

Software Requirements Specification

for

Traffic Monitoring with Machine Learning

Version 1.0 approved

Prepared by Hue Ngo, Russell Carter, Javier Hernandez,
Daniel Caceres, Vrezh Khalatyan, Gracie Zamora

LA Data Science Federation and LADOT

May 11, 2019



Los Angeles, California

Table of Contents

- Table of Contents **ii**
- Revision History **ii**
- 1. Introduction 1**
 - 1.1 Purpose 1
 - 1.2 Document Conventions 1
 - 1.3 Intended Audience and Reading Suggestions 1
 - 1.4 Product Scope 1
 - 1.5 References 2
- 2. Overall Description 2**
 - 2.1 Product Perspective 2
 - 2.2 Product Functions 2
 - 2.3 User Classes and Characteristics 3
 - 2.4 Operating Environment 3
 - 2.5 Design and Implementation Constraints 3
 - 2.6 User Documentation 3
 - 2.7 Assumptions and Dependencies 3
- 3. External Interface Requirements 3**
 - 3.1 User Interfaces 4
 - 3.2 Hardware Interfaces 4
 - 3.3 Software Interfaces 4
 - 3.4 Communications Interfaces 4
- 4. System Features 4**
 - 4.1 Data Preparation Engine 4
 - 4.2 Data Analytics Engine 5
 - 4.3 Data Visualization Engine 5
- 5. Other Nonfunctional Requirements 6**
 - 5.1 Performance Requirements 6
 - 5.2 Safety Requirements 6
 - 5.3 Security Requirements 6
 - 5.4 Software Quality Attributes **Error! Bookmark not defined.**
 - 5.5 Business Rules 6
- Appendix A: Glossary 6**
- Appendix B: Analysis Models 8**
- Appendix C: To Be Determined List 8**

Revision History

Name	Date	Reason For Changes	Version
Hue	5/11/19	Revised sections 1, 2, 3, 5	1.0
Hue, Daniel, Gracie, Javier, Russell, Vrezh	10/26/18	Revised sections 1, 2, 3, 5	1.0
Hue	11/4/18	Revised sections 1, 2, 3, 5, 6, Appendix	1.0
Russell Carter	11/5/18	Revised sections 2.1, 2.2	1.0
Hue, Russell, Javier	11/8/18	Revised sections 3.1, 4, Appendix	1.0
Hue	2/23/19	Revised all sections for final draft	1.0

1. Introduction

The Traffic Monitoring with Machine Learning application is used to predict and visualize traffic conditions in real-time. The application utilizes machine learning fundamentals to find traffic propagation patterns and make predictions based on various features. The application is also developed as a web application to visualize traffic conditions in real-time.

1.1 Purpose

The purpose of this document is to explain in detail the functions that the application will perform. The version being referred to is version 1.0.0 as there are no other revisions or updates that have been processed. There will be a complete overview of all aspects of the application, including the functions of the application, the operating requirements, and the database requirements. The document will inform readers as to what the application will do.

1.2 Document Conventions

The terms *product* and *application* shall be used interchangeably to refer to the Traffic Monitoring with Machine Learning application. External links will be underlined and have blue color text. A list may be in the format of a table or bullet points. Illustrations may accompany some technical specifications for easy understanding and simplification.

1.3 Intended Audience and Reading Suggestions

The main audience of the software requirements specifications document are developers, project managers, testers, marketing staffs, and researchers. It is suggested that if this is your first read through to quickly skim through the entire document to further your knowledge and the usage of the application. For developers, some main points to focus on can be the product functions and the database requirements. For project managers the user interface, functional and database requirements are good points to read over. Testers may go over the user documentation, the assumptions and dependencies, user interfaces, and performance requirements sections. The marketing staff and researchers may go over the user interface and performance requirements to start off with. But all should go over the user, hardware, software, and communications interfaces as well as the safety requirements of the application.

1.4 Product Scope

The scope of this application is to model the road network using various geographic data. This road network is then transformed into a Network Linkage, which provides the available routes vehicles can travel on. The data from the Network Linkage may provide insightful propagation patterns of traffic congestions, which is used in a machine learning algorithm to predict future traffic jams. These traffic jams are visualized in a web application.

