

Mock user statement:

My name is Thomas and I work for the Minnesota Department of Transportation. I'm reaching out for you to help us build a pipeline to evaluate traffic impacts due to large events. They cause a lot of inefficiency in our roadways. The online system needs to show a map with live conditions and areas of high potential traffic in the future. Can you help us with this?

Also, we have some other developers that want to get the results in coded format. They can receive geojson. Can you do that?

Slightly alternate approach:

- If predicting traffic flow at a certain area given an event is too difficult, then select a location, and list what events regularly impact the traffic there

## Traffic Impact Prediction Specification

<b>Product Owner</b>	Metro Transit	
<b>Engineering Lead</b>	Greg Kohler	kohle147@umn.edu
<b>Geocomputing Lead</b>	Laure Briol	briol009@umn.edu
<b>Data Lead</b>	Logan Gall	gall0487@umn.edu
<b>Required Approvers</b>	Thomas Tankerson	tankert@dot

## Overview

### Objectives

Develop a system that:

- Predicts traffic congestion caused by large events.
- Provides information to users about potential traffic events.
- Integrates with existing traffic visualizations.
- Allows users to customize their based on location, event type, and severity of traffic.
- Provides an API service for data requests.

### Problems:

- Little predictive analysis: Current consumer systems lack the ability to predict traffic caused by large events.
- Reactive systems: Existing traffic controls are mostly reactive to live conditions.
- Limited: Users can't easily preview traffic conditions according to their specific needs and preferences.

- Significant impacts: The impact of large events on local traffic is often underestimated, leading to insufficient planning and management.

#### Who It Directly Affects:

- Commuters and residents: Those who travel regularly and live close to event locations.
- Attendees: People attending the events who need information for better travel planning.
- Traffic controllers: Government and private entities responsible for traffic management.
- Organizers: Those organizing large events can use traffic flow information for better coordination.

#### Why it is Important Solve for MnDOT:

- Efficient management: Proactively managing traffic flow, especially during large events, can significantly reduce congestion and improve commute times.
- User experience: Providing customized traffic updates will improve the overall experience for commuters and event attendees.
- Resource allocation: Traffic authorities can allocate necessary resources based on predicted impacts.
- Data-driven planning: Organizers can use traffic predictions for more efficient planning of events, leading to better safety and smooth operations.

## Motivation

Traffic is a major inefficiency in transportation that wastes time and causes excess pollution. While daily traffic is expected and common, large events don't happen every day and the traffic impacts are difficult to plan around. A better way to predict traffic conditions can be useful for many to optimize hassle and time.

Our expected outcome is to build a practical user web service. The service will allow users to view traffic conditions and prepare for travel based on upcoming events. Local residents, event organizers, and passing commuters can plan their travel knowing the traffic impacts of an upcoming event.

## Definitions

*This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendixes in the SRS or by reference to other documents.*

## Scope

*This section describes the scope of the project. It describes the functional and nonfunctional requirements as well as what items are out of scope for this iteration of the project.*

# Functional Requirements

## Traffic Data

- [ESRI World Traffic Service](#)
  - Pulls data from [HERE](#)
- Live traffic flow
- Updates every 5 minutes
- Historical traffic flow
- [API](#)

## Event Data

- Source TBD
- Essential -- a small area with many major events (UMN)
- Nice to have -- Entire TC area
- Optional -- Even larger area

## Traffic prediction model

- Takes input of event location, time, and date
- Takes input of historical traffic on that date
- Outputs an area with high probability of slow traffic flow (linestring/polyline or other)
- Essential -- Outputs some area prediction of traffic (heatmap)
- Nice to have -- Outputs specific roads with poor traffic (lines)
- Optional -- Outputs an entire area's traffic prediction, both good and bad roads (lines)

## Web Map Interface

- Road network map
- Essential -- Highlights areas with poor traffic based on user input
- Nice to have/Optional -- Live data and predictions of when traffic will start to get bad

## GeoJson API

- User input date/time/event of interest
- Google Cloud API
- Returns areas with potential congestion

*Functional requirements are capabilities that the product must do to satisfy specific user needs.*

*Functional requirements are sometimes referred to as business requirements. They describe capabilities that the intended product can perform to enable business users to do some part of their work and carry on with their business (operational) work. Each requirement should be prioritized with Essential, Nice to have, or optional priorities.*

