Even Smarter Data: Using Crowdsourcing to Improve Accessibility in Real-Time

A Smart Data Framework to Provide Live Updates for Public Transport Accessibility

Paloma Cáceres, Carlos E. Cuesta, Almudena Sierra-Alonso, Belén Vela, José María Cavero, Miguel A. Garrido
Escuela Técnica Superior de Ingeniería Informática
Universidad Rey Juan Carlos, URJC
Móstoles (Madrid), Spain
e-mail: {paloma.caceres,carlos.cuesta,almudena.sierra,belen.vela,josemaria.cavero,miguel.garrido}@urjc.es

***Abstract*—Smart cities initiatives introduce the challenge of a more inclusive society that provides equal opportunities. Nevertheless, there are not many systems or apps designed to generate routes in the public transport for people with special needs. For instance, Google Maps is often used to calculate routes and find points of interest. However, it does not provide detailed information regarding specific facilities such as accessibility in transit. Even worse, it is not easy to find, to download or even to process specific and detailed accessibility information, in order to develop services for users. In this paper, we propose a smart data framework able to manage accessibility data in public transport and provide such user-level services. Moreover, this framework is also able to deal with real-time information by gathering notifications about incidents in the transport network. These data are obtained in a crowdsourcing process, and provides live updates for information, so that they can be used to recalculate accessible routes. In order to illustrate the process, we describe a case study with which we validate the framework, consisting of a smart app to generate accessible subway routes and to notify accessibility incidents.**

Keywords - Public transport; smart city; accessibility; incident management; crowdsourcing; linked open datasets.

I. INTRODUCTION

Smart cities appear as the main approach to tackle one of the greatest challenges in the first half of the century - urban mobility. This is a complex problem with many facets and nuances, but our strategy is always the same: to get better, more detailed data about many issues, and to use it in a intelligent way, to provide better focused answers to these problems.

The source of any problems in urban mobility is the flow of people and vehicles trying to use the same spaces at the same time. One improvement is to stress the importance of using the public transport - however, this choice would be much more efficient if people was able to avoid colliding with other people when traversing the cities. They could use their own smart devices to do that, by finding specific routes within the urban space.

A software system which provides maps and calculates routes in public transport networks is not a novelty anymore - however, most of the current routing systems still do not consider accessibility. Even the most popular among them, Google Maps, does not provide accessible routes. Our previous research [1]-[3] intends to use semantic information to fill this niche.

Building upon this infrastructure, our platform is able to evolve beyond its original constraints, and then to consider higher accessibility levels. Our initial architecture had to deal with mostly non-volatile data, i.e., data with a small rate change. When our system had to compute a route, it usually uses very stable data (structure of the network, predefined timetables), and even their alterations have usually a certain durability (planned works, changes in the infrastructure or the rolling stock). However, other sources of change are much less predictable: collapses, floods, accidents, even the traffic flow. These are usually easy to perceive once they happen - but many citizens are unaware of those *incidents* until they bump into them.

Thus, our system must include incident management to be complete. This would allow our smart data to include live information and to provide an even more intelligent response in real-time, which takes the current situation into account. This is particularly important when considering accessibility and the relevant groups of interest. This issue effectively offers an advantage over other approaches, such as those discussed in the related works section.

This approach uses a *crowdsourcing* strategy: users themselves are who provide live information about incidents as they find them. This information is sent through mobile devices, and stored in our routing platform, probably from many coincidental sources. These new data are collapsed and integrated with pre-existing accessibility information - so semantic annotations are also used for their processing. In summary, their combination can be conceived as *real-time accessibility data*.

This integration presents several challenges: their specific meanings and vocabulary; the incident lifecycle management; or the handling of simultaneous sources, and even their scale. Our evolved architecture must deal with all this, and this paper presents our solution.

The paper is structured as follows: Section 2 presents some related works. Section 3 describes our CoMobility and Access@City projects, which are the context of this work. Section 4 presents our smart data framework, describing our proposed client-server, scaling architecture, a semantic vocabulary to annotate events and the workflows between client and server sides. Section 5 describes a case study. Our conclusions and future works are presented in Section 6.

