

WYDZIAŁ MATEMATYKI I NAUK INFORMACYJNYCH
POLITECHNIKI WARSZAWSKIEJ

Hurtownie danych

Dokumentacja projektowa

Hurtownia danych dla firmy
zarządzającej rowerami miejskimi
w Nowym Jorku

Szymon Rećko
Mikołaj Spytek

14 czerwca 2022

Spis treści

1	Opis biznesowy projektu	3
1.1	Uzasadnienie potrzeby powstania	3
1.2	Sposób generowania zysku dla użytkownika końcowego	3
1.3	Źródła analizowanych danych	4
2	Opis źródła danych	4
3	Opis rozwiązania	5
3.1	Warstwy modelu	5
3.2	Warstwa ETL	6
3.3	Ostateczny model hurtowni	9
3.4	Warstwa raportowa	9
3.4.1	Przykładowe raporty	9
4	Testy funkcjonalności	10
4.1	Test skryptu generującego hurtownię	10
4.2	Testy procesu ETL	11
4.2.1	Populowanie tabel wymiarowych	11
4.2.2	Populowanie tabeli faktowej	11
4.2.3	Dodawanie wierszy do tabel wymiarowych w cyklicznych aktualizacjach	12
4.2.4	Dodawanie wierszy do tabeli faktowej w cyklicznych aktualizacjach	13
4.3	Testy warstwy raportowej	14
5	Podsumowanie	16
6	Podział pracy w zespole	16

1 Opis biznesowy projektu

1.1 Uzasadnienie potrzeby powstania

Hurtownia danych i system Business Intelligence służący do prezentacji raportów to jedno z najlepszych narzędzi wspomagających podejmowanie strategicznych decyzji biznesowych.

Dzięki zebraniu i uspołnieniu danych z różnych źródeł w bardzo prosty sposób można stworzyć raport i wizualizację która pomoże osobom decyzyjnym w dostrzeżeniu problemów w obecnym działaniu organizacji, czy też w podjęciu decyzji w jaki sposób rozwijać, czy też modyfikować działanie firmy.

Ten sposób przechowywania danych jest również przystosowany do wprowadzania zmian w przyszłości, ponieważ system jest podzielony na warstwy. Zmiana w formacie źródłowym jednego zbioru wejściowego nie powoduje konieczności zmian całej hurtowni danych, lecz jedynie poprawkę w warstwie ETL. Podobnie w prosty sposób można zmienić dostawcę oprogramowania używanego w warstwie raportowania jeśli taka potrzeba zajdzie.

Projekt ten powstaje, aby w intuicyjny sposób umożliwić pracownikom organizacji analizę danych (które i tak są już zbierane, w celu zapewnienia poprawności działania systemu) i wyciąganie z nich wniosków mających wpływ na zyski, koszty utrzymania biznesu lub ważne decyzje biznesowe.

1.2 Sposób generowania zysku dla użytkownika końcowego

Końcowy użytkownik systemu będzie miał dostęp do raportów dotyczących sposobu korzystania klientów z dostarczanej przez firmę usługi. Dzięki temu będzie mógł lepiej zrozumieć ich motywacje, oraz lepiej dopasować kierunki rozwoju firmy, oraz inne podejmowane decyzje. Poniżej zaprezentowano kilka przykładowych raportów, oraz wyjaśnienie, dlaczego mogą być one przydatne:

- Jaka jest liczba kilometrów przejechanych przez dany rower? (w postaci wykresu słupkowego dla 100 rowerów o największej przejechanej liczbie kilometrów od serwisowania) – pozwala stwierdzić, które pojazdy należy serwisować w pierwszej kolejności
- Z których stacji wypożycza się najczęściej rowerów? Czy zmienia się to z godziną wypożyczenia? Być może z dniem tygodnia? Lub pogodą? (w postaci hierarchicznej mapy, z możliwością filtrowania po okresie, pogodzie, dniu tygodnia) – pozwala podjąć decyzję dokąd transportować rowery i w jakich ilościach, tak aby ich nie brakowało
- Jaki jest średni czas wypożyczenia roweru? Czy zależy od lokalizacji? Typu użytkownika – pozwala na szacowanie liczby rowerów, które są używane w danej chwili, co jest potrzebne do obliczenia wolumenu zakupu nowych pojazdów.
- W jakich godzinach użytkownicy korzystają z rowerów (w postaci wykresu słupkowego) – pozwala szacować, kiedy można zabrać rowery do serwisu.
- Jak zmienia się ilość wypożyczeń na przestrzeni tygodnia/miesiąca (np. w postaci wykresu kaskadowego) – pozwala stwierdzić, jak bardzo święta i okresy wolne od pracy wpływają na zainteresowanie wypożyczeniem rowerów (wprowadzenie promocji, jeśli zainteresowanie spada itd.)
- Jakie odległości pokonują użytkownicy w zależności od typu roweru? (wykres słupkowy) – pomaga w podjęciu decyzji, czy należy powiększyć flotę rowerów elektrycznych.

1.3 Źródła analizowanych danych

W ramach projektu analizujemy wystawione publicznie dane raportowe dotyczące wypożyczeń rowerów. Dane te są udostępniane po każdym miesiącu kalendarzowym. Jeden wiersz odpowiada jednokrotnemu wypożyczeniu roweru przez jednego użytkownika. Znajdują się w nim takie dane jak:

- stacja początkowa i końcowa wypożyczenia,
- godzina rozpoczęcia i zakończenia wypożyczenia,
- informacje o użytkowniku (płeć, rok urodzenia, typ: subskrybent/wypożyczenie okazjonalne)
- identyfikator roweru

W przypadku, gdyby rozwiązanie powstawało w porozumieniu z klientem, dane mogłyby być odświeżane częściej niż co miesiąc z transakcyjnej bazy danych (na przykład co tydzień, ponieważ miesiąc w sezonie rowerowym to długi okres, co sprawia, że jest mało czasu na decyzje). Wtedy również rozwiązalibyśmy problem braku przypisania wypożyczenia do konkretnego użytkownika - w danych publicznych ta informacja musiała zostać usunięta, w celu zachowania anonimowości klientów, ale w hurtowni danych pozwoliłaby na jeszcze dogłębsze analizy – na przykład, czy ktoś nie korzysta z systemu w sposób niezgodny z regulaminem, czy też opracowanie zniżek dla użytkowników korzystających z systemu w sposób regularny.

