DWH KM1

Michał Iwaniuk, Bartłomiej Borycki

May 2025

1 Cel projektu i planowane korzyści

Celem projektu jest stworzenie hurtowni danych służącej do kompleksowej analizy wypadków drogowych w Wielkiej Brytanii z lat 2019–2023. Głównymi odbiorcami rozwiązania są instytucje państwowe odpowiedzialne za bezpieczeństwo ruchu drogowego. Projekt ma na celu dostarczenie narzędzia umożliwiającego monitorowanie i badanie wypadków w kontekście wielu czynników jednocześnie — takich jak lokalizacja zdarzenia, warunki pogodowe, liczba ofiar oraz typy pojazdów uczestniczących w kolizjach.

Dzięki integracji danych policyjnych o wypadkach z danymi meteorologicznymi możliwe będzie odkrywanie korelacji, na przykład wpływu pogody na liczbę i ciężkość wypadków, a także identyfikowanie szczególnie niebezpiecznych miejsc lub warunków.

Planowane korzyści: Hurtownia danych umożliwi generowanie raportów i dashboardów prezentujących kluczowe wskaźniki bezpieczeństwa drogowego. Instytucje będą mogły z łatwością śledzić trendy (np. wzrost lub spadek liczby wypadków w ujęciu rocznym), identyfikować obszary wysokiego ryzyka na mapie oraz podejmować świadome decyzje dotyczące poprawy infrastruktury czy wdrażania działań prewencyjnych.

2 Planowane raporty i dashboard BI (6 stron)

Na potrzeby użytkowników końcowych zostanie przygotowany interaktywny dashboard BI, składający się z sześciu stron tematycznych. Każda strona będzie skupiać się na innym aspekcie danych o wypadkach drogowych, prezentując kluczowe wskaźniki oraz umożliwiając ich filtrowanie. Poniżej przedstawiono planowaną zawartość każdej strony:

Strona 1: Podsumowanie ogólne

Strona startowa będzie zawierać syntetyczny przegląd sytuacji na drogach w analizowanym okresie. Znajdą się tu:

- Kafelkowe KPI: łączna liczba wypadków, liczba ofiar, liczba pojazdów.
- Wykres liniowy: liczba wypadków w latach 2019–2023.
- Wykres kołowy lub słupkowy: podział wypadków według ciężkości (śmiertelne, poważne, lekkie).
- Mapa punktowa lub heatmapa UK: lokalizacja wypadków z podziałem przestrzennym.

Strona 2: Analiza lokalizacji

Ta strona będzie umożliwiać analizę przestrzennego rozkładu wypadków:

- Choropleth map: intensywność wypadków w regionach administracyjnych (np. hrabstwa).
- Wykres słupkowy / tabela: ranking 10 najbardziej niebezpiecznych lokalizacji.
- Filtry interaktywne: rok, typ drogi (np. droga główna, lokalna).

Strona 3: Warunki pogodowe a wypadki

Sekcja ta połączy dane meteorologiczne z informacjami o wypadkach:

- Wykres kolumnowy: liczba wypadków przy różnych warunkach pogodowych.
- Wskaźnik i wykres: średnia temperatura podczas wypadków i jej rozkład.
- Wykres udziału: stan nawierzchni (sucha, mokra, oblodzona).

Strona 4: Ofiary wypadków

Strona ta będzie poświęcona analizie ofiar i ich charakterystyce:

- Wskaźnik: średnia liczba ofiar na wypadek.
- Wykres słupkowy: liczba ofiar wg ciężkości obrażeń.
- Wykres warstwowy / procentowy: typ ofiar (piesi, kierowcy, pasażerowie).
- Piramida wieku: wiek i płeć poszkodowanych.

Strona 5: Pojazdy i sprawcy

Ta część będzie analizować dane dotyczące pojazdów i kierowców uczestniczących w wypadkach:

- Wykres kolumnowy: udział różnych typów pojazdów w wypadkach.
- Wskaźnik i histogram: liczba pojazdów na wypadek.
- Wykresy: wiek pojazdu, typ paliwa, pojemność silnika.
- Wykresy demograficzne: wiek, płeć i status społeczno-ekonomiczny kierowców.

Strona 6: Analiza czasowa i sezonowość

Ostatnia strona umożliwi analizę czasową zdarzeń drogowych:

- Wykres liniowy: miesięczna liczba wypadków w latach 2019–2023.
- Wykres słupkowy: liczba wypadków wg dnia tygodnia i godziny.
- Suwak daty: wybór zakresu czasowego do analizy.

3 Wykorzystywane zbiory danych

Dane publikowane są corocznie przez Department for Transport (DfT), zwykle we wrześniu za rok poprzedni (np. dane za 2023 rok zostały udostępnione we wrześniu 2024). Publikacje obejmują finalne, zweryfikowane dane i są dostępne publicznie w formacie CSV. W projekcie wykorzystywane są dane z lat 2019–2023.

Źródłem danych dla hurtowni są zestawy opublikowane przez **Department for Transport (DfT)** na portalu `https://data.gov.uk/dataset/road-safety-data`, obejmujące lata 2019–2023 oraz dane meteorologiczne z biblioteki **Meteostat** (`https://meteostat.net`). Poniżej przedstawiono szczegóły każdego ze zbiorów, wraz z opisem ich struktury.

3.1 Dane o wypadkach (Collisions)

Zbiór `Collisions` zawiera jeden rekord dla każdego zgłoszonego wypadku drogowego. Dane obejmują informacje o dacie, lokalizacji, typie drogi, warunkach atmosferycznych i skutkach zdarzenia. Wartości są kodowane liczbowo zgodnie ze słownikiem STATS19.

