**Software Requirements Specification**

**for**

**Capital Project Development Map**

**LA City Data Science**

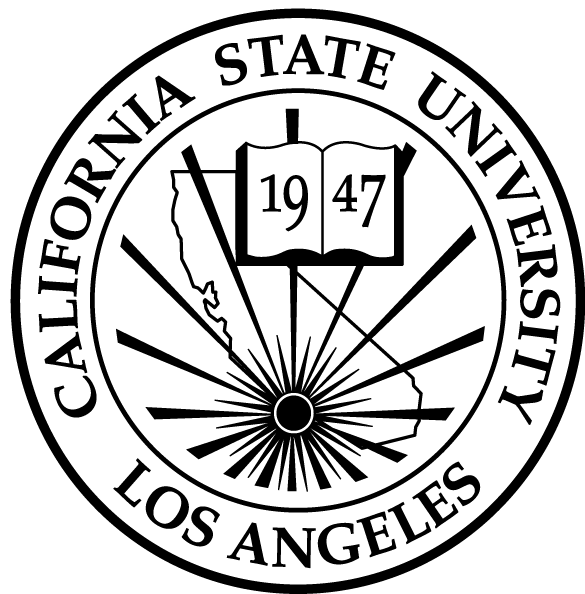
**Version 1 approved**

**Prepared by**

**Kevin Gamboa, Kevin Lam, Alvin Nguyen, Elizaveta Sokolova**

**LA City Data Science**

**October 15, 2017**



Approved By:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Mohammad Pourhomayoun Date

# Table of Contents

Table of Contents...................................................................................................................<pg 3>

Revision History.................................................................................................................... <pg 5>

1. Introduction................................................................................................................<pg 6>

1.1 Purpose...........................................................................................................<pg 6>

1.2 Intended Audience and Reading Suggestions................................................<pg 6>

1.3 Product Scope.................................................................................................<pg 7>

1.4 Definitions, Acronyms, and Abbreviations ...................................................<pg 7>

1.5 References......................................................................................................<pg 7>

1. Overall Description.................................................................................................... <pg 8>

2.1 Product Perspective........................................................................................ <pg 8>

2.2 Product Functions...........................................................................................<pg 9>

2.3 User Classes and Characteristics....................................................................<pg 9>

2.4 Operating Environment................................................................................<pg 10>

2.5 Design and Implementation Constraints......................................................<pg 10>

2.6 User Documentation.....................................................................................<pg 10>

2.7 Assumptions and Dependencies...................................................................<pg 10>

2.8 Apportioning of Requirements.....................................................................<pg 11>

1. External Interface Requirements..............................................................................<pg 12>

3.1 User Interfaces..............................................................................................<pg 12>

3.1.1 Point Feature.....................................................................................<pg 12>

3.1.2 Line Feature......................................................................................<pg 13>

3.1.3 Polygon Feature................................................................................<pg 14>

3.1.4 Scoring Feature.................................................................................<pg 15>

3.1.5 Adding shapefiles Feature................................................................<pg 16>

3.2 Hardware Interfaces.....................................................................................<pg 16>

3.3 Software Interfaces......................................................................................<pg 16>

3.4 Communications Interfaces.........................................................................<pg 16>

1. Requirements Specification.....................................................................................<pg 17>

4.1 Functional Requirements..............................................................................<pg 17>

4.2 External Interface Requirements..................................................................<pg 17>

4.3 Logical Database Requirements...................................................................<pg 17>

4.4 Design Constraints.......................................................................................<pg 17>

5. Other Nonfunctional Requirements..........................................................................<pg 18>

5.1 Performance Requirements..........................................................................<pg 18>

5.2 Safety Requirements.....................................................................................<pg 18>

5.3 Security Requirements.................................................................................<pg 18>

5.4 Software Quality Attributes..........................................................................<pg 18>

5.5 Business Rules..............................................................................................<pg 18>

6. Other Requirements..................................................................................................<pg 19>