Drugim źródłem informacji jest API zawierające historyczne dane pogodowe. Dzięki niemu uzyskujemy informacje jak warunki pogodowe wpływają na korzystanie z systemu przez użytkowników. Dodatkowo używając API geograficznego (reverse geocoding) jesteśmy w stanie dodać atrybuty dla stacji, co pozwoli lepiej zrozumieć na wypożyczenia rowerów wpływa dzielnica wypożyczenia, czy też sąsiedztwo.

2 Opis źródła danych

Dane z pierwszego źródła to pobrane z sieci pliki płaskie (<https://citibikenyc.com/system-data>). Jeden plik .csv odpowiada jednemu miesiącowi wypożyczeń, co powoduje comiesięczne wprowadzenie rekordów do hurtowni. Każdy z nich zawiera następujące kolumny:

- `tripduration` – długość wypożyczenia w sekundach
- `starttime` – dokładna data i godzina wypożyczenia roweru
- `stoptime` – dokładna data i godzina zwrócenia roweru
- `start/stop station id` – identyfikator stacji wypożyczenia/zwrotu
- `start/stop station name` – nazwa stacji wypożyczenia/zwrotu
- `start/stop station latitude` – szerokość geograficzna stacji wypożyczenia/zwrotu
- `start/stop station longitude` – długość geograficzna stacji wypożyczenia/zwrotu
- `bikeid` – identyfikator roweru
- `usertype` – typ użytkownika (okazjonalny lub subskrybent)
- `birth year` – rok urodzenia użytkownika

- **gender** – płeć podana przez użytkownika

Drugie źródło danych, z którego korzystamy przy tworzeniu hurtowni to API pogodowe (<https://www.visualcrossing.com/>), dostarczające historyczne dane pogodowe z Nowego Jorku. Jest ono płatne, ale pozwala na wygenerowanie 1000 rekordów dziennie w planie darmowym, co dla tego projektu jest wystarczające. Dostarcza wiele kolumn z informacjami o pogodzie, jednak nie korzystamy ze wszystkich z nich. Poniżej przedstawiamy te, których używamy:

- **name** – lokalizacja, której dotyczy wiersz
- **datetime** – data, której dotyczy dany wiersz
- **temp** – średnia temperatura z danego dnia w stopniach Celsjusza
- **feelslike** – średnia odczuwalna temperatura z danego dnia
- **precip** – ilość opadu w mm
- **precipcover** – procent doby, w którym występował opad
- **preciptype** – typ opadu (deszcz/śnieg/inne)
- **windspeed** – prędkość wiatru
- **cloudcover** – procent zachmurzenia nieba
- **conditions** – słowny opis warunków atmosferycznych.

Zapytania do API wykonujemy za pomocą skryptu napisanego w języku Python, a wyniki zapisujemy do pliku `.csv`, z którego korzysta potem proces ETL.

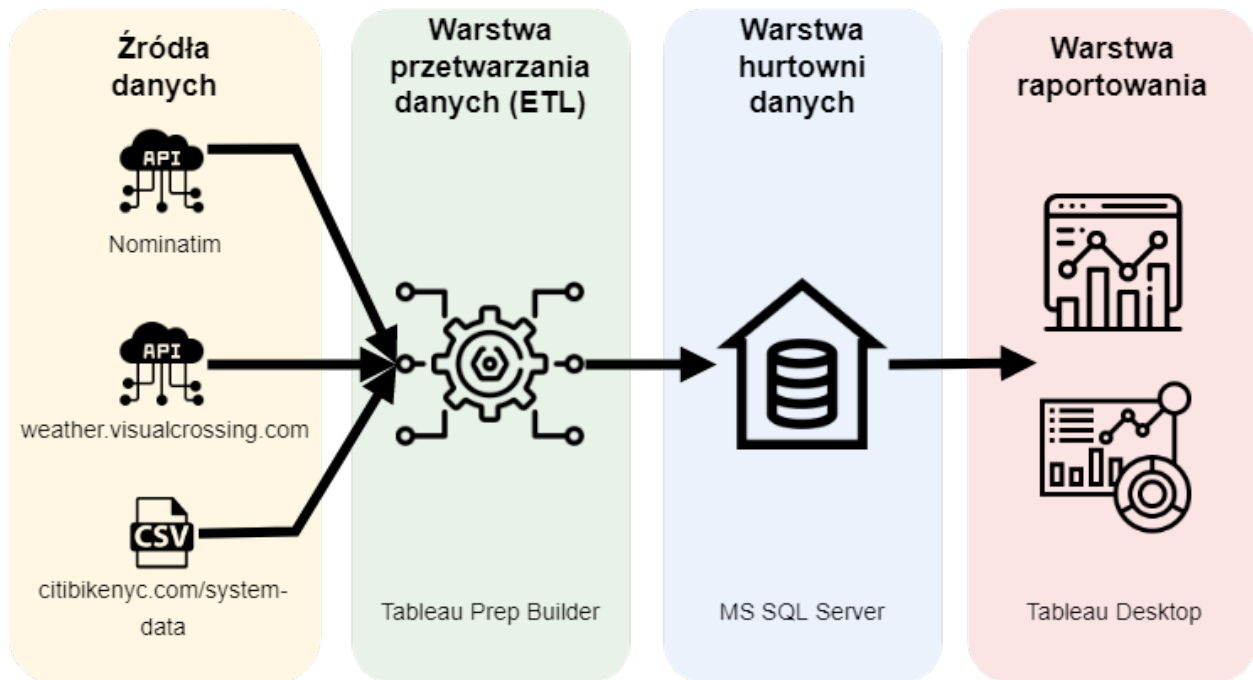
Ostatnim źródłem, z którego korzystamy jest API **Nominatim**, które na podstawie współrzędnych geograficznych dostarcza informacji o danej lokalizacji, takiej jak kraj, miasto oraz dokładny adres. API jest darmowe, jednak warunkiem korzystania jest wykonywanie nie więcej niż jednego zapytania na sekundę, co niestety znacznie wydłuża proces ETL przy pierwszym wprowadzaniu danych. Wykonując zapytania dla stacji rowerowych dostajemy informacje o dzielnicy, sąsiedztwie oraz ulicy na której się znajduje. Tak jak w przypadku poprzedniego API, zapytania wywoływane są za pomocą skryptu w Pythonie, i zapisywane do pliku `.csv`.

