Traffic Monitoring Service

Group No. 7

Report 2

Team Members

Name	Email
Kevin Hsieh	**hsieh63@eden.rutgers.edu**
John Reed	**johnreed@eden.rutgers.edu**
Geoff Oh	**geoffrey.oh@rutgers.edu**
Mike Simio	**michaelsimio@gmail.com**
Peter Lin	**Peterlin741@gmail.com**
Matt Araneta	**maraneta@eden.rutgers.edu**

Instructor: Prof. Ivan Marsic

Project URL: http://traffichistory.co.nf/

Revision History:

Version No.	Date of Revision
v.1	03/03/2013

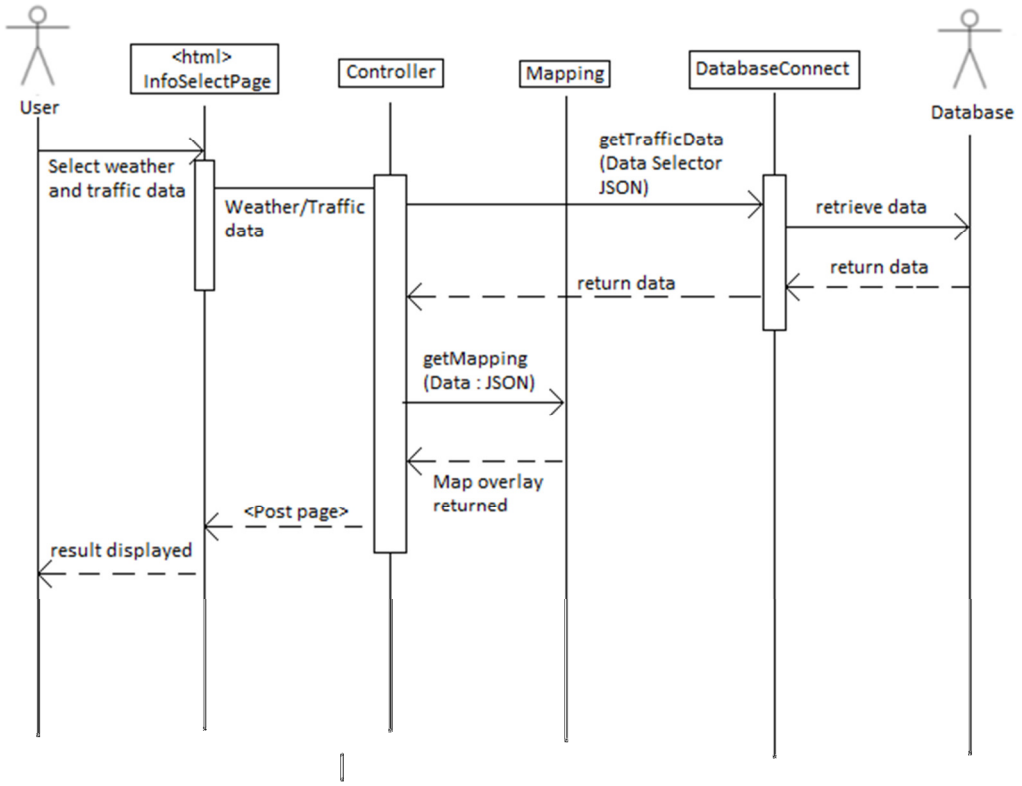
Breakdown of Contributions

All members contributed equally for this report.