Appendix A: Glossary..........................................................................................................<pg 20>

Appendix B: Analysis Models.............................................................................................<pg 21>

Appendix C: To Be Determined List...................................................................................<pg 22>

# 

# 

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Kevin Gamboa | 10/15/17 | Update Requirements | 1 |
| Elizaveta Sokolova | 11/3/2017 | Update Requirements | 1 |
| Kevin Gamboa | 12/7/2017 | Update / Finalize Requirements | 1 |
| Kevin Gamboa | 4/3/2018 | Update / Finalize Requirements | 1 |

# 1. Introduction

## LADOT Transpo Model is a web application that will provide a user with capabilities of viewing different types of layers that help prioritize projects to advance for funding and implementation according to their relative advancement of City policy priorities (Safety, Access, Sustainability, etc.). This map could be based on a web-map tool DCP built for the Great Streets Challenge that helped to prioritize applications in accordance to policy priorities (MP 2035, Visions Zero, Plan for Healthy Los Angeles, etc.).

## 1.1 Purpose

The purpose of this document is to define a full set of requirements for our LADOT Transpo Model. The complete definition of all LADOT Transpo Model requirements provides the source requirement inputs for the development of the subsequent supporting software subsystems documents.

## 1.2 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, users, testers, and documentation writers. The software requirement documents contain the project’s introduction, overall description, external interface requirements, requirements specification, and other nonfunctional requirements in the same order as the list.

Users and testers will find interest in requirements specification list.

Developers and project managers will find interest in the overall description section and external interface requirements list.

Documentation writers will find interest in the overall description section.

## 1.3 Product Scope

* The LADOT Transpo Model is a Parameterized Model that would advance the web map by automating the scoring process, based on project overlay with multiple map layers.
* This model could the project list developed Active Transportation Planner and expand to other LADOT / Public Works capital programs.
* The project will enable LADOT staff to justify mobility investment according to their relative advancement of City mobility policy priorities and anticipated benefits to Angelenos in terms of safety, access, sustainability, and livability.
* The project would use data on existing conditions to forecast the influence of individual projects to street users’ experience by travel mode.

## 1.4 Definitions, Acronyms, and Abbreviations

* LADOT – Los Angeles Department of Transportation
* Transpo Model – Transportation Model
* ArcGIS – Arc Geographic Information System

## 1.5 References

* ArcGIS API Reference

- https://developers.arcgis.com/javascript/3/jsapi/

* LADOT Front End Styling Guide

- https://github.com/datala/dot-planner/files/768067/LADOT\_STYLE\_GUIDE.APRIL.2016.pdf

# 

# 2. Overall Description

The LADOT MIP Interactive Web Map was developed to help LADOT staff estimate how individual projects advance the Department’s core values and the City’s adopted vision for the transportation system. The Map translates the needs the Department aims to address with mobility investments into five categories of spatial data layers:

1. Mobility Plan Alignment

2. Safe & Healthy

3. Equitable and Inclusive

4. Accessible and Affordable

5. Sustainable and Resilient

Using the Map, project managers will complete a Project Scoring Form for each individual project, which provides LADOT Transportation Planning & Policy staff with project-specific information for each of the 5 data categories based on its scope and site context in order to calculate a Project Score.

## 2.1 Product Perspective

* This software is based off of ESRI’s ArcGIS mapping tools, and is dependent on their system.
  + It is made specifically for the Los Angeles County and its geographical region.
  + This software centralizes significant data layers using ArcGIS, to help LADOT staff estimate the multiple benefits individual projects may have based on the built environment and socioeconomic context of the project site.

