# RESTFUL SERVICES FOR ORIENTATION AND ACCESSIBILITY IN A UNIVERSITY CAMPUS

Ioan DAMIAN[1], Anca Daniela IONITA[2]

*Finding the right navigation route in a university campus represents a regular need for various stakeholders who participate to the academic life. For the outdoor space, this can be obtained with the general-purpose mapping services. However, daily activities involve frequently passing from outdoor to indoor spaces, as well as being aware of educational-specific information and accessibility constraints for students or teachers with special needs. This article presents a set of RESTful services conceived and realized to add such supplementary functionality to the current route planning based on mapping. It also provides a comparison between the route directions provided by such services and those obtained from other web mapping services delivered for free.*

**Keywords**: smart campus, RESTful web services, accessibility for special needs

## 1. Introduction

Attending university is a highly advanced stage of education, hence, in a new student's eyes, its campus is very large, new, and different from what he or she had been experiencing so far. It is different than the educational institutions attended before; it is not the school in the neighborhood, it does not have a well recognizable school yard, or the familiar school halls. The main reason of creating a university campus was to offer students access to what was represented by a traditional university, once an exclusivist service, where only the wealthy families could send their children to. Nowadays, campus universities attract students from everywhere and from any background [1]. The feeling of being somewhere new for the first time might be overwhelming to an incoming student, who needs some sort of confidence on knowing how to get from one part of the campus to another, or how to go to a specific class, held in one of the university buildings. The students' need to orientate themselves is very important; in the end, it gives them a sense of belonging, and the opportunity to plan their activities in order to make most out of them. Furthermore, if one considers navigating through campus, this may be challenging because many universities were built when the "smart

[1] PhD Student, Eng., Automatic Control and Computers Faculty, University POLITEHNICA of Bucharest, Romania, e-mail: ioan.damian@stud.acs.upb.ro
[2] Professor, Automatic Control and Computers Faculty, University POLITEHNICA of Bucharest, Romania, e-mail: anca.ionita@upb.ro

campus" concept was inexistent. Often the infrastructure (i.e. the architecture of the buildings, the roads, the paths students can take to get to other buildings and attend courses) was not built to scale, nor to meet future needs without destroying the former ones. As an addition, a student with locomotor disabilities would consider very hard to find the right route that allows him or her to get from one side of a campus to another.

A university campus generally corresponds to a large area of land, where both academic and social activities are conducted. It is characterized by complex topologies, making the experience to navigate through it difficult for visitors, students or full-time employees. The solution is to provide individuals with maps and tools to guide them through the campus and support them to make decisions [2]. Due to the specific infrastructure that each university implemented upon its construction, the requirements for orientation and navigation present themselves as being custom to that specific university. At an abstract level, navigation means knowing how to go from one point to another, but on complex topologies other issues arise. The navigation inside the buildings is different from the one outside the buildings, the difference in level (stairs/floors), the time or the difficulty constrains must be taken into consideration to provide end users with a custom mobility service. For developing such solutions, research was conducted on a variety of topics, such as: landscape of the campus [2], i-campus [3], smart city [4][5], event-based navigation [6], iBeacon navigation [7], mobile social network in a smart campus [8]. Our approach to this custom need was to design services that take into account the model of the accessibility facilities, in addition to the topological representation of the university campus.

The remainder of the paper is organized as follows: section 2 analyzes related work. Section 3 explains the functionality and the design for the proposed solution for orientation and accessibility in a university campus. Section 4 presents the services based on REST (Representational State Transfer) that were developed and integrated. Then, section 4 gives an example for a specific university, including indoor and outdoor navigation functionality and criteria for the access of people with locomotory disabilities. The route directions obtained in this way are compared with those available for the same locations from a consecrated free mapping service.

## 2. Related Work

The problem of navigation in the context of organizing events inside a campus is addressed in [6]. Events are common inside a campus life, so organizing them consists of registration, providing their description, inviting people and guiding the audience to the specific place where the event is held. Bendre et al. look for the most dynamic and flexible way of guiding people around places that are not commonly accessed. Their solution is a system run on mobile devices equipped with GPS tracking, mapping applications and accelerometers, to provide a smooth and accurate experience.