3 Opis rozwiązania

3.1 Warstwy modelu

Architektura naszego rozwiązania została przedstawiona graficznie na rysunku 1. Składa się ona z trzech głównych warstw:

- **Warstwa przetwarzania danych** – jej zadaniem jest zebranie danych z różnych źródeł, uspoźnienie ich oraz przetransformowanie do oczekiwanej postaci wyjściowej oraz załadowanie właściwych wynikowych tabel do hurtowni danych. Przygotowana jest w narzędziu Tableau Prep Builder (dokładniejszy opis w 3.2).



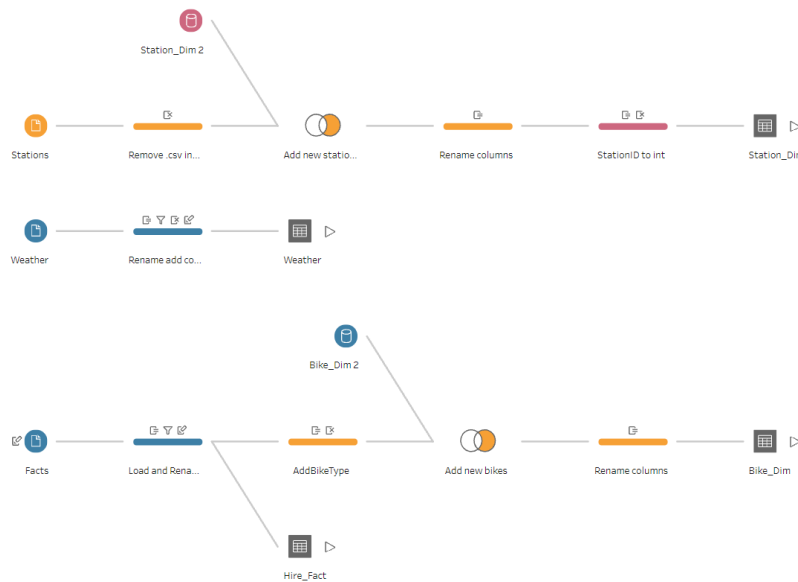
Rysunek 1: Schemat warstw rozwiązania

- **Warstwa hurtowni danych** – odpowiada za przygotowany model hurtowni, jego implementację. Przechowuje i agreguje dane potrzebne do sporządzenia raportów w kolejnej warstwie. Przygotowana w MS SQL Server, z użyciem SQL Server Management Studio (dokładniejszy opis w sekcji 3.3).
- **Warstwa raportowania** – jej celem jest prezentacja użytkownikom końcowym raportów i wizualizacji, na podstawie których mogą podejmować decyzje. Przygotowana została w narzędziu Tableau Desktop.

3.2 Warstwa ETL

Warstwa przetwarzania i uspójniania danych może być zobrazowana schematem przedstawionym na rysunku 2. Dokonujemy w niej następujących przekształceń:

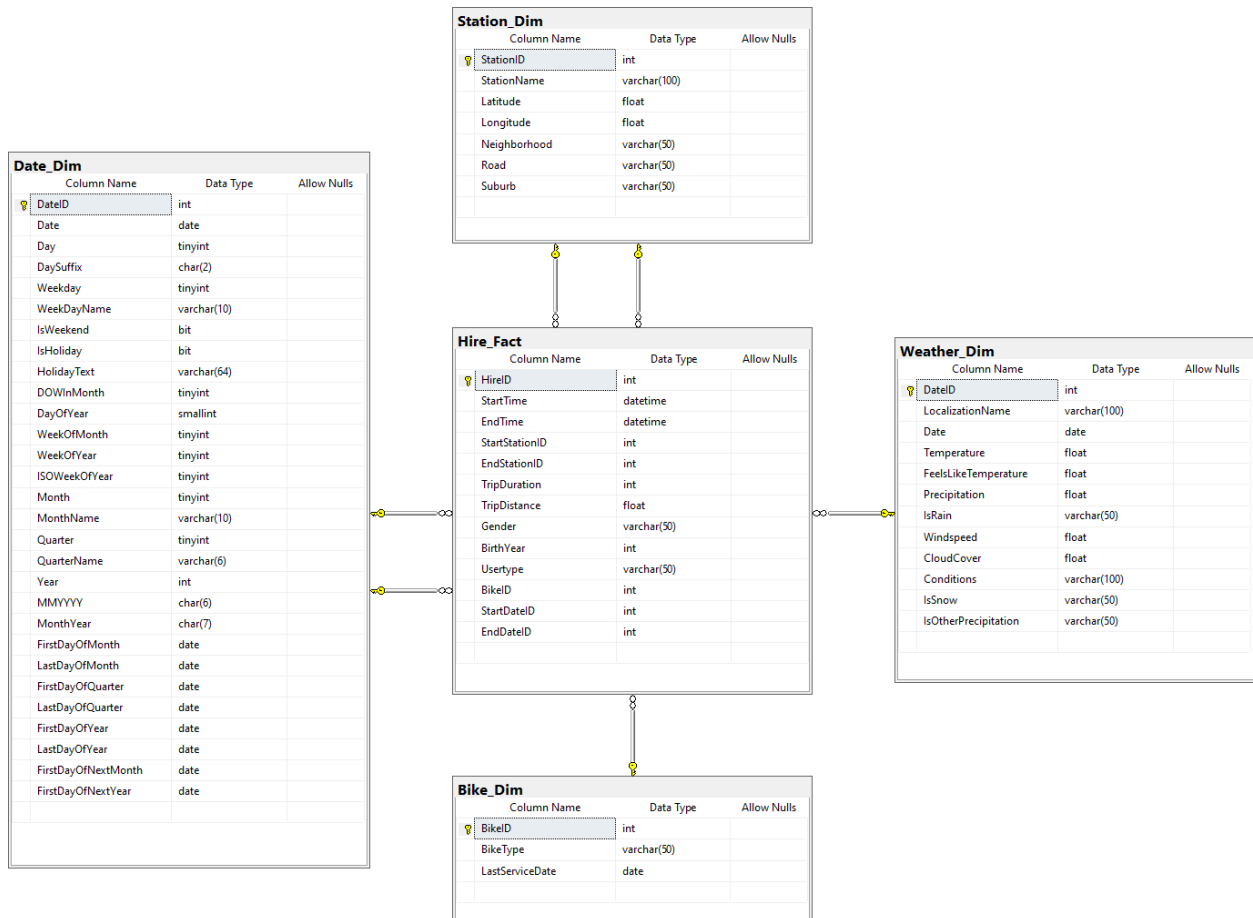
- Wczytanie danych dotyczących wypożyczeń z plików płaskich `.csv`
- Zmiana nazw kolumn tak, aby dokładnie opisywały, co zawierają (np. `feelslike` przekształcone w `FeelsLikeTemperature`, `lat` zmienione na `Latitude` itd.),
- Ujednolicenie stylu nazewnictwa kolumn (zdecydowaliśmy się na `PascalCase`),
- Przekształcenia tabeli faktów:
 - na podstawie pól startu i zwrotu roweru wygenerowanie pól typu `int` -i identyfikatorów daty, aby połączyć wymiar kalendarza (klucze obce),
 - stworzenie nowej miarki – długość wycieczki obliczanej na podstawie współrzędnych geograficznych stacji początkowej i końcowej,
 - przekształcenie kodu płci na tekst (zmiana 1 na "Male", 2 na "Female" i 0 na "Unknown")