Nazwa kolumny	**Opis**
accident_index	Unikalny identyfikator wypadku
accident_year	Rok zdarzenia
accident_reference	Alternatywny kod wypadku
location_easting_osgr	Współrzędna w osi wschód (OSGB)
location_northing_osgr	Współrzędna w osi północ (OSGB)
longitude	Długość geograficzna
latitude	Szerokość geograficzna
police_force	Jednostka policji rejestrująca
accident_severity	Ciężkość wypadku
number_of_vehicles	Liczba pojazdów
number_of_casualties	Liczba ofiar
date	Data zdarzenia
day_of_week	Dzień tygodnia
time	Godzina zdarzenia
local_authority_district	Kod jednostki samorządowej
local_authority_ons_district	Kod ONS jednostki
local_authority_highway	Kod drogi
first_road_class	Klasa głównej drogi
first_road_number	Numer drogi
road_type	Typ drogi
speed_limit	Ograniczenie prędkości
junction_detail	Szczegóły skrzyżowania
junction_control	Kontrola ruchu na skrzyżowaniu
second_road_class	Klasa drugorzędnej drogi
second_road_number	Numer drogi drugorzędnej
pedestrian_crossing_human_control	Przejście z nadzorem
pedestrian_crossing_physical_facilities	Rodzaj przejścia
light_conditions	Warunki oświetleniowe
weather_conditions	Warunki pogodowe
road_surface_conditions	Stan nawierzchni
special_conditions_at_site	Szczególne warunki miejsca

Nazwa kolumny	Opis
carriageway_hazards	Utrudnienia na jezdni
urban_or_rural_area	Typ terenu (miejski/wiejski)
did_police_officer_attend_scene_of_accident	Obecność policji na miejscu
trunk_road_flag	Flaga drogi głównej
lsoa_of_accident_location	Kod LSOA lokalizacji
enhanced_severity_collision	Znormalizowana ciężkość wypadku

3.2 Dane o pojazdach (Vehicles)

Zbiór `Vehicles` zawiera jeden rekord dla każdego pojazdu biorącego udział w wypadku. Dane obejmują typ pojazdu, manewr w chwili zdarzenia, jego stan techniczny, napęd oraz informacje o kierowcy.

Nazwa kolumny	Opis
accident_index	Identyfikator wypadku
accident_year	Rok wypadku
accident_reference	Alternatywny identyfikator
vehicle_reference	Numer pojazdu w zdarzeniu
vehicle_type	Typ pojazdu
towing_and_articulation	Czy pojazd ciągnął coś
vehicle_manoeuvre	Wykonywany manewr
vehicle_direction_from	Kierunek początkowy
vehicle_direction_to	Kierunek docelowy
vehicle_location_restricted_lane	Pas ograniczony
junction_location	Położenie względem skrzyżowania
skidding_and_overturning	Poślizg/dachowanie
hit_object_in_carriageway	Obiekt na jezdni
vehicle_leaving_carriageway	Zjazd z jezdni
hit_object_off_carriageway	Obiekt poza jezdnią
first_point_of_impact	Punkt pierwszego kontaktu
vehicle_left_hand_drive	Czy pojazd z kierownicą po lewej
journey_purpose_of_driver	Cel podróży
sex_of_driver	Płeć kierowcy
age_of_driver	Wiek kierowcy
age_band_of_driver	Przedział wiekowy kierowcy
engine_capacity_cc	Pojemność silnika
propulsion_code	Typ napędu
age_of_vehicle	Wiek pojazdu
generic_make_model	Marka i model (jeśli znane)
driver_imd_decile	Decyla deprywacji kierowcy
driver_home_area_type	Typ miejsca zamieszkania
lsoa_of_driver	Kod LSOA kierowcy
escooter_flag	Flaga hulajnogi
dir_from_e/n, dir_to_e/n	Współrzędne kierunku ruchu
driver_distance_banding	Przedział odległości od domu

3.3 Dane o ofiarach (Casualties)

Zbiór `Casualties` zawiera jeden rekord dla każdej osoby poszkodowanej w wypadku. Uwzględnia dane demograficzne, typ uczestnika ruchu i kontekst zdarzenia.

Nazwa kolumny	**Opis**
accident_index	Identyfikator wypadku
accident_year	Rok wypadku
accident_reference	Alternatywny identyfikator
vehicle_reference	Numer pojazdu
casualty_reference	Numer ofiary w wypadku
casualty_class	Klasa ofiary (pieszy, pasażer itd.)
sex_of_casualty	Płeć ofiary
age_of_casualty	Wiek ofiary
age_band_of_casualty	Przedział wiekowy ofiary
casualty_severity	Ciężkość obrażeń
pedestrian_location	Pozycja pieszego
pedestrian_movement	Ruch pieszego
car_passenger	Miejsce pasażera w samochodzie
bus_or_coach_passenger	Rodzaj pasażera autobusu
pedestrian_road_maintenance_worker	Czy pieszy to pracownik drogowy
casualty_type	Typ użytkownika drogi
casualty_home_area_type	Typ miejsca zamieszkania
casualty_imd_decile	Decyla deprywacji
lsoa_of_casualty	Kod LSOA ofiary
enhanced_casualty_severity	Znormalizowana ciężkość
casualty_distance_banding	Odległość od domu

3.4 Dane meteorologiczne (Meteostat)

Dane pogodowe zostały pobrane z biblioteki Meteostat za pomocą API w języku Python. Pozwalają na powiązanie każdego wypadku z warunkami atmosferycznymi panującymi w danym czasie i lokalizacji. Dane są pobierane dynamicznie na etapie procesu ETL.