Another navigation system is described in [7], based on Bluetooth Low Energy (BLE) technology. BLE has already been successfully used in advertisement and retail sectors, by implementing beacons in stores to send clients advertisements about products or services. The way it can be implemented in a campus is by designing an infrastructure where an administrator would set up these beacons to guide teachers and students through the campus. The principle is that the beacon sends information to other Bluetooth clients and they are guided back to the beacon. Several advantages provided by this solution are:

- Bluetooth is supported by many different platforms (iOS and Android);
- BLE uses very little energy, so navigation is not felt like a drawback;
- BLE is already implemented and documented;
- BLE works efficiently indoor.

Disadvantages in choosing this solution are:

- The position is not always precise;
- The strength of the signal is not precise;
- Interferences may occur, due to the presence of walls or people.

Another research regarding friendly directions for accessing a point of interest on the map is described in [8]. Yim et al. present an Android app that uses the mobile device to determine the location or find a building, and it plays a video with information about that building. The client side only provides the location through GPS and an image of the building; further processing is made on the server side, to store the client position, recognize the building using camera sensors and electronic maps computed in AutoCad, and stream the video back to the client. This approach was successfully implemented in museums, with advantages like precise location and video on demand. Some disadvantages are given below:

- Guiding is not deterministic;
- In universities this application would lack clear instructions for the user about buildings' rooms or specific points of interest.

Yu et al. integrate the social networking aspect to make the experience in a campus more pleasant [9]; students can connect, chat or study together. The hardware layer is made of sensors and mobile device tools. The middle layer consists of data storage, communication management and location management - both outdoor - achievable through GPS - and indoor - through Wi-fi. The advantages in this implementation are:

- The location is precise, using both GPS and Wi-fi;

- Social networking is approached;
- It provides data about classrooms, such as empty classrooms to study, or friends' position.

One can also mention some disadvantages:

- A new person will not be able to find a new classroom;
- The guiding system is obtained with the help of the social component, even if the positioning is precise indoors and outdoors;
- It can be better used by students who have been for some time in the campus and know their way around.

## 3. Proposed Solution

The solution is conceived for the use of professors, students and visitors of a university campus. The main functionality that was targeted consists in:

- A visual map of the campus, with points of interest like library, canteen, secretariat, which can be managed by authorized employees of the university with the role of administrators;
- The capability to filter points of interest shown on visual map, based on criteria detailed in Section 4;
- The generation and visualization of a recommended route between 2 points of interest, providing: a trajectory outlined on the visual map to indicate the general direction (as shown in Fig.2-a) and a text summary of the given route (as shown in Fig.3-a).
- The provision of accessibility information for the entire campus map, useful for persons with special locomotory needs caused by a permanent handicap or by temporary health issues.

The map and the points of interest may be configured by administrators who belong to the campus personnel responsible with the evolution of the system. They have the right to add to the system buildings or rooms based on the model detailed in Section 4, but also to update the existing ones. Users like professors or students can then query the system and have access to the information uploaded by administrators in regard with the university campus.

### *3.1 Service-Oriented architecture*

The system architecture, depicted in Fig. 1, contains 5 nodes: USER Web Page, ADMIN Web Page, Back-end server, Data Storage, Map Cloud. The Diagram also depicts 5 services: Map Service, Storage Service, Services for Managing Points of interest, Orientation Services and Accessibility Services.

ADMIN Web Page consumes Services for Managing Points of interest, to add new information about the university campus, like new handicap access ramp. USER Web Page consumes Map Services to obtain a visual map of the campus; it also consumes Orientation and Accessibility Services to get information about the points of interest. The back-end sever node provides services consumed by the USER Web Page and the ADMIN Web Page. The Back-end Server uses the Data Storage node to save data. The Map Cloud provides Map services to the USER Web Page node.