Rysunek 2: Schemat procesu ETL

- Stworzenie tabel wymiarowych na podstawie danych o rowerach:
 - do wymiaru roweru stworzone pole typu roweru (elektryczny, tandem, zwykły) – niestety nie ma tego w wersji danych z których korzystamy, więc ustalamy sztucznie na podstawie identyfikatora roweru: kończące się na 9 są elektryczne, a kończące się na 8 to tandemy, reszta zwykłe,
 - do wymiaru roweru dodajemy pole z datą ostatniego serwisowania (sztucznie generowana),
 - do wymiaru stacji przenosimy kolumny związane z jej nazwą oraz współrzędnymi geograficznymi,
 - do wymiaru stacji dodajemy kolumny **Road**, **Neighbourhood**, **Suburb** – korzystając z API Nominatim,
 - zaokrąglenie wszystkich współrzędnych geograficznych stacji do 4 miejsc po przecinku, ponieważ w danych dokładność nie jest spójna; przekłada się to na dokładność lokalizacji ok. 1 m
- Stworzenie i przekształcenia tabeli wymiarowej z danych pogodowych
 - wysłanie zapytania do API i wczytanie danych z otrzymanego pliku **.csv**,
 - wybór tylko interesujących nas kolumn (tych opisanych w sekcji 2), zignorowanie pozostałych,
 - rozdzielenie kolumny **precipitype**, która ma wylistowane rodzaje opadów w ciągu danego dnia na trzy kolumny **IsRain**, **IsSnow**, **IsOtherPrecipitation**, które mówią czy w danym dniu występował opad tego rodzaju

Dzięki temu procesowi do hurtowni trafiają dane spójne, z których można w prosty sposób w narzędziu klasy Business Intelligence przygotować raporty i wizualizacje dla użytkownika końcowego.



Rysunek 3: Schemat hurtowni danych

3.3 Ostateczny model hurtowni

Schemat hurtowni został przedstawiony na rysunku 3. Tabelą faktów jest **Hire.Fact**. Oprócz atrybutu faktów i dwóch miar będzie ona zawierać zdegenerowany wymiar użytkownika oraz klucze obce do tabel wymiarów. **StartStationID** oraz **EndStationID** wskazują na wymiar stacji **Station.Dim**, który zawiera dodatkowo współrzędne geograficzne stacji oraz jej nazwę. **BikeID** łączy tabele faktów z wymiarem **Bike.Dim** zawierającym typ wypożyczonego roweru oraz datę ostatniego serwisowania. Wymiar **Date.Dim** jest podłączony zarówno do daty wypożyczenia jak i zwrotu roweru. W przypadku wymiarów daty oraz stacji mamy więc do czynienia z role-playing dimension - są one połączone z tabelą faktową więcej niż jednym kluczem obcym. Wymiar **Weather.Dim** jest połączony z tabelą faktową przy użyciu pola **StartDateID**. Podjęliśmy taką decyzję, gdyż to pogoda w czasie wypożyczania roweru ma wpływ na decyzję użytkownika, a więc także na biznes. Zawiera on najważniejsze informacje pogodowe na temat najważniejszych czynników pogodowych, czyli temperatury, realnej i odczuwalnej, rodzaju i intensywności opadów, prędkości wiatru i zachmurzenia.

Ze względu na specyfikę danych wymiar użytkownika jest zdegenerowany - znajduje się w tabeli faktów. Ponieważ fakty są publikowane w internecie, muszą być anonimizowane. Gdyby hurtownia powstawała we współpracy z firmą, z tych pól mógłby powstać osobny wymiar.

Najważniejszymi miarami z punktu widzenia biznesu są: czas trwania jednego wypożyczenia (domyślnie agregowany jako średnia, ale w niektórych raportach wykorzystywana jest też suma), ilość wypożyczeń (zliczanie wystąpień) oraz przejechany dystans (agregowany jako suma).

3.4 Warstwa raportowa

Warstwa raportowa obsługiwana jest przez narzędzie Tableau Desktop. Jedynym źródłem danych jest połączenie z hurtownią w Microsoft SQL Server. Model danych, z którego korzysta ta warstwa, jest nieznacznie różny od tego w hurtowni. Zostały dodane kalkulacje bieżące - np. wyznaczające czas trwania wypożyczenia roweru w minutach (w hurtowni dana ta przechowywana jest w sekundach) oraz stworzenie hierarchii geograficznej pozwalającej na „zagłębianie się” w dane (ang. drilldown), tzn. dzielnica → sąsiedztwo → ulica. Plik .pdf z wygenerowanymi statycznie raportami znajduje się w repozytorium rozwiązania.

3.4.1 Przykładowe raporty

Dla użytkownika końcowego zostało przygotowane 5 raportów dotyczących:

- Podsumowania przebiegu rowerów
- Analizy wypożyczeń ze względu na położenie geograficzne i pogodę
- Analizy długości ze względu na dzień tygodnia i typ użytkownika
- Analizy ilości wypożyczeń w rozkładzie czasowym
- Podsumowania ze względu na osiedla i dzielnice

Podsumowanie przebiegu rowerów przedstawia wykres słupkowy 20 rowerów z największą ilością przejechanych kilometrów. Możliwe jest kontrolowanie zakresu czasowego z jakiego przedstawiane są dane. Konkretnie rowery oznaczone są kolorem ze względu na swój typ (normalny, elektryczny, tandem). Dzięki tej wizualizacji możemy wyróżnić najczęściej używane rowery, które mogą wymagać serwisowania.

