332:452: Software Engineering

Report 1: System Specification

Group 7
Aditya Devarakonda
Vamshi Chilukamari
Akhilesh Madalli
Vladimir Samokhin
Sanket Wagle

Project: Traffic Monitoring
URL: https://sites.google.com/site/452trafficmonitor/
2/18/2011

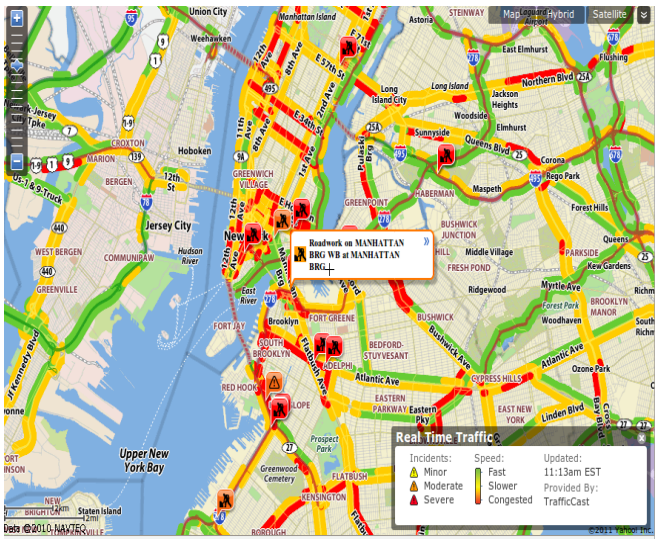
Breakdown of Contributions

All members contributed equally for this report.

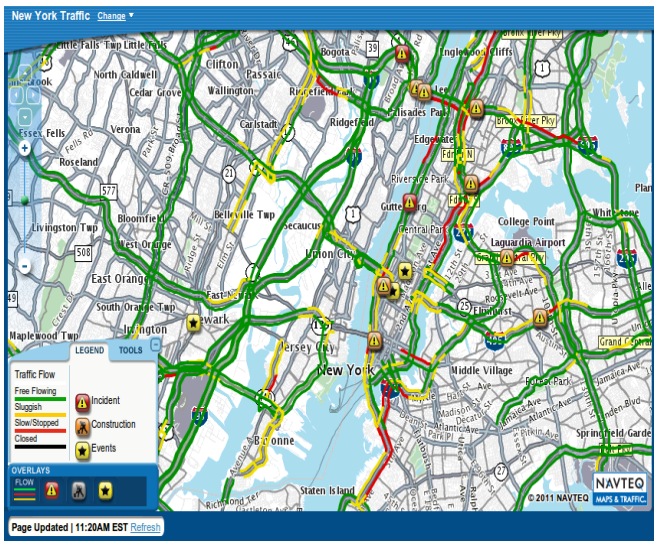
Table of Contents

Customer Statement of Requirements

Many traffic monitoring services only generate information about current, live traffic incidents. In general, most of these services only collect data at specific intervals and use live traffic information to generate the incidence map. Current services such as, "Yahoo! Maps Live Traffic"[1] and "Traffic.com"[4] use this method to monitor traffic. Below are examples of two live traffic systems with incidents reporting on the Manhattan area of New York City.



Yahoo! Maps Live Traffic



Traffic.com Live Traffic

This project attempts to use historical traffic data to show the traffic trends along a particular route/area during various weather conditions and times of day. This new approach to traffic monitoring is necessary when considering the fact that, live traffic only reports incidents as they occur. If, for example, a route consistently has accidents and heavy traffic then it would be a bad idea to take that route. However, a live traffic monitoring system will only report incidents as they occur without notifying users that accidents occur along the route daily. Clearly this would be bad for the users of such a monitoring system. To rectify this problem, we intend to develop a system that will take historical traffic patterns into account. This represents a better monitoring method because it takes past trends into account in order to show the likelihood of encountering heavy traffic along the route/area regardless of whether there is currently an incident or not. In addition, to using the historical traffic data, one can use live traffic so that the users have all the necessary information in order to make a good route choice. In this project we seek to implement the historical traffic portion and, time allowing, extend the system to support live traffic. Google Maps[5] has a similar traffic monitoring system that combines historical traffic and live traffic.

Our system will support two methods of traffic reporting that users can choose from:

(1) "Incidents reports within an area", where the user is given the choices to,
  A) "Select target area". This target area will be one of
    I. "North"
    II. "Central"

III. “South”

B) “Time”. The time interval can be one of
   I. Intervals of 1 hour (example: “3AM”)
   II. “All”. Traffic reports for all time periods within the selected area are displayed.

C) “Type of day”. This can be one of
   I. “Weekday”. The traffic reports spanning Monday-Friday are considered.
   II. “Weekend”. The traffic reports on Saturday and Sunday are considered.