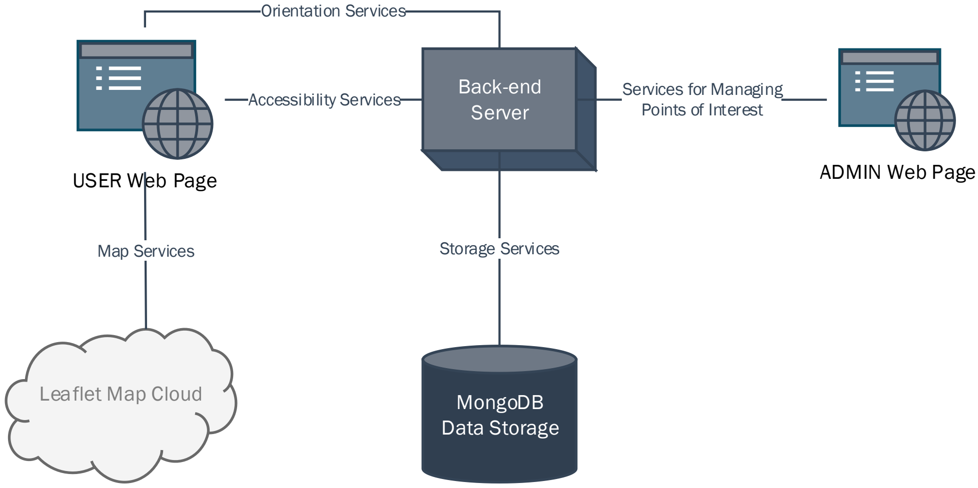



Fig. 1 System Architecture

### *3.2 Implementation details*

To implement the solution presented above, we developed the Back-end server to provide **RESTfull** services with HTTP technologies [11]: Services for Managing Points of interest, Orientation Services and Accessibility Services. Internally, they are split into a set of services of smaller granularity (described in Section 4.2), to respect data grouping and modeling.

The **RESTfull** services are consumed by front-ends implemented by the USER Web Page and the ADMIN Web Page. The Front-ends are developed with HTML, JavaScript and CSS, to ease the adoption and portability for the end-users.

The Back-end Server creates and retrieves data required by users and administrators and it stores them in a central Data Storage, assuring consistent offshore data. For our implementation of Data Storage, we used MongoDB - a collection-based, NoSQL database that is useful if the administrators want to extend the granularity of the saved data.

The front-ends plot data provided through the Back-end server services on the visual map of the university campus. The campus map is obtained externally from the Leaflet cloud provider. Leaflet provides extensibility, easiness of development and smaller size than most popular mapping solutions, like Google Maps. It offers JavaScript APIs for map and building structure, enabling our application to build data in the front-ends.

Therefore, the application integrates the third-party map service with in-house cloud developed services. The requests are made via an HTTP [11] client that acts as a middleware between the client device and the cloud services.

## 4. Orientation and Accessibility Services

This section describes the solution proposed by us and provides details about chosen points of interest and the design of the REST API for providing orientation and accessibility services. The services presented in Fig. 1, Points of interest Management Services, Orientation Services and Accessibility Services are split internally in other services described in Section 4.2. These outer-services re-use the same inner-services, but with different verbs (for example Points of interest Management Services use a write operation on the inner-services, whereas Orientation and Accessibility Services use a read operation on the inner-services)

### *4.1. Accessibility Information*

We believe that the most value to obtain from our system is a route between two points offered to users, this being the reason of macro-modeling stored data as a graph. Apart from the route calculated with typical algorithms based on graphs, we considered a scenario for persons with temporary or permanent locomotory issues, who need information about obstacles of the terrain or campus architecture to plan their navigation. Our solution provides *Vertex* and *Graph Entry* services used for setting the geospatial position of fixed resources; the output, shipped from backend to frontend, contains the information needed to place markers on the map. Furthermore, the *Path* service is used for orientation, based on an algorithm that computes the easiest path. *Vertex* and *Graph Entry* services determine if the source and the destination of navigation are spatially close. Having the visual representation of the path and the lexical description of it, the user can choose to use just the direction information in case the two points are close to each other (on the same level, or on different levels, but one above the other)

In our approach one has to define multiple models [10] for the university campus, not only for mapping, but also to consider the accessibility and other issues. Thus, the service also achieves accessibility data, shipped to the user at the same time with the orientation data, i.e. the user can have both the path and the description of how difficult it is.

The actors specified for this application, who need to go from point A to point B within the campus, may be students, teachers, other employees, parents etc. Apart from computing the existent paths from one point to another, the designed services also consider special needs of such actors, like: walking on foot, riding a bicycle, traveling by car, encompassing locomotor difficulties, carrying heavy objects, having problems with the elevation differences.