 ISBN: 978-1-61208-692-7

II. Related Works

There already are several software applications which inform about specific aspects of the public transport domain. In some cases, they include accessible wayfinding information or/and accessibility features, or elements for people with special needs or disabilities. Other software provides data about public transport using a crowdsourcing approach. In this section, we discuss some of the more representative of them.

Among those related to accessibility, Landmark Ontology for Hiking [4] is focused on elderly people and helps them to walk less by using wheelchairs. It formally represents landmarks for hiking. Wheelmate [5] and/or Wheel Map [6] provide some information about accessible places for people riding a wheelchair. Access Map [7] provides accessible routes to people with mobility needs. All of these just consider mobility-related disabilities. On the other hand, "Ciudades Patrimonio de la Humanidad" [8] has a web application which provides accessible routes to people with special mobility needs, including blind or hearing-impaired people. However, none of these applications are customizable.

Among those using a crowdsourcing approach, some applications use crowdsourcing to improve the experience of using the public transport and to provide real-time information about the public transport status. For example, Tiramisu Transit [9] provides data, such as how full the bus is, or whether there is some wheelchair space left. Moovit [11] is able to plan routes and indicate when to get off, or the status of the service. Swiftly [12] works with transport agencies, rather than the general public. It provides more accurate vehicle arrival data for these transport agencies, so that they can give better information to their users. The OneBusAway [10] project consists of a set of tools to improve the user experience in the public transport, by achieving on-time performance of buses and other transit systems, decreasing wait times, increasing feelings of safety or even increasing transit trips per week. OneBusAway provides several feedback mechanisms that allow users to make comments about these tools.

Other initiatives to use crowdsourcing are the BUSUP [13] and CIVITAS [14] projects. The first one allows users to book crowdsourced buses on demand and the second one intends to achieve a cleaner, better transport in Europe. A CIVITAS subproject is dedicated to mobility strategies for vulnerable groups.

OpenTripPlanner (OTP) is another project to provide services for passenger information and transport network analysis. It computes routes combining transit, pedestrian, bicycle and car segments traversing networks built from OpenStreetMap [15] and GTFS [16] data. OTP also takes (transport) accessibility into account.

As we see, the number of systems intending to improve the user experience in the public transport are increasing. However, most of them (with some exceptions, such as Tiramisu Transit or CIVITAS) do not take the information about accessibility elements into account; but these elements are necessary for users with special needs, to also improve their own experience in the public transport. Even the mentioned systems [9][14], though providing accessibility features, do not allow the users to inform about the actual state of these elements: whether a lift is operative, if there are works that prevent or hinder the access of (for instance) blind people, etc.

To the best of our knowledge, currently there are no software applications which analyze the status of the public transport network to estimate the availability of accessibility features, while also using crowdsourcing to update their data.

III. The Context: CoMobility and Access@City Projects

This work is developed in the context of two research projects. The first one, called CoMobility [18], defines a multimodal architecture based on linked open data for sustainable mobility. Its main goals are to improve citizens' mobility and to optimize their trips by combining public transport and car sharing. The second one, called Access@City [19], is a coordinated project that defines a technological framework in which to process, manage and use open data concerning public transport with the goal of promoting its accessibility. One of its subprojects is Multiply@City [20]. Figure 1 provides a general depiction of this latter project. Multiply@City focus on processing and harmonizing public transport accessibility data in a semantic manner by means an ontology, considering that data are provided by different sources and have different formats (as represented on the top of Figure 1). The accessibility data are obtained from open data by means of Web scraping and here, they can also be updated via crowdsourcing techniques (from smartphones, as represented on the bottom of Figure 1). The left blue square represents the engine for extracting and processing the data from different sources, both from public transport companies and from crowdsourcing.

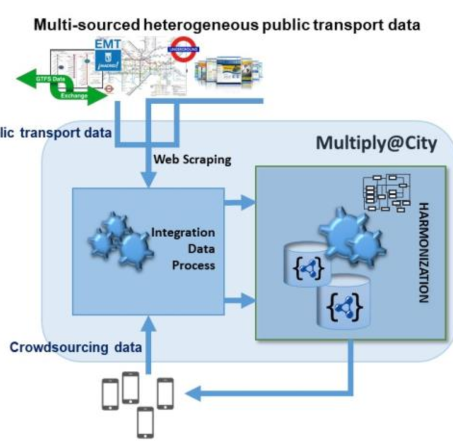


Figure 1 Multiply@City project architecture.