1.5 References

The following table contains the references that are most often referred to for this application.

Alias	Description
GRNN	Graph Recurrent Neural Network (GRNN) was developed by six doctorates at the Shanghai Jiao Tong University, and the new methodology was published in November of 2018. GRNN aimed to reduce the redundancy of a traditional road network by transforming the road segments and its neighboring road segments into nodes and linkage network. The road segments are represented as nodes and the edges in the network demonstrate that the nodes are neighbors, meaning that there's an available path from one node to other node.
ArcGIS	All references to ArcGIS services. https://doc.arcgis.com/en/
ArcGIS Web AppBuilder: Widget	ArcGIS Widget provide functionalities, customizations, and configurations for ArcGIS map application. https://doc.arcgis.com/en/web-appbuilder/create-apps/widget-add-data.htm
CSV	Comma-separated-value file
TSV	Tab-separated-value file

2. Overall Description

The overall product shall satisfy the Product Perspective, Functions, User Classes, and Operating Environment. The application shall provide Documentation and inform users of any Constraints, and Assumptions and Dependencies.

2.1 Product Perspective

This product is an implementation of a the newly created GRNN approach. The machine learning model of this application will mimic similar data input and output of the original GRNN. The expected output of the model is a traffic prediction based on average speed data of the various locations in the city of Los Angeles. The average speed data and other additional data may be provided from Los Angeles Department of Transportation (LADOT) or from ITA in the form of local storage (CSV, TSV) or a data stream.

2.2 Product Functions

The following list contains the major functionalities of the application:

- The application shall predict average speed of a location.
- The application shall display prediction output on a map.

2.3 User Classes and Characteristics

The different classes and their characteristics that will use application are:

- Employee: these users will be able to view traffic visualizations.
- Developer: these users will be able to view traffic visualizations and modify the application source code.

2.4 Operating Environment

End-users and developers can use the application on Windows and/or Mac operating systems. From developers' perspectives, the application will require at least Python 3.6.x to be installed and all of the required libraries and its dependencies.

2.5 Design and Implementation Constraints

The following various operations and tasks may affect the product's timetable:

- Insufficient or erroneous data for training model
- Local machines with poor specifications or insufficient power to execute the application
- Frameworks or technologies with deep learning curve
- Difficult developing environment that is required for creating the application
- Mastering data science and machine learning concepts

2.6 User Documentation

A tutorial shall be provided to show step-by-step how to utilize the code.

2.7 Assumptions and Dependencies

The following factors are assumptions that could affect the requirements:

- Application developers are not liable for erroneous data from ETL processes
- Users of this application are registered with a valid ArcGIS account to use their services.
- Any application dependencies with their correct versions are maintained by their creators and developers and are working as intended.
- Only immediate and known user classes will be included in this SRS.

3. External Interface Requirements

The external interface requirements detail the User Interfaces, Hardware Interfaces, Software Interfaces, and Communication Interfaces of the application.

3.1 User Interfaces

The list below contains the requirements for the user interfaces:

- Feature layer shall be used to render coordinates as streets in a map.
- Polyline objects shall be used to display the paths of a road network.
- Time slider shall be used to filter traffic data by time.

3.2 Hardware Interfaces

This application does not have a hardware interface requirement.

3.3 Software Interfaces

Listed are the requirements for software interfaces:

- ArcGIS JavaScript shall be used to visualize traffic data received from Node.js.
- Pytorch shall be used to train Graph Recurrent Neural Network for machine learning.
- Pandas and other data science supported libraries shall be used for data manipulations.

3.4 Communications Interfaces

Listed are the requirements for the communication interface:

- The application shall send data in the form of JSON via a HTTP Response.
- The application shall retrieve data in the form of JSON via a HTTP Request.

4. System Features

The features of this application are called Engines and each engine is composed of sub-features. These engines are Data Preparation Engine, Data Analytic Engine, and Data Visualization Engine.

This section is written for developers, testers, designers, and users with administrative authority over this application.

