CITY Boost

A realistic view of traffic density.

// PICTURE here: On coming.

**Team 07(WebFlow): Zoe Fu, Ried Stagemeyer, Dat Nguyen.**

# 1. Introduction

## 1.1 Purpose

City Boost is a web application that lets the user, Mayor Mann, manage traffic. It simulates the flow of traffic within a preloaded city map. Relevant goals and desired features of the application are explained below:

* Structure and organize the entire traffic flow for the city.
* Facilitate fast, unbiased, and accurate interactions at each intersection.
* Help reduce cost and time of construction projects in the city.
* Provide a flexible, automated, and interactive interface between the user and the program by showing live statistics of the traffic flow.

## 1.2 Scope

Construction projects in high density urban areas have a large impact on traffic flows. The city can develop more efficient solutions for expansion/redevelopment projects by simulating potential traffic flows. This will allow them to investigate the impact of changing traffic component placement (stop lights, stop signs, etc.) at intersections to determine their optimal combinations/locations. The ability to run simulations to determine the optimal placement of traffic components will minimize delays and the loss of productivity resulting from them. This means the developed application will give people and companies more time and money across the whole city.

## 1.3 Core of the system

Application will be able to:

* Allow user to dynamically create city map or load map from csv files.
* Allow user to adjust the start/stop locations and number of cars.
* Allow user to adjust the location and combinations of traffic components (traffic lights, stop signs).
* Allow user to run multiple traffic simulations.
* Provide a GUI that allows user to drag and drop traffic components.
* Provide a window to show live traffic flow statistics.
* Display a full report of the traffic flow at the completion of a simulation run.

## 1.4 Objectives and Success Criteria

Our product will closely simulate actual city traffic for the Mayor. This will give him the most realistic and accurate look of the traffic flows to find the optimal placement of traffic components. The optimal placement of components is defined as getting the most cars to their destination the fastest. The system should allow multiple simulations to be run with various combinations/placements to be analyzed. A simulation is finished once every car on the map reaches their destination, however the user can terminate the current simulation at any time. Many constraints will be included, first, all moving traffic components (i.e cars) will be moving at the same speed and making consistent turning decisions at each intersection. The simulation will be running in a one-time thread.

## 1.5 Definitions, Acronyms, and Abbreviations

OOP: Programming language model organized around objects rather than “actions” and data rather than logics

UML: Diagram based on the UML (Unified Modeling language) with the purpose of visually representing a system along with its main actors, role.

Intersection: An area shared by two or more roads.

Tile: a smallest countable unit on the City Boost simulation map.

GUI: Graphical User Interface.

Time thread: the smallest sequence of programmed instruction that can be managed independently by a scheduler.

Turn: Action that each car agents will take at intersection

Q Learning : Learning process of an agents(cars) which agents are givens history and current state , possible actions, rewards from start, and next actions as learning process. Where Q is a policy which optimize the state and action pair.

Convolution Neural Network(CNN): Type of classification technique used in image classification. Where learned parameter will be convolution filter.

## 1.6 References

<https://searchmicroservices.techtarget.com/definition/object-oriented-programming-OOP>

<https://tallyfy.com/uml-diagram/>

# 2. Current System

N/A

# 3. Proposed System

## 3.1 Overview

The proposed system must allow Mayor Mann to efficiently determine the optimal placement of traffic components (traffic lights, stop signs) throughout the city. This will be done by creating an application that lets him run multiple simulations using various locations and combinations of those traffic components. He will also be able to change the number of cars and their start/end points. Various statistics and reports will be output to allow the user to determine what combination is most desirable.

## 3.2 Functional Requirements

F1. Run multiple simulations

F2. Let user change the location of traffic components

F3. Let user change the combination of traffic components

F4. Analyze traffic performance

F5. Allow user to change number of cars

F6. Allow user to change start/end locations for each car

F7. Allow user to change the layout of the map

## 3.3 Nonfunctional Requirements

N1. Be sure to provide unique requirement identifiers for traceability

N2. Yada

N3. Yada

## 3.4 System Models

### Use Case Model

TBD

### Structural Model

TBD

### Behavioral Model

TBD

### User Interface: Navigational Paths and Screen Mockups

TBD

# Glossary

Yada