TRAFFIC MANAGEMENT SYSTEM

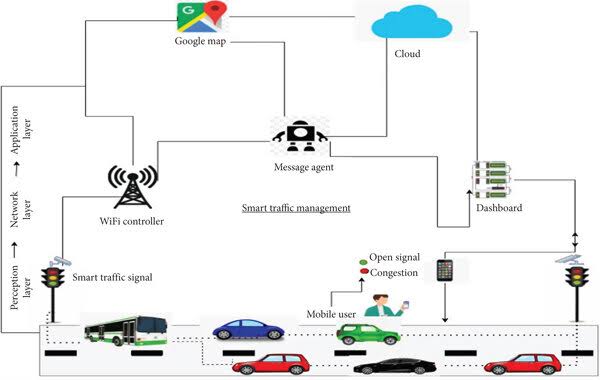
BATCH MEMBER

962121104022: SHIBINA S S

Phase 4: Submission document

Project Title : Traffic Management System

Topic: Continue building the project by the traffic information platform and mobile apps



# Introduction

# 

* . A constant increase in demand for transport make city authorities turn to solutions of Intelligent Transportation Systems (ITS), which are the basis for the design of intelligent transport networks.
* Their importance is highlighted, inter alia, in one of the classic definitions of urban logistics, proposed by E. Taniguchi, RG Thompson, and T. Yamada, according to which “city logistics is the process for totally optimising the logistics and transport activities by private companies with the support of advanced information systems in urban areas considering the traffic environment, its congestion, safety and energy savings within the framework of market economy” [1].
* In recent years, the increasing importance has been gained by V2V (Vehicle-toVehicle), V2I (Vehcle-to-Infrastructure) and I2V (Infrastructure-to-Vehicle) systems, which are a direct implementation of the concept called “Cloud computing” in road transport systems. Under an increasing dominance of information technology in most areas of life, their further expansion and increased importance of telematics systems should be expected also in relation to urban freight transportUrban transport telematics

**Web development technology :**

1. **HTML**

* HTML stands for Hyper Text Markup Language. It’s one of the fundamental technologies required for web development.
* It provides the base structure for a web page. HTML code ensures that all the content on a website is properly formatted. This is so your Internet browser can display the content as intended. Without HTML, a browser couldn’t display text or load images and other elements.
* HTML5, the most current version of HTML, specifies a large number of Application Programming Interfaces (API) that can be used with JavaScript for a more interactive and dynamic website:
* Canvas: Canvas is an HTML5 element used to draw images and shapes and manipulate them. It can also be used for more complex cases such as game graphics and animations.
* Web Storage: Web Storage is used to store information right in the browser. Some examples of this would be storing user login information and saving user preferences for a website.
* Service workers: Service workers enable a script that keeps running in the background when a web page is opened and is mainly used in websites with offline capabilities. It makes pages available offline and allows for the use of web push notifications. It can send these notifications even when your browser isn’t open.

## 2. CSS

## Cascading Style Sheets, abbreviated as CSS, define the style and aesthetics of a web page. While HTML is used to structure a web page, CSS specifies the appearance of that structure.

## This includes page layouts, colours, fonts and element positioning. If HTML is the bones of the web page, CSS is the skin. It makes the Internet, and your website, look good.

## 3. Programming languages

## Programming languages are ways to communicate with computers and tell them what to do. They provide a vocabulary and set of grammatical rules to instruct a computer to perform specific functions.

## There are lots of different programming languages just as there are lots of human languages. Developers typically specialize in a few languages.

## Below are a few examples of different languages:

## JavaScript:

## JavaScript is the most used language by startups for a few reasons. It can be used as both a front-end and back-end programming language.

## Compared to other languages, JavaScript is comparatively easy-to-learn (although all languages come with some difficulties), and it’s used everywhere in custom web application development. It’s also constantly being updated and expanded with new features.

## All the following are back-end languages that make a website or application work:

**Python:**

* Python is another popular programming language. It’s relatively simple and uses a unique syntax that focuses on readability. Python is good for artificial intelligence (AI) applications and projects, statistics, data science work, and machine learning. It can also be used to create web applications.