Nazwa kolumny	**Opis**
time	Data i godzina pomiaru
temp	Temperatura powietrza [°C]
dwpt	Temperatura punktu rosy [°C]
rhum	Wilgotność względna [%]]
prcp	Opady [mm/h]
snow	Pokrywa śnieżna [mm]
wdir	Kierunek wiatru [°]
wspd	Prędkość wiatru [km/h]
wpgt	Poryw wiatru [km/h]
pres	Ciśnienie atmosferyczne [hPa]
tsun	Czas nasłonecznienia [min]
coco	Kod warunków pogodowych (Weather Condition Code)

Uwaga dotycząca integracji danych pogodowych. Dla każdego wypadku pobierana jest godzinna obserwacja meteorologiczna odpowiadająca dokładnemu czasowi i lokalizacji zdarzenia. W tym celu wykorzystywana jest funkcjonalność `Point` biblioteki Meteostat, która umożliwia uzyskanie danych nawet dla lokalizacji bez bezpośredniej stacji pogodowej – poprzez interpolację na podstawie sąsiednich stacji oraz wysokości terenu. Najistotniejsze dla modelu kolumny to: `temp`, `prcp`, `snow`, `wspd`, `wdir` oraz `coco`. Dane te są pobierane dynamicznie podczas procesu ETL, a ich formatem pośrednim jest obiekt `Pandas DataFrame`. Dane pogodowe są historyczne i statyczne – dlatego można je pobrać jednorazowo dla zakresu lat 2019–2023 i aktualizować tylko przy dodawaniu nowych wypadków (np. dla roku 2024).

4 Model fizyczny hurtowni danych (schemat gwiazdy)

Projekt hurtowni danych został oparty o klasyczny schemat gwiazdy (ang. star schema), który jest jednym z najczęściej stosowanych podejść w budowie systemów analitycznych typu Business Intelligence (BI). Struktura modelu zakłada istnienie jednej centralnej tabeli faktów (`Fact_Collisions`), przechowującej zagregowane dane o wypadkach drogowych, oraz zestawu tabel wymiarów opisujących różne konteksty tych zdarzeń — takie jak data, lokalizacja czy warunki pogodower.

Główne założenia modelu:

- Poziomem szczegółowości (granularnością) tabeli faktów jest pojedynczy wypadek drogowy (jeden rekord = jedno zdarzenie).
- Tabela faktów zawiera zarówno dane bezpośrednio pochodzące z zestawu `Collisions`, jak również agregaty wyliczane na podstawie tabel `Vehicles` i `Casualties`.
- Do modelu włączono dane pogodowe pochodzące z biblioteki Meteostat, przypisane do konkretnego miejsca i godziny wypadku.
- Tabele wymiarów zawierają szczegółowe atrybuty opisowe wykorzystywane do filtrowania, grupowania i analiz przekrojowych.
- Model został zoptymalizowany pod kątem zastosowań raportowych i eksploracyjnych — np. w Power BI lub SSRS — dlatego nie uwzględnia pełnej normalizacji, lecz priorytetem jest wydajność odczytu oraz prostota zapytań.

W kolejnych podsekcjach przedstawiono strukturę wszystkich tabel wchodzących w skład modelu: `Fact_Collisions` oraz wymiarów `DimDate`, `DimLocation`, `DimWeather`, `DimVehicleType`, `DimCasualtyType`, `DimRoadCondition` i innych opcjonalnych.

Tabela faktów: `fact_collision`

Table 5: Struktura tabeli faktów: `fact_collision`

`Kolumna`	**Typ**	**Opis**
Klucze główne i obce		

Kolumna	Typ	Opis
accident_index	string	Unikalny klucz wypadku
date_id	string	Klucz do dim_date
location_id	string	Klucz do dim_location
weather_id	string	Klucz do dim_weather
Dane ogólne		
accident_severity	smallint	Stopień powagi (1=śmiertelny, 2=poważny, 3=lekki)
number_of_vehicles	smallint	Liczba pojazdów
number_of_casualties	smallint	Liczba ofiar
police_attended	boolean	Czy policja była obecna
escooter_involved	boolean	Czy uczestniczyła hulajnoga elektryczna
enhanced_severity_collision	smallint	Rozszerzona klasyfikacja wypadku
Dane o pojazdach (zagregowane)		
vehicle_type_car	int	Liczba samochodów osobowych
vehicle_type_bus	int	Liczba autobusów
vehicle_type_motorcycle	int	Liczba motocykli
vehicle_type_goods	int	Liczba pojazdów dostawczych/-ciężarówek
vehicle_type_other	int	Pozostałe typy pojazdów
vehicle_manoeuvre_turning_left	int	Liczba pojazdów skręcających w lewo
vehicle_manoeuvre_turning_right	int	Liczba pojazdów skręcających w prawo
vehicle_manoeuvre_overtaking	int	Liczba pojazdów wyprzedzających
vehicle_left_hand_drive_count	int	Liczba pojazdów z kierownicą po lewej
avg_vehicle_age	decimal	Średni wiek pojazdu
avg_engine_capacity_cc	decimal	Średnia pojemność silnika (cc)
Dane o kierowcach		
driver_sex_male	int	Liczba kierowców mężczyzn
driver_sex_female	int	Liczba kierowców kobiet
driver_age_band_0_25	int	Kierowcy do 25 lat
driver_age_band_26_50	int	Kierowcy 26–50 lat
driver_age_band_51_plus	int	Kierowcy powyżej 50 lat
driver_purpose_commute	int	Kierowcy w drodze do/z pracy
driver_purpose_education	int	Kierowcy do/z uczelni/szkoły
driver_purpose_other	int	Inny cel podróży
Dane o ofiarach		
casualty_class_driver	int	Liczba ofiar – kierowców
casualty_class_passenger	int	Liczba ofiar – pasażerów
casualty_class_pedestrian	int	Liczba ofiar – pieszych
casualty_severity_fatal	int	Liczba ofiar śmiertelnych
casualty_severity_serious	int	Liczba ofiar poważnych
casualty_severity_slight	int	Liczba ofiar lekkich
casualty_age_band_0_15	int	Ofiary w wieku 0–15
casualty_age_band_16_30	int	Ofiary w wieku 16–30
casualty_age_band_31_60	int	Ofiary w wieku 31–60