The accessibility data provided by the application contain:

- if an entry point has a disability ramp
- how many stairs the entry point has
- if the route is only indoors or outdoors
- if a route segment is accessible by stairs, by elevator or none of those two
- if the doorway is large or small
- if the path has an inclination

### *4.2. API Requirements for the RESTful Web Services*

The REST API provides nine routes managed by standard HTTP [11] verbs (GET, PUT, POST, DELETE) to obtain the campus data and the most efficient route for the user. The general flow is that an HTTP client will send request to the server, these requests expose the Webservices, integrating the functionality of an HTTP client, a WEB server and a Data Base server. Table 1 presents the data consumed by the REST API internal objects. Orientation and accessibility are obtained with the following services:

- *Path Finder* Service, using /graphb/path route
- *Edge Management* Service, using /graphb/edge and /graphb/edges routes
- *Vertex Management* Service, using /graphb/vertex and /graphb/vertexes routes
- *Graph Management* Service, using /graphb/graph and /graphb/graphs routes
- *Entry Management* Service, using /graphb/entry and /graphb/entries routes

*Table 1*

**Data consumed by the REST API**

| REST API Internal objects | Data specifications |
|---|---|
| Graph management | • *id* – the general name of the graph<br>• *type* – the granular category of the graph that holds the current graph<br>• *container* – the id that holds the current graph<br>• *level* – necessary to describe if the graph is included in another graph |
| Graph entrance management | • *id* – graph's entrance<br>• *container* – the graph the entrance belongs to<br>• *difficulty* – the difficulty in accessing the entrance<br>• *door size* – "large" or "small" |

| | |
|---|---|
| | • *disability ramp* – if the entrance has a disability ramp or not<br>• *lat* and *long* – latitude and longitude of the entrance to place it on the map |
| Vertex management | • *id* – every vertex has a unique id.<br>• *floor* – every vertex is located on a building floor.<br>• *type* - every vertex has a type.<br>• *parent* – the graphs id that contains the vertex<br>• *lat* and *long* - latitude and longitude of the vertex. |
| Edge management | • *id* – collision proof id formed from the details of the edge.<br>• *source* – possible values: vertex id or entrance id. It models one end of an edge.<br>• *target* – possible values: vertex id or entrance id. It models one end of an edge.<br>• *outdoor* – if the edge involves outdoor or indoors navigation.<br>• *elevation* – the difference in level between ends of the edge. Applies for both indoor and outdoor edges.<br>• *elevator* – if the edge contains navigation with the elevator.<br>• *stairs* – if the edge contains navigation on stairs.<br>• *general* - if True, the edge connects the vertex with an entrance. If False, the edge connects one vertex with another. |

## 5. Comparative Analysis Regarding Accessibility

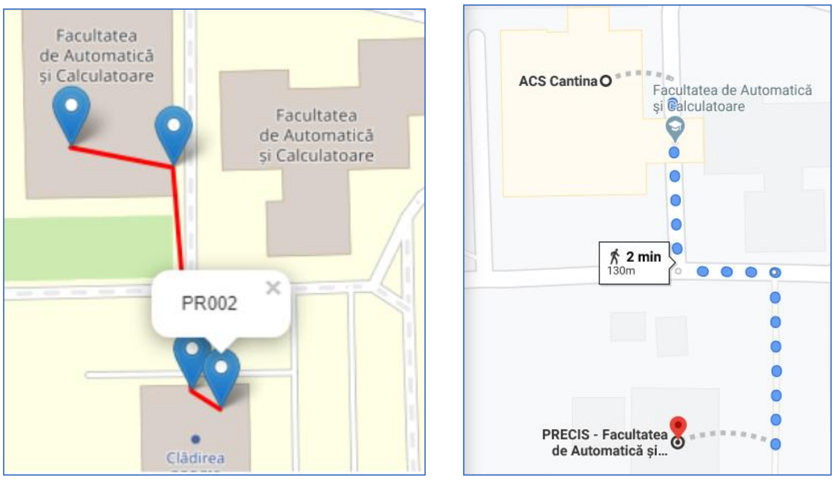
Based on the services presented in the Section 3, we configured the application to provide orientation and accessibility for a part of our university campus, with details in regard with the Faculty of Automatic Control and Computer Science. This section analyzes an output obtained for this configuration: the visual depiction of the route to the destination, and detailed directions to help the user while taking the computed route.