**Java:**

* Java is a language that’s popular in enterprise companies. It can be run on multiple platforms, has extensive documentation, and is supported by Oracle (a large and respected software company) in case you run into trouble. It can also be used to write Android applications.

**Rust:**

* Rust is a newer language that focuses on safety and especially safe concurrency. Concurrency in software engineering means the collection of techniques that enable a computer program to perform several different tasks simultaneously. Safe concurrency gives you the efficiency needed to execute multiple tasks while keeping your code safe and secure. Rust is high performing and has gained market traction even as a newer language. by length and weight, speed) without interfering the safety and behaviour of drivers;

**4. Web development frameworks**:

* Web development frameworks are tools and libraries that developers use to make developing in a particular language easier and more efficient. They provide interfaces to access commonly-used functionalities as well as abstractions that make complicated things easier to understand and handle.
* Essentially, they make the development process more efficient because developers have a pool of code resources to draw from when developing. There’s no need to develop software functions from scratch when frameworks give you all the necessary tools to build a website or application. Because frameworks give you a headstart on your development, they’re more cost effective than developing entirely new code.

## Toll Collection Systems for Entry to the City Centre – the likeliho. The most common use cases include chats and notifications in web apps.Areas of telematics application in urban transport system assistance

The purpose of the use of telematics in urban logistics is to optimize access to logistics hubs and linear infrastructure. Its use implies, among other things [2]:

* reduction of freight distribution costs

­ increase in productivity of local delivery vehicles;

­ increase in reliability of commercial vehicle operations;

­ increase in safety;

* increase in the capacity of urban freight systems (without providing additional traffic infrastructure).

Telematics solutions support many aspects of urban transport systems, including vehicles, infrastructure, and organization and freight management. Among the solutions developed in recent years in the world, particularly important are [3]:

* Traffic Management Systems – task of telematics systems is to control “working” vehicles by responding to any changes in communication processes;
* Systems for Classifying and Weighing Vehicles – classification systems allowing for counting vehicles and their distribution depending on the speed of movement and length, as well as systems for weighing vehicles in motion providing a variety of parameters (e.g. pressure on the axles, weight, distance between vehicles, classification by length and weight, speed) without interfering the safety and behaviour of drivers;
* City Cards – modern and multi-functional medium for electronic products and services, the implementation of which in transport systems simplifies toll systems for urban services (such as entry into the restricted traffic zone);
* Park-and-Drive Systems – parking systems, designed for people commuting from greater distances, enabling quick change to the means of public transport;
* Variable Message Signs – road signs and boards showing information on a given road section and at given time which changes depending on circumstances, such as speed limits, diversions for traffic, delays, functioning of public transportation, available parking spaces, warning of the road dangers, etc.; they can be shown using a variety of technologies (e.g. LED, electromechanical systems, liquid crystal displays); their use improves the level of road safety and increases the efficiency of transport management;
* Passenger Information Systems – are among the most modern transport solutions, enabling for a rapid delivery of information using video and audio signals for passengers; major components of these systems are aggregated timetables, information boards at bus stops, clocks;
* Toll Collection Systems for Entry to the City Centre – the likelihood of transport congestion increases in the realities of dynamic growth in the number of vehicles within urban areas, which is a consequence of the weak "expansion" of urban infrastructure. These systems belong to the administrative solutions improving the situation in this respect; automatic toll collection systems are based on the use of solutions for the diagnosis and classification of vehicles.

The main task of telematics solutions as tools to enable effective support for urban freight transport is the management of information flows generated within this transport, and the most important result of this process is to improve the quality of functioning of the urban logistics system by enhancing the ability to control and impact on the present data flows. Among the many practical solutions, there are three main categories, which are based directly on telematics and where the use of ICT tools determines their proper functioning:

* systems directing commercial vehicles entering downtown on scheduled routes, e.g. by setting specific traffic signs (usually signs and variable message signs - VMS) or providing maps of planned routes for trucks with relevant traffic information;
* intelligent systems for managing transport routes, integrating the planned route and information for vehicles with navigation software, where the data obtained from trucks on their location, load and planned destinations can be linked with data on road traffic in real time;
* integrated logistical tools, which are solutions based primarily on the use of web technologies (mainly the Internet and websites) that allow combining and coordinating producers, buyers and logistics operators in terms of placing orders for optimization of logistics flows.