Kolumna	Typ	Opis
casualty_age_band_60_plus	int	Ofiary w wieku 60+

Wymiar: dim_time

Table 6: Struktura wymiaru: dim_time

Kolumna	Typ	Opis
time_id	string	Klucz wymiaru czasu (np. 2023010101 = 1 stycznia 2023, godz. 01)
full_datetime	datetime	Dokładna data i godzina
date	date	Data dzienna (YYYY-MM-DD)
year	int	Rok
month	int	Miesiąc (1–12)
month_name	string	Nazwa miesiąca
day	int	Dzień miesiąca
day_of_week	string	Nazwa dnia tygodnia
day_of_week_num	int	Numer dnia tygodnia (1=Pon, 7=Nd)
is_weekend	boolean	Czy to weekend
week_number	int	Numer tygodnia w roku
quarter	int	Kwartal
hour	int	Godzina (0–23)
hour_band	string	Przedział czasowy
part_of_day	string	Pora dnia
is_night	boolean	Czy to noc (0–5)
is_rush_hour	boolean	Czy godzina to szczyt komunikacyjny
is_holiday	boolean	Czy to święto

Wymiar: dim_location

Table 7: Struktura wymiaru: dim_location

Kolumna	Typ	Opis
location_id	string	Klucz lokalizacji
location_easting_osgr	int	UK Grid Easting
location_northing_osgr	int	UK Grid Northing
longitude	float	Długość geograficzna
latitude	float	Szerokość geograficzna
lsoav_of_accident_location	string	Kod LSOA
local_authority_district	string	Kod jednostki samorządowej
road_type	string	Typ drogi
speed_limit	int	Ograniczenie prędkości
junction_detail	string	Typ skrzyżowania
junction_control	string	Rodzaj kontroli skrzyżowania
carriageway_hazards	string	Zagrożenia na drodze
urban_or_rural_area	string	Miasto lub wieś

Kolumna	Typ	Opis
trunk_road_flag	boolean	Czy droga krajowa
first_road_class	string	Klasa głównej drogi
first_road_number	int	Numer głównej drogi
second_road_class	string	Klasa drogi skrzyżowanej
second_road_number	int	Numer drogi skrzyżowanej

Wymiar: dim_weather

Table 8: Struktura wymiaru: dim_weather

Kolumna	Typ	Opis
weather_id	string	Klucz wymiaru pogody
temperature	decimal	Temperatura [°C]
precipitation	decimal	Opady [mm]
snow_depth	decimal	Pokrywa śnieżna [mm]
wind_speed	decimal	Prędkość wiatru [km/h]
wind_direction	int	Kierunek wiatru [°]
pressure	decimal	Ciśnienie atmosferyczne [hPa]
weather_coco	string	Kod warunków pogodowych
weather_desc	string	Opis pogody

```
-- Tabela: dim_time
CREATE TABLE dim_time (
    time_id VARCHAR(20) NOT NULL PRIMARY KEY,
    full_datetime DATETIME NOT NULL,
    date DATE NOT NULL,
    year INT NOT NULL,
    month INT NOT NULL,
    month_name VARCHAR(20) NOT NULL,
    day INT NOT NULL,
    day_of_week VARCHAR(20) NOT NULL,
    day_of_week_num INT NOT NULL,
    is_weekend BIT NOT NULL,
    week_number INT NOT NULL,
    quarter INT NOT NULL,
    hour INT NOT NULL,
    hour_band VARCHAR(20) NOT NULL,
    part_of_day VARCHAR(20) NOT NULL,
    is_night BIT NOT NULL,
    is_rush_hour BIT NOT NULL,
    is_holiday BIT NOT NULL
);
GO

-- Tabela: dim_location
CREATE TABLE dim_location (
```

```
    location_id VARCHAR(50) NOT NULL PRIMARY KEY,
    location_easting_osgr INT NOT NULL,
    location_northing_osgr INT NOT NULL,
    longitude FLOAT NOT NULL,
    latitude FLOAT NOT NULL,
    lsoav_of_accident_location VARCHAR(20),
    local_authority_district VARCHAR(50) NOT NULL,
    road_type VARCHAR(50) NOT NULL,
    speed_limit INT NOT NULL,
    junction_detail VARCHAR(50),
    junction_control VARCHAR(50),
    carriageway_hazards VARCHAR(100),
    urban_or_rural_area VARCHAR(20) NOT NULL,
    trunk_road_flag BIT NOT NULL,
    first_road_class VARCHAR(10) NOT NULL,
    first_road_number INT NOT NULL,
    second_road_class VARCHAR(10),
    second_road_number INT
);
GO

-- Tabela: dim_weather
CREATE TABLE dim_weather (
    weather_id VARCHAR(50) NOT NULL PRIMARY KEY,
    temperature DECIMAL(5,2),
    precipitation DECIMAL(5,2),
    snow_depth DECIMAL(5,2),
    wind_speed DECIMAL(5,2),
    wind_direction INT,
    pressure DECIMAL(6,2),
    weather_coco VARCHAR(20),
    weather_desc VARCHAR(100)
);
GO

-- Tabela: fact_collision
CREATE TABLE fact_collision (
    accident_index VARCHAR(50) NOT NULL PRIMARY KEY,
    date_id VARCHAR(20) NOT NULL,
    location_id VARCHAR(50) NOT NULL,
    weather_id VARCHAR(50), NOT NULL
    accident_severity SMALLINT NOT NULL,
    number_of_vehicles SMALLINT NOT NULL,
    number_of_casualties SMALLINT NOT NULL,
    police_attended BIT NOT NULL,
    escooter_involved BIT NOT NULL,
    enhanced_severity_collision SMALLINT NOT NULL,

    vehicle_type_car INT NOT NULL,
    vehicle_type_bus INT NOT NULL,
    vehicle_type_motorcycle INT NOT NULL,
```

```
    vehicle_type_goods INT NOT NULL,
    vehicle_type_other INT NOT NULL,
    vehicle_manoeuvre_turning_left INT NOT NULL,
    vehicle_manoeuvre_turning_right INT NOT NULL,
    vehicle_manoeuvre_overtaking INT NOT NULL,
    vehicle_left_hand_drive_count INT NOT NULL,
    avg_vehicle_age DECIMAL(5,2),
    avg_engine_capacity_cc DECIMAL(6,2),

    driver_sex_male INT NOT NULL,
    driver_sex_female INT NOT NULL,
    driver_age_band_0_25 INT NOT NULL,
    driver_age_band_26_50 INT NOT NULL,
    driver_age_band_51_plus INT NOT NULL,
    driver_purpose_commute INT NOT NULL,
    driver_purpose_education INT NOT NULL,
    driver_purpose_other INT NOT NULL,

    casualty_class_driver INT NOT NULL,
    casualty_class_passenger INT NOT NULL,
    casualty_class_pedestrian INT NOT NULL,
    casualty_severity_fatal INT NOT NULL,
    casualty_severity_serious INT NOT NULL,
    casualty_severity_slight INT NOT NULL,
    casualty_age_band_0_15 INT NOT NULL,
    casualty_age_band_16_30 INT NOT NULL,
    casualty_age_band_31_60 INT NOT NULL,
    casualty_age_band_60_plus INT NOT NULL,

    CONSTRAINT FK_fact_time FOREIGN KEY (date_id) REFERENCES dim_time(time_id),
    CONSTRAINT FK_fact_location FOREIGN KEY (location_id) REFERENCES
   dim_location(location_id),
    CONSTRAINT FK_fact_weather FOREIGN KEY (weather_id) REFERENCES dim_weather(
   weather_id)
);
GO
```