D) “Show incident severities of”. Can be a choice from 1-5 (least-severe). This will hide all incidents below the user selected threshold.

Although the suggestion for this project was to focus on target area(s) in New Jersey, our system will be able to map all of New Jersey, except it will reduce the number of roads considered to major highways and thruways. As such, we intend to implement this service by partitioning New Jersey into 3 target sectors. This will allow us to sufficiently represent the traffic in a given area and give the users enough granularity so that they can monitor the incidents reports in their region. The idea here is that local roads are, in general, less congested than major highways and where accidents will have less impact due to lower speed limits and ample detours. This assumption will be verified throughout the course of the project but the principles behind the assumption seem accurate.

(2) “Incident reports along a route”. The route maybe selected from a drop down menu which lists several major highways and thruways.

A) “Time”. The time interval can be one of
   I. Intervals of 1 hour (example: “3AM”)
   II. “All”. Traffic reports for all time periods within the selected area are displayed.

B) “Type of day”. This can be one of
   I. “Weekday”. The traffic reports spanning Monday-Friday are considered.
   II. “Weekend”. The traffic reports on Saturday and Sunday are considered.

C) “Show incident severities of”. Can be a choice from 1-5 (least-severe). This will hide all incidents below the user selected threshold.

This service behaves in the same manner as the “Incident reports within an area”. The only difference is that it shows the traffic incidents along single routes. Given our idea of showing traffic along major highways, the “Incident reports along a route” service will give the option to look at the traffic patterns on one major highway. We intend to extend this portion such that the user will be able to zoom into their region (North, Central or South) while only looking at one route. However, this extension will depend on our progress throughout the semester.

The above services represent the user accessible front-end of our traffic monitoring system. The back-end of our system encompasses the “Weather collection” and “Traffic collection” services. These services will augment the front-end services by collecting the data, performing statistical analysis and returning the processed data to the front-end. Both of the back-end services are accessible and configurable by the administrator only, so users will not have access to the raw data nor the components that collect and process the data and requests.

The “Weather collection” service requires and allows update to the following parameters:

(1) “Name of highway”. This parameter is used to retrieve weather along this route.
(2) “Time interval”. This parameter will determine how long the collection service waits before requesting for more data.

The “Weather collection” service utilizes the administrator parameters to retrieve data from a weather forecasting service such as “weather.com” or “AccuWeather.com” along the specified highway. The service then parses the information that is retrieved and stores it in the database for future use in the statistical analysis by the system. The frequency of data retrieval is determined by the second administrator parameter (Time interval). The weather collection service will gather data at every time interval. Because our system is based on major highways, there is really no need to coordinate with the Traffic collection service because there are weather services that track weather conditions along major routes. This gives greater flexibility to our system because the weather and traffic collection services are independent and can perform their task simultaneously.

The “Traffic collection” service requires the same two parameters as the weather collection service.

(1) “Name of highway”. This parameter is used to retrieve data along a specific highway.
(2) “Time interval”. This parameter is used set the delay(sleep time) between two data retrieval cycles.

The “Traffic collection” service will access a live traffic monitoring service, such as “511nj.org” or “Yahoo! Live Traffic” in order to retrieve the incidents data. When the data retrieval is done, there are a few checks that the collection service needs to do before adding the data to the database. The first should be a check to see whether the incident already exists within the database. If the incident does exist then the service needs to check if the current incident report just retrieved was updated since the previous reference to the same incident. If this is also true then the database entry referring to the same incident is modified to reflect the update. If the condition was false and there was no update then the system should not add a new database entry and move on to the next task or next incident on the list. There is no dependency on weather so the traffic collection service can perform its task in parallel with the weather service. The only requirement is that the administrator should configure both weather and traffic collection services to retrieve data on the same highway.

Glossary of Terms

Job Scheduler:	A logical component in our monitoring system which collects weather and traffic data at defined intervals.
Graphical User Interface (GUI):	A type of interface that allows users to interact with the website using graphical components (buttons, checkboxes, etc.) and images instead of text-based commands.
Mapping Service:	A website or web service which can provide customizable geographic maps that can be used in our graphical user interface.
Weather Service:	A website or web service that contains weather data that can be collected, parsed and stored.
Traffic Service:	A website or web service that contains live traffic data that can be collected, parsed and stored.
Database:	A storage device that can be used to store the weather data and traffic data being collected.
Administrator:	An entity which has privileges to customize, alter and update all facets of the traffic monitoring system.

