Málaga, 12 de Diciembre de 2012

Informe Ejecutivo

Título:	VANET-3.0-2012: Búsqueda de rutas óptimas basadas en el estado del tráfico
Resumen:	En este documento se describe una aplicación distribuida que permite a los usuarios conocer el estado del tráfico y obtener las rutas más cortas para ir a su destino en coche. La aplicación está disponible tanto a través de la Web como para dispositivos móviles con Android. El estado del tráfico se calcula a partir de los datos geográficos recolectados por los dispositivos móviles.
Objetivos:	1. Presentar la arquitectura de una aplicación distribuida para la recolección de datos de tráfico y el cálculo de rutas óptimas.
Conclusiones:	1. La aplicación aquí presentada puede ser muy útil para quienes tienen que circular en coche por una ciudad con problemas de congestión de tráfico. 2. Esta aplicación es un buen ejemplo de transferencia de conocimiento de la Academia a la Sociedad.
Relación con entregables:	No tiene dependencias.

DIRICOM

Diseño Inteligente de Redes Inalámbricas de Comunicación
Conserjería de Innovación, Ciencia y Empresa
P07-TIC-03044
http://diricom.lcc.uma.es

Málaga, December 12th, 2012

Executive Summary

TITLE: VANET-3.0-2012: Optimal Route Search based on Traffic Information

ABSTRACT: This document describes a distributed application that allows the users to know the road traffic state and to compute the shortest routes in car from an origin to a destination. The applicacion is available on-line and can be installed in mobile phones with Android. The traffic state is computed from geographic data sent by the mobile phone application.

GOALS:

1. We present the software architecture of a distributed application to gather traffic data and compute optimal routes in car.

CONCLUSIONS:

1. The application can be very useful for people driving in a city with congestion problems.
2. This application is a good example of transference of knowledge from Academy to Society.

RELATION WITH DELIVERABLES: No dependencies.

Optimal Route Search based on Traffic Information

DIRICOM

December 12^{th}, 2012

1 Software Architecture

The application has been decomposed in five different subsystems that are detailed in the following:

- **Traffic data gathering**. This component is in charge of gathering the geographic data generated by the mobile devices. The information is stored in a database to be processed. It is composed by two software pieces: the client, which is in the mobile device and is gathering the GPD data to send them to the server; and the server, which stores the data from the client in the database. Each user must have an account in the system in order to associate their data to this account for future queries. Anyway, when the information is used for the purpose of computing the traffic state or optimal routes, the data is shared in aggregated form in order to avoid the identification of particular users.

- **Traffic information processing**. The information sent by the mobile devices is raw (time and geographic coordinates). This information tells nothing about the traffic congestion. It is necessary to process the information stored in the database in order to obtain high level information about the traffic state. This component process the raw data and compute an estimated traversing time for each street of the city.

- **Optimal route computation**. It computes the optimal route from an origin to a destination. We distinguish two modules: one in the mobile device, used by the user to submit a query; and one in the server, which makes the computations. In order to find the shortest route the server uses the Dijkstra algorithm. The city is considered as a large graph where the vertices are crossings and traffic rounds while the arcs are the streets or part of them. Each arc (street) has an associated cost, that is related to the time required by a car to traverse that street. This number is computed by the traffic information processing subsystem and the traffic forecast subsystem.

- **Traffic forecast**. The time to traverse a street changes dynamically, especially when a traffic jam is starting or stopping. In order to take this issue into account we need to predict the traffic state in the near future. The Dijkstra algorithm will ask for future forecasts on demand. It considers the case that after a car has traversed an arc, the time to traverse the next arc change, is not the current one. The traffic forecast subsystem is in charge of making these predictions about traffic in the future. This is done using methods of time series prediction based on the past data stored in the database.

- **Past routes and traffic monitoring**. The application shows the traffic state in the past, present and also future predictions of the traffic state. In order to do this it draws on a google map lines covering the streets in the city with colors depending on the time to traverse these streets. For fluent traffic the color is green and for severe jams the color is red. Yellow and orange are used to represent moderate congestion. In addition, the registered users can see and manage the routes they search for in the past and the ones that they really followed. This information is available in both, the Web application and the mobile phone application.