CREATION, IMPLEMENTATION AND EVALUATION OF AN INDOOR NAVIGATION SYSTEM FOR USERS WITH DISABILITIES

Y. M. Montilla1∗, C. León-Sánchez2, I. Lizarazo Salcedo3

1 National University of Colombia - ymontilla@unal.edu.co
2 3D Geoinformation group, Delft University of Technology, Faculty of Architecture and the Built Environment, the Netherlands - c.a.leonsanchez@tudelft.nl
3 Spatial Analysis of the Territory and Global Change (AET-CG) Research Group. National University of Colombia Bogotá, Colombia - ializarazos@unal.edu.co

Commission IV, WG IV/9

KEY WORDS: 3D Indoor Navigation, Indoor Positioning, Visually Impaired, 3D Modeling, IndoorGML, CityGML, BLE, RSSI, WPL

ABSTRACT:

Indoor navigation is a complex task for people and especially for visually impaired ones. This research proposes an indoor navigation system oriented to visually impaired users, integrating the OGC IndoorGML and CityGML standards with Bluetooth Low Energy devices (BLE) by using the RSSI signal loss value and the Weighted Path Loss techniques to calculate the user’s location. This paper describes the design of the system and its implementation as a functional prototype in a mobile web application. Several operational tests were conducted to determine both the accuracy and precision of the user location. The user’s positioning results show a root-mean-squared error (RMSE) of 0.88 m in a scenario with obstacles and no height difference of the BLE devices location, and a RMSE of 1.06 m in a scenario with obstacles and height difference. These results confirm the potential of the implemented prototype to grow into a fully operational system.

1. INTRODUCTION

Navigation is a basic action in people’s daily lives (Tippe et al., 1992), which allows the recognition of indoor and outdoor spaces (Yan et al., 2019). Indoor navigation is becoming more complex due to the increasing size of buildings (Alattas et al., 2017) and their elements such as stairs, doors and elevators (Li, 2008, Kontarinis et al., 2019). The challenge increases for people with disabilities (Park et al., 2020) because they rarely have access to maps, signs, or other orientation devices (Coret Gorgonio et al., 2015). Visually impaired people usually require assistance and orientation to reach unfamiliar destination indoor spaces (Alattas et al., 2017).

Nowadays, the availability of new technologies has sparked a great deal of interest on indoor navigation systems (Zafari et al., 2019, Simões et al., 2020). As a result, geographic information systems are no longer limited to manage outdoor space, but also indoor spaces (Li, 2008). This emerging trend has generated multiple proposals for indoor navigation, which incorporate different representation models, technologies and positioning techniques (Simões et al., 2020). Many proposals focus on the use of communication technologies such as Wi-Fi, Bluetooth, RFID, (Coret Gorgonio et al., 2015), in order to overcome Global Navigation Satellite Systems (GNSS) inefficiency in enclosed spaces due to signal attenuation and losses (Pérez-Navarro et al., 2018, Simões et al., 2020).

Several standards can encompass the representation of interior spaces from a geometric approach, such as OGC’s CityGML and KML 2.0 and ISO 16739-1:2018 IFC (Industry Foundation Classes). However, these standards lack semantic features required for navigation purposes in an indoor space (Kim et al., 2014). IFC and CityGML are semantically poor because they do not implement explicitly topological relations as neighborhoods between spaces (Isikdag et al., 2013). In contrast, the OGC IndoorGML standard is designed to materialize the semantic representation of indoor spaces to support indoor navigation systems (OGC, 2014). The differences between the mentioned standards are basically because each standard is created for a different purpose (Wong and Lee, 2019); nevertheless, IndoorGML models can be derived from CityGML models in LoD4 or IFC (Kim et al., 2014).

In this paper, we propose an indoor navigation system aimed at facilitating the mobility of visually impaired users. We describe the process to integrate the CityGML and IndoorGML standards with BLE technology and specify the technical characteristics of the system’s components. Integrating the standards is essential to replicate the proposed approach, to have a geometric representation of the space, to provide context information, and to enable a topological and semantic component. Then, we propose the use of BLE devices to determine the user location in real time. This approach allowed us to implement and test a prototype which provides a balanced solution between cost and positional accuracy.

2. DATA AND METHODS

2.1 Study Area

The navigation system is implemented for the Birmania building, at the municipality of Fusagasugá, department of Cundinamarca, in the Andean region of Colombia. It is a four-story residential building with an approximate usable area of 202.5 m² in total. The building contains four dwellings, one per floor.

* Corresponding author
2.2 Analysis of user requirements

User requirements for the indoor navigation system is conducted according to well-established criteria suggested by several authors (Afyouni et al., 2012, Simões et al., 2020, Miao et al., 2011). Based on the literature review, generic requirements of an indoor navigation system and specific requirements for addressing needs of visually impaired users were established. The specification is organized around the following three criteria:

Generic service criteria:

- **Localization**: The system should provide a location with respect to a coordinate system (Afyouni et al., 2012), with accuracy and timeliness (Wirola et al., 2010, Simões et al., 2020). The accuracy refers to both vertical and horizontal coordinates (Wirola et al., 2010). The quality of localization should balance location accuracy and speed of response (Simões et al., 2020).
- **Navigation**: The route should be planned according to user’s capabilities and preferences, and should track the route followed by the user to provide relevant information for guidance and de-tour notifications (Afyouni et al., 2012).
- **Visualization**: Provide visual information to users of their position (Simões et al., 2020) as well as to their surroundings and to context information (Afyouni et al., 2012).