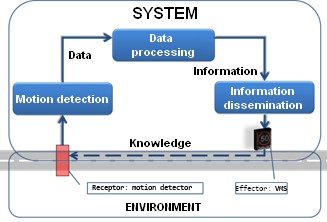
In addition, many other solutions in the field of urban freight transport use telematics systems as additional support tools to improve their effectiveness.

## Variable Message Signs as a key technology supporting traffic control

Particularly important in technical solutions used in telematics systems are Variable Message Signs (VMS), which next to traffic light, are key tools for traffic control. For a specific road section and time VMS can display information varied and dependent on the current traffic situation, such as: speed limits, diversions for traffic, delays, functioning of public transportation, parking available, warning of the road dangers, etc. Information is presented in the form of text or symbols, using special screens or rotating drum signs (solution, which is used less).

VMS can be run by an operator or by an automatic control system. In the second case, the combination of VMS with motion detection subsystems enables for a full implementation of the idea of intelligent transport systems. The system configured in such way includes three key elements (Fig. 1):

* data acquisition subsystem (motion detection) - is a key element deciding on the accuracy of the presented information, and elimination of the human factor (operator) and reduces susceptibility to errors; its task is to transfer data to the information control subsystem concerning e.g. traffic on selected road sections;
* data processing and communication subsystem - is responsible for carrying out data analysis and determining the messages transferred to the information dissemination subsystem;
* information dissemination subsystem - based on VMS, which provides information to road users being a result of the analysis of data from the detection subsystem.



**Fig. 1.** Interaction of key subsystems in an intelligent traffic control systems. Source: [4].

In this context, there are four fundamental processes in the system which should be highlighted (Fig. 1):

* inputs, which are system supply, by which the system interacts with the environment (motion detectors);
* outputs, which show the results of operation of the system and through which the system interacts with the environment (VMS);
* transformation, including the processes within the system, providing transformation of input signals into output signals;
* feedback, being the foundation of system control processes, which is the influence of the final signal (output) generated by the effectors of the system, on the reference signals (input), received by the receptors.

# Intelligent traffic control system in Szczecin

The main objective of the implementation of the traffic management system in Szczecin was to increase the flow of traffic on the approach roads to the city centre within the existing transport corridors along Gdansk Route, by providing road users with real-time information about the current overcrowding and projected states as well as information about traffic incidents or possible alternative routes. As part of this project, its authors designed, supplied, installed, integrated and commissioned ITS technology elements compromising the system, which allows to actively manage traffic in the area of implementation with the impact on the area of the Szczecin Metropolitan Area. The system consists of two main layers:

* application layer, which consists of Urban Transportation Management System (UMTS) with the individual subsystems designed to monitor and manage traffic actively.
* executive layer, consisting of VMS, which allow for the presentation of complex information, signs and diagrams.

The system uses various types of sensors to collect data, including:

* video detectors used for segmental measurement and evaluation of traffic conditions with the ability to determine travel times;  laser detectors to assess the condition of the surface;  optical sensors for adaptive traffic control.

All data are transmitted to the server system located in the Traffic Management Centre. For the purpose of data transmission, the project includes mixed communication system based on wired communications - fibre and wireless communication of GPS / UMTS type. The system is powered by the urban grid and in the data centre part by the backup power based on UPS / generator of the Szczecin Town Hall.

System operators in the Traffic Management Centre define, implement and configure parameters which, thanks to special system algorithms and after processing the measurement data, are used e.g. to provide real-time information to road user about road conditions, including overcrowding, the average travel time on individual road sections and weather conditions. Additionally, it is used to identify traffic incidents and other situations that may affect the traffic, such as mass events.

The system has the functions of traffic forecasting, based on the traffic model, which allows for generating forecasts useful in the planning of routes for certain time intervals. Operators have access to a specialized engineering and simulation tools to support traffic management functions. The system includes cooperation of fully adaptive traffic control algorithms by which it is possible to increase traffic flow at selected intersections in situations where the drivers use the alternative route.

