

CITY TRAFFIC SIMULATOR

REQUIREMENTS ANALYSIS DOCUMENT

Team Name
Chavez, Bryan
Majid, Mohammad
Nguyen, Brian
Vuong, Nguyen Huy

INTRODUCTION

PURPOSE

Organize the city by increasing the functionality of city traffic lights and stop signs in order to maximize the number of vehicles to a particular destination and minimize the travel time. The following document details the design and structure of the City Traffic Simulator.

SCOPE

This application is a simulation of the appropriate amount of traffic lights and stop signs for the city of Pacopolis. The mayor of the city is looking to maximize the flow of traffic from point A to point B while minimizing the travel time. Being able to provide an effective simulator in order complete the task mayor Mann has given us.

OBJECTIVES AND SUCCESS CRITERIA

The objective is to maximize flow of traffic from start to destination and minimizing the time of travel.

DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

OOP

GUI : Graphical user interface.

UML : Unified Modelling Language

RID: Requirement Identifier.

RAD : Requirement Analysis Document

REFERENCES

City map of Pacopolis: <http://pacman.wikia.com/wiki/Maze>

OVERVIEW

The overview of this project is mainly organizing enough traffic lights and stop signs to be processed correctly and set in the city of Pacopolis. The city of Pacopolis goal is achieving the necessary help in setting the traffic lights and stop signs to help minimize the amount of time needed for each vehicle to get to their specific destination.

CURRENT SYSTEM

The current system will be built in Python.

PROPOSED SYSTEM

OVERVIEW

This application uses object-oriented programming. The program utilizes a few main objects which model traffic lights, stop signs, vehicles and the city map.

The Map class is the GUI class which has inherited all the methods of a GUI application. Map class will simulate the city and movement of vehicles and traffic lights.

Vehicle class will be the class to control Vehicle movement and functionality. It will make use of different type of data structures and algorithm to maximize a vehicle performances.

FUNCTIONAL REQUIREMENTS

- F1. The system will make use different types of algorithm and data structures to maximize program efficiency, performances and response time.
- F2. The system will be able to minimize traffic jams and vehicle travels times without creating any accidents.
- F3. The system will successfully implement the appropriate order and location of traffic lights and stop signs.

NONFUNCTIONAL REQUIREMENTS

- N1. Changes logs and git branches will be use for keep track of different version of the simulations.
- N2. A GUI application will be created representing the traffic simulations.
- N3. The application will be runnable on any devices.
- N4. The application can send alert if there might be conflicts going on.

SYSTEM MODELS

USE CASE MODEL

Use Case 1

Actor: Mayor Mann

Basic Flow: The mayor will use a spreadsheet to represent traffic lights and stop signs to quest for the best combination of lights and signs from the desired starting point to ending point.

STRUCTURAL MODEL

The structure of this simulation will have a organized map. The map will include several vehicles and the vehicles will be tested to find the least amount of time needed to go to a specific destination. These cars will abide by the traffic rules and abide by the stop sign to stop at a specific amount of time in the preset of traffic lights or stop signs.

BEHAVIORAL MODEL

When the car interacts with the traffic light, depending on what the traffic light is set too the vehicle will stop for a certain amount of time. Also when the vehicle object is interacting with the stop sign object, the vehicle will abide by the rules presented in the program by stopping for a certain amount of time.

USER INTERFACE: NAVIGATIONAL PATHS AND SCREEN MOCKUPS

User will be able to interact with the GUI application by adding / removing cars.

Additional application functionality will added more over time.

GLOSSARY