Listing 1: Deklaracja modelu hurtowni (schemat gwiazdy)

5 Kluczowe miary i atrybuty w modelu

W zaprojektowanym modelu hurtowni danych wyróżniamy kluczowe **miary** (*fact measures*) oraz istotne **atrybuty wymiarów**, które stanowią podstawę analiz i raportów. Poniżej przedstawiono ich listę wraz z krótkim opisem:

Główne miary analityczne

- **Liczba wypadków** – podstawowa miara faktu, najczęściej obliczana jako zliczenie rekordów w `fact_collision`. Może być reprezentowana explicite przez kolumnę `accident_count = 1`, co ułatwia agregację.

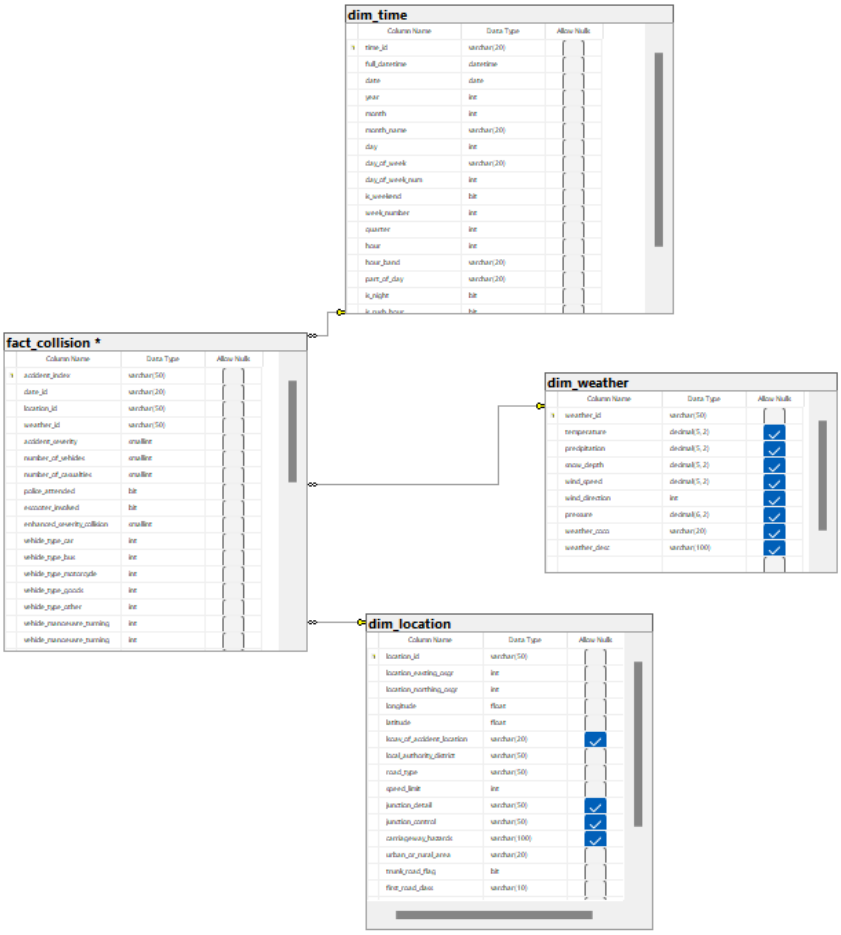

Figure 1: Diagram hurtowni

- **Liczba ofiar (ogółem)** – suma kolumny `number_of_casualties`; informuje o łącznej liczbie poszkodowanych.
- **Liczba ofiar śmiertelnych / ciężko / lekko rannych** – oddzielne miary na podstawie agregacji odpowiednich kolumn w `fact_collision`, np. `casualty_severity_fatal`, `...serious`, `...slight`.
- **Liczba pojazdów** – suma kolumny `number_of_vehicles`; wykorzystywana m.in. do analizy przeciętnego rozmiaru kolizji.
- **Średnia liczba pojazdów na wypadek** – wartość wyliczana jako: $\frac{\sum \texttt{number\_of\_vehicles}}{\sum \texttt{accident\_count}}$.
- **Średnia liczba ofiar na wypadek** – $\frac{\sum \texttt{number\_of\_casualties}}{\sum \texttt{accident\_count}}$; pozwala określić przeciętną dotkliwość zdarzenia.
- **Procent wypadków śmiertelnych** – udział liczby wypadków o kategorii `fatal` względem wszystkich: $\frac{\texttt{count\_fatal}}{\texttt{total\_accidents}} \cdot 100\%$.
- **Liczba wypadków w danych warunkach** – wynik filtrowania po wymiarach (np. `weather_desc = 'Deszcz'` lub `road_type = 'A'`).