Analiza wypożyczeń ze względu na położenie geograficzne i pogodę zawiera wizualizację mapy Nowego Yorku z naniesionymi stacjami. Każdą stację można analizować w wybranym oknie czasowym pod względem ilości wypożyczeń, które miały na niej miejsce. Używając filtrów pogodowych jesteśmy w stanie dowiedzieć się jak różne warunki atmosferyczne, takie jak deszcz i śnieg, wpływały na zainteresowanie usługą w danym dniu. Analiza mogłaby pomóc w podjęciu decyzji o przeniesieniu rowerów z mniej popularnych stacji do tych częściej używanych.

Analiza długości ze względu na dzień tygodnia i typ użytkownika przedstawia na wykresie słupkowym całkowity i średni czas jazdy. Dane zostały zagregowane po dniach tygodnia i typie użytkownika (Customer, Subscriber). Możliwe jest filtrowanie ze względu na dzielnice Nowego Jorku. Raport ten może pomóc przy określaniu różnic w zachowaniach typów użytkowników w zależności od dnia i położenia.

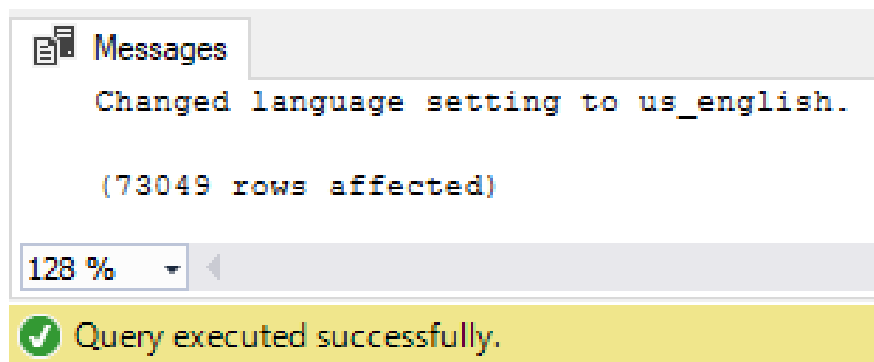
Analiza ilości wypożyczeń w rozkładzie czasowym zawiera zagregowaną ilość wypożyczeń, zgrupowaną po godzinie w której zostały wykonane. Filtr umożliwia obserwację dla dowolnie wybranego podzbioru dni tygodnia. Dzięki wizualizacji wykresem słupkowym jesteśmy w stanie stwierdzić, że w dniach pracujących największy ruch jest o godzinie 8 i 18 gdy ludzie przemieszczają się między pracą i miejscem zamieszkania. W weekendy największa ilość wypożyczeń jest w przedziale godzinowym 12-19.

Podsumowanie ze względu na osiedla i dzielnice jest dashboardem składającym się z tabelarycznych danych z kolumnami zawierającymi średnią długość przejazdu, średni czas przejazdu oraz sumę wypożyczeń w danej lokalizacji. Lokalizacja ma charakter hierarchii składającej się z przedmieścia (np. Brooklyn, Manhattan) i przynależącego do niego sąsiedztwa (np. Battery Park City, Hell's Kitchen). Obok tabeli znajduje się wykres kafelkowy przedstawiający te dane w bardziej przystępny sposób, gdzie wielkość kafelka odpowiada za ilość wypożyczeń a natężenie koloru za średni czas wypożyczeń. Tabela i wykres są skoordynowane w taki sposób, że zmiana dokładności lokalizacji na jednym pokazuje się też na drugim.

4 Testy funkcjonalności

4.1 Test skryptu generującego hurtownię

Testowany jest skrypt języka SQL generujący bazę danych zawierającą hurtownię – tabele, wraz z kluczami głównymi i ograniczeniami kluczy obcych, poprzez wykonanie skryptu SQL. Oczekiwany efekt: Skrypt wykonuje się poprawnie, informuje o wstawieniu do bazy danych 73049 rekordów (generowane wiersze z wymiaru daty). Powstają wszystkie tabele odpowiednio ze sobą połączone. Wynik wykonania przedstawiony na rysunku 4, powstały model hurtowni prezentowany był już wcześniej na rysunku 3. Test przebiegł poprawnie

