CS-E4460 WWW Applications Final Report

Matteo Rizzi,* Zhongyuan Jin,† and Ruoyan Meng‡

MULTIUSER REAL-TIME GEOLOCATION SERVICE

(Dated: December 17, 2019)

This document is a description of a basic web application developed mainly exploiting Geolocation technology. It is analyzed in the following six different sections: introduction, related work, results & application, analysis, conclusions, references.

I. INTRODUCTION

Nowadays, Geolocation is becoming a kind of "must technology" in web development. Most of the recent web applications are also called Location-Based Service (LBS) because of the use of Geolocation as a crucial part of the whole application [1]. In fact, in the last few years, there was a proliferation of Global Positioning Systems (GPS), wireless mobile networks and IP location identification techniques with a consequent increase of all types of cross-platform mobile apps.

Although location-based services are popular, most developers are not familiar with their development because in the past it was quite complicated and difficult. However, since the introduction of HTML5 Geolocation, many IT developers became familiar with it and the result is a proliferation of LBS. In order to exploit the Geolocation potential fully, a simple LBS application by using both HTML5 Geolocation API and Web Socket API was developed. It aims to provide a web platform to track in real-time the location of more than one user at the same time. Detailed design and implementation are described in the following chapters.

The rest of this paper is organized as follows. Section 2 gives a generic overview of current web applications which are connected to the project. The application is shown in Section 3 through both screenshots and comments. Section 4 has an analysis about Geolocation in many browsers. Section 5 includes a conclusion and possible future works, while the final section summarizes all the references we used.

II. RELATED WORK

As mentioned in the previous section, nowadays there are a lot of web applications based on Geolocation. Here there is a short roundup of useful example among this ocean of app:

- **Pokemon Go:** players use the app to explore their surroundings on foot, using the smartphone as a map to discover and collect Pokemon characters. Each place in the app is tied to real-world locations that the user has to visit physically, encouraging players to travel in order to collect more characters and visit new neighbourhoods;
- **Uber:** it is a prime example of using LBS technology to provide a more user-friendly experience. The app uses two-way LBS technology so the driver knows where to pick up the user, and the user is able to see a live view of the driver's location on a map. The Uber app also provides two-way communication between the user and driver through the use of Web Socket API. For example, the user can give a description in real-time to the driver (e.g., "I am wearing a red hat"), and the driver can let the user know if there is a delay (e.g., "Traffic is stopped — be there soon").

These are typical examples of how much important is the cooperation between different API like Geolocation and Web Socket to reach a fully-operational real-time experience.

III. RESULTS & APPLICATION

A. Geolocation API

When used, HTML5 Geolocation is a wonderful feature which allows your web browser to inform the web server where you are so it can provide locally relevant content to you [2]. However, there are a large number of factors that this feature is dependent on, from technological, geographical and even to physical which will influence how accurate this feature is in the real world.

The following list aims to sort location sources services from most accurate to least accurate:

- **GPS (Global Positioning System:)**This is available for any device which has GPS capabilities such are smartphones produced in the last 10 years. Accuracy is excellent - typically a smartphone GPS chip can establish your location anywhere on earth within 3 metres. To do this the smartphone needs to receive a signal from at least 4 of the 6 satellites that are visible at any point on the Earth's surface. Less accurate results will occur if the smartphone

* matteo.rizzi@aalto.fi - 808914
† zhongyuan.jin@aalto.fi - 767783
‡ ruoyan.meng@aalto.fi - 766085

can only see 3 satellites because in that scenario the GPS receiver can still get an approximate position by making the assumption that smartphone is at sea level. If it really is at sea level, the position will be reasonably accurate but otherwise, the location will be not so much accurate.

- **WiFi Positioning System:** When a user travel along each road, it works by gathering WiFi signals and then smartphone regularly sends WiFi network and location data to Google and Apple's servers. Accuracy varies but is typically within 5-15 metres. If the GPS signal is too weak to determine the location, the location of nearby networks is used (since it is known) and the location can still be gathered even though GPS isn't working. The beauty of this is that it is public and faster than GPS.
- **Mobile Network Location:** Mobile phone tracking is normally used when GPS and WiFi positioning aren't available. The way this works is like a 2-dimensional version of GPS (or GPS with 3 satellites where the assumption is that you are at sea level). Using cell tower ID triangulation with at least 3 towers, it is possible to determine a phone location to within an area of about 3/4 of a square mile [3].
- **IP Geolocation:** The final method used is IP based Geolocation. Here, location is caught based on the public IP address of the device. This is the most inaccurate positioning method and the typical scenario is a desktop computer with a wired-only connection to the internet. The position is established based on databases of lookup tables which map IP addresses to locations. The accuracy is typically determined according to a database which will only return location data accurate to a City or an area within a City. Therefore, in many cases, IP is hidden behind Internet Service Provider NAT and the accuracy is only to the level of a city

