# Goal

**To provide an interactive means for analysts in the mayor’s office and other stakeholders from New York City (NYC) to easily access motor vehicle collision information interactively thereby aiding them to make informed decisions.**

# System Description

## Stakeholders

* Mayors office
* Ordinary citizens (secondary stakeholder)

## Main Systems of Interest

* A Power BI Dashboard that allows **analysts** in the mayor's office to visualize the number of crashes by period and location.
* Also, our Power BI Dashboard should provide some statistical information about crashes in New York City.
* Finally, our Power BI Dashboard will show the closest police station as well as closest hospitals to crashes.

## Enabling Systems

Systems that will enable us to build our primary system are automated cron jobs or schedulers that will fetch, clean, and store data from the sources listed below

## **Data Sources**

* Motor Vehicle Collisions: <https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95>
* Historical Traffic Data <https://data.cityofnewyork.us/Transportation/Traffic-Volume-Counts-2012-2013-/p424-amsu>
* List of hospitals as of 2011 <https://data.cityofnewyork.us/Health/NYC-Health-Hospitals-Facilities-2011/ymhw-9cz9>
* List of police departments <https://www1.nyc.gov/site/nypd/bureaus/patrol/precincts-landing.page>

## **ETL**

* Azure Data Factory

## **Storage**

* Azure Data Lake Storage

## Dashboards/Reports

* Azure Power BI
* Azure Synapse Analytics

# System Boundaries

## **Roles and Responsibilities**

* Automated Software i.e Background Processes
* Data Owners:
  + Police Department (NYPD): Motor Vehicle Collisions
  + Department of Transportation (DOT): Historical Traffic Data
  + NYC Health + Hospitals: List of hospitals
  + Department of City Planning (DCP): List of police stations
* Data Analysts in Mayors office
* Developers/System Administrators
* General Public

## **Functionalities of System**

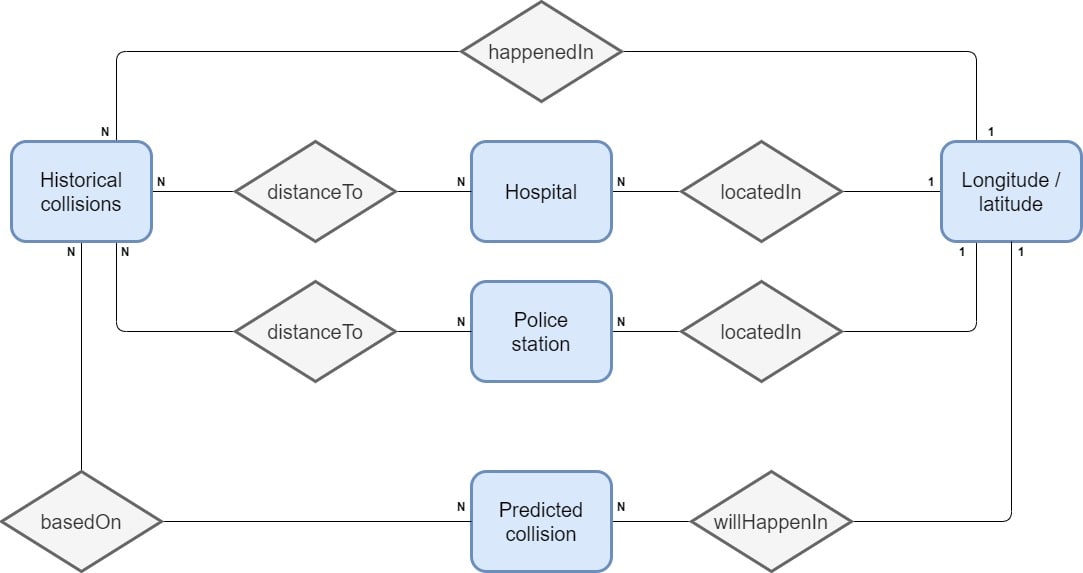
* Easily query data about crushes with little to no technical knowledge
* Easily create and share informative visualizations with people
* Access to basic informative visualizations:
  + Visualize places where accidents happened for the last several days
  + Visualize nearest hospitals to the places of accidents
* Provide statistics and predictions:
  + Get information about what are the districts/crossroads where accidents occurred most frequently for last week/month
  + Predict districts where an accident is more likely to occur for the next week
  + Get information about how often do crashes take place on a specific street/crossroad
* Easily export data, visualizations, or reports in some desired format
* Easily import current information about accidents