The right blue square represents the engine to semantically annotated previous data, following our MAnto vocabulary (a vocabulary to identify the different infrastructure elements of public transport like lines or stops, and their corresponding accessibility features like lifts, escalators, etc.). Next, semantic data are stored into a semantic repository: one collection of data is for the infrastructure elements of the public transport and the other is for the incidents about the accessibility features of the infrastructure.

The Regional Consortium for Public Transport in Madrid (CRTM) [21], Madrid public bus company (EMT Madrid) [22], and the Spanish Society for Blind People (ONCE) [23] have expressed their interest in the results of our CoMobility and Access@City projects.

IV. PROPOSED SMART DATA FRAMEWORK

The intent of our proposal is to improve accessibility information about public transport in order to support new social accessibility services, such as calculating public transport routes which are accessible for all. These routes are based on stable data about the infrastructure of the transport network (i.e., stations, lines and stops) and on continuously updated data about accessibility features (i.e., a lift or an escalator does not work). To consider these last features, we had to develop a solution which also includes information about the current state of the public transport network, using a crowdsourcing approach.

This proposal applies a client-server architectural style, with features akin to the microservices approach. On the client side, our software is able to update the accessibility information and to calculate a route based on the currently available accessibility features, taking into account the user's accessibility needs. On the server side, we have developed and deployed a datastore which holds the relevant information about the transport network and its accessibility features, listens and processes different notifications from our client software and returns any data which is requested by the same client.

A. The Proposed Architecture

As we mentioned above, our framework implements a client-server architecture, with some additional features.

In the client side, we have developed a smart device software (still a prototype), called MMA4A, which offers two different alternatives to users: (i) to request an accessible route considering their specific needs; (ii) to collect events or incidents about unavailable accessibility features. To compute the route (i), it is previously necessary to obtain the infrastructure data about the public transport network, and to request the current state of this infrastructure – provided by the server. To communicate the events collected by the users (ii), it is also necessary to send these incidents to the server, in order to update the current accessibility features in the network. Therefore, it is required to develop a specific server architecture that holds the infrastructure data and its accessibility status, updated with a crowdsourcing approach. This sequence of activities and their data exchanges are shown in Figure 2.

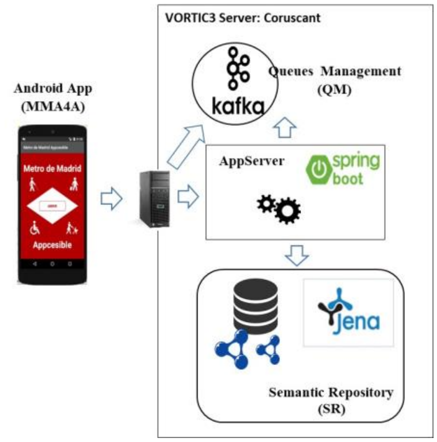


Figure 2 Proposed architecture (smart client-semantic server).

Concerning (i), the server must send the infrastructure data of the transport network, such as stations, lines and stops; and the last updated data about their accessibility features. Our implementation uses Spring Boot [24] as an applications server (AppServer) and a Jena Semantic Repository (SR) [25]. In previous works, [1][2], we developed a semantic data set about the infrastructure, which is now stored in the Jena SR.

Regarding (ii), the server should simultaneously listen to the (potentially many) notifications of incidents or events about the accessibility features that, at this moment, are not available for the public transport network. For this reason, we have implemented an Apache Kakfa server [26] as a Queuing Manager (QM). The QM gathers the different events notified from the crowd of sources, i.e., the users through their smart devices. This information, once processed, must be stored in the semantic repository, so the AppServer also manages and controls the communication between the QM and the SR.

To sum up, we have implemented the following elements in the server (Coruscant, a HP multicore high-performance server, with a direct high-speed Internet connection): a Spring Boot as the application server (AppServer), an Apache Kakfa server as a QM and a Jena SR. Then, the Jena repository maintains two separate data collections: one of them provides the (mostly static) data about station, lines and stops, called the *infrastructure collection* (i) and the other one provides the (dynamic) data of the network state about the accessibility features, called the *events collection* (ii).