## 

## 2.2 Product Functions

* Scoring Feature:
  + The positioning of projects, in terms of its layers, determines how the scoring will be calculated.
* Point Feature:
  + Selecting the point feature enables the user to click anywhere to propose a project that can be represented as a point on a map (stop signs, traffic lights).
* Line Feature:
  + Selecting the Lines feature enables the user to click anywhere on the map multiple times to propose a project that would have a physical representation of a line on a map (road construction, bicycle lanes).
* Polygon Feature:
  + Selecting the Polygons feature enables the user to click anywhere on the map multiple times to propose a project that has a physical representation of a polygon on a map (schools, parks).
* Multipoint Feature:
  + Selecting the Multipoints feature enables the user to click anywhere on the map multiple times to propose a project that has a physical representation of multiple points on a map (work on multiple stop signs, work on multiple freeway entrances).
* **Adding Shapefiles:**
  + Allows the user to upload a shapefile with pre-existing projects.
* **Layers:**
  + Usershave the option to toggle on and off any layer they have selected.

## 2.3 User Classes and Characteristics

* Developers may update code to add new attributes and layers into the LADOT Transpo Model.
* Users and testers would use the LADOT Transpo Model’s implementation to calculate and observe the city’s projects and policy priorities.

## 

## 2.4 Operating Environment

* The LADOT Transpo Model will be actively running through ESRI’s database systems.

## 

## 2.5 Design and Implementation Constraints

There are limitations that can affect ability of the software developers to implement the product. These can include things such as:

* There are regulatory policies within the organization that may affect development the software.
* Accessibility / license limitations.
* Interfaces with other applications.
* Database issues.
* Server issues.

## 

## 2.6 User Documentation

* The API Reference contains detailed descriptions for each class in the ArcGIS API for JavaScript:

<https://developers.arcgis.com/javascript/3/jsapi/>

## 2.7 Assumptions and Dependencies

## Implemented ESRI’s components with LADOT Transpo Model software.

* Re-used data from LA City’s geohub database.

## 2.8 Apportioning of Requirements

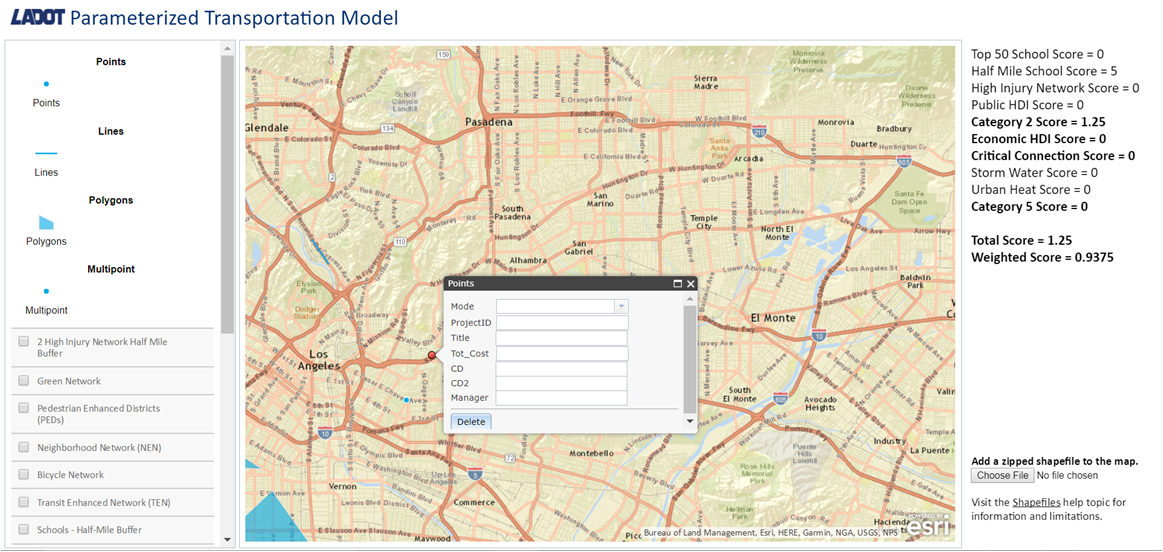
**Delayed requirements until future versions of the system.**

|  |  |
| --- | --- |
| **1.)** | **Calculating scores for all 4 sections.** |
| **2.)** | **Implementing ESRI multi-points.** |
| **3.)** | **Collecting data for predictive model.** |

# 3. External Interface Requirements

## 3.1 User Interfaces

## 3.1.1 Point Feature

Selecting the Polygons feature enables the user to click anywhere on the map multiple times to propose a project that has a physical representation of a polygon on a map(schools, parks).