- **Średnia temperatura podczas wypadków** – średnia wartość z kolumny `temperature` w `dim_weather`, powiązana z rekordami wypadków.
- **Liczba wypadków przy opadach** – suma wypadków, w których `precipitation > 0`; może być uproszczona przez binarny atrybut `was_precipitation`.
- **Liczba wypadków z udziałem pieszych** – suma wypadków, w których wystąpiła ofiara typu `pedestrian`, np. na podstawie flagi `involved_pedestrian_flag`.
- **Liczba wypadków z udziałem wybranego typu pojazdu** – np. motocykli, rowerów, ciężarówek; możliwe do realizacji przez flagi w fakcie lub filtrację po `vehicle_type`.
- **Liczba wypadków w godzinach szczytu** – zliczenia bazujące na filtrze po atrybucie `is_rush_hour` w `dim_time`.

Najczęściej wykorzystywane atrybuty wymiarów

- **Z wymiaru czasu** (`dim_time`)**:** `year`, `quarter`, `month`, `day_of_week`, `is_holiday`, `hour`, `is_rush_hour`.
- **Z wymiaru lokalizacji** (`dim_location`)**:** `region`, `urban_or_rural`, `road_type`, `speed_limit`.
- **Z wymiaru pogody** (`dim_weather`)**:** `weather_desc`, `precipitation`, `temperature`.
- **Z wymiaru typu pojazdu** (`dim_vehicle_type`)**:** `vehicle_type_desc` – np. do filtrowania wypadków z rowerami, motocyklami itd.
- **Z wymiaru typu ofiary** (`dim_casualty_type`)**:** `casualty_type_desc` – do przeglądu kolizji z pieszymi, pasażerami itd.
- **Z wymiaru stanu nawierzchni** (`dim_road_condition`)**:** `road_surface_desc`.
- **Z wymiaru ciężkości zdarzenia** (`dim_accident_severity`)**:** `severity_desc` – pozwala analizować np. tylko wypadki śmiertelne.

Uwagi dodatkowe. Niektóre atrybuty demograficzne, takie jak płeć czy wiek kierowców/ofiar, nie są obecnie częścią pełnych wymiarów w modelu gwiazdy. Można je agregować na poziomie źródłowych tabel `Vehicles` i `Casualties`, lub — opcjonalnie — dodać do tabeli faktów uproszczone wskaźniki (np. `pct_male_drivers`, `avg_driver_age`). Tego rodzaju agregaty warto jednak stosować tylko w uzasadnionych przypadkach, z uwagi na potencjalne zniekształcenie danych przy dużych zróżnicowaniach.

6 Architektura rozwiązania – komponenty i przepływ danych

Architektura projektowanej hurtowni danych opiera się na klasycznym podejściu warstwowym, w którym dane przepływają od źródła, przez warstwę przetwarzania, aż do końcowej warstwy prezentacyjnej. Główne komponenty systemu to:

Źródła danych

- **Pliki CSV (2019–2023)** – dane o wypadkach drogowych publikowane przez *Department for Transport (DfT)* na portalu `https://data.gov.uk`. Każdy rok zawiera trzy zestawy: `Collisions`, `Vehicles` oraz `Casualties`.
- **API Meteostat** – zewnętrzne źródło danych pogodowych, dostępne online i wykorzystywane dynamicznie podczas przetwarzania danych (ETL) za pomocą skryptów Python. Dane nie są pobierane w formie plików statycznych, lecz na żądanie, dla konkretnego czasu i lokalizacji wypadku.

Strefa pośrednia (Staging Area)

W celu ułatwienia kontroli jakości i przekształceń danych, wykorzystywana jest warstwa staging, zaimplementowana w relacyjnej bazie danych (np. SQL Server). Dane z plików CSV i API Meteostat są tymczasowo ładowane do następujących tabel:

- `Stg_Collisions_Raw` – dane o wypadkach.
- `Stg_Vehicles_Raw` – dane o pojazdach.
- `Stg_Casualties_Raw` – dane o poszkodowanych.
- `Stg_Weather_Raw` – dane pogodowe (zobserwowane warunki w momencie zdarzenia).

Warstwa integracyjna i analityczna (Data Warehouse)

Po wstępnym załadowaniu i przekształceniu danych, trafiają one do hurtowni danych zbudowanej w modelu gwiazdy. Składa się ona z:

- jednej tabeli faktów: `fact_collision`, zawierającej informacje zagregowane na poziomie pojedynczego wypadku,
- wielu tabel wymiarów: `dim_date`, `dim_location`, `dim_weather`, `dim_vehicle_type`, `dim_casualty_type`, `dim_road_condition`, `dim_accident_severity`, itd.

Wszystkie relacje pomiędzy tabelą faktów a wymiarami są realizowane za pomocą kluczy obcych. Kluczowe kolumny są indeksowane, co zapewnia wysoką wydajność zapytań analitycznych. Docelowa baza danych jest hostowana w systemie RDBMS, takim jak Microsoft SQL Server.

Warstwa prezentacyjna (BI)

Dane z hurtowni są udostępniane końcowym użytkownikom poprzez narzędzia Business Intelligence, takie jak Power BI lub SQL Server Reporting Services (SSRS). Dashboard BI składający się z sześciu stron umożliwia interaktywną analizę danych, w tym:

- eksplorację danych przez filtry i przekroje czasowe, przestrzenne oraz demograficzne,
- generowanie raportów zbiorczych (np. liczba wypadków w regionach),
- monitorowanie wskaźników bezpieczeństwa drogowego (KPI),
- analizę trendów oraz sezonowości zdarzeń.

Opis przepływu danych (pipeline)

1. Pobranie danych źródłowych (CSV i API Meteostat).
2. Załadunek danych do tabel staging w bazie danych.
3. Przekształcenie i wzbogacenie danych (agregacja, czyszczenie, dołączenie pogody).
4. Ładowanie wymiarów i tabeli faktów do hurtowni danych.
5. Udostępnienie danych do narzędzia BI w celu prezentacji i analizy.

Rola komponentów:

- **SSIS (SQL Server Integration Services)** – orkiestracja przepływu danych (ETL).
- **Python (Meteostat API)** – pobieranie danych pogodowych na żądanie.
- **SQL Server (DWH)** – magazyn danych oraz warstwa modelu analitycznego.
- **Power BI / SSRS** – wizualizacja i eksploracja danych przez użytkownika końcowego.