An important element of the system is a web portal, consisting of a few key modules that cover most of the system command and control operations, several external interfaces, and several independent modules for functions that are implemented independently. The portal core consists of:

* a common graphical user interface (CGUI), which displays data on the system using GIS data provided by the GIS module;
* data collection module that is used to collect data from the field devices and store them for later processing;
* road network module, which allows for dynamic changes in the network;
* traffic forecasting module that uses data collected by the data collection module and road network model in order to provide proper forecasts for road traffic;
* video monitoring module, which allows to display real-time video images of specific devices in a common graphical user interface (CGUI), and also to control video cameras remotely;
* VMS module to manage VMS installed in the system;
* external interface subsystem, consisting of three main elements:

­ WWW web interface, which is used by Internet users of the system

­ DATEX and RDS / TMC interfaces, used to communicate with other systems;

* GIS module, which is responsible for managing geographic data as well as the presentation of data on demand;
* module for recording and storing video, which is responsible for recording and storing video signals from the video cameras.

The software is designed on the basis of an open architecture using several free software and publicly available products. The system is developed in J2EE technology based on the free ESB (Enterprise Service Bus) software implementation. As a general feature of the system, all modules are integrated in ESB, allowing for flexibility in the future expansion.

Currently, there are 19 electronic signs in Szczecin, 5 of which show travel time between selected sections on the map of the part of the city (mapped to the needs of navigation systems) and inform about traffic. These signs inform drivers about the actual travel time between the signs and target points (Most Długi, Trasa Zamkowa, Rondo Uniwersyteckie and the A6 motorway in the case of exiting the city), and between control points (e.g. between Szosa Stargardzka and intersection of Struga and Granitowa streets). The signs show information in real time, calculated on the basis of data read from the license plates of passing cars on two points of detection. The other information communicated to drivers via VMS shows the traffic flow on given sections and is based on colour codes, representing respectively: green – information about free traffic on the given road section, yellow – slow down of traffic, red – significant traffic problems. Moreover, signs inform about the traffic restrictions, surface condition or weather conditions.



**Fig. 2.** Map of variable message signs in Szczecin. Source: [5].

The major part of the system functionality has been confirmed by analyses conducted by the Szczecin Town Hall. The research was conducted on measurement sections associated with the choice of an alternative route for getting to the centre. On the basis of measurements and calculations comparing the state prior to the implementation of the system (as of October 2011) with the state after its implementation (as of November 2012), the average travel time in analysed sections was reduced despite of the increase in traffic, which led to the increase in traffic flow by 36%.

As part of the analysis carried out during the C-LIEGE project, further assessment was conducted for the possibility of using a mobile application that would allow for even bigger system improvement.

# Mobile application supporting traffic control system in Szczecin

## General assumptions of application

Szczecin Traffic Manager application has been developed to support the needs of drivers traveling around Szczecin, particularly in terms of early display of messages shown on each variable message sign installed within the city as part of the abovedescribed Traffic Management System for Szczecin. Proper operation of application requires constant access to the Internet, both in terms of packet data transmission and Wi-Fi connection. It also requires GPS module. Additionally, the device supporting the application (smartphone or tablet) must be equipped with an accelerometer and magnetometer. Application uses class diagram in UML modelling (Figure 3) in order to illustrate the scope of the types of objects used in applications designed for a user. Its key components include:

* DistanceCalculator - is responsible for checking and calculating a distance from the signs. If the current location is within the sign, class object Sign is captures;
* Sign - is responsible for the storage and handling of sign data collected from

XML;

* ApiClient - is responsible for the connection to the server and includes a method of connecting to the server using headers and SSL;
* SZRApiClient - support class, ApiClient class, calls the ApiClient class with additional parameters such as the server address;
* GoogleMapManager - class responsible for managing map, includes methods for initializing markers and animating map;
* XMLParser - class responsible for parsing XML data, captures a hash table with table images.