# BPM Diagam

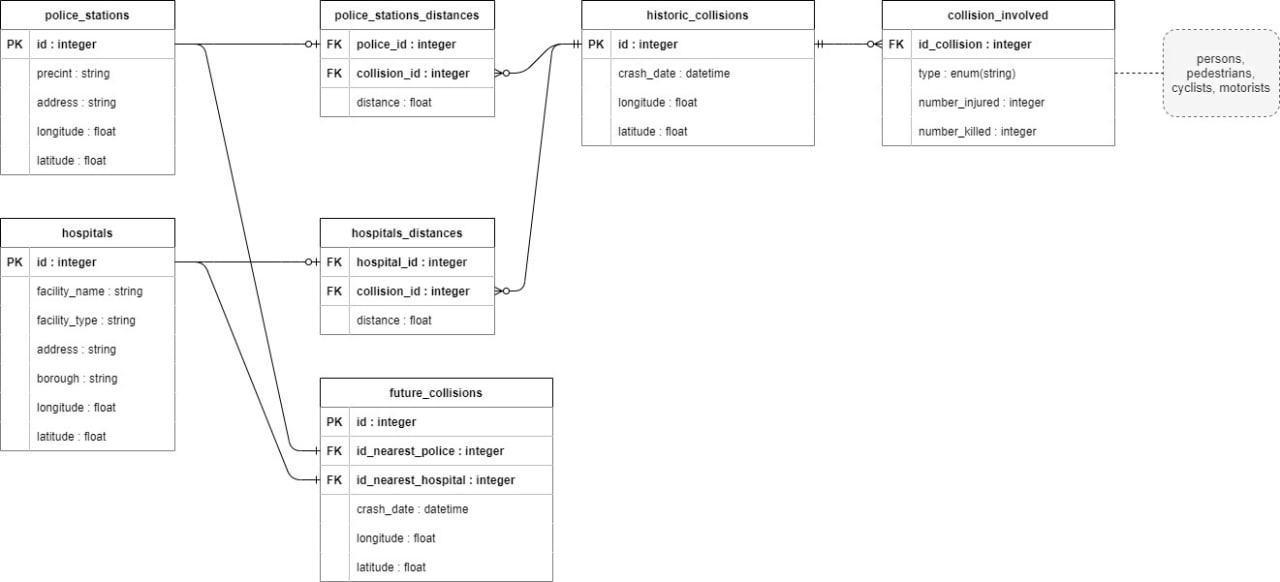
Diagram, schematic

Description automatically generated

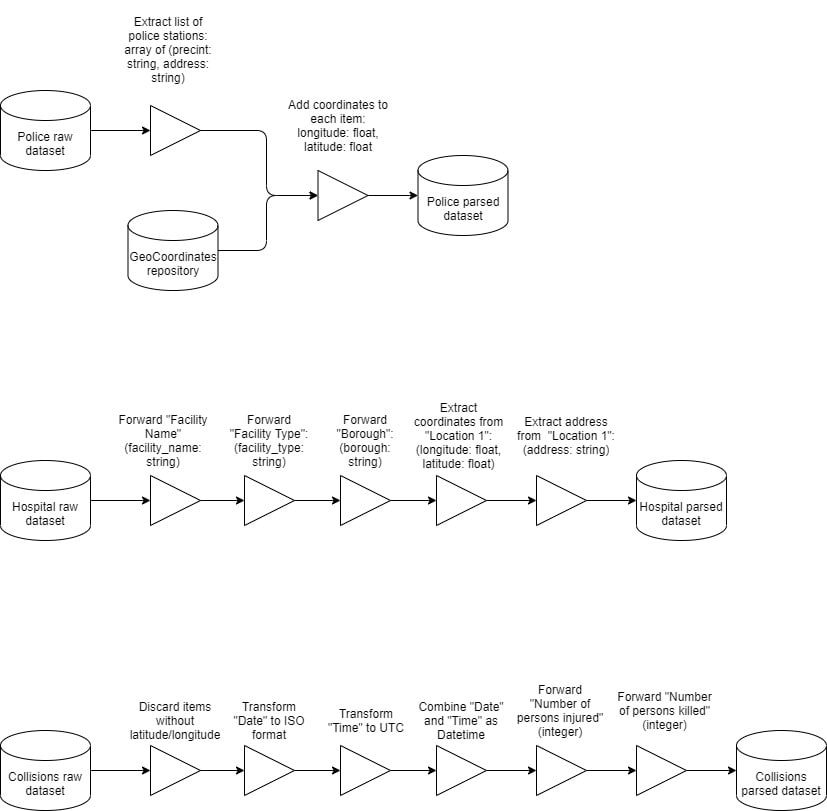
# Abstract Schema Diagram



# Schema Diagram



# Data Flow Diagram



# Analytical Diagram

Diagram

Description automatically generated

# SRS

We prepared an SRS document that can be found [here](https://github.com/acquayefrank/nyc-crashes/blob/main/documents/srs_nyc-crashes.doc)