4.1 Data Preparation Engine

4.1.1 Description and Priority

Data Preparation Engine is composed of a series of Python scripts that shall preprocess the Waze dataset. This engine has high priority because it forms the baseline to properly train and test the predictive model.

4.1.2 Functional Requirements

The followings are the functional requirements of the Data Preparation Engine:

DPE-1: The engine shall average integer or decimal values.

DPE-2: The engine shall find the unique strings that exist in a column.

DPE-3: The engine shall replace empty fields with NULL values.

- DPE-4:** The engine shall reformat the column headers with values in all uppercase.
- DPE-5:** The engine shall replace NULL values in a non-id number column with the average of its sum.
- DPE-6:** The engine shall reformat data values to their correct data types.

4.2 Data Analytics Engine

4.2.1 Description and Priority

Data Analytics Engine is composed of a series of Python scripts to perform Feature Selection, Feature Extraction, and Machine Learning. This engine has a high priority because it trains and tests the model required to perform traffic predictions.

4.2.2 Functional Requirements

The followings are the functional requirements of the Data Analytics Engine:

- DAE-1:** The engine shall group data based on time, day of the week, special events, and others to form the extracted features.
- DAE-2:** The engine shall select specific features from the extracted features to train the predictive model.
- DAE-3:** The engine shall train a model based on the dataset with the selected features.
- DAE-4:** The engine shall test the trained model for accuracy.

4.3 Data Visualization Engine

4.3.1 Description and Priority

Data Visualization Engine is an Express and Nodejs web application that communicates with the database and serves visualizations to the end-users. This engine has a medium priority because the Data Analytics Engine needs to be established before we can create visualizations for the analyzed data.

4.3.2 Stimulus/Response Sequences

End-users shall be able to select different data fields for visualizations.

4.3.3 Functional Requirements

The followings are the functional requirements of the Data Visualization Engine:

- DVE-1:** The engine shall retrieve data from the database.
- DVE-2:** The engine shall visualize traffic data in a map as feature layers.
- DVE-3:** The engine shall summarize traffic data in forms of charts and graphs.
- DVE-4:** The engine shall visualize traffic data dynamically.
- DVE-5:** The engine shall query selected data per user requests via widgets.

5. Other Nonfunctional Requirements

In other nonfunctional requirements, the application shall detail the performance, safety, and security requirements. The application shall include the software quality attributes and business rules.

5.1 Performance Requirements

Listed are the performance requirements:

- The machine learning model may be optimized to process real-time data.
- The web application shall be responsive.

5.2 Safety Requirements

No safety requirements were identified for this application.

5.3 Security Requirements

Developers agree of the NDA before, during, and after the application is implemented wherein developers may not share any of the Waze data or traffic data provided by the City of Los Angeles to anyone outside of the project scope.

5.4 Software Quality Attributes

Listed are the software quality attributes of the application:

- The machine learning model shall be validated.

5.5 Business Rules

The followings are business rules about the product:

- Any data source used for this application is private unless given permission by the original providers.
- The application is a private property as requested by the LADSF and LADOT.