Additionally, the application includes two additional classes that should be emphasized:

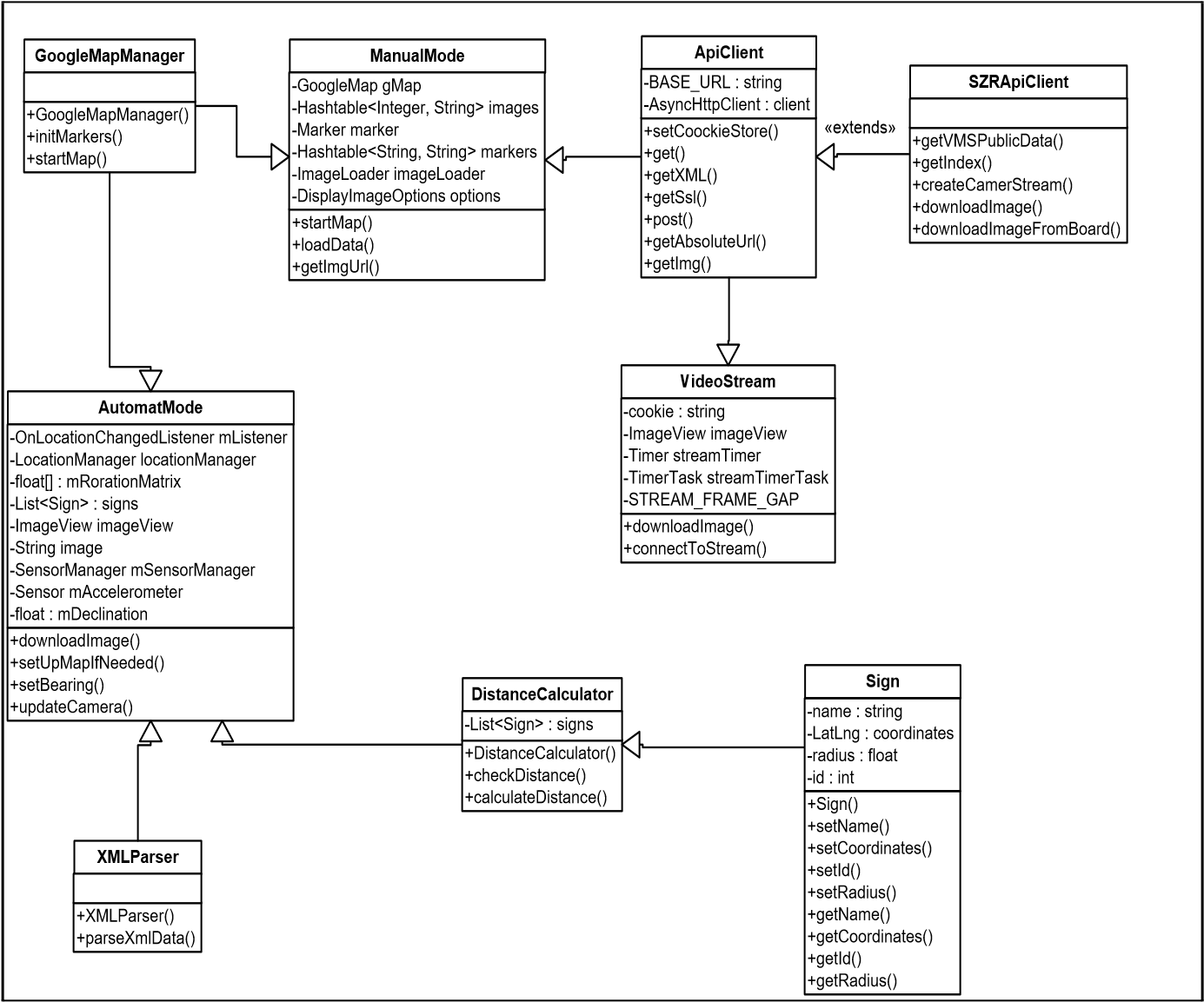
* ManualMode – class responsible for the manual mode of application, which calls the method initializing map and the method connecting to the server, which captures table images. It also has an internal class

CustomInfoWindowAdapter, the task of which is to place image in the cloud called after clicking on the symbol on the map;

* AutomatMode – class responsible for the automatic mode, implementing interfaces such as: LocationListener, LocationSource and SensorEventListener. The first interface is responsible for listening for the current GPS position, using the events:

­ onLocationChanged() – called after the change in location;

­ onProviderDisabled() –called when the GPS is turned off; ­ onProviderEnabled() –called when the GPS is turned on; ­ onStatusChanged() –called at the change in status.