Function Requirements

Stakeholders

- Users
- Administrator

Actors and Goals

- User
  - Initiating Actor
  - The user's goal is to access weather and traffic information pertaining to a chosen region or along a chosen route.
- Administrator
  - Initiating Actor
  - The administrator's goal is to set up the weather collection and traffic collection components of the traffic monitoring system to collect data for use in the statistical analysis process.
- Job Scheduler
  - Initiating Actor
  - The goal of the job scheduler is to automatically schedule the data collection system to collect and process data at administrator defined intervals.
- Database
  - Participating Actor
  - The goal of the database is to store the parsed weather and traffic information for future use in the statistical analysis process. The database is an intermediary type since the data collection side and the user interface must have access to it.
- Mapping Service (Google Maps)
  - Participating Actor
  - The goal of the mapping service is to present an interactive map used to display traffic information to the user.
- Weather Service (weather.com)
  - Participating Actor
  - The goal of the weather service is to provide weather forecast data, which the data collection system can process and store for future use.
- Traffic Service (511nj.com)
  - Participating Actor
  - The goal of the weather service is to provide live traffic data, which the data collection system can process and store for future use.

Use Cases

Casual Description:

- UC1: ViewTrafficStatistics
  - Provides user with traffic conditions along a chosen highway or chosen region. See System Sequence Diagrams: UC1

- UC2: ScheduleWeatherCollection
  - Collect weather data based on administrator configurations at intervals specified, once again, by the administrator.
- UC3: ScheduleTrafficCollection
  - Collect traffic data based on administrator configurations at intervals specified, once again, by the administrator.
- UC4: AdministratorSettings
  - Let the administrator configure the database and job scheduler for collection and storage.

Fully-Dressed Description:

- Use Case 1: View Traffic Statistics on the Highway

Primary Actor: User.
Goal: To view traffic conditions at a particular time along the highway.
Stakeholders: Supporting actors are Google Maps, Database.
Precondition: Nothing important to mention.
Post condition: Nothing important to mention.
Main Success Scenario:

1. User enters his/hers location (North, Central, South) in New Jersey.
2. User enters the name of the highway he wants to travel on, day of the week, time of the day, weather condition, and incident severity.
3. System request Google maps for latitude and longitude using user's location.
4. System sends request to database for traffic and weather condition data along the highway.
5. System calculates statistics on the highway and retrieves the map from Google maps.
6. System overlaps statistics and incident marks on the given map.

Extensions: Given location or highways do not exist in the data base and it ask user to enter the location or the highway again.

- Use Case 2: Job Scheduler for Weather Conditions

Primary Actor: Job Scheduler
Goal: To collect weather data along the highway and store the data in our data base at certain interval of time.
Stakeholders: Supporting actors are weather channel, data base and administrator.
Precondition: Administrator.
Post condition: Current weather data is stored into the database.
Main Success Scenario:

1. Job scheduler initiates weather collection process.
2. System request weather channel for weather conditions along the highway at a particular interval of time.

3. System gathers data and puts the data into the database.

Extensions: Nothing to mention.

- Use Case 3: Job Scheduler for Traffic Conditions

Primary Actor: Job Scheduler.
Goal: To collect traffic data along the highway and store the data in our data base at certain interval of time.
Stakeholders: Supporting actors are weather channel, data base and administrator.
Precondition: Administrator.
Post condition: Current weather data is stored into the database.

Main Success Scenario:

1. Job scheduler initiates traffic collection process.
2. System request Yahoo traffic for traffic conditions along the highway at a particular interval of time.
3. System verifies that each incident reported is new and then adds into database otherwise just ignores that incident so there is no duplication of data.
4. System gathers data and puts the data into the database.

Extension: Nothing important to mention.