Geolocation API is used in a standard way both to locate the user's position and to get the position of other users [4]. It is supported by the following browser versions:

API	Chrome	Internet Explorer	Firefox	Safari	Opera
Geolocation	5.0+	9.0+	3.5+	5.0+	10.60+

Figure 1. Browser compatibility

Geolocation API is really easy to be implemented in any code. Basically, just by adding the following lines to the code [5], it is possible to make Geolocation works.

```
if (navigator.geolocation ) {
    navigator.geolocation.getCurrentPosition(
        showPosition);
    }
function showPosition(position){
    x.innerHTML = "Latitude: " + position.coords.
        latitude + "Longitude : " + position.coords
        .longitude ;
}


}
```

The Geolocation interface represents an object able to obtain the location of the device. It gives Web content access to the location of the device. This allows a Web site or app to offer customized results based on the user's location. An object with this interface is obtained using the navigator. Geolocation property implemented by the Navigator object.

First of all the application must check if Geolocation is supported. If Yes, it runs the *getCurrentPosition()* method; if Not, a message will be shown to users. Then, if the *getCurrentPosition()* method correctly worked, it returns an object containing Longitude and Latitude coordinates asked by the *showPosition()*.

B. Web Socket API

The rise of HTML5 and the Web Socket protocol achieve the real-time data transmission in web-based systems, so far, they are considered as the best solution to resolve this subject. Web Socket is defined as a technology that enables web pages to use the Web Socket protocol for full-duplex communication with a remote host. Web Socket efficiently provides a socket connection to the internet with the smallest overhead. It produces a huge reduction in network traffic and latency; therefore, it is the perfect technology for developing scalable and real-time web communication system. To use HTML5 Web Socket, a new Web Socket instance should be initialized with a valid URL that signifies the remote endpoint to be connected. A Web Socket connection is established when refreshing an HTTP protocol to Web Socket protocol during the initial handshake between client and server. Web Socket connections work on standard HTTP ports (80 and 443), therefore, it is called "proxy server and firewall-friendly protocol" [6].

Web Socket API is used in this application to carry out the "real-time" tracking thanks to which a user can follow and track the position of other users without open/close communications with the server.

C. React JS

React is a famous open-source front-end JavaScript library developed by Facebook. React is popular among developer communities because of its simplicity but efficient developing process. React offers an easy way to create interactive user interfaces. Basically, it updates (using rendering) the components to the view of each

state and makes the data changes in the application. In ReactJS, every component manages its state and composes them to the user interfaces.

D. Application

In this section, there are some screenshots of the web application. Considering the following figures, the main idea is that once a user belongs to a "room" it is allowed to track in real-time the location of the other users belonging to the same room (e.g. In Figure 5, room number is '1234'). The list of locations on the left is used just for demo purposes because it allows changing the users' location without necessarily moving the devices.

The Web Application can be reached at the following link: https://www-app.herokuapp.com/

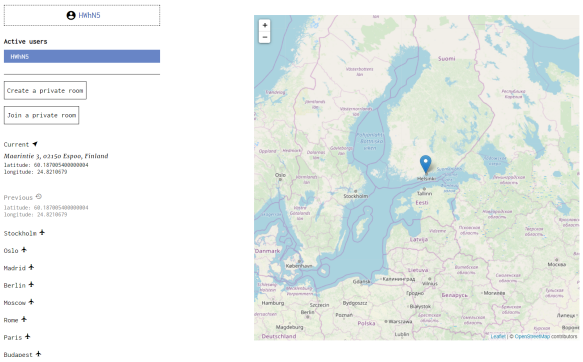

Figure 2. User not in a Room

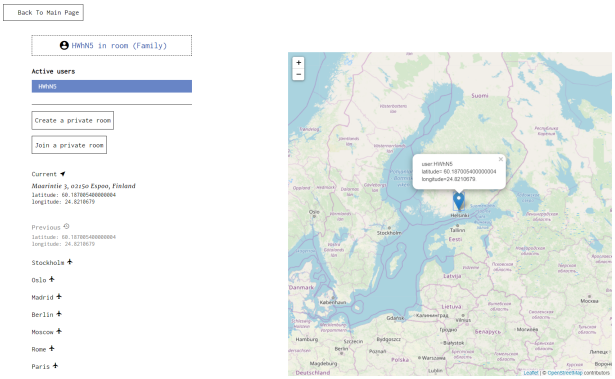

Figure 3. Popup information

A typical scenario for this web-app is a family which needs a "private room" where parents can follow children and vice-versa. Thanks to this functionality, privacy issues can be prevented since users are able to track only the location of other users belonging to the same room. A blue marker is placed to indicate the user location's and, if clicked, it shows the user's name and both its latitude and longitude. Once these coordinates change, of course, the marker will be moved to the new ones proving the real-time feature of the application.