# Prototype

For our prototype we developed a solution using docker, which will allow the analysts in the mayors office to analyze crush information:

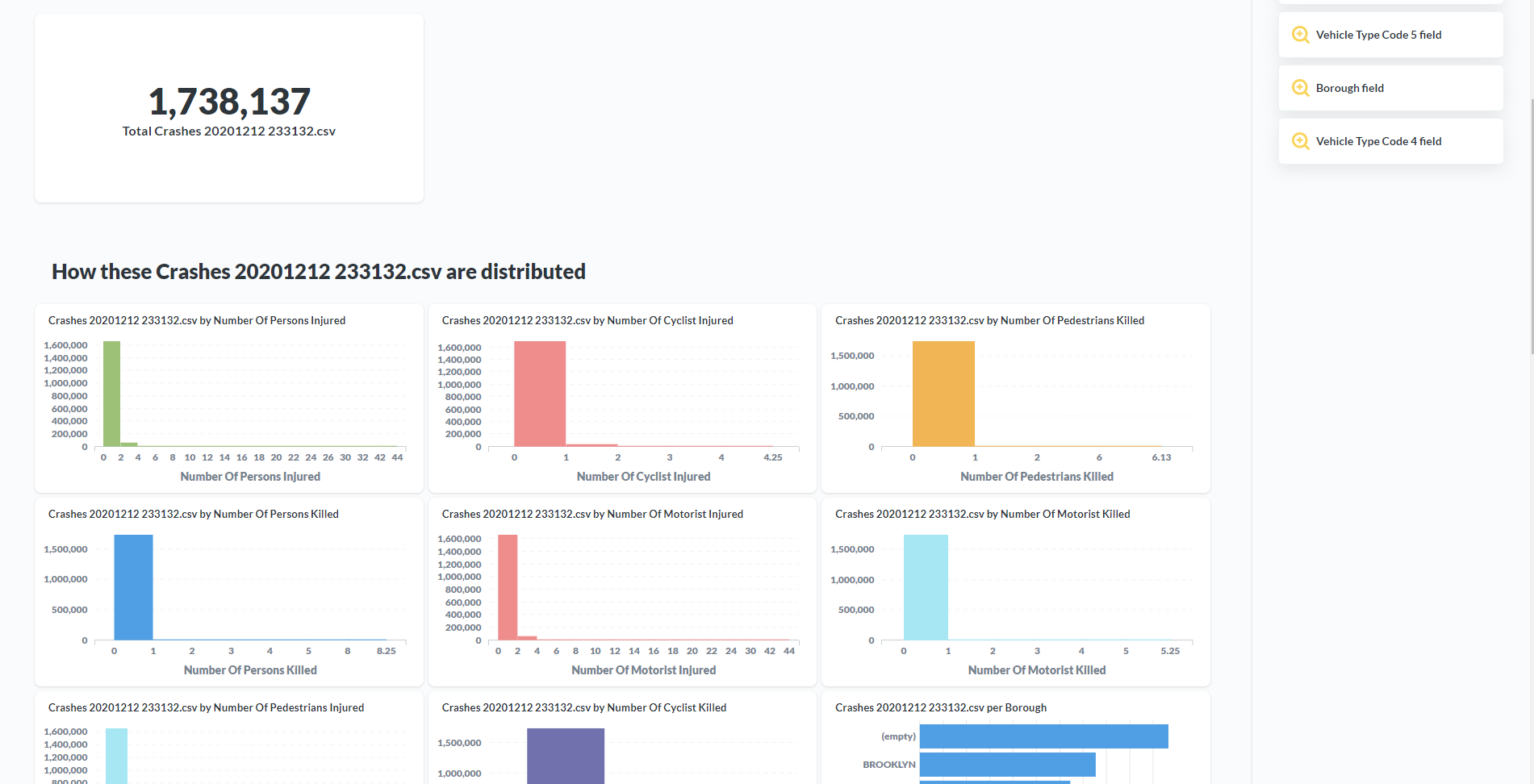
In our docker deployment we have the following:

**metbase dashboard:**

Metabase is an open source solution which allows people with little to no technical knowledge to generate and share analytics within an organization.

With metabase we provide the analysts in the mayors office the opportunity to interactively create reports and dashboards through SQL or a WYSIWYG editor.