7 Szczegółowy opis procesu ETL w SSIS

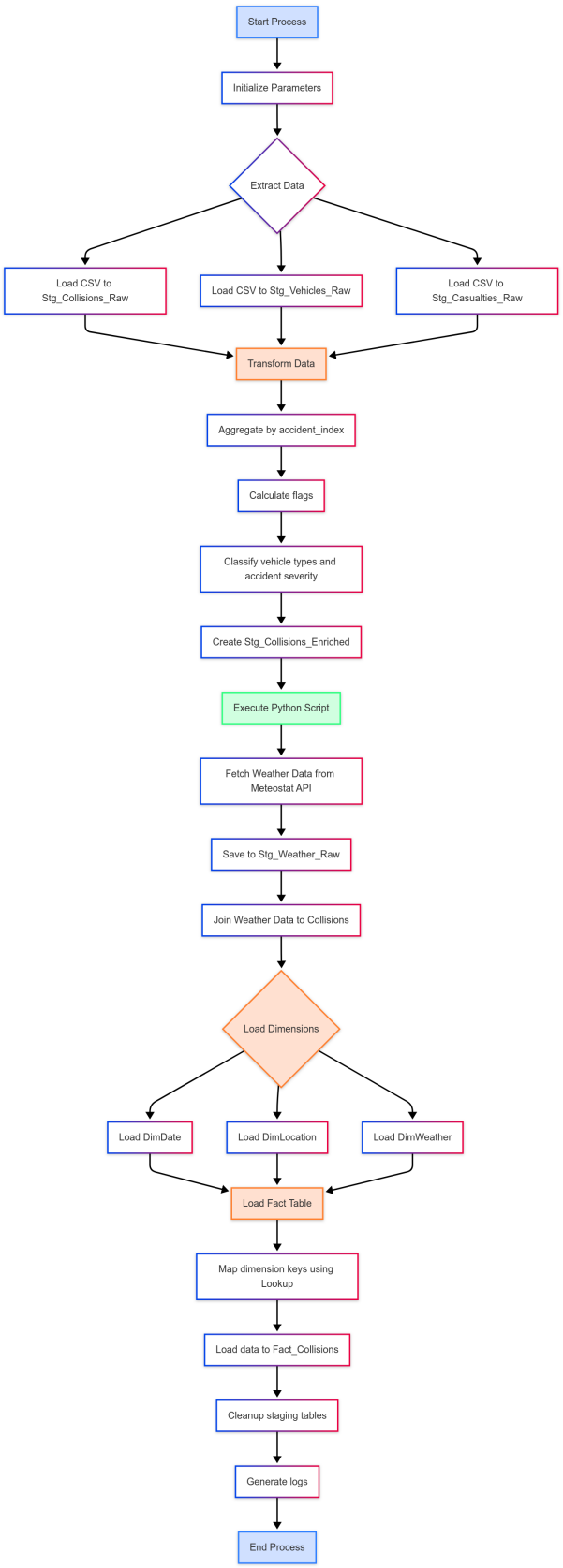

Figure 2: Diagram aktywności

Proces ETL został zaprojektowany jako zautomatyzowany pipeline, realizowany głównie w środowisku SSIS (SQL Server Integration Services), z wykorzystaniem dodatkowego skryptu Python do integracji danych pogodowych. Całość opiera się na przetwarzaniu danych z plików źródłowych i API do hurtowni danych zgodnej ze schematem gwiazdy.

Krok 1: Inicjalizacja procesu

Proces rozpoczyna się od uruchomienia paczki SSIS – manualnie lub automatycznie (np. przez SQL Server Agent). W ramach inicjalizacji ustawiane są parametry środowiskowe:

- rok przetwarzanych danych (dla przetwarzania przyrostowego),
- ścieżki do plików CSV (dla dynamicznego ładowania),
- klucz API Meteostat (przekazywany jako zmienna środowiskowa lub parametr wejściowy do skryptu Python).

Krok 2: Załadunek danych źródłowych (Extract)

Dla każdego zbioru (Collisions, Vehicles, Casualties) tworzony jest osobny Data Flow Task w SSIS. Dane z plików CSV są ładowane do tabel staging:

- `Stg_Collisions_Raw`
- `Stg_Vehicles_Raw`
- `Stg_Casualties_Raw`

Zastosowano komponenty Flat File Source (CSV) oraz OLE DB Destination (SQL Server). W razie potrzeby możliwe jest wcześniejsze scalanie plików CSV lub przetwarzanie roczne z wykorzystaniem pętli `Foreach Loop Container`.

Krok 3: Agregacja i wzbogacenie danych (Transform)

Po załadowaniu danych surowych wykonywane są następujące transformacje:

- agregacja danych o ofiarach i pojazdach do poziomu wypadku (grupowanie po `accident_index`),
- wyliczenie flag (np. udział pieszych, obecność motocykli),
- klasyfikacja typu pojazdu i ciężkości wypadku na podstawie reguł logicznych,
- uzupełnienie rekordów wypadków o wartości skumulowane i atrybuty analityczne.

Wynikiem tego kroku jest tabela `Stg_Collisions_Enriched`, zawierająca wzbogacone dane o wypadkach, gotowe do połączenia z pogodą i wymiarami.

Krok 4: Integracja danych pogodowych (Python)

Dane meteorologiczne są pobierane dynamicznie z API Meteostat za pomocą zewnętrznego skryptu Python, uruchamianego z poziomu SSIS (komponent Execute Process Task). Logika działania obejmuje:

- wczytanie listy wypadków z lokalizacją i czasem (eksport z `Stg_Collisions_Enriched`),
- dla każdego wypadku pobranie godzinowego rekordu pogodowego (temperatura, opady, wiatr, itp.),
- zapis wyników do pliku CSV i import do tabeli `Stg_Weather_Raw`.