B. Semantic Annotation of Data obtained from Crowdsourcing

When a user notifies an event or incident about an accessibility feature by means of the MMA4A app, the corresponding data are to be stored to the *events collection* of our SR. These accessibility features are part of the infrastructure of the public transport and must be fitting with the user's needs. For example, a person with a mobility disability or temporary issue needs a lift to access to the public transport network. For this reason, we need to know which accessibility features are associated to each station or stop place.

To identify the specific accessibility features of any public transport network, we have carefully studied the Identification of Fixed Objects in Public Transport (IFOPT [27]) reference datamodel. It is a standard, which defines a model for the main *fixed objects* related to the access to Public Transport (e.g., stop points, stop areas, stations, connection links, entrances, etc.). IFOPT extends a previous model, i.e., the European Reference Data Model for Public Transport Information (Transmodel [28]). Transmodel is the European reference data model in the field, which provides a model of both public transport concepts and data structures that may be useful when building information systems related to the different kinds of public transport. It does not, however, provide any information about accessibility. IFOPT includes specific structures with which to describe accessibility data concerning the equipment of vehicles, stops and access areas. In this work, we have studied the different users' needs and the corresponding accessibility features able to satisfy them. In fact, we developed a proposal to annotate data of the bus network in a semantic way based on IFOPT in previous work [3]. Now, we have based on it to identify the users' needs. TABLE I resumes the elements and their correspondence with these users' needs.

TABLE I RELATIONSHIP BETWEEN FEATURES AND USERS' NEEDS

Accessibility features of public transport (based on IFOPT)	Accessibility needs of users (based on IFOPT)			
	Auditory and visual	Mobility	Phobia to lifts	Phobia to escalators
Lift	✓	✓	✗	✓
Escalator	✓	✗	✓	✗
Ramp	✓	✓	✓	✓
Stairs	✓	✗	✓	✓

The previous table summarizes the events or incidents about accessibility features we consider in this work. Relative to this, we have also analyzed both Transmodel and IFOPT standards to identify the specific characteristics of such events, in order to include them. Based on both models, we have developed a specific *vocabulary* as a RDF Schema [29], which comprises the necessary information to annotate, in a semantic way, these events about accessibility, obtained via crowdsourcing. Figure 3 shows this terminology and their relationships.

Using this vocabulary, we can register whether a lift does not work (*hasLift FALSE*) in a specific *StopPlace* (i.e., station) of a specific *ofLine* (line of transport). This event has also an associated opening date (*openDate*). When the incident is solved, we can close it indicating the final date (*closeDate*) and then assigning a TRUE value to both *hasLift* (the original feature) and *closedEvent*.

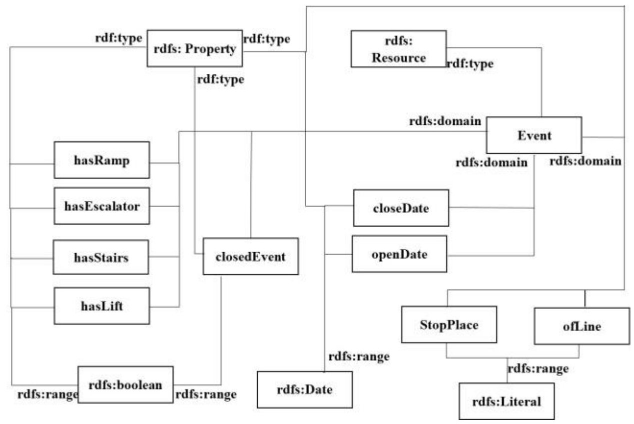


Figure 3 Events vocabulary in RDF Schema.

In summary, the event data about accessibility features, processed from many sources (*crowd*), is semantically annotated based on the vocabulary and it is then stored in the Jena semantic repository, in the events collection.

C. The workflows between client and server side

As mentioned before, MMA4A is a software for smart devices, able to compute a specific public transport route based on the users' needs and to notify the events or failures related to accessibility features of the network. We want to underline that the MMA4A app is still a prototype concerning the accessibility of the user interface (it does not include a fully accessible UI).