*Screenshots from metabase below:*



***Figure 1: Sample Dashbaord from metabase***

Graphical user interface, text, application, email

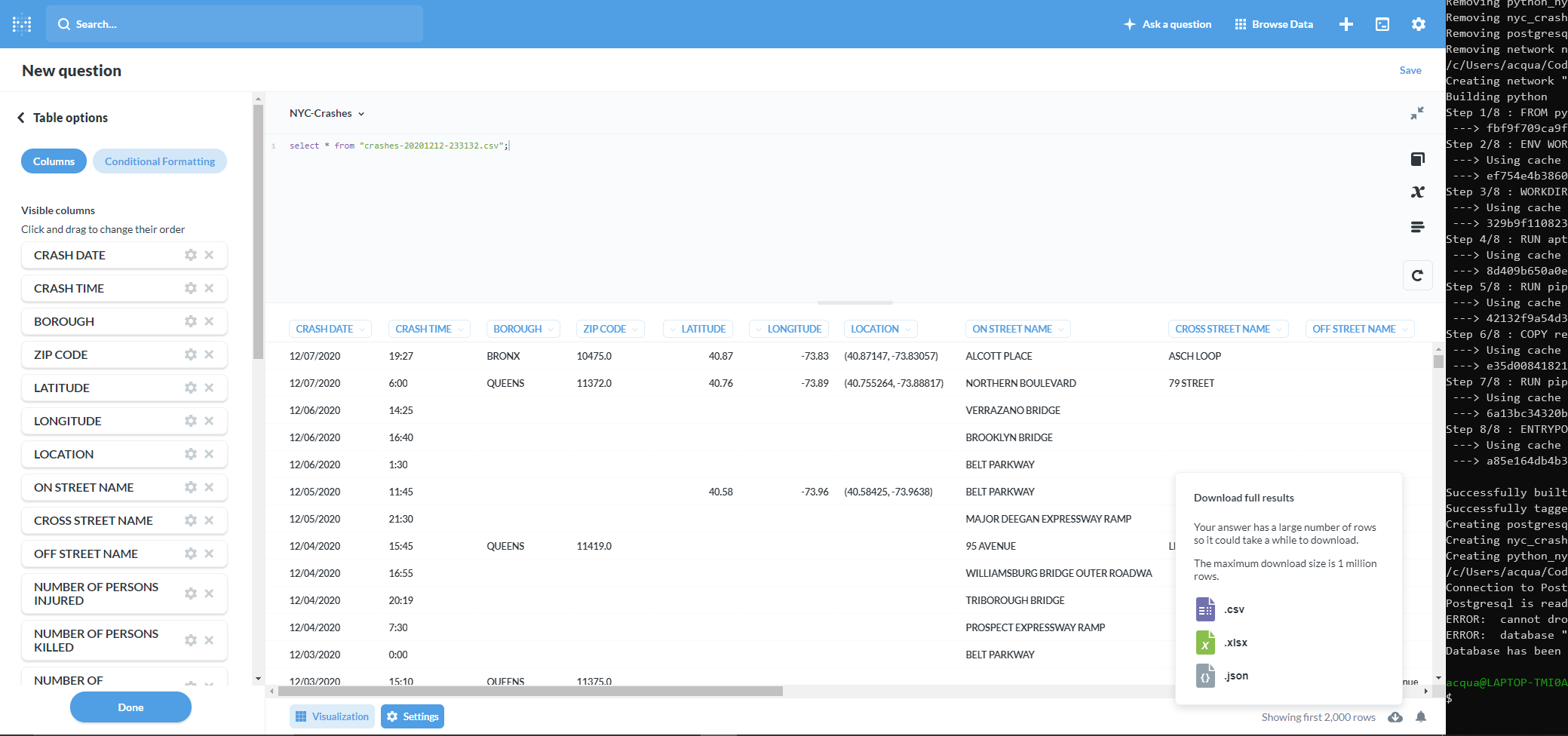
Description automatically generated

***Figure 2: Exploring data in metabase through SQL***

Graphical user interface, text, application, email

Description automatically generated

***Figure 3: Exploring metabase with WYSIWYG Editor***

******

***Figure 4: Easily export data from metabase.***

**Jupyter notebook:**

We also provide jupyter notebooks for analysts to easily interact with data from python and generate solutions.

Map

Description automatically generated

***Figure 5: A map of accidents in NYC generated in a jupyter notebook.***

Chart, bar chart

Description automatically generated

***Figure 6: Some reports generated in a jupyter notebook.***

**Automated Reports:**

**In addition to the jupyter notebook and metabase we generate a monthly, weekly report for the mayor and citizens who have no technical knowledge:**

* + - **A map of crashes slong with close hospitals and police stations**
    - **A report for citizens**
    - **A report ofr mayor**

**A sample of these reports can be found** [here](https://github.com/acquayefrank/nyc-crashes/tree/main/src/result_folder)

**NB// All information pertaining to the project can be found on the github repo including source code for our prototype:**

**https://github.com/acquayefrank/nyc-crashes**