# Non-Functional Requirements

- Easy to use interface

- Essential -- Usable by someone with desktop software and knows how to read time-series traffic maps
- Nice to have -- Usable by power users of typical map applications i.e. Google/Apple maps
- Low processing time
  - Essential -- < 5min for a new event area/request
  - Nice to have -- < 1 min
  - Optional -- Near instant for any upcoming events
- Reliable
  - Will be able to consistently predict bad traffic for a given location/event
  - Essential -- A few limited events (i.e. UMN sports)
  - Nice to have -- TC very large recurring events (i.e. professional sports)
  - Optional -- Smaller events (Shows/Plays, other events)

*Non-functional requirements include usability, performance, reliability and security requirements. These are qualities that the product must have. Technical requirements also fall under the non-functional category. Each requirement should be prioritized with Essential, Nice to have, or optional priorities.*

## Out of Scope Requirements

- Any scale larger than Twin Cities (or even just UMN campuses)
- Predictive modeling for events that do not occur regularly
- Prescriptive analysis / giving optimal routes that avoid traffic
- (Maybe) Filtering out noise from regular daily traffic

*These are requirements that we are deeming out of scope for this iteration of the project. We list them here in order to be unambiguous and entirely clear with respect to project scope.*

## Persona Acceptance Criteria

Who are the stakeholders impacted by the project's success? What are they trying to achieve?

### **As a I developer I ...**

- Require access to API's so that I can download the data and perform analysis
- Require .... so that I ...

### **As an Operator I...**

- Require reliable data streams so that I can maintain database results without error
- Require .... so that I ...

### **As an end user I**

- Require a web interface so that I can utilize the data effectively
- Require regular updates so that I can view live data when planning events

# Open Questions

*What are assumptions we are making and known risks with respect to the feasibility of a project? This can be with respect to licensing, staffing, or how a particular requirement will be achieved. All open questions must be addressed by the design stage.*

- How do we go about making decent models for traffic flow?
  - How to handle input?
    - Traffic data
      - Line strings
    - Events occur over a period of time
      - Spatio-temporal analysis
      - Look at hour or two before and after
  - How to handle output?
    - Heatmap
    - Line string(s)
- Gathering event information
  - API?
    - Limited to API events
  - Manual lists?
    - Significantly reduces scope
    - But ensures high data quality

# Dependencies

*What dependencies does this specification have on other projects, components, or software? All dependencies must be highlighted here as every dependency is a potential risk.*

Closed Source Dependencies:

- ESRI Traffic data
  - Built upon HERE API
- ESRI Infrastructure
  - ArcPro
  - ArcOnline
- Google Cloud
- GitHub

Open Source Dependencies:

- Flask
- PostGIS

# References

ESRI World Traffic Service:

<https://umh.maps.arcgis.com/home/item.html?id=ff11eb5b930b4fabba15c47feb130de4>

ESRI API Documentation:

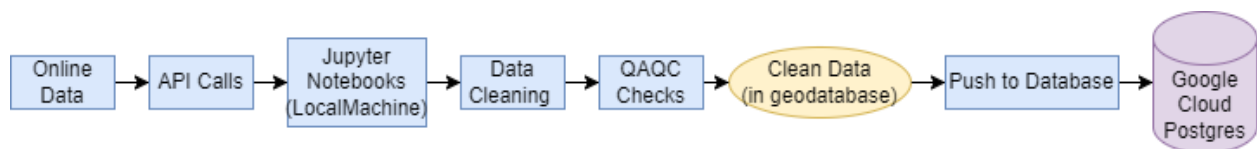
<https://developers.arcgis.com/documentation/mapping-apis-and-services/routing/>

This subsection should provide a complete list of all documents referenced elsewhere in the SRS; This information may be provided by reference to an appendix or to another document.