### *5.1. Results for the Analyzed Example*

For the purpose of this analysis, let us consider the starting point: *EC001* - a room at the ground floor of the *EC* building, and the destination: *PR002* - a room at the ground floor in *Precis* building. After the user introduces these two details in the web form, the computations will provide 2 pieces of information: the graphical representation of the route in Fig. 2 a) and the detailed directions in Fig. 3 a). The application recognizes that the two points of interest (*EC001* and *PR002*) are not in the same building, therefore it looks for an easy path for the user to take. Using the underlaying graph-based modeling, if the two subgraphs represented for *EC* and *Precis* buildings respectively are connected, the algorithm implemented within the *Path* Service determines the easiest path that connects these 2 points of interest.

Thus, our application provides the visual orientation information through the orientation services. The blue markers in Fig. 2 a) are intermediate points for the route, i.e. in this scenario: starting point, destination, EC building entrance point, PR building entrance point.

Therefore, the user can click on them and recognize the current location; they also act as checkpoints. The red line connecting these blue markers help the user with a general direction where he or she must go; the blue markers and the connecting red line help for the navigation.



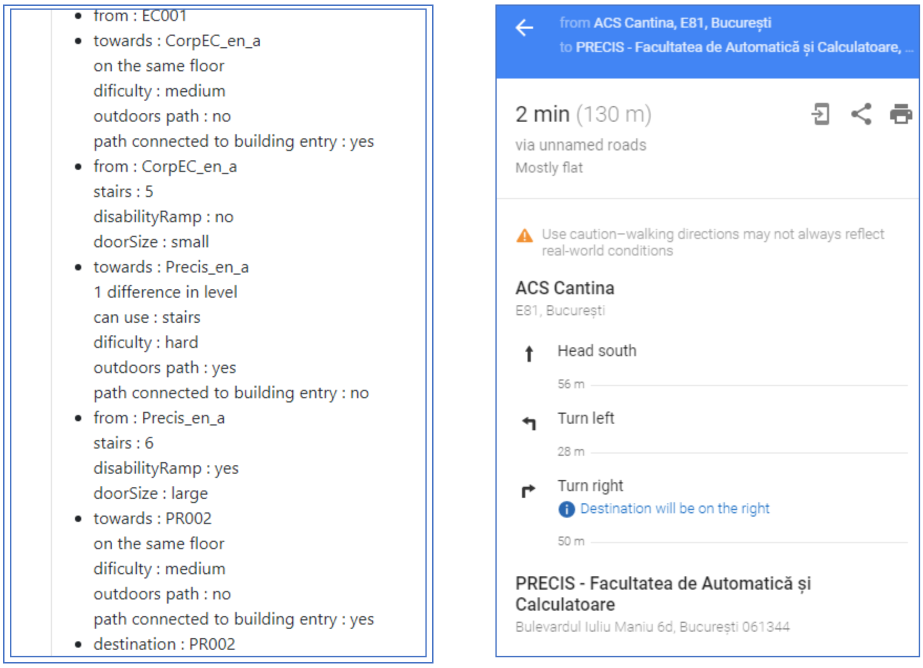

a) b)

Fig. 2. The visual directions for inter-building navigation towards the destination: a) in our application; b) in Google Maps (from Precis to ACS cafeteria)

More details about the accessibility path to the destination are provided through the accessibility service. The details from Fig. 3 a) are provided to instruct the user on how to access the destination, as described bellow:

- the directions with the syntax *from* and *towards* for every segment of the route (i.e. *from: EC001*, *towards: CorpEC_en_A*);
- whether a segment of the route is indoor;
- whether a segment is connected to a building entry;
- whether a segment of the route is situated on the same level;
- if a segment of the route is not on the same level, then options are provided: taking the stairs or the elevator;
- whether an intermediate point has a disability ramp or not;
- how many stairs an intermediate point has;
- how heavy is the door of an intermediate point;

- what is the destination - marked with *destination*, e.g. *destination: PR002*.




a) b)

Fig. 3. Details presented as text a) in our application; b) in Google Maps

All the information provided by the application are meant to help the user to be aware of the route and to plan it in advance in case of an issue.

### *5.1. Comparison with Google Maps*

We hereby compare our results with the navigation directions obtained from a consecrated web mapping service like Google Maps. For this purpose, let us consider the route from *Precis* building to *ACS cafeteria* in Fig. 2 b) and the correspondent navigation details in Fig. 3 b).