Generic efficiency criteria:

- **Modeling effort**: It should focus on the building and it should be updated according to the current situation of the interior space (Afyouni et al., 2012).
- **Flexibility**: Possibility of expanding the system to navigate a larger space and incorporate other types of sensors and technologies (Afyouni et al., 2012).
- **Efficiency**: Speed of response to user requests (Afyouni et al., 2012).
- **Scalability**: Ease of scaling up to a larger system (Simões et al., 2020), while maintaining its capacity (Afyouni et al., 2012).
- **Others**: Low implementation cost (Simões et al., 2020).

Specific requirements for supporting visually impaired people:

- **Localization**: It must be provided with high spatial accuracy (Martinez-Sala et al., 2015), and in real time (Nakajima and Haruyama, 2013).
- **Navigation**: The planned route should address the user’s particular preferences and needs (Miao et al., 2011), and provide contextual information that allows them to be aware of eventual obstacles along the way (Simões et al., 2020, Miao et al., 2011, Nakajima and Haruyama, 2013). Other relevant navigation data should include direction of travel and distance to the destination (Nakajima and Haruyama, 2013).
- **Application**: It should be simple to use and include commands easy to memorize, preferably with voice commands. It should include both a mobile and a desktop version for planning routes in advance and options to select preferences for route calculation and directions in terms of right, back, left and distance in meters (Miao et al., 2011).
- **Data model**: It should include attributes on critical spaces for navigation of impaired people, for example, space width, slope, changes in elevation, eventual voids or promontories on the floor; type and opening direction of doors and doorlocks; location of handrails, elevators, wheelchair elevator and ramps (Park et al., 2020).

2.3 System architecture

The architecture defines the system components and how they interact with each other (ISO, 2011). We propose four components: 1) A network of BLE sensors in the building broadcast their RSSI signal. 2) A database that provides the semantic and topological information of the building; it stores part of the model, containing the topological information that provides the navigation network for routing; it also stores the CityGML model that provides the context information of the space and provides the known information of the sensors (location, name, identification). 3) A web service, to allow the communication between the application and the database. 4) A mobile application, to provide a graphical user interface (GUI) to interact with the user, allowing the visualization of the 3D model of the building.

2.4 Construction of a 3D representation model

As the building lacked a digital or analogue architectural plan, it is necessary to create it from scratch and then to store the corresponding data in a database. This architectural data includes only those elements required to generate the CityGML v2.0 model at level of detail 4 (LoD4). We use the satellite image provided by the Bing Maps service to assign the geographic coordinates of the building. Once the structuring of the data is completed, the following layers were obtained: room, Muro_R, Muro_L, door, window, staircase, floor, ceiling, plate and roof. The Muro_L layer includes all the vertical elements needed to have an accurate representation of indoor spaces in the building, including walls and columns. To facilitate the construction of the CityGML model, the attributes described in the table 1 were assigned to each layer. Likewise, the attributes of interest for people with disabilities were assigned to the room (Table 2) and door (Table 3) layers.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Unique object identifier</td>
</tr>
<tr>
<td>ParentID</td>
<td>Parent object identifier following CityGML</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the space</td>
</tr>
<tr>
<td>No. Floor</td>
<td>Storey above the ground where the object is</td>
</tr>
<tr>
<td>Height</td>
<td>Architectural height of the object</td>
</tr>
<tr>
<td>AlturaZ</td>
<td>Altitude above sea level of the object</td>
</tr>
<tr>
<td>Type</td>
<td>Type of navigation in the spaces: transitional</td>
</tr>
<tr>
<td>Function</td>
<td>or general (applies only to the room layer)</td>
</tr>
<tr>
<td>Color</td>
<td>Color of the object to assign the appearance</td>
</tr>
<tr>
<td>Inclination</td>
<td>Defines whether the object is tilted or non-</td>
</tr>
</tbody>
</table>

Table 1. General attributes

The development of the CityGML and IndoorGML models is carried out following the steps proposed by (Montilla and León-Sánchez, 2020).

We created the LoD4 CityGML v2.0 file from the architectural plan database using FME, the gml_description attribute, which stores all the physical characteristics of the space. We also used
Table 2. Attributes in room layer

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transit_Type</td>
<td>Indicates whether the space corresponds to a room, stairs, hallway, etc.</td>
</tr>
<tr>
<td>Width</td>
<td>Width of the space if it has a transit function</td>
</tr>
<tr>
<td>Slope</td>
<td>Slope of the space.</td>
</tr>
<tr>
<td>Caract_Steps</td>
<td>If the space corresponds to stairs, indicate the number of steps and the average height.</td>
</tr>
<tr>
<td>Caract_Stairs</td>
<td>If the space corresponds to stairs, indicate the characteristics of the stairs, the presence of handrails, stair lifts, etc.</td>
</tr>
</tbody>
</table>

Table 3. Attributes in Door layer

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type_Door</td>
<td>indicates whether the door is hinged, folding or electronic, additionally specifies the material</td>
</tr>
<tr>
<td>Width</td>
<td>Width of the door</td>
</tr>
<tr>
<td>Height</td>
<td>Height of the door</td>
</tr>
<tr>
<td>Door_opening</td>
<td>Defines the type of door opening, e.g. Left Inside</td>
</tr>
</tbody>
</table>

FME to create the IndoorGML file derived from the CityGML model. To obtain the connections between spaces we used a topological analysis and from this we constructed the Node Relationship Graph (NRG) using the Poincaré duality. This process outputs the class State, a node of each space and the class Transition, which establishes the connection between nodes.