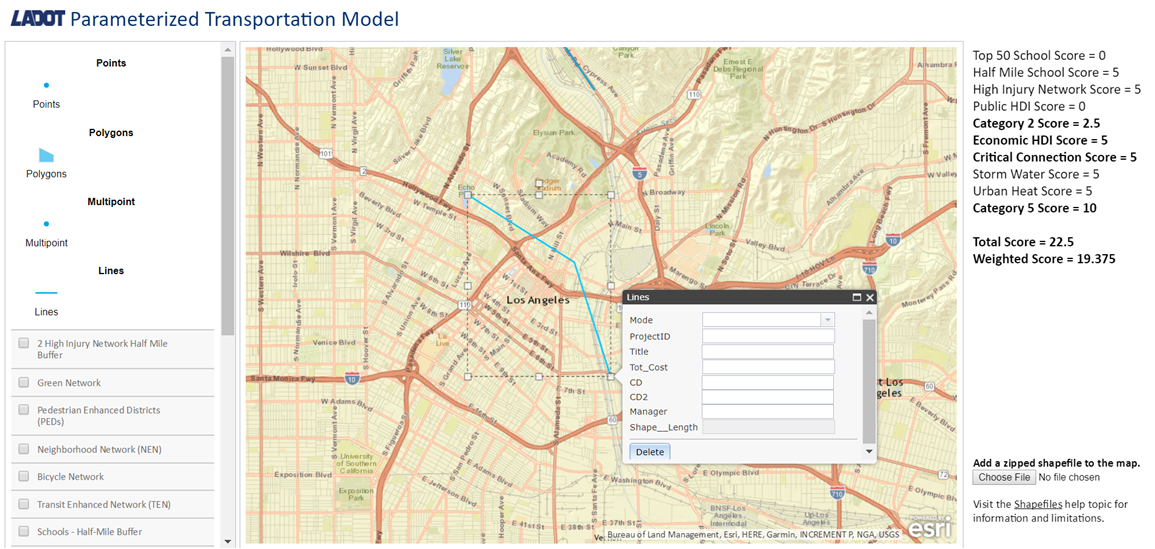
## 

## 

## 

## 

## 3.1.2 Line Feature

Selecting the Lines feature enables the user to click anywhere on the map multiple times to propose a project that would have a physical representation of a line on a map (road construction, bicycle lanes).