**Fig. 3.** Class diagram for the Szczecin Traffic Manager application. Source: own study.

The application is equipped with a special LocationSource interface which includes a method associated with activation and deactivation of the map. SensorEventListner module listens to changes associated with the motion sensor. Additional important features are provided by the following modules:

* VideoStream, downloading the images from the video cameras in the form of frames, calling cyclic clock, which captures a single frame from the video camera at equal time intervals;
* VideoStreamList, containing a list of video cameras available in the application.

## Application functioning

The application is available on the Android platform. The proper functioning of the application requires constant access to the Internet (for both packet data of the phone and Wi-Fi) and GPS receiver switched on. In addition, the smartphone must be equipped with an accelerometer and magnetometer.

The application can operate in three basic modes (Figure 4):

* manual - giving the opportunity to get familiar with the information provided on individual VMS signs, by indicating a specific sign; in addition, it is possible to hide the display of large (with information on suggested routes) or small (with information about the speed limit) VMS signs;
* automatic - the current GPS position of the vehicle is displayed while driving and the application displays VMS signs, which are closest to the user (the range is defined in the program); additional functionality is speed measurement;
* video-streaming – images displayed in real-time from cameras placed in six locations in Szczecin, which allows for orientation in the current traffic situation.

c)

a)

b)

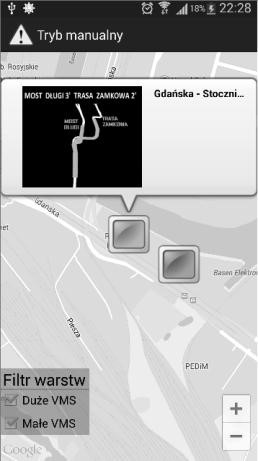


Fig. 4. Szczecin Traffic Manager application modes: (a) manual, (b) automatic, (c) video streaming.

The program uses Google Maps API, access to which is based on a library of Google Play Service. Presentation of data is based on collecting data from the variable message sign system and their presentation on the map using features available in Google Maps. The application runs on multiple threads executed asynchronously. Querying is supported by Asynchronous Http Client library that allows querying outside of the main thread of the program by which all happens concurrently. Information from electronic signs is presented in XML format on the site called UMTS, which are parsed (translated) using the built-in

DocumentBuilderFactory library and displayed in designated areas.

The application has access to cameras via AssyncHttpClienta and properly set thread using cyclic timer, which collects and returns the image of the 6 cameras in Szczecin.

The program has also the ability to track the user and display the characters appropriate to its location. The GPS, which is built in phone, tracks user's location and then displays the closest, electronic board located within a radius of 1,000 m.

Moving the phone rotates camera on the "z" axis plane by using compass function of the phone. Additional feature is speed measurement determined by the changing GPS location.

**Create a platform that displays real time traffic information using html and javascript**

Creating a real-time traffic information platform using HTML and JavaScript involves fetching and displaying data from a traffic API. Here’s a simplified example using the Google Maps JavaScript API:

1. \*\*Set up the HTML structure\*\*:

```html

<!DOCTYPE html>

<html>

<head>

<title>Real-time Traffic Information</title>

<style>

#map {

Height: 500px;

Width: 100%;

}

</style>

</head>

<body>

<div id=”map”></div>

<script src=”script.js”></script>

</body>

</html>

```

1. \*\*Set up JavaScript\*\*:

Create a JavaScript file named `script.js` and include the following code:

```javascript

Function initMap() {

Const map = new google.maps.Map(document.getElementById(‘map’), {

Center: {lat: 37.7749, lng: -122.4194}, // Default to San Francisco

Zoom: 13

});

Const trafficLayer = new google.maps.TrafficLayer();

trafficLayer.setMap(map);

}

```

This code sets up a basic HTML structure with a map container and includes the JavaScript file `script.js`.

1. \*\*Integrate Google Maps API\*\*:

Make sure to include the Google Maps JavaScript API in your HTML file. Add this line inside the `<head>` section:

```html

<script src=<https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap> async defer></script>

```

Remember to replace `YOUR\_API\_KEY` with your actual Google Maps API key.

1. \*\*Testing\*\*:

Open this HTML file in a web browser. You should see a map centered around San Francisco with real-time traffic information.