To calculate the route, the software needs to work with updated data (both about the network infrastructure and the state of the accessibility features). To provide an updated bootstrap, we have decided that when the app starts up, it requests to download this information to the server.

Also, from the client perspective, we have identified the different actors and use cases (see Figure 4).

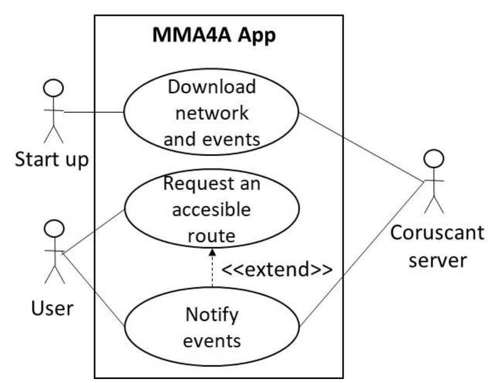


Figure 4 Use case diagram for MMA4A.

Now, we describe the workflows between client and server in more detail (see Figure 5).

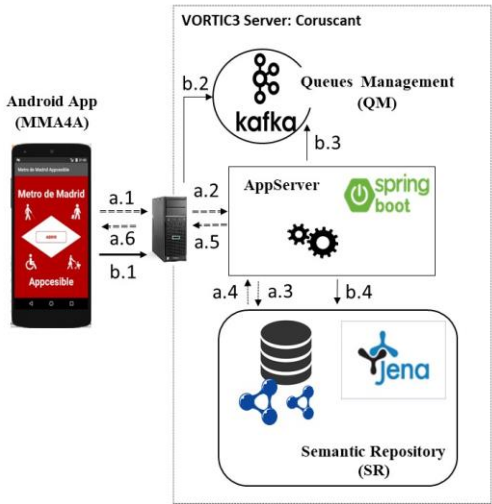


Figure 5 Workflows of the proposed architecture.

These are the workflows corresponding to the relationships between the Coruscant server actor and the associated use cases (*Download network and events*; and *Notify events*, respectively): (a) When the MMA4A starts up, it requests to download the data about the public transport network and the accessibility events stored in the Coruscant server. These two downloading requests are sent to the RESTful AppServer API; consequently, it generates the respective RDF/XML [30] files from the Jena Semantic Repository to return them to the app; (b) When a user notifies an event or failure about an accessibility feature, the app sends the event to the QM in the server side. Then, the AppServer consumes this information from the QM and next, it writes this incident in the SR as a RDF triple. Figure 5 shows (a) and (b) workflows between client and server side.

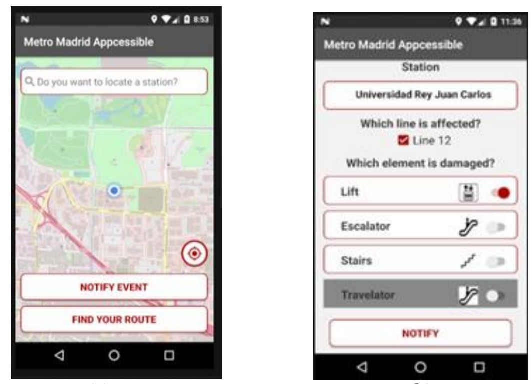
V. THE CASE STUDY

In the following section, we validate our smart data framework against real data from the subway in the city of Madrid, Spain (Metro Madrid [31]), used by means of the prototype MMA4A app. Specifically, we want to validate the incident management ("crowdsourcing") facility.

Before using this semantic dataset, it has been validated by means of SPARQL queries. These were defined to verify the correctness of data. It is relevant to highlight that this information was the outcome of an integration process from several sources of public transport data.

To validate our proposal, we have defined the following process. First, the user selects the "*Notify an event*" option; second, the user indicates what accessibility feature does not work in a specific station; third, we validate the RDF code generated for this event; fourth, a different user, who has a specific user need, request a route; this route is composed by stations where some accessibility features do not work, including one related to the previously notified event.

Next, we show this process for a specific case study in a detailed way. First, a user (UserA) selects to notify an event (*Notify events* use case). Figure 6a depicts the user interface where the user can select this. Second, the user notifies that the *lift* of the Universidad Rey Juan Carlos *station* in Metro Line12 does not work (Figure 6b).