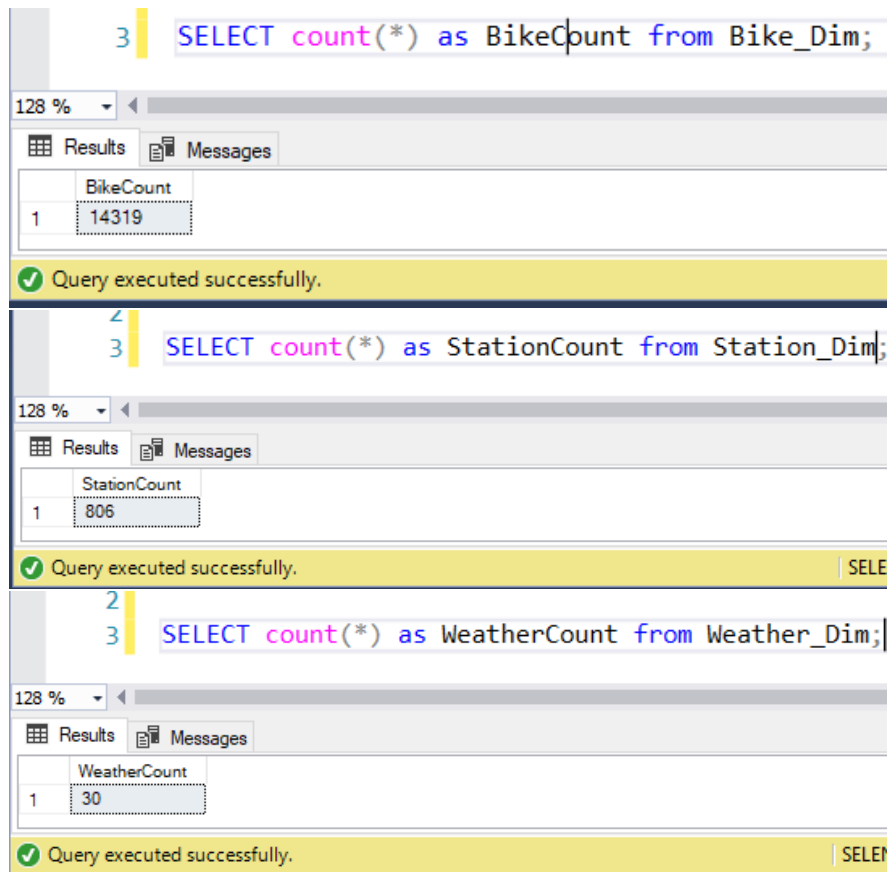


Rysunek 4: Test skryptu generującego hurtownię

4.2 Testy procesu ETL

4.2.1 Populowanie tabel wymiarowych

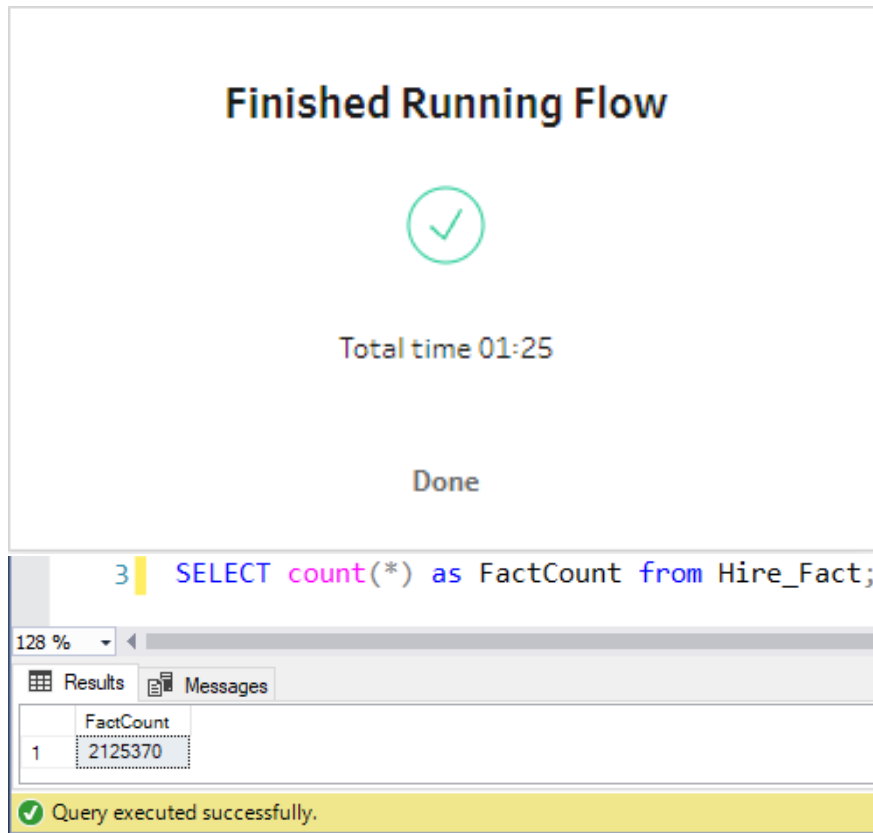
Testowane jest wprowadzenie wierszy do pustych tabel wymiarowych, poprzez uruchomienie odpowiednich funkcji w Tableau Prep. Oczekiwany efekt: poprawne przejście flow w Tableau Prep Builder oraz pojawienie się w bazie danych 30 rekordów w tabeli `Weather_Dim`, 806 rekordów w tabeli `Station_Dim` oraz 14319 rekordów w tabeli `Bike_Dim`. Wyniki przedstawiono na rysunku 5. Test przebiegł poprawnie.



Rysunek 5: Wprowadzanie danych wymiarów do pustej hurtowni

4.2.2 Populowanie tabeli faktowej

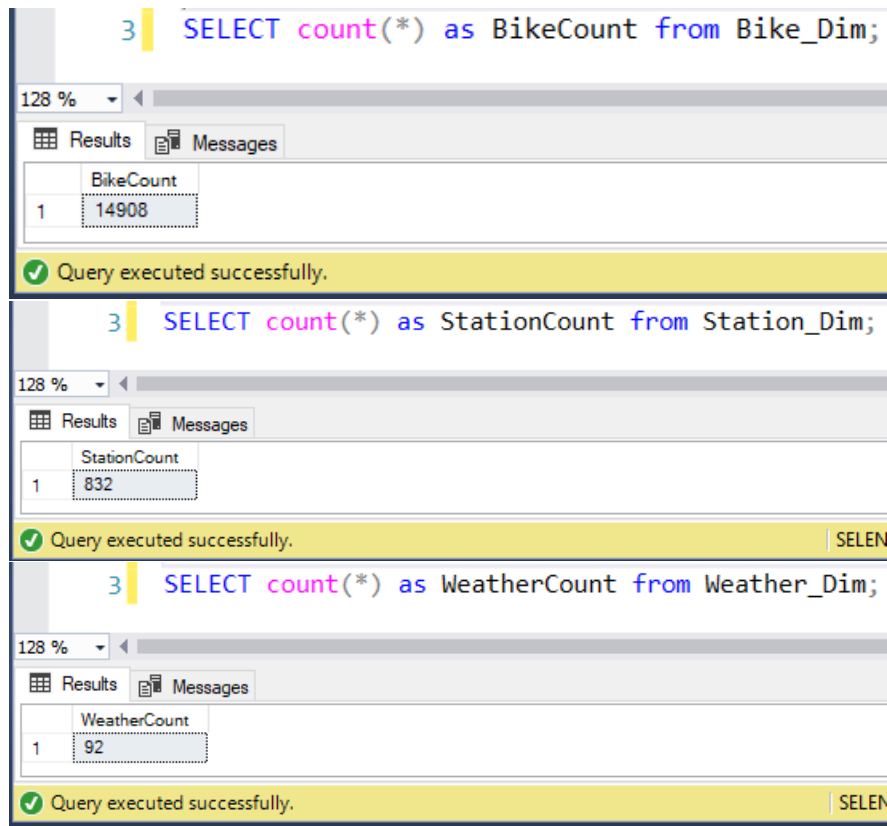
Testowane jest wprowadzanie wierszy do pustej tabeli faktowej, poprzez odpalenie odpowiedniej funkcji w Tableau Prep. Oczekiwany efekt: poprawne przejście flow w procesie ETL, pojawienie się 2125370 rekordów w tabeli faktowej. Wyniki przedstawione są na rysunku 6. Test przebiegł poprawnie.