Table of Contents

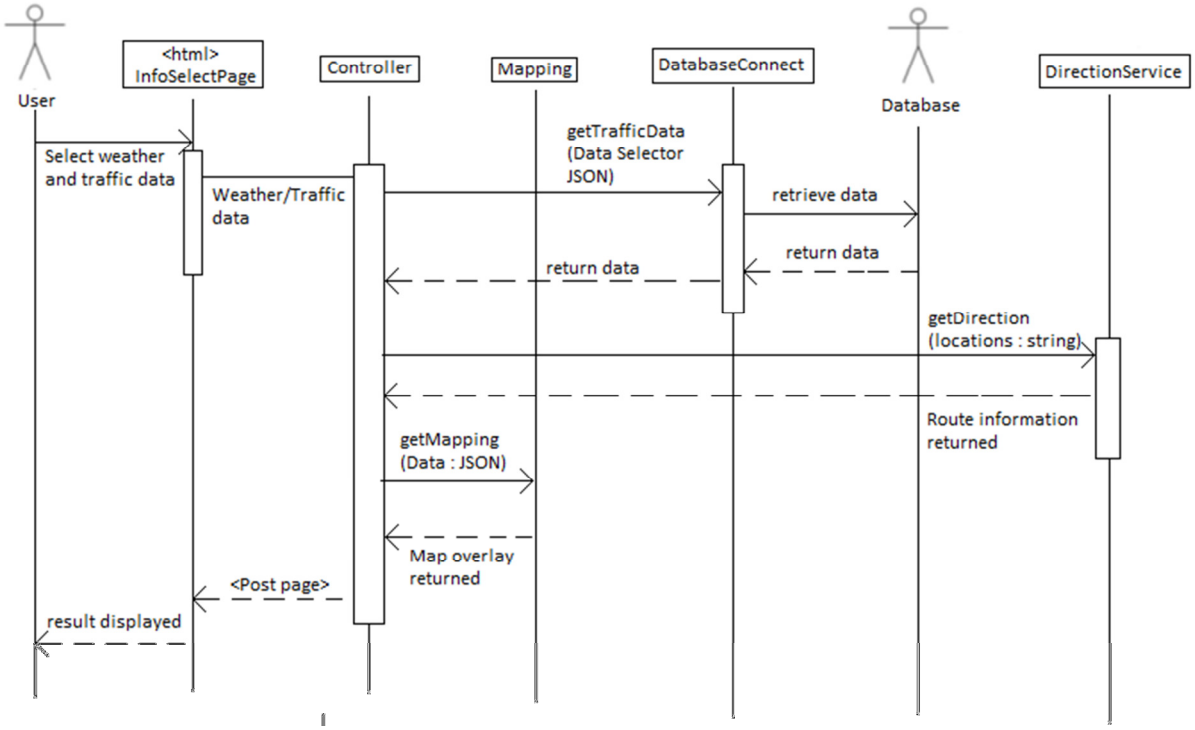
Interaction Diagrams

Use Case 1:



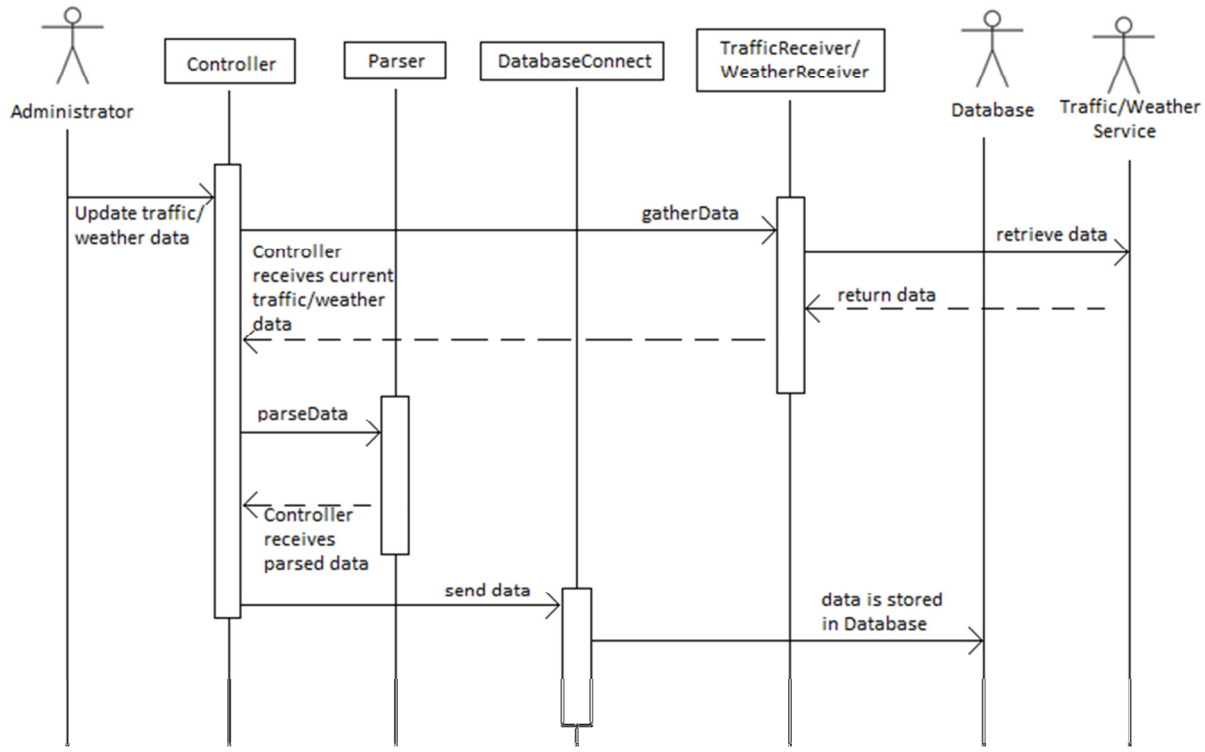
This use case has the responsibility of analyzing the inputs entered by the user to show the relevant traffic history for the area they desire. The web Application receives the data entered by the user, and using the Expert Doer principle, sends the data to the Controller. Using the High Cohesion principle, DatabaseConnect is used as an intermediate between the controller and the database. This ensures that the controller does not do too many computations when attempting to access data from the Database. The Database then returns the information through DatabaseConnect back to the Controller. The Controller sends the Database information to the Mapping service, and it returns a map overlay with the information posted. The Controller posts the map on the web Application, and the information is made available for the user.

Use Case 2:



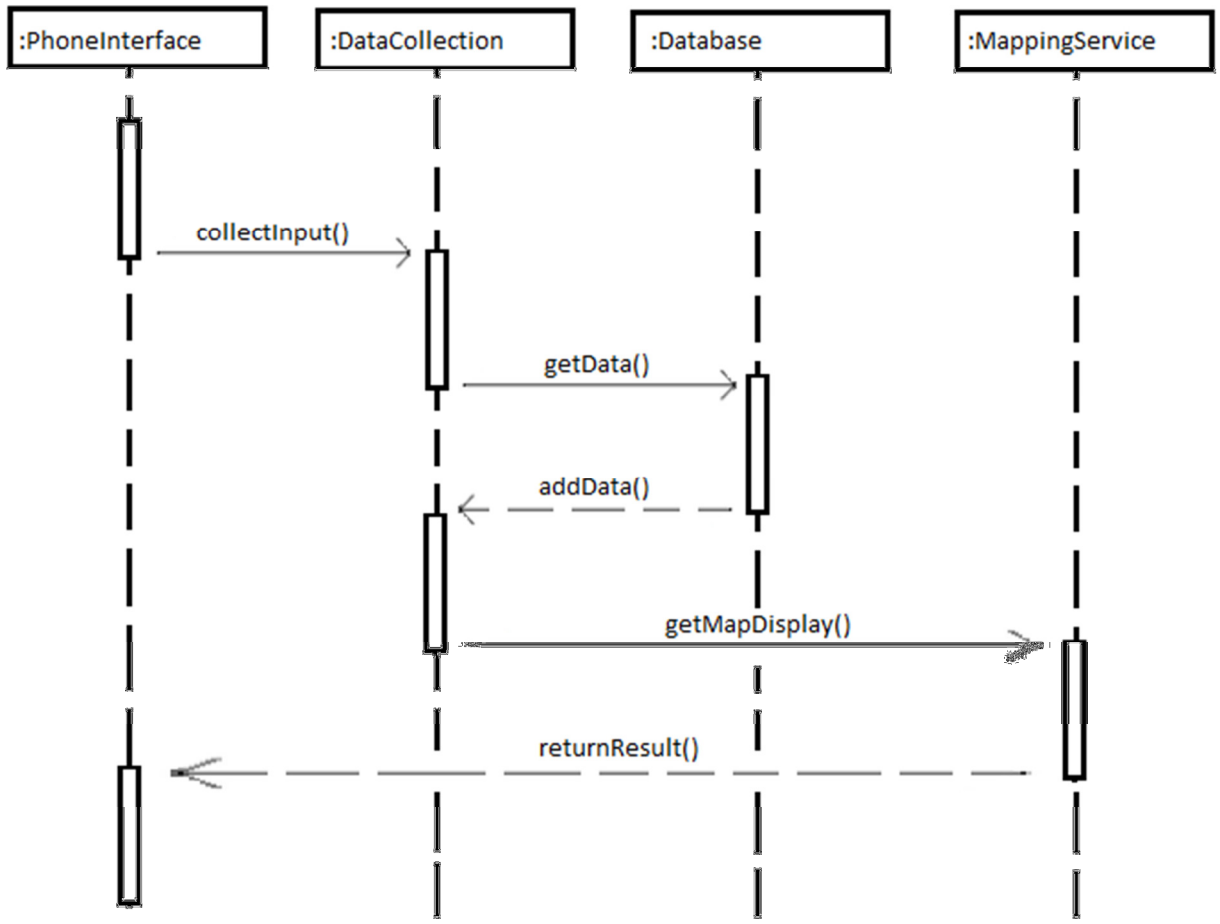
The methodology for the second use case is similar to that of Use Case 1. It employs the Expert Doer principle for the information sent, and it uses the High Cohesion principle in the form of DatabaseConnect. The difference from the first use case is the addition of a DirectionService. After the Database information is sent back to the Controller, the Controller then accesses the DirectionService to get the route information desired by the user. A list of directions is sent back to the Controller, and the Controller then adds the list of the directions to the object it sends to the Mapping service. The Mapping service returns the overlay of traffic history along with a route the user can follow to reach his or her destination. This information is sent to the web Application so that it can be viewed by the user.

Use Case 4 and 5:



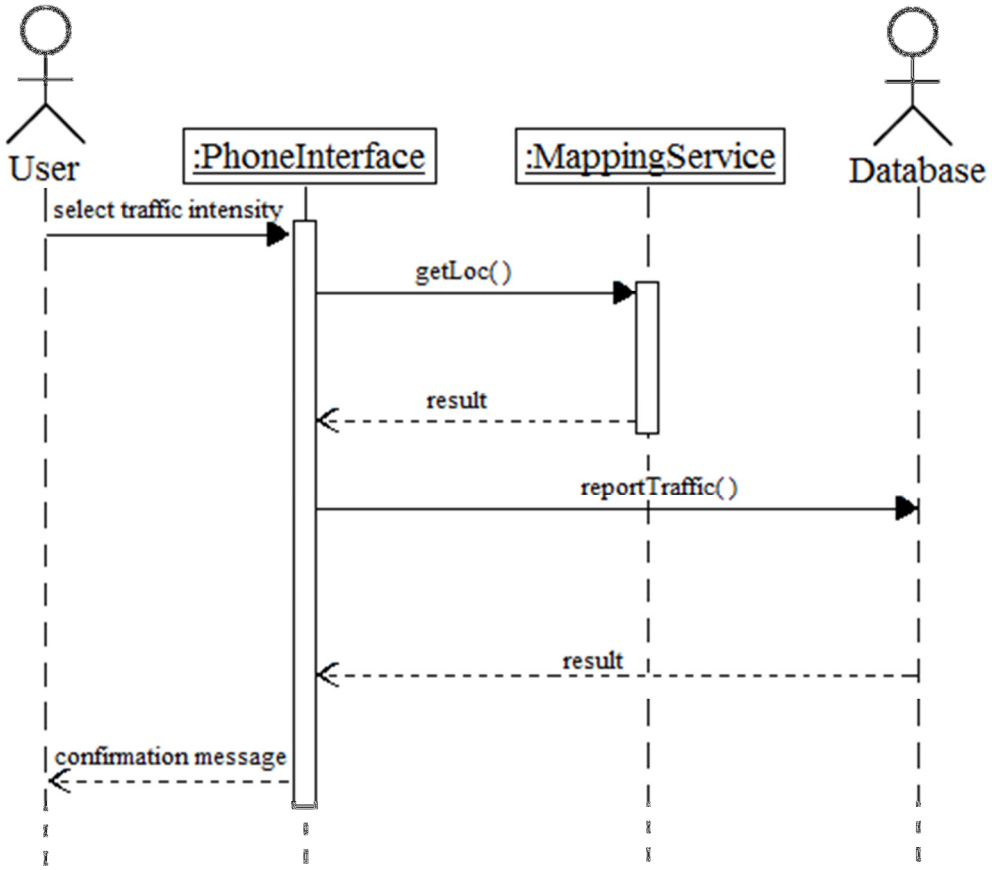
These use cases have the responsibility of accessing the Traffic and Weather Service websites to retrieve data at constant intervals. At every time interval, the Controller is told to update the traffic and weather data. Through the High Cohesion principle, the Controller sends for the TrafficReceiver or WeatherReceiver to access the data from the web services. This data is then sent back to the Controller via the TrafficReceiver and WeatherReceiver. Using the High Cohesion principle again, the Controller sends the data to a Parser to format the data in a fashion that is easier to use. The Controller then uses this parsed data and sends it to the DatabaseConnect. The Low Coupling principle is used, to ensure that the Database has the least number of connections possible. The only pieces of the system that should interact with the Database should be those specifically made to do so. With the database storage, this use case is complete.

Use Case 6:



This use case has the responsibility of getting traffic history data shown to a mobile user. Similar to Use Case 1, it employs the Expert Doer principle. The user enters his or her data to the Mobile Application and sends it to the controller of this system, DataCollection. The DataCollection uses the data sent by the user to retrieve the Database information relevant to his or her request. The data is sent back to DataCollection, and this data is then sent to the MappingService. The MappingService uses the data sent by DataCollection to show a map overlay that shows the relevant traffic history requested by the user. This overlay is sent to the Mobile Application to be seen by the user.

Use Case 7:



This use case handles the responsibility of receiving the traffic intensity data entered by the user. This use case employs the Expert Doer principle, as the user inputs the traffic intensity for his or her location, and sends it to the controller of this system, the PhoneInterface. The PhoneInterface must then retrieve the user's location, so it accesses the MappingService to do so. After the user's location is returned, the PhoneInterface can send the traffic intensity and user's location to the Database to be stored for future use. The user is sent a confirmation message to show that his or her data has been received.

Project Management

Peter Lin and Matt Araneta are a team that will be working on web design and gathering data for traffic and weather. They will be in charge of use case 4 and 5. They will collaborate with Kevin and John on use cases 1, 2, and 3.

Geoff Oh and Mike Simio are a team that will be working on the mobile application of the project. They will be in charge of use case 6 and 7.

Kevin Hsieh and John Reed will be working on implementing an algorithm for finding the fastest route for directions, and managing the data gathered. They will work with Peter and Matt on use case 1, 2, and 3.

References

1. Marsic, Ivan. *Software Engineering*. 2012.
2. View Traffic Incidents. <511nj.org /IncidentList.aspx>.
3. View weather. <weather.com>.
4. Google Maps. <maps.google.com>.
5. Yahoo! Live Maps. <maps.yahoo.com>.
6. View Traffic Reports. <traffic.com>.