(a) (b)

Figure 6 App user interface: (a) "*Notify an event*" and (b) "*Which accessibility elements do not work*".

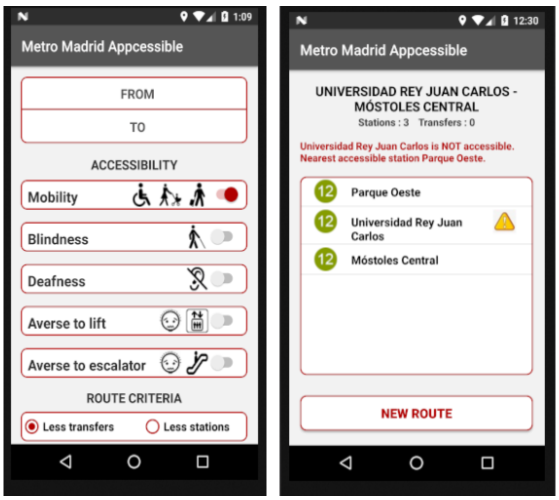
Third, when the UserA selects the NOTIFY bottom, the app generates an RDF event notification to send to the server. Figure 7 shows the RDF code describing the event generated before, about the lift accessibility feature in the Universidad Rey Juan Carlos station. This information is stored in the Jena SR by the AppServer component.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:mao="http://com.vortic3.MANTO#"
   xmlns:foaf="http://xmlns.com/foaf/0.1/" >
  <rdf:Description rdf:about="http://com.vortic3.MANTO#Universidad Rey Juan Carlos-Linea12MetroSur-lift">
   <rdf:type rdf:resource="http://com.vortic3.MANTO#Event"/>
   <mao:ofLine>https://www.metromadrid.es//es/viaja_en_metro/red_de_metro/lineas_y_horarios/linea12.html</mao:ofLine>
   <mao:hasLift>FALSE</mao:hasLift>
   <mao:openDate>2018-06-22T13:40:36.686</mao:openDate>
   <mao:closedEvent>FALSE</mao:closedEvent>
   <mao:StopPlace>https://www.metromadrid.es//es/viaja_en_metro/red_de_metro/estaciones/UniversidadReyJuanCarlos.html</mao:StopPlace>
  </rdf:Description>
</rdf:RDF>
```

Figure 7 RDF code about the event: Lift failure at the Universidad Rey Juan Carlos station in Line12

Fourth, another user (i.e., UserB) will request a route. Previously, when the MMA4A was started up in UserB's smart device (e.g., after a reboot), the app requests the infrastructure data and the stored events to the server. Regarding the calculation of the route, there are several possibilities: for instance, it is possible for users to obtain an accessible route in the Metro by means of a specific algorithm, considering the different users' needs. Perhaps no accessible route exists, but if one does, then this app will be able to calculate it. The user interface in Figure 8a shows the information that the app requires in order to establish the route. First, the origin and the destination of the trip; second, the kind of user's special needs (defined by groups: people with special mobility needs, blind people, hearing-impaired people); and third, a decision about the route (minimizing commutes or stations). In our case study, UserB indicates a mobility need. Then, the app computes the route, taking these choices into account.

It is important to stress that the algorithm provides a route, only after examining that the start, destination and any transfer stations have the required accessibility features. In our case study, we have requested a route starting at the Universidad Rey Juan Carlos station. Then, as previously indicated, the app notifies a lift failure in this station (see Figure 8b) and offers as an alternative the closest accessible stations (in our case study, the closest accessible stations are Parque Oeste and Mostoles Central as also shown in Figure 8b).



(a) (b)

Figure 8 App user interface: (a) Request a route and (b) Notify an event to the user.

Figure 9 provides a partial image of the Metro Madrid map, to show that the closest stations offered to the user are the best alternative.

Figure 9 Partial image of the network of Metro Madrid.

The app also asks to any other user who passes by this station whether the lift works now. Figure 10 shows this alert message.

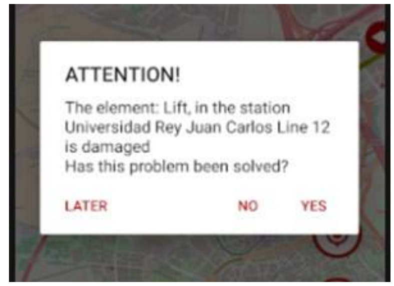


Figure 10 About Metro Madrid. Alert message of MMA4A.

We have carried out many other tests according to the different user needs identified in this solution (auditory and visual disabilities, mobility disability, phobia to lifts and phobia to escalators). We have also verified that the routes provided by MMA4A are correct for these kinds of users.

In summary, our experience with this app proves that data in the semantic repository can be used to generate the routes for users with special needs, while the incidents are registered and considered while elaborating those routes. The experiment described here shows that the behavior of the system is correct at this moment. We cannot guarantee that event notifications were always true.