Rysunek 6: Wprowadzenie faktów do pustej hurtowni

4.2.3 Dodawanie wierszy do tabel wymiarowych w cyklicznych aktualizacjach

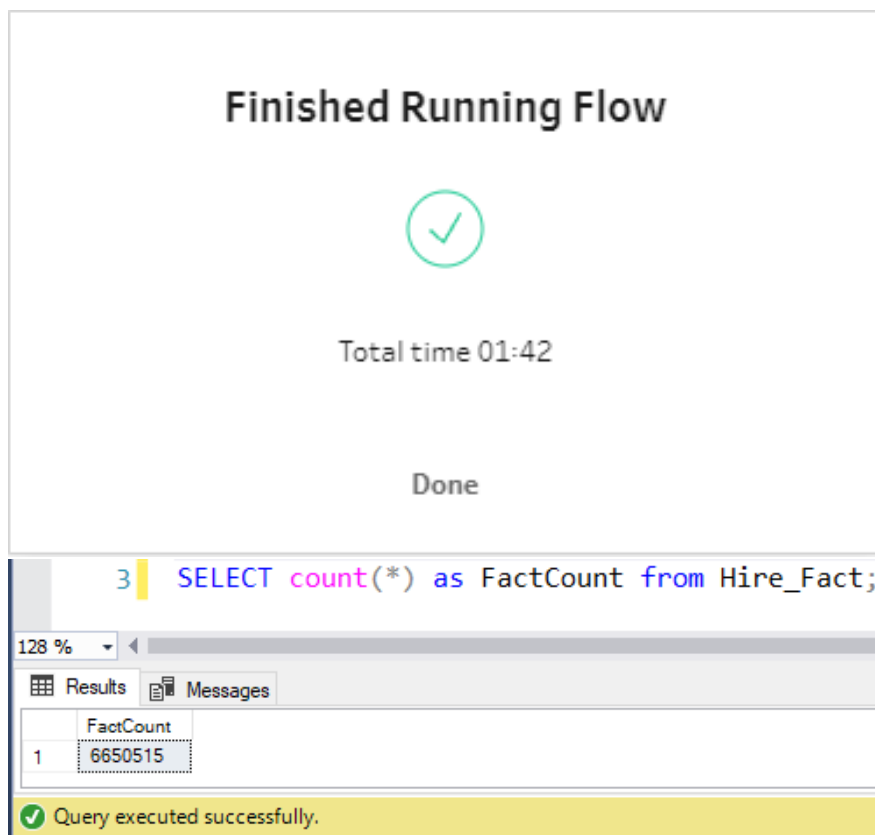
Testowane jest dodawanie kolejnych wierszy do zawierających już rekordy tabel wymiarowych, przez odpalenie flow w Tableau Prep. Oczekiwany rezultat: poprawne wykonanie się procesu ładowania danych. Po wykonaniu tabele `Weather_Dim`, `Station_Dim` i `Bike_Dim` zawierają odpowiednio 92, 832 i 14908 rekordów. Wyniki przedstawiono na rysunku 7. Test przebiegł poprawnie.



Rysunek 7: Dodawanie rekordów do wypełnionych już tabel wymiarowych

4.2.4 Dodawanie wierszy do tabeli faktowej w cyklicznych aktualizacjach

Testowane jest dodawanie kolejnych wierszy do tabeli faktów, kiedy jest ona niepusta. Oczekiwany rezultat: 6650515 wierszy w tabeli faktów po wykonaniu procesu. Wynik przedstawiony na rysunku 8. Test przebiegł poprawnie.