Please note that this is a basic example and a real-world application would require more features, error handling, and possibly a more advanced API for traffic data. Also, make sure to follow any terms of use or licensing agreements associated with the APIs you are using.

**Designing mobile apps for both iOS and Android platforms to provide real-time traffic updates and route recommendations** :

Designing mobile apps for both iOS and Android platforms to provide real-time traffic updates and route recommendations involves several steps. Here’s a high-level outline of what you’d need to do:

### 1. \*\*Market Research and Planning:\*\*

- \*\*Identify User Needs:\*\* Understand what features are essential for your users. This might include real-time traffic updates, route planning, alternate routes, voice-guided navigation, etc.

- \*\*Competitive Analysis:\*\* Study existing apps like Google Maps, Waze, Apple Maps, etc., to identify strengths and areas for improvement.

- \*\*Define Features and Functionality:\*\* Based on your research, create a list of features and functionalities that your app will offer.

### 2. \*\*UI/UX Design:\*\*

- \*\*Wireframing and Prototyping:\*\* Create wireframes to outline the app’s structure and user flow. Tools like Sketch, Adobe XD, or Figma can be used.

- \*\*Visual Design:\*\* Design the actual user interface, including color schemes, typography, and iconography.

- \*\*User Testing:\*\* Conduct usability testing to ensure that the design is intuitive and user-friendly.

### 3. \*\*Development:\*\*

- \*\*Choose Development Tools and Frameworks:\*\*

- For Android: Java/Kotlin with Android Studio.

- For iOS: Swift/Objective-C with Xcode.

- \*\*Real-Time Traffic Data API Integration:\*\*

- Integrate with a reliable traffic data provider’s API (like Google Maps API, TomTom, HERE, etc.) to get real-time updates.

- \*\*Implement Route Recommendations:\*\*

- Use the data obtained from the traffic API to suggest optimal routes based on current conditions.

### 4. \*\*Backend Development (if required):\*\*

- If your app requires user accounts, saving preferences, or any kind of server interaction, you’ll need a backend. You could use a backend-as-a-service (BaaS) platform or set up your own server.

### 5. \*\*Testing:\*\*

- \*\*Unit and Integration Testing:\*\* Ensure that each component of your app functions correctly.

- \*\*Compatibility Testing:\*\* Test the app on different devices and screen sizes to ensure it works properly.

- \*\*User Acceptance Testing (UAT):\*\* Involve real users to test the app and provide feedback.

### 6. \*\*Deployment:\*\*

- \*\*App Store and Play Store Submission:\*\*

- For iOS, submit the app to the Apple App Store.

- For Android, submit the app to the Google Play Store.

### 7. \*\*Post-Launch:\*\*

- \*\*Monitoring and Maintenance:\*\*

- Continuously monitor app performance, fix bugs, and release updates.

- \*\*User Feedback and Iteration:\*\*

- Listen to user feedback and make improvements accordingly.

### 8. \*\*Legal Considerations:\*\*

- \*\*Terms of Service and Privacy Policy:\*\* Ensure you have proper legal documentation for your app.

- \*\*Compliance:\*\* Adhere to any legal and regulatory requirements for traffic and navigation apps.

Remember, this is a high-level overview and each of these steps can be quite detailed. Additionally, working with real-time traffic data may require careful consideration of data privacy and compliance with local laws. Always ensure that your app complies with any relevant legal requirements.

**Conclusion:**

* In conclusion, effective traffic management is paramount for ensuring smooth and safe movement of vehicles on roadways.
* Utilizing advanced technologies and real-time data has become instrumental in achieving this goal.
* Mobile applications, like the one we discussed, play a crucial role in providing users with up-to-date traffic information and intelligent route recommendations.
* By integrating reliable traffic data APIs and employing user-friendly interfaces, these apps empower commuters to make informed decisions, ultimately reducing congestion and travel times.
* Additionally, continuous user feedback and iterative improvements are essential in maintaining the app’s relevance and usefulness.
* Moreover, traffic management solutions contribute significantly to urban planning, environmental sustainability, and public safety.
* They enable cities to optimize infrastructure and transportation systems, leading to more efficient and livable urban environments.