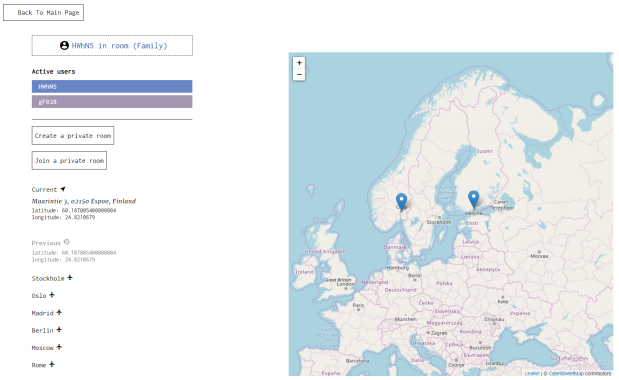

Figure 4. Two users in the same room

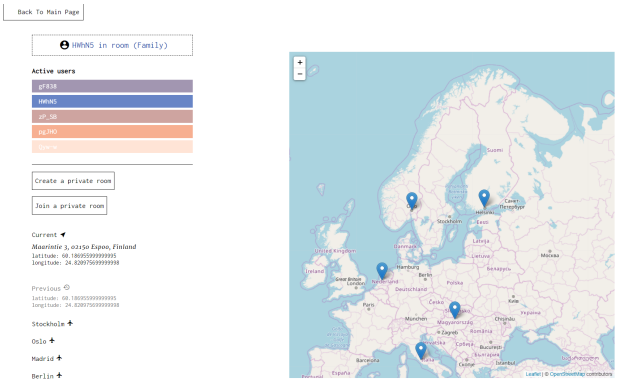

Figure 5. Five users in the same room

IV. ANALYSIS

A. Response Time of Geolocation API

Page loading and interactive response are the key factors affecting user experience [7]. The interactive process where a user performs an operation and displays data on the page account for almost 80% of all application interaction scenarios. Although this is a very common interaction process, the delay of the API response will directly affect whether the function can be used smoothly. Hence, we did some tests on the response time of the Geolocation API when applying different browsers.

From the results, without caching, it takes a longer

Table I. Geolocation API performance in browsers

	Firefox	Chrome	Safari
cached\ms	16	10	service blocked
not cached\ms	500∼4000	500∼4000	service blocked

time for the Geolocation API to send the request to the server to return the data. The time interval will also be affected by network conditions such as network delay, network bandwidth, etc. In this case, the performance in the browser is highly random. In the case of browser caching, the performance of the Geolocation API is significantly more stable and far better than without caching. Caches store server-returned data in temporary storage locations so that they can be accessed faster, which also explains why the Geolocation API can perform better when cached.

V. CONCLUSIONS

In this paper, we considered the HTML5 Geolocation API in all its features. Even though this API is very easy to be implemented in almost all codes, it is currently a crucial topic in modern Web development. In future, Geolocation could be improved more in terms of "device-accuracy" than in terms of API since, as said before, it is just a matter of few lines of codes that contain a great potential.

VI. REFERENCES

As our application providing real-time location information between different user, Kwangseob and Kiwon[3] provided a solution to manage this. Wen-Chen[2] present their idea in Location-Based Services using HTML5 Geolocation and Google Map API, which give us useful reference. Giacomo and his colleagues' work[1] give us a clear idea of the framework of peer-to-peer location-based services. The overview from Thomas and his colleagues[4] provide both historical and future efforts of Geolocation in the browser, they also discussed this from a privacy angle.

[1] G. Brambilla, M. Amoretti, and F. Zanichelli, An open web application framework for peer-to-peer location-based services, Concurrency and Computation: Practice and Experience **30**, e4254 (2018).

[2] W.-C. Hu, N. Kaabouch, H.-J. Yang, and X. Wang, Location based services using html5 geolocation and google maps apis, in *Proceedings of the Midwest Instruction and Computing Symposium (MICS 2013)* (2013).

[3] K. Kim and K. Lee, Real-time processing of spatial attribute information for mobile web based on standard web framework and html5, Spatial Information Research **24**, 93 (2016).

[4] T. Steiner, A. Kostiainen, and M. Kruisselbrink, Geolocation in the browser, in *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19 (ACM, New York, NY, USA, 2019) pp. 913–918.

[5] A. T. Holdener, *HTML5 geolocation* (" O'Reilly Media, Inc.", 2011).

[6] Q. Liu and X. Sun, Research of web real-time communication based on web socket, International Journal of Communications, Network and System Sciences **5**, 797 (2012).

[7] P. Lubbers, B. Albers, F. Salim, and T. Pye, *Pro HTML5 programming* (Springer, 2011).