## Appendix

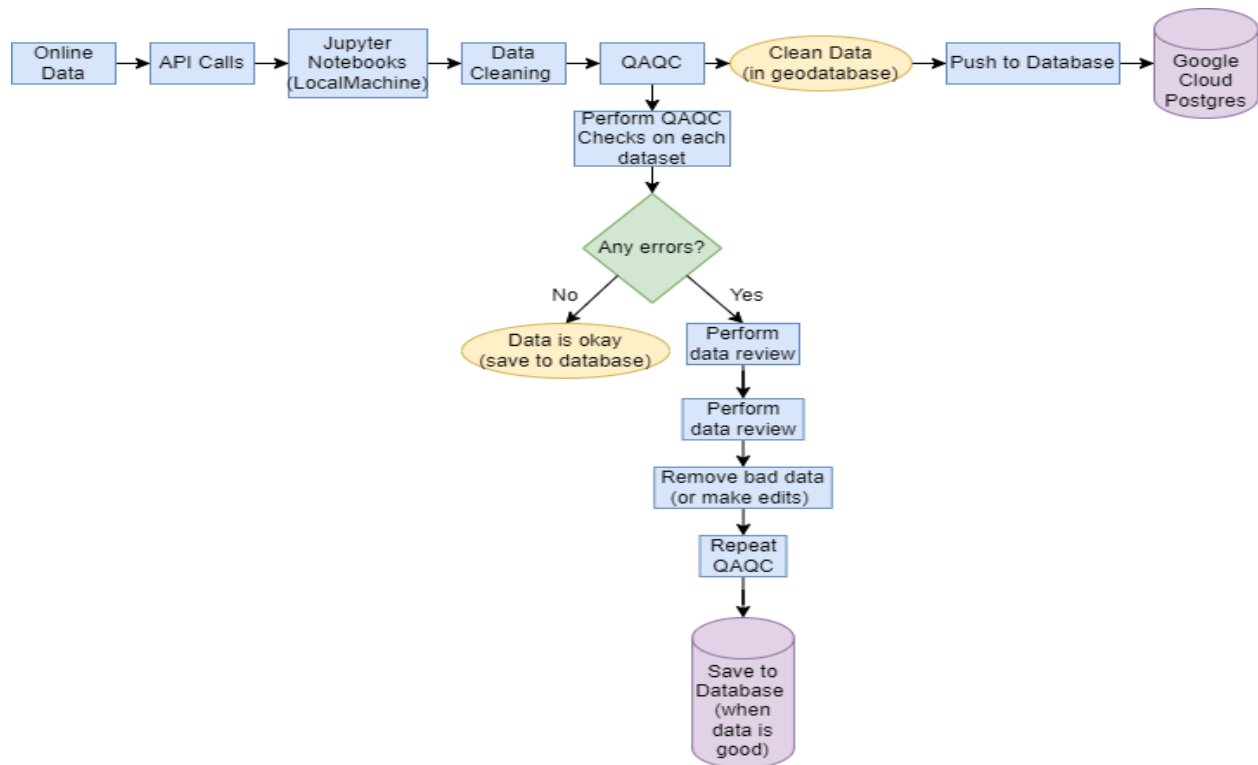
\*When appendices are included, the SRS should explicitly state whether or not the appendices are to be considered part of the requirements.

Traffic Impact Prediction Final Pipeline



The structured pipeline provides stakeholders and project members with clear visibility into both the development process and final objective. Further, the Traffic Impact Prediction Final Pipeline promotes transparency and accountability by documenting each stage of the data processing and analysis workflow, enabling stakeholders to trace the lineage of predictions and insights generated by the system. The pipeline's modular design facilitates; scalability, flexibility, and maintainability, ensuring that the pipeline remains adaptable to emerging challenges and evolving needs within the realm of traffic prediction and management.

## QA QC Pipeline



When evaluating the quality of the data, the approach differs between statewide data—such as temperature, land cover, and elevation—and more localized data, specifically sports and roads, which are focused around the University of Minnesota (UMN) campus area.

The QA/QC protocols employed are varied and rigorously justified to match the unique characteristics of each dataset. For elevation and land cover data, checks such as completeness (gap analysis), positional accuracy (specifically referencing the Nad\_1983\_UTM\_Zone\_15N spatial reference), thematic accuracy, and logical consistency (including cell size dimensions) are utilized.

Temperature data undergo a different set of QA/QC checks, including completeness (identifying missing values), logical consistency (evaluating temperature ranges and minimum-maximum temperature relationships), and the identification of duplicate dates or gaps in data.

Roads and sports schedules are evaluated with a focus on consistency, completeness, thematic accuracy, and logical consistency, among other checks. For roads, this includes ensuring that traffic patterns are accurately represented and that there are no detached road segments. For sports schedules, the focus is on filtering data by years and standardizing stadium names.