Appendix A: Glossary

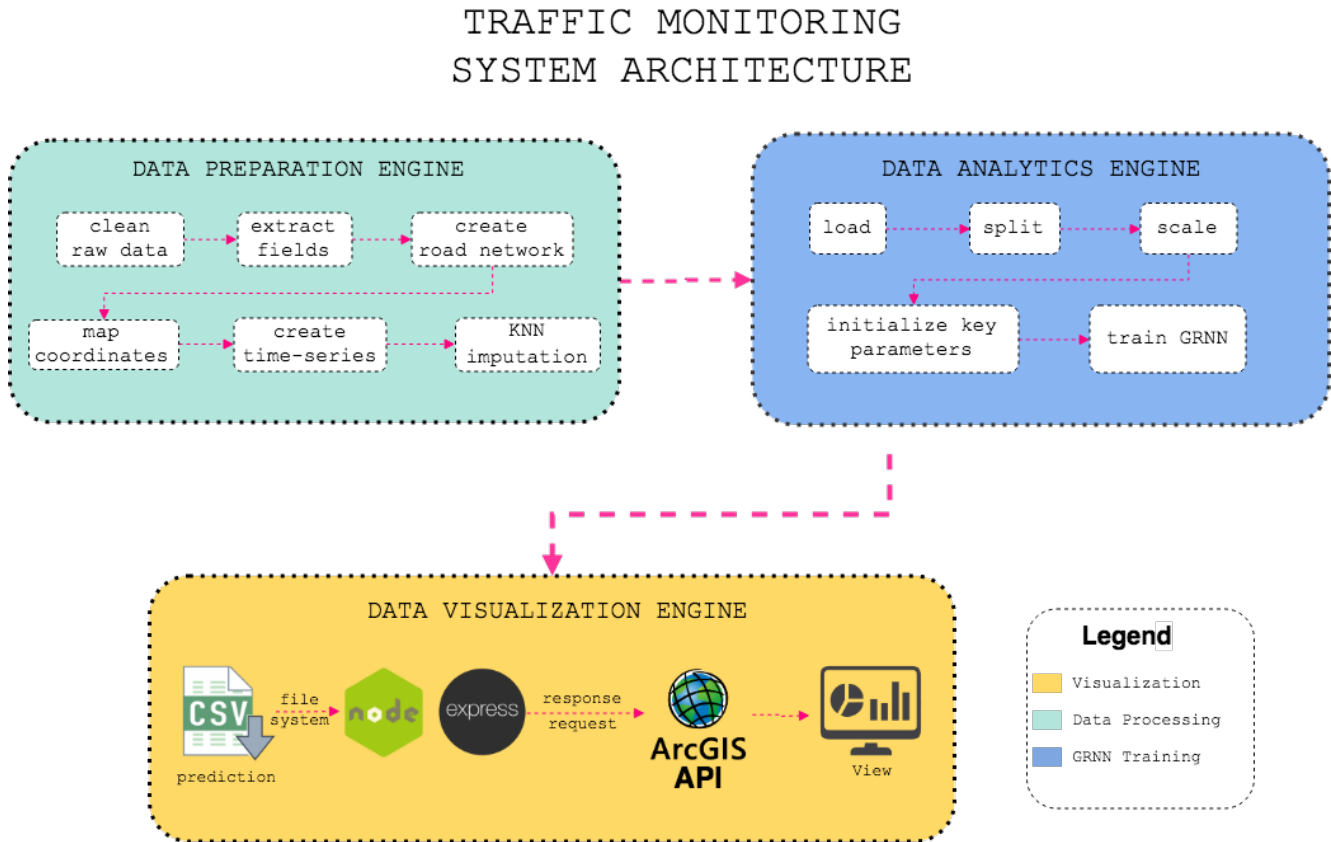
The following table contains the terms and definitions to properly interpret the application software requirement specification.

Term	Definition
ArcGIS	Esri's all-in-one solution to work with geographic information.
CSS	Cascading Style Sheet is a style sheet that is used to describe the presentation of a markup language.
CSV	Comma Separated Values. File format that is used to store tabular data such as spreadsheets or databases.

DSF	Data Science Federation is a specialized department in the City of Los Angeles.
FE	Feature Extraction is a method to convert data into features usable in a model.
FS	Feature Selection is a method to remove unnecessary features in a dataset.
HTML	Hypertext Markup Language is the standard markup language for creating web pages.
HTTP	Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems.
JavaScript	A programming language that is heavily used for web applications.
LADOT	Los Angeles Department of Transportation is a specialized department of Los Angeles City.
Machine Learning	Predictive mathematical model used for making predictions.
Operating System	The software that allows any computer to communicate, modify, and terminate any hardware and software communications based on end-users decisions.
Python	A general-purpose programming language that can also be used to program web application and data analytics application.
Runtime	The time when an application is executed.
SDD	Software Design Document is a document that specifies the design of the application on a technical and detailed level.
SRS	Software Requirement Specification is a document that specifies the functional and nonfunctional requirements of the application.
Pytorch	Open-source software library for training and testing machine learning models.

Appendix B: Analysis Models

The diagram illustrates the system architecture:



Appendix C: To Be Determined List

The following requirements and conditions are to be determined:

- The data source that is used to provide online predictions for the machine learning model.