You can download both the data infrastructure and the data events (RDF files) whose structure we describe in this paper in the following links: http://coruscant.my.to:8080/download/metro.xml, http://coruscant.my.to:8080/download/events.xml.

VI. Conclusion and Future Work

One of the major challenges of smart cities initiatives is to achieve an inclusive society for all citizens, including those with special needs, such as mobility issues. If this challenge is to be met, more thorough information about the means of transport and their accessibility features is required, to be able to arrange and provide accessible routes for everybody, including those special users.

There already are several web applications and tools which provide information and services for transport users. We have studied some of them and to the best of our knowledge, currently no software application takes fully into account the current status of the public transport network with respect to the accessibility features; in particular, updating data with a crowdsourcing strategy, to compute accessible routes for special needs users.

In order to solve this issue, we have proposed a *smart data architecture* to manage both data sources about the infrastructure of the public transport.

Our proposal considers accessibility features when it computes a route. But it goes beyond that, by taking live information into account, such as incidents that may occur at any time. Thus, our dataset provides a smarter response in real-time. These live data, coming from crowdsourcing, are collapsed and integrated with existing accessibility information, providing a *real-time accessibility dataset*.

We provide a Kafka Queuing Manager to gather users' notifications about the state of accessibility features in the public transport, and a Jena Semantic Repository to store both these data and the infrastructure data about the transport network. We have also developed a smart app (still a prototype) for public transport users, which is able to compute accessible routes taking the user's needs into account, and also to notify incidents or events in the accessibility features of the network. In order to semantically annotate the data before storing them, we have also defined a specific vocabulary as a domain-specific RDF Schema.

Regarding future work in this area, we intend to include more public transport information from other sources and means of transport, and to integrate them with the current architecture. For this purpose, it will be necessary to semantically harmonize them using our MAnto ontology, which has already been used to semantically annotate the data stored in the infrastructure collection of our Jena repository. This will make possible to provide fully open datasets for different public transport networks. Moreover, these datasets must be published in an open platform, providing free access to accessibility and special needs data. We have also worked in gathering information about accessible pedestrian routes in the city, obtained via crowdsourcing techniques, capturing the geographical information and accessibility features of these routes. This information will be also incorporated as smart data into the Multiply@City platform.

ACKNOWLEDGMENT

This work is supported by the Multiply@City and the Access@City projects (TIN2016-78103-C2-1-R), funded by the Spanish Ministry of Science, Innovation and University. We also would like to thank Iván Pérez Muñoz for his assistance in the development process.

REFERENCES

[1] P. Cáceres, A. Sierra-Alonso, C. E. Cuesta, J. M. Cavero, and B. Vela, "Enhancing Accessibility Information in Google Maps: Adding new pieces of information to GTFS to improve accessibility" In Proceedings on the Second International Conference on Universal Accessibility in the Internet of Things and Smart Environments (SMART ACCESSIBILITY 2017), pp. 6-9, Nice (France), 2017.