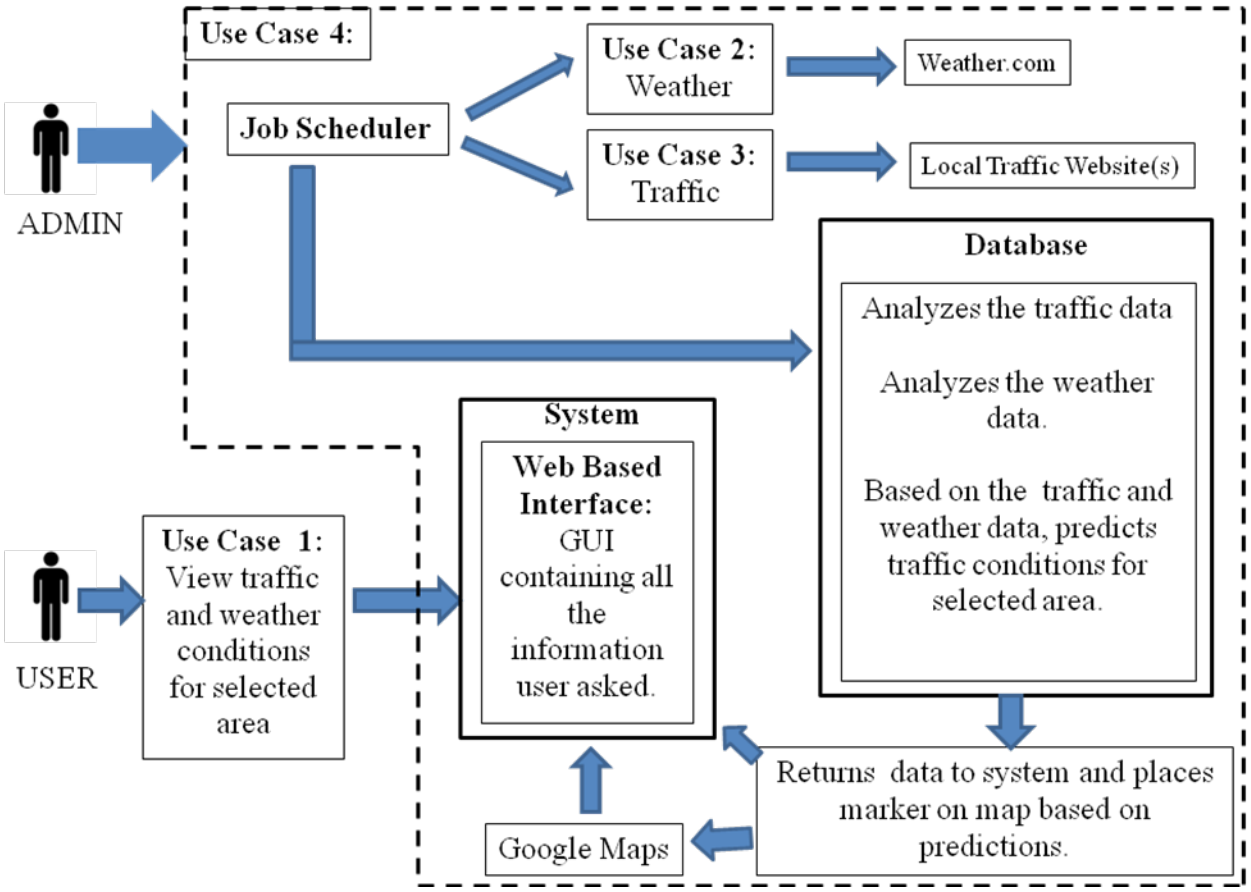
- Use Case 4: Administration

Primary Actor: Administrator.
Goal: To design traffic and weather collection criteria and create database to store the data.
Stakeholders: Supporting actors are database and job scheduler.
Precondition: Nothing important to mention.
Post condition: Criteria are finalized and database is created.
Main Success Scenario:

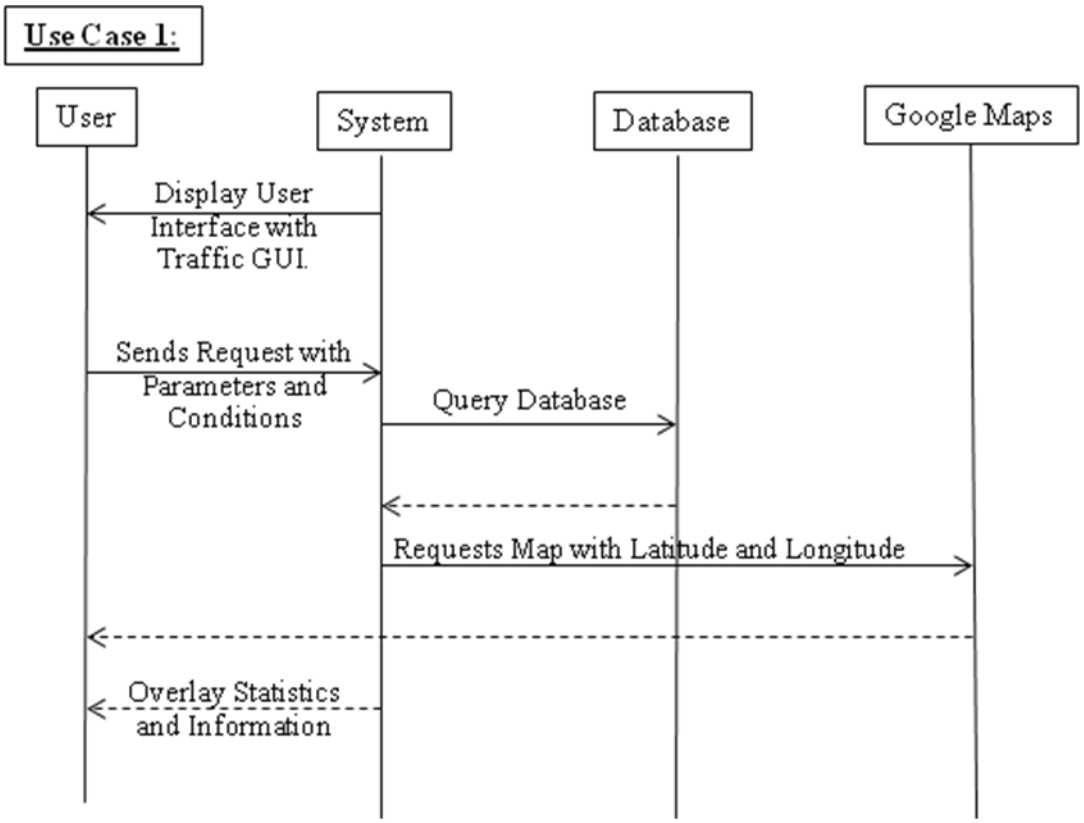
1. Administrator supplies Yahoo traffic and weather channel to supply data.
2. Configuring the database and then store data into it using Use Case 2 and Use Case 3.

