się z serwisem internetowym `developer.forecast.io` pobierając z niego dane pogodowe. Następnie konwertuje odebrane dane do formatu obsługiwanego przez klasę składową obiektu Miasto, `QJsonObject`. W obiekcie Miasto są metody umożliwiające dostęp do fragmentów klasy `QJsonObject` wycinając z niego opis pogody, dane pogodowe, itp.

Docelowo obiekt ten ma być rozszerzony o wyszukiwanie danych pogodowych na podstawie nazwy miasta. Teraz istnieje jedynie możliwość wyszukiwania danych pogodowych na podstawie koordynatów.

- Widżet OknoZZakładkami Widżet ten modeluje fragment głównego okna programu będący widżetem z zakładkami. Ma możliwość zmiany swojego rozmiaru wraz ze zmianą rozmiaru głównego okna programu.

Docelowo ma się tworzyć z dwiema zakładkami: mapą oraz zakładką miasta, która wyświetli zapytanie o nazwę miasta w celu wyświetlenia danych pogodowych. Będzie jeszcze trzecia zakładka ze znakiem + dodająca kolejne zakładki z miastami. Po dodaniu nowej zakładki z miastem trzeba będzie ustawić miasto tej zakładki. Po tym zostaną pobrane informacje. Skasować będzie można tylko dodane zakładki. Dwóch pierwszych nie będzie dało się skasować. Ilość zakładek w sumie nie będzie mogła być większa niż 5.

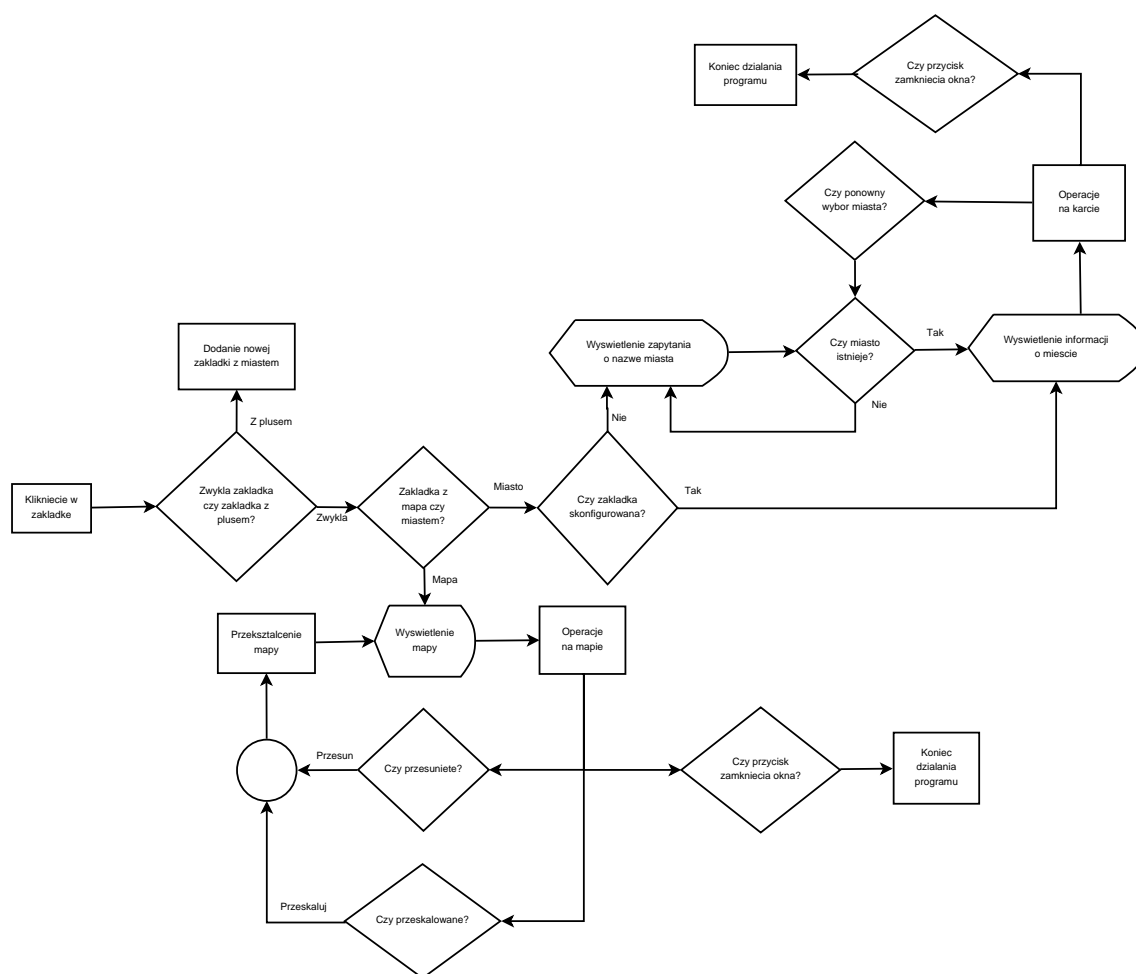
8. Opis funkcjonalności do implementacji:

- Widżet ZakładkaMiasta Widżet ten będzie wyświetlać aktualną temperaturę, ciśnienie i wilgotność w pierwszym rzędzie. W drugim rzędzie będzie wyświetlać wykres zmiany temperatury w przeciągu 48 godzin. W trzecim rzędzie będzie wyświetlać wykres zmiany temperatury w przeciągu 7 dni.

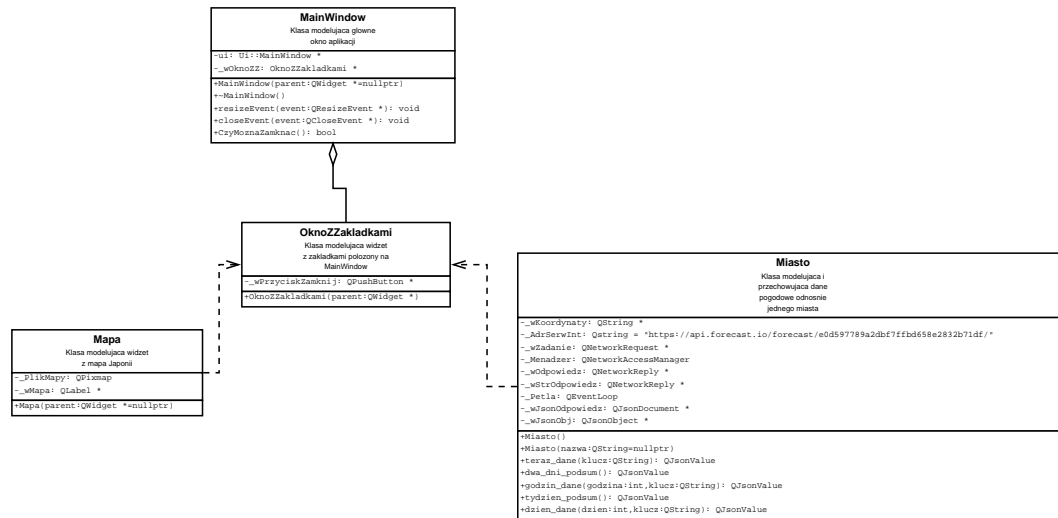
9. Harmonogram:

- 9 kwietnia - zrealizowane** — przeprowadzenie badań na temat najlepszych źródeł informacji oraz odpowiedniej struktury programu;
- 22 kwietnia - zrealizowane** — napisanie wersji alpha programu realizującej podstawowe funkcje programu. Podstawowymi funkcjami do realizacji są: wyświetlenie mapy Japonii, pobranie danych z serwisów pogodowych i zapisanie ich w klasie danego miasta;
- 23 kwietnia - zrealizowane** — podsumowanie wstępnej wersji programu oraz ewentualne poprawki w planach projektu;
- 29 kwietnia - zrealizowane** — przekazanie wstępnej wersji programu do oceny;
- 6 maja - w trakcie** — dodanie dwóch kolejnych funkcjonalności: zakładki z obiektem Miasto oraz wyszukiwanie koordynatów miasta na podstawie podanej nazwy.

10. Diagram przepływu sterowania:



11. Diagram klas:



12. Zarządzanie projektem:

- Git - do przechowywania i rozwijania dokumentacji oraz oprogramowania związanego z projektem. Adres internetowy repozytorium.
- LaTeX - do tworzenia dokumentacji projektu