[2] P. Cáceres, A. Sierra-Alonso, B. Vela, J. M. Cavero, and C. E. Cuesta, "Towards Smart Public Transport Data: A Specific Process to Generate Datasets Containing Public Transport Accessibility Information" In Proceedings on the Third International Conference on Universal Accessibility in the Internet of Things and Smart Environments (SMART ACCESSIBILITY 2018), pp. 66-71, Rome (Italy), 2018.

[3] P. Cáceres, A. Sierra-Alonso, C. E. Cuesta, B. Vela, and J. M. Cavero, "Modelling and Linking Accessibility Data in the Public Bus Network", Journal of Universal Computer Science, vol. 21, no. 6, pp. 777-795, 2015.

[4] T. Sarjakoski et al., "Landmarks and a Hiking Ontology to Support Wayfinding in a National Park During Different Seasons". In M. Raubal, D. M. Mark, and A. U. Frank (Eds.), Cognitive and Linguistic Aspects of Geographic Space, Lecture Notes in Geoinformation and Cartography. Springer, 2013. DOI:10.1007/978-3-642-34359-9.

[5] Wheelmate,http://myhealthapps.net/app/details/13/wheelmate, [retrieved: Dec. 2018].

[6] Wheelmap, https://wheelmap.org/, [retrieved: Dec. 2018].

[7] Accessmap, https://www.accessmap.io/, [retrieved: Dec. 2018].

[8] Ciudades Patrimonio de la Humanidad, http://www.ciudadespatrimonio.org/accesibilidad/?idioma=en, [retrieved: Dec. 2018].

[9] J. Zimmerman et al., "Field Trial of Tiramisu: Crowd-Sourcing Bus Arrival Times to Spur Codesign". In Proc. CHI 2011, New York, pp. 1677–1686, 2011.

[10] B. Ferris, K. Watkins, and A. Borning. "OneBusAway: Results fromProviding Real-Time Arrival Information for Public Transit". In Proc. CHI 2010, Atlanta, Georgia, April 10–15, 2010.

[11] Moovit, https://moovit.com/, [retrieved: Dec. 2018].

[12] Swiftly, https://www.goswift.ly/, [retrieved: Dec. 2018].

[13] BUSUP, https://www.busup.com/, [retrieved: Dec. 2018].

[14] CIVITAS, http://civitas.eu/, [retrieved: Dec. 2018].

[15] OpenTRipPlanner, http://www.opentripplanner.org/, [retrieved: Dec. 2018].

[16] OpenStreetMap, https://www.openstreetmap.org, [retrieved: Dec. 2018].

[17] GTFS, Google Transit Feed Specification, https://developers.google.com/transit/gtfs/, [retrieved: Dec. 2018].

[18] C. E. Cuesta, P. Cáceres, B. Vela, and J. M. Cavero, "CoMobility: A Mobile Platform for Transport Sharing", ERCIM News, no. 93, pp. 22-23, 2013.

[19] Access@City project (TIN2016-78103-C2-1-R), funded by the Spanish Ministry of Science, Innov. & University.

[20] Multiply@City project (TIN2016-78103-C2-1-R), funded by the Spanish Ministry of Science, Innov. & University.

[21] Regional Consortium for Public Transports of Madrid, http://datos.crtm.es/, [retrieved: Dec. 2018].

[22] EMT Madrid home page, http://www.emtmadrid.es/Index.aspx?lang=en-GB , [retrieved: Dec. 2018].

[23] Spanish Society of Blind People (ONCE, Organización Nacional de Ciegos de España) http://www.once.es/new/otras-webs/english, [retrieved: Dec. 2018].

[24] Spring Boot, http://spring.io/projects/spring-boot, [retrieved: Dec. 2018].

[25] Apache Jena, https://jena.apache.org/index.html, [retrieved: Dec. 2018].

[26] Apache Kafka. https://kafka.apache.org/intro, [retrieved: Dec. 2018].

[27] CEN/TC 278. Intelligent transport systems - Public transport - Identification of Fixed Objects In Public Transport (IFOPT), EN 28701:2012.

[28] Transmodel, standardization, E. C. (2016). Transmodel, Road Transport and Traffic Telematics. Public Transport. Ref. Data Model, EN 12896, http://www.transmodel.org/en/cadre1.html, [retrieved: Dec. 2018].

[29] RDF Schema, https://www.w3.org/TR/rdf-schema/, [retrieved: Dec. 2018].

[30] RDF/XML, https://www.w3.org/TR/rdf-syntax-grammar/, [retrieved: Dec. 2018].

[31] Metro Madrid home page, http://www.metromadrid.es/en/, [retrieved: Dec. 2018].

 ISBN: 978-1-61208-692-7