## 

## 

## 

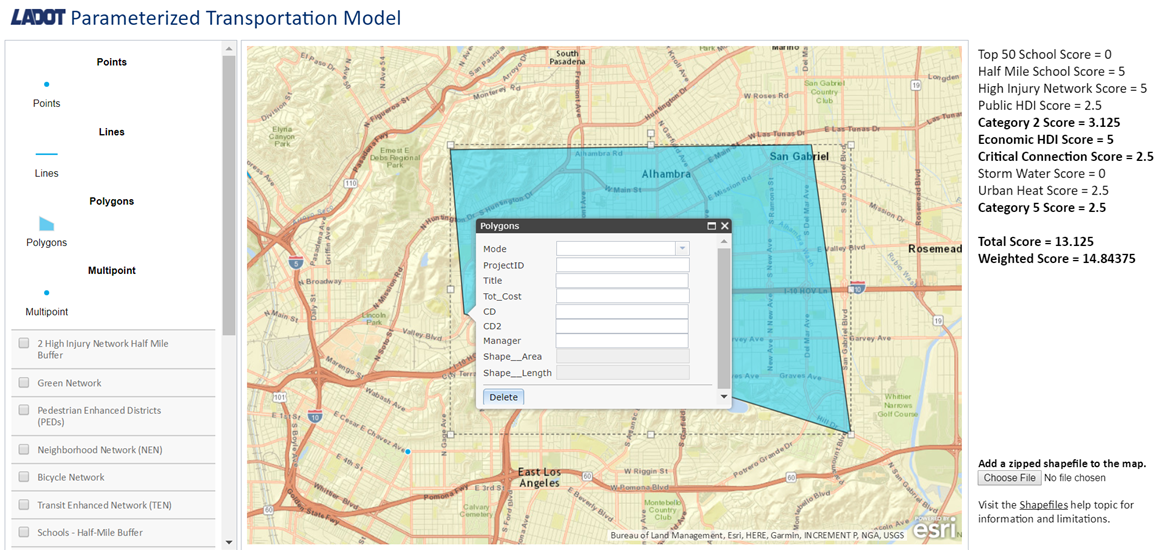
## 

## 

## 

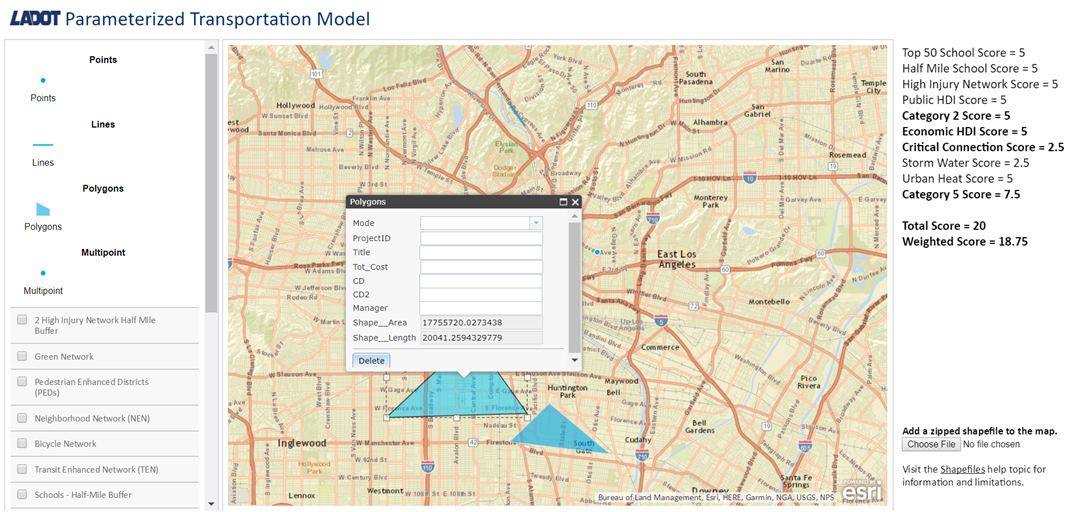
## 

## 3.1.3 Polygon Feature

Selecting the Polygons feature enables the user to click anywhere on the map multiple times to propose a project that has a physical representation of a polygon on a map(schools, parks).

## 

## 3.1.4 Scoring Feature

The positioning of projects, in terms of its layers, determines how the scoring will be calculated. The calculations are based upon LADOT priorities and City mobility goals.