***Criterion 1*** of our comparison is the ease to scale the granularity of searching. Google Maps is a reliable tool for getting general directions over long distances, thus it does not contain all the details for the local navigation. This can be observed from the possibilities to choose the precise starting point or destination. In our application, a node can be easily added and maintained by the local community (teachers, students, staff of the campus), whereas Google Maps contains the names for a reduced set of locations; to map an entire campus and other public spaces with all the points of interest would be a costly solution at a large scale. This is the reason why Google Maps cannot find the rooms inside the buildings, i.e. *PR002* and *EC001*; it only offers access to locations like *Precis* and *ACS cafeteria*, as shown in Fig. 2 b).

***Criterion 2*** for our comparison stands in the details of the navigation directions. Fig. 3 b) does not take into consideration elements for persons with locomotory difficulties, it only offers general directions like *go South*, *turn left* /*turn right*; it does not contain intermediate points that can be recognized on a short distance, nor sources of potential dangers when accessing / navigating inside the buildings, as depicted in Fig. 3 a).

***Criterion 3*** is whether the directions conform to reality. Fig. 2 b) presents a last segment of the navigation route (where one is instructed to turn left), leading to an entrance that has been closed for more many years. This drawback is not present in a community-maintained application like the one presented in this paper.

***Criterion 4*** of the comparison is the specific information for users with special locomotory needs. Fig. 2 a) presents a route that takes this into consideration and provides a solution that uses a wheelchair ramp, whereas Fig. 2 b) gives indications towards an exit point from the *PRECIS* building that is only accessed by stairs.

## 6. Conclusions

To conclude, the paper presented the problem of orientation and accessibility in a university campus and proposed a solution that offers services to provide the user with the needed information, i.e. topological, orientation and accessibility information. This solution involves storage and representation of geographical data, and a software architecture that provides orientation and accessibility services. The application can function for navigation from one classroom to another and from one building to another, for students, visitors and all kinds of university employees. In computing the most cost-efficient route, allowing the user to plan the journey between two locations, our approach also considers the fact that a user can have locomotory issues. To prove the efficacy of this solution, the paper analyzed a specific example, configured for the topological environment of a faculty deployed on two buildings, within our university campus. The comparison with results obtained from Google Maps showed that this solution can provide customized and more detailed outputs according to several criteria: granularity of searching, navigation directions, conformity to reality and information for users with special locomotory needs.

The system may be extended with services for indoor and outdoor localization to add dynamic navigation, in addition to the directions based of fixed points of interest that are currently given for users with locomotory challenges.

## R E F E R E N C E S

[1]. *S.N. Ciolofan, M. Mocanu, A.D. Ionita*, Cyberinfrastructure architecture to support decision taking in natural resources management, 19th International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, pp. 617 - 623, 29-31 May 2013.
[2]. *R. Drober*, Campus Landscape – Functions, Forms, Features, John Wiley & Sons, 2000, pp. 10-60
[3]. *L. Kwok*, "A vision for the development of i-campus," Smart Learning Environments, vol. 2, pp. 1-12, 2015
[4]. *M. Hamblen*, "Just what is a smart City?," 2015. [Online]. Available: https://www.computerworld.com/article/ 2986403/ just-what-is-a-smart-city.html. [Accessed 17 December 2017].
[5]. *T. Nam et al*, "Conceptualizing smart city with dimensions of technology, people, and institutions" in 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times, 2011.
[6]. *S. Bendre et al*, "Event Based Campus Navigation System" International Journal of Computer Science and Information Technologies, **vol. 7**, no. 1, pp. 462-464, 2016.
[7]. *D. Merode et al*, "Smart Campus based on iBeacon Technology" in International Symposium on Ambient Inteligence end Embedded System, 2015.
[8]. *J. Yim et al*, "Design and Implementation of a Smart Campus guide Android App", in Appl Math Inf Sci, **vol. 8**, no. 1, pp.47-53, 2014.
[9]. *Z. Yu et al*, "Towards a Smart Campus with Mobile Social Networking" in 2011 Internet of Things and 4$^{th}$ International Conference on Cyber, Physical and Social Computing, IEEE, 2011, pp. 162-169.
[10]. A.*D. Ionita, M. Mocanu*, Multiple Modeling Paradigms Applied for Accidental Pollution Management, Environmental Engineering and Management Journal, **Vol.14**, No. 9, pp. 2051-2060, 2015.
[11]. *D. Gourley, Brian Totty*, HTTP: The Definitive Guide, O'Reily & Associates, chapters 1-5, 2002.