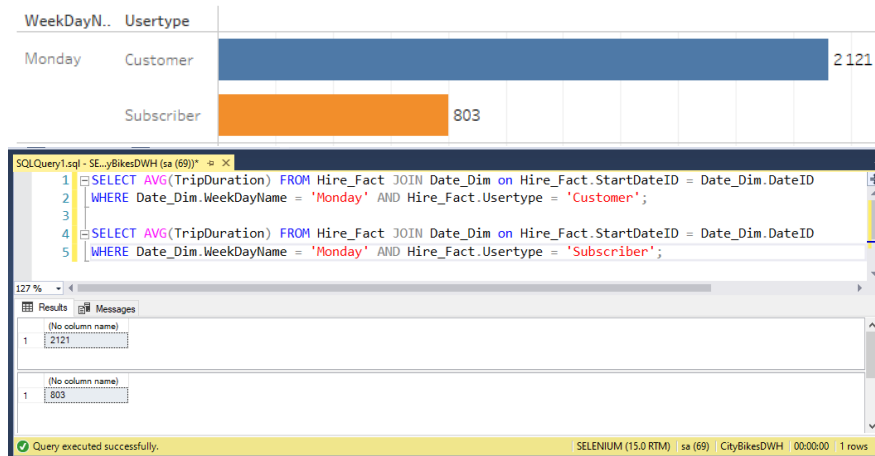


Rysunek 8: Dodawanie wierszy do tabeli faktowej

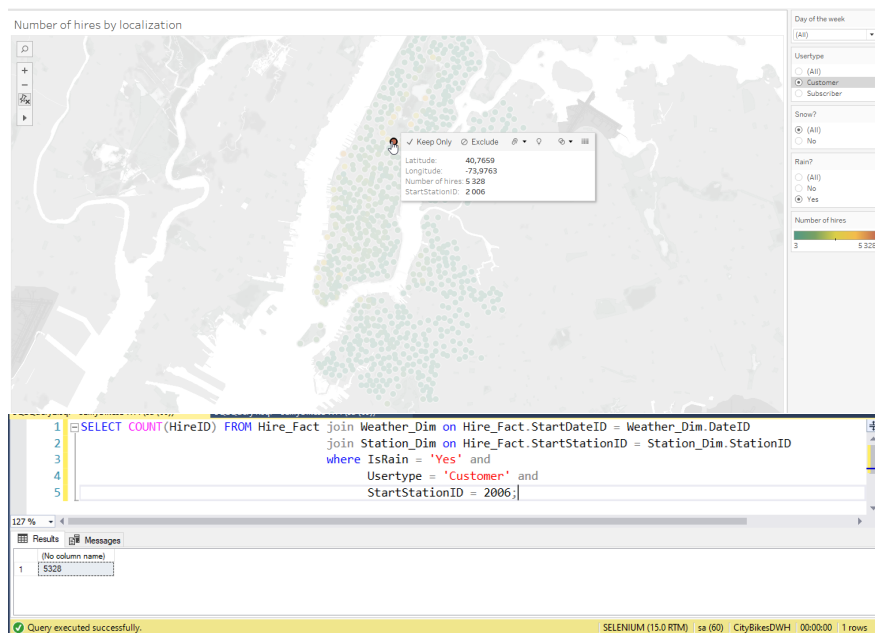
4.3 Testy warstwy raportowej

Zostały również przeprowadzone testy warstwy raportowej, aby sprawdzić, czy prezentowane użytkownikom wartości zgadzają się z rzeczywistością. Miały one postać porównania wartości na wykresach, z wynikami odpowiednio skonstruowanych kwerend w warstwie hurtowni danych. Oczekiwana jest zgodność liczb na wizualizacjach z wynikami zapytań SQL. Rezultaty testów widoczne są na rysunkach 9, 10 oraz 11. Testy przebiegły prawidłowo.

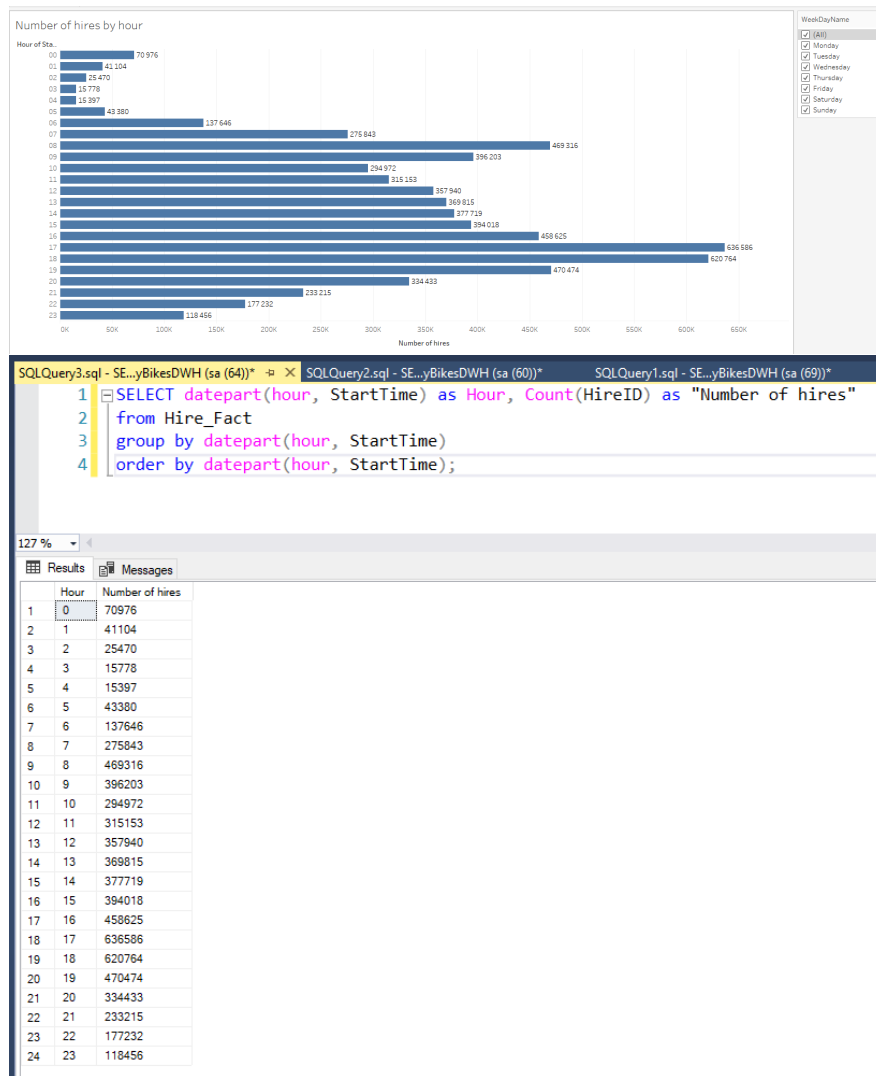
Average and total trip duration by day of week and user type



Rysunek 9: Test wizualizacji dotyczącej długości średniej wycieczki



Rysunek 10: Test wizualizacji zawierającej mapę



Rysunek 11: Test wizualizacji pokazującej wypożyczenia w zależności od godziny

5 Podsumowanie

Założenia projektowe zostały spełnione. Udało się przygotować system obsługujący hurtownię danych dla firmy zajmującej się udostępnianiem rowerów miejskich w Nowym Jorku. Jest on sprawny we wszystkich warstwach, od warstwy importu i przekształcania danych po warstwę raportowania, co zademonstrowały testy funkcjonalne. Z perspektywy klienta biznesowego najważniejsza jest właśnie warstwa raportowa, ponieważ to ona bezpośrednio wspiera w podejmowaniu decyzji biznesowych. Hurtownia pozwala na rozwój rozwiązania w prosty sposób, bez konieczności modyfikacji całej architektury.

6 Podział pracy w zespole

Praca nad projektem była rozłożona równomiernie pomiędzy członków zespołu. Większość rozwiązania powstała przy jednoczesnej, wspólnej pracy, ale główny podział zadań był następujący: Mikołaj Spytek

odpowiedzialny był za model hurtowni oraz przygotowanie danych z dwóch API. Szymon Rećko zajął się pozostałą częścią procesu ETL oraz warstwą raportową. Dokumentacja projektowa została przygotowana we współpracy.

Literatura

- [1] Abelski J., Wykład przedmiotu Hurtownie danych i systemy Business Intelligence, 2022
- [2] Kimball R., Ross M., The data warehouse toolkit: the definitive guide to dimensional modeling, 3rd Edition, Wiley, 2013