## 

## 

## 

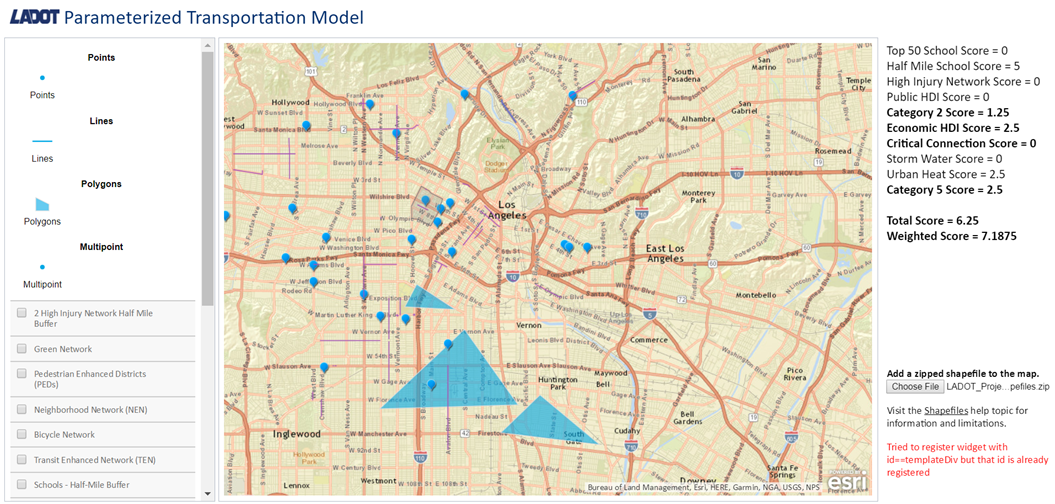
## 

## 

## 

## 

## 3.1.5 Adding Shapefiles Feature

The “Choose File” button allows the user to upload a shapefile with pre-existing projects.

## 3.2 Hardware Interfaces

* Software does not have any hardware interface requirements.

## 3.3 Software Interfaces

* Implemented ESRI’s ArcGIS with LADOT Transpo Model.

https://developers.arcgis.com/javascript/3/jsapi/

* This is a customer-specific requirement that is needed to add and edit layers and its corresponding data within the web app.

## 3.4 Communications Interfaces

# 

The Transpo Model shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

# 4. Requirements Specification

## 4.1 Functional Requirements

4.1.1 The system shall allow project managers to draw project shape on the map (point, line, polygon).

4.1.2 The system may limit access to authorized users.

4.1.3 Data about the project should be entered after the project is drawn on the map.

4.1.4 The system shall support the multi-point features.

4.1.5 It shall calculate and output scores based on their location on the map and intersection with layers.

4.1.6 It shall allow Admin/Users accounts.

4.1.7 It shall allow Admins/Codebase to change weights.

## 4.2 External Interface Requirements

4.2.1 The system shall receive information about the project from project managers.

4.2.2 It shall receive the weights for calculating the score for the project.

4.2.3 It shall receive shapefiles of existing projects via upload and display them on map with their scores.

4.2.4 It shall output on the screen the score of the projects based on what layers it intersects.

4.2.5 It shall output the file with the scores and detailed calculations.

4.2.6 It shall prevent the identical project upload.

## 4.3 Logical Database Requirements

4.3.1 The system shall store projects’ information into a database on GeoHub side.

4.3.2 The database should include all projects’ information such as Mode, Project ID, title, total cost, project date, manager and so on.

4.3.3 The database shall be updated every time the project manager enters a new a project into the system, updates or deletes one.

## 4.4 Design Constraints

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

* The web app’s backend is integrated with ESRI’s servers and is dependent on them.

## 5.2 Safety Requirements

* No necessary safety requirements needed with the Transpo Model.

## 5.3 Security Requirements

* Admin / user accounts will need identity authentication when configuring with the score and weights inside the web app.

## 5.4 Software Quality Attributes

* Future developers may find comments within important lines of code to help them understand each implementation.

## 5.5 Business Rules

* Administrators can add and edit data without security clearance.
* Users have more restrictions and can’t change scores / weights of projects.

# 

# 

# 6. Other Requirements

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# Appendix A: Glossary

# Appendix B: Analysis Models

# Appendix C: To Be Determined List

* Implementing ESRI multi-points.
* Calculating scores for the remaining layers.