Pobieranie danych odbywa się z dokładnością do godziny i miejsca, a API zapewnia interpolację wartości w razie braku lokalnej stacji meteorologicznej.

Krok 5: Dołączenie danych pogodowych

Tabela `Stg_Collisions_Enriched` zostaje zaktualizowana o dane pogodowe. Łączenie odbywa się po `accident_index`. W razie potrzeby możliwe jest wcześniejsze uwzględnienie pogody już w etapie tworzenia enriched.

Krok 6: Ładowanie wymiarów

Dla każdego z wymiarów stosowane są odpowiednie strategie ładowania:

- **DimDate** – generowanie pełnego kalendarza na lata 2019–2023,
- **DimLocation** – wybór unikalnych kombinacji lokalizacji z enriched,
- **DimWeather** – unikalne kombinacje cech pogodowych,

Wszystkie wymiary posiadają techniczne klucze główne (surrogate keys) typu `identity` oraz indeksy na kolumnach naturalnych.

Krok 7: Ładowanie tabeli faktów

Dane z tabeli `Stg_Collisions_Enriched` są ładowane do tabeli `Fact_Collisions` za pomocą Data Flow Task. Proces obejmuje:

- łańcuch transformacji typu `Lookup` w celu pozyskania kluczy obcych do wymiarów,
- uzupełnienie kolumn miar oraz flag,
- zapis danych do `Fact_Collisions` z wykorzystaniem trybu `FastLoad`.

Dla zapewnienia integralności danych rekomenduje się ładowanie wymiarów przed faktami oraz weryfikację spójności (np. liczba rekordów, obecność kluczy obcych).

Krok 8: Finalizacja procesu

Na końcu procesu wykonywane są opcjonalne czynności porządkowe:

- czyszczenie tymczasowych tabel staging,
- generowanie logów, komunikatów o sukcesie lub błędach,
- (opcjonalnie) obsługa wersjonowania lub inkrementalnego przetwarzania w kolejnych latach.

Uwagi końcowe. Proces został zaprojektowany z myślą o skalowalności i wydajności — obsługuje dane z kilku lat, zakłada łatwość przyszłych aktualizacji (np. za rok 2024) oraz zapewnia spójność danych między wymiarami a faktem. W ramach ewolucji rozwiązania możliwe jest także dodanie przetwarzania różnicowego (np. aktualizacja danych przy korektach DfT) lub integracja z innymi źródłami zewnętrznymi.

8 Testowanie i zapewnienie jakości danych (opcjonalnie)

Aby zagwarantować poprawność działania zaprojektowanego rozwiązania oraz wiarygodność danych w hurtowni, przewidziano zestaw testów weryfikujących jakość danych na różnych etapach przetwarzania – od plików źródłowych, przez proces ETL, po dane końcowe w modelu.

1. Testy poprawności danych wejściowych

- **Struktura plików CSV** – weryfikacja zgodności liczby i nazw kolumn z dokumentacją STATS19. Błędne nazwy kolumn mogą skutkować błędnym mapowaniem w SSIS.
- **Spójność danych testowych** – losowa kontrola kilku rekordów załadowanych do staging w porównaniu z plikami źródłowymi.
- **Zakresy i wartości graniczne** – np. sprawdzenie, czy `accident_severity` przyjmuje tylko wartości 1–3, a współrzędne geograficzne mieszczą się w granicach UK.

2. Testy procesu ETL

- **Testy jednostkowe kroków transformacji** – np. ręczne porównanie liczby ofiar dla wybranych wypadków z agregacjami w `Stg_Casualties_Agg`.
- **Testy integracyjne** – pełne uruchomienie procesu dla jednego roku (np. 2019) i weryfikacja danych w tabeli faktów.
- **Testy wydajnościowe** – pomiar czasu wykonania poszczególnych etapów (szczególnie pobierania pogody dla dużych zbiorów).

3. Testy kompletności i spójności danych

- **Sumy kontrolne** – np. suma pojazdów z pliku collisions powinna pokrywać się z sumą z tabeli `Stg_Vehicles_Raw`.
- **Weryfikacja losowych wypadków** – sprawdzenie, czy:
  - każdy `accident_index` występuje w tabeli faktów dokładnie raz,
  - wartości miar (liczba ofiar, pojazdów) są zgodne z danymi źródłowymi,
  - dane pogodowe zostały poprawnie przypisane i są realistyczne.
- **Poprawność kluczy obcych** – np. brak wierszy w `Fact_Collisions`, które nie mają dopasowania w wymiarach (test referencyjny `LEFT JOIN`).
- **Unikalność rekordów** – sprawdzenie, czy `accident_index` jest unikalny w tabeli faktów (można dodatkowo wymusić ograniczenie `UNIQUE`).

4. Testy integracji danych pogodowych

- **Spójność między danymi policyjnymi i meteostat** – np. jeżeli w `weather_conditions` występuje kod oznaczający opady, to wartość `prcp` z Meteostat powinna być większa od zera.
- **Kompletność danych pogodowych** – liczba wierszy w `Fact_Collisions` powinna odpowiadać liczbie wierszy w `Stg_Collisions_Raw`.

5. Testy użytkowe (User Acceptance Testing)

- **Weryfikacja interpretacji danych** – z udziałem użytkowników końcowych (np. analityków), którzy oceniają czy wyniki raportów są spójne z ich wiedzą (np. liczba wypadków w regionie X).
- **Testy dashboardu BI** – sprawdzenie funkcjonalności filtrów, poprawności prezentowanych wskaźników oraz czasu odpowiedzi raportów.

Podsumowanie

Wdrożone mechanizmy testowe i walidacyjne pozwalają zachować wysoką jakość danych na każdym etapie procesu — od importu danych źródłowych po końcową prezentację w narzędziu BI. Hurtownia danych została zaprojektowana w sposób umożliwiający szybką identyfikację błędów, analizę ich przyczyn oraz wdrożenie działań naprawczych. Dzięki temu użytkownicy końcowi mogą mieć pewność, że prezentowane dane są rzetelne i wspierają podejmowanie trafnych decyzji analitycznych.