Extension: Nothing important to mention.

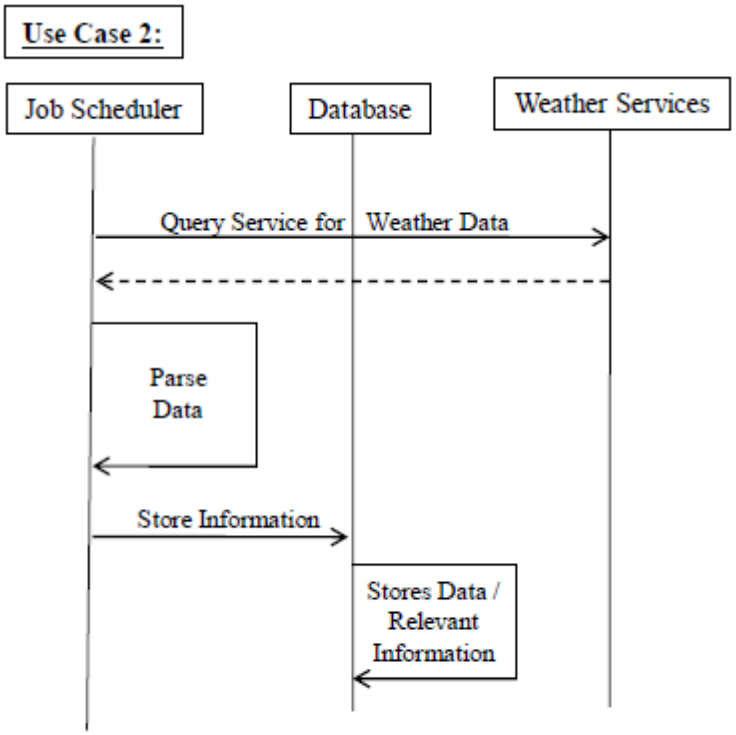
Use Case Diagram:



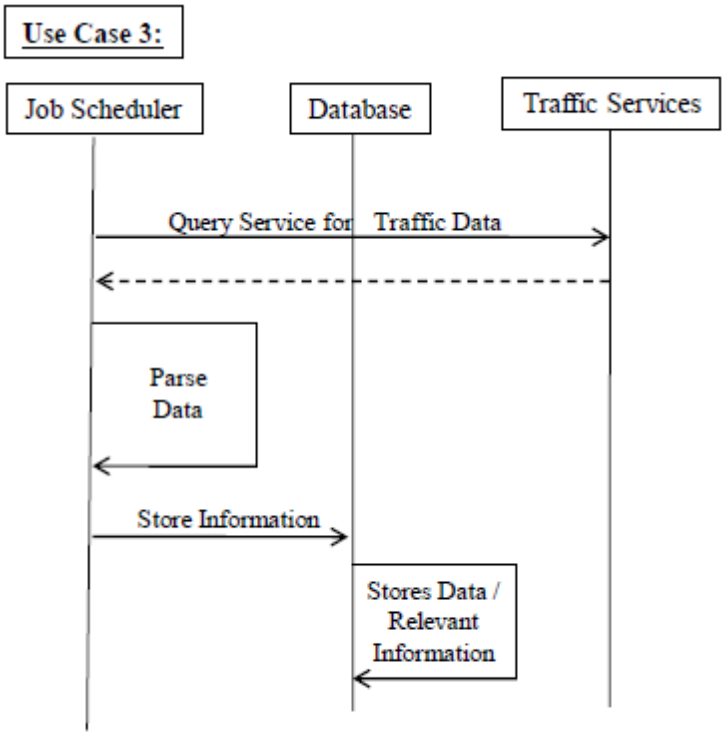
System Sequence Diagrams



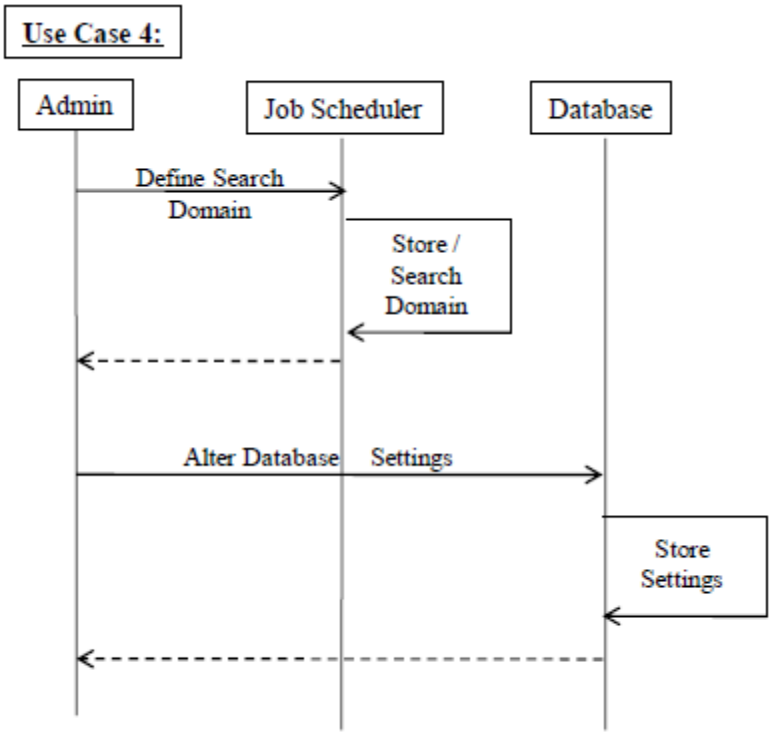
In UC 1 the system displays the user interface to the customers and requests the information. The user then enters the information and sends it back to the system. Then the system then does a query search in the database for the requested traffic data, and retrieves them information. The system then pings Google maps with the latitude and longitude to recover the map of the selected region. Next, Google sends this information back to the system, which is finally returned to the original user.



In UC 2 the job scheduler pings the weather services for the data for defined intervals. Then the weather service sends the data back to the job scheduler, which then stored it into the database. The job scheduler then parses through the data, takes the relevant data and sends it to the database. Finally the database stores the information into the database.



In UC 3 the job scheduler pings the traffic services for the data for defined intervals. Then the traffic service sends the data back to the job scheduler, which then stored it into the database. The job scheduler then parses through the data, takes the relevant data and sends it to the database. Finally the database stores the information into the database.



In UC4 the administrator provides search criteria to the job scheduler so it can gather traffic and weather condition data. Job Scheduler store search domain setting and collects the data. Administrator has the access to database to fix the database if necessary.

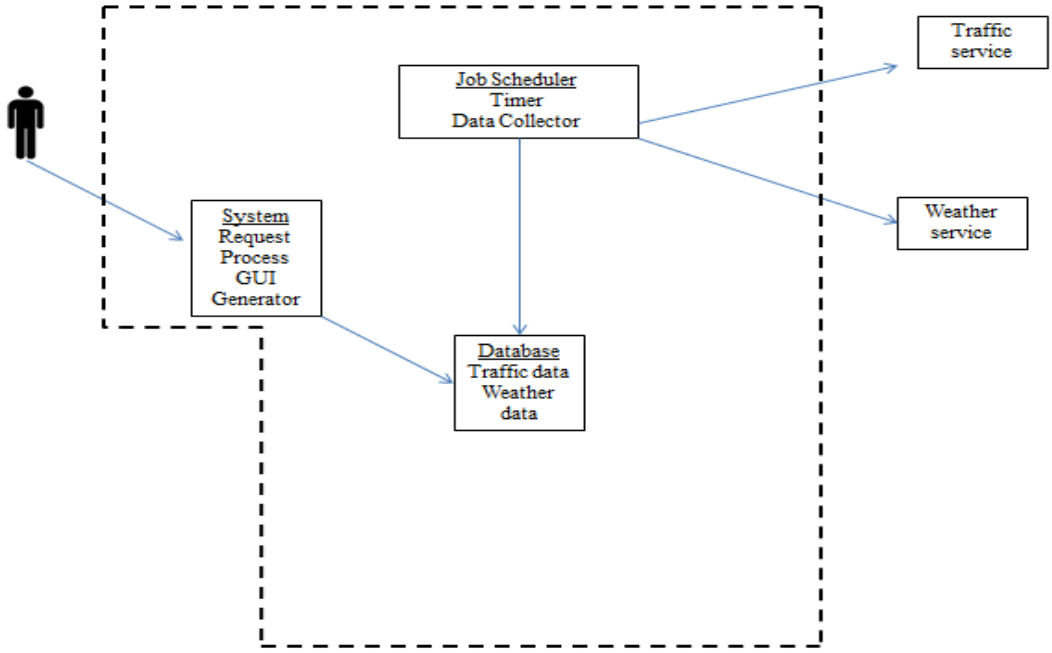
Nonfunctional Requirements

- Quality Requirements for “Traffic Monitoring System”:
  - “Traffic Monitoring System” database is not going to collect any irrelevant data.
  - “Traffic Monitoring System” should process the data and display the result within 15 seconds.
  - “Traffic Monitoring System” database should have sufficient amount of space for the data but not too large.
  - Multiple user should be able to access the website.

- Constraints:
  - The data collection from 12 AM to 4 AM will be collected in greater intervals due to less traffic.
  - “Traffic Monitoring System” should respond efficiently even when there are multiple users accessing the website.
  - “Traffic Monitoring System” should be user friendly and easy to navigate

Domain Analysis

Domain Model:



Concept Definition:

System:

- It is used to generate graphical user interface so that user can interact with the system
- System processes request made by the user

Database:

- It is used to store traffic and weather data

Job Scheduler:

- Job scheduler is used to schedule the data collection system to gather traffic and weather conditions at particular intervals of time.

Association Definition:

System:
System accesses data from the database and provides traffic condition at that point of the day to the user so that user can reach his destination using the highway with least traffic and reach safe on time.

Database:
It is used to store traffic and weather condition data along the highway and is accessed by the system to provide user with traffic conditions so it takes user least time to reach his destination safe.

Job Scheduler:
Job Scheduler is used to initiate traffic data and weather data gathering so that they can be parsed by the collection system and stored in the database.

Attribute Definition:

System:

1) Request processor: it is used to process the inputs provided by the user and access the data from the database to provide it to the users so user can take the highway with least traffic and reach their final destination on time.

2)GUI Generator: This is used to provide an interface so user can interact with the traffic monitoring system and supply the highway, day of the week , incident severity and location to the system for processing. GUI is used by the monitoring system to give users traffic predictions along the highway/region they want to view.

Database:

1) Traffic data: This is used to store traffic data.
2) Weather data: This is used to store weather data.

Job Scheduler:

1) Data Collector: Data Collector initiates traffic and weather data for collection and processing, once approved by the timer and saves the data in it. Then, data collector stores this data into the database.
2) Timer: it is used by Job Scheduler so that the data collector may process data at defined time interval provided by the administrator.

System Operation Contracts:

- Parse data:
  Precondition: Data is stored in the database
  Post condition: it makes sure data is parsed into weather conditions data and traffic condition data

- Service request:
  Precondition: There is enough data in the database to be provided to the user
  Post condition: This operation returns user with markers overlaying on the map. If there is no data in database for a particular highway or location, it returns map with no markers on it suggesting to the user that there is no data for the criteria provided by the user.

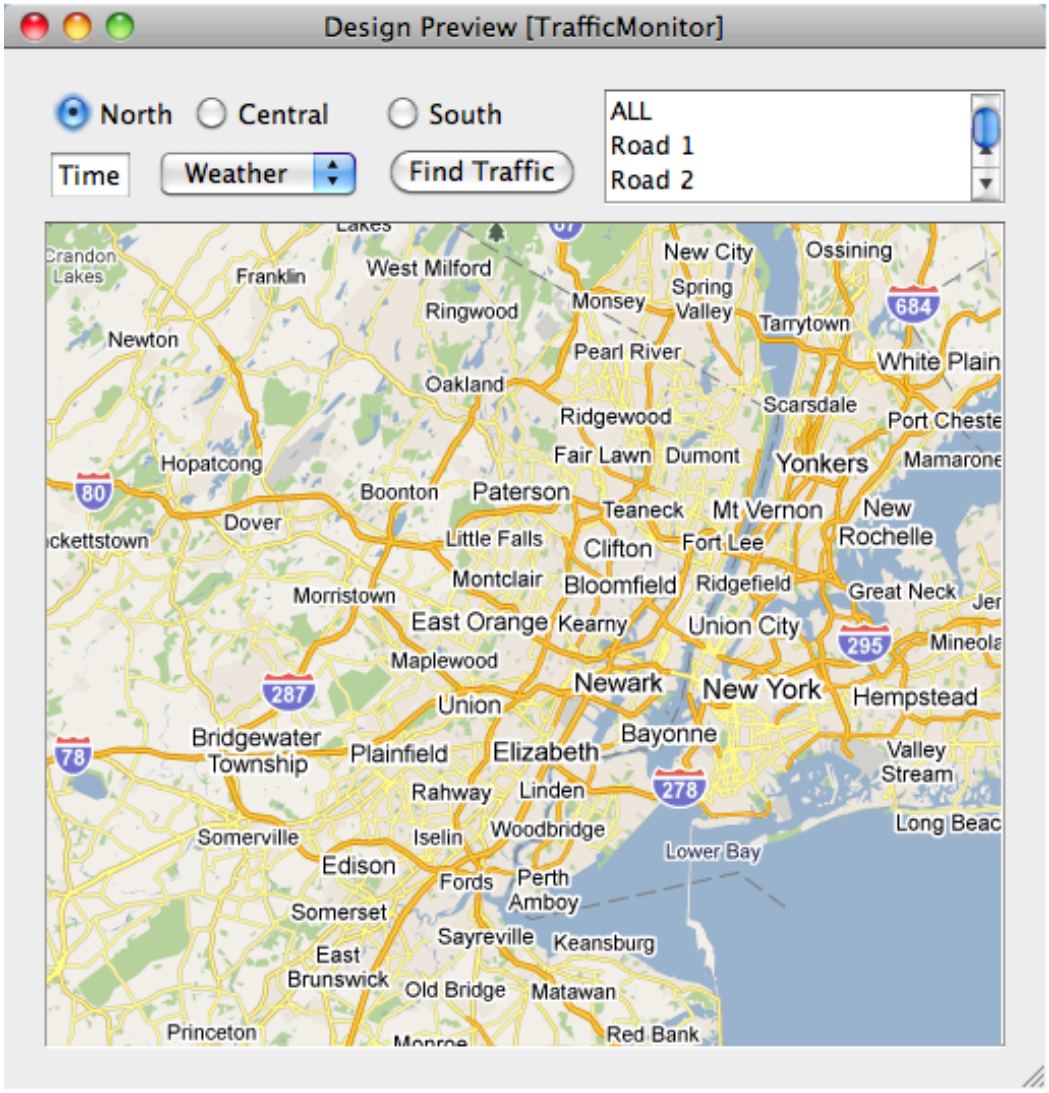
- Calculate Results:

Precondition: Criteria provided by the user must be available in the database.
Post condition: Statistical averaging of the target traffic area/route is computed and returned.

User Interface

Preliminary design:

To access and use the Traffic Monitoring System, the user first opens their preferred internet browsing application / program and loads the project's website. The user enters all pertinent information on this page to find traffic and weather information. The user clicks one of the three radio buttons designated to the three different regions of the state of New Jersey - North, Central, and South to find information about the desired area the user chooses to travel. The user then enters the time of day. Using a drop down menu, the weather type is selected: rain, snow, sleet etc. or can select display all types of weather. The user then selects the highway or all highways to travel on using the scroll menu. The user then submits this information to be queried by the website to display information.
The user has an option to view three different sets of information. One choice is to see traffic predictions for the selected route. The second choice is to see traffic predictions for the entire selected region. The third option is to overlay the map with live traffic condition from a live website (such as Google traffic/NJ traffic etc.).



User effort estimation:

The effort put in by the user to get all the information required, including mouse clicks and keystrokes. (***Using an example time period of 9:00A.M.***)

- **NAVIGATION**: Total of two mouse clicks, as follows
  - Open preferred web browser to load project website

  ---*After completing data entries as shown below*---
  - Click *"Show Traffic"* to display the traffic in the selected region.

- **DATA ENTRY:** Total of 5 mouse clicks and 6 keystrokes
  - Click the radio button to select region to travel in.
  - Press the *"Tab"* key to move to the text field (*"Time"*).
  - Press the keys *"9", "0", "0", "A", "M"* (Enter time in a 12-hour format without the colon for nine in the morning).
  - Click the drop down menu named *"Weather"* and select a weather condition by clicking on a choice.
  - Click on the preferred highway OR click on ALL to select all roads.
  - Click the radio button to display which map the user wants to see.

Plan of Work

Project Milestones	Dates
Set up data collection system including weather and traffic statistics	March 6th 2011
Set up a database server to store processed weather and traffic data	March 6th 2011
Assemble GUI Interface	March 10th 2011
Develop the statistical averaging model for traffic prediction	April 17th 2011
Integrate GUI and collection system	April 3rd 2011

References

1. http://maps.yahoo.com/
2. http://www.511nj.org/
3. http://www.weather.com/
4. http://www.traffic.com/
5. http://maps.google.com/
6. Software Engineering by Ivan Marsic Statement of Requirement Page 64
7. Software Engineering by Ivan Marsic Use Case Modeling Page 71
8. Software Engineering by Ivan Marsic System Specification Page 78
9. Software Engineering by Ivan Marsic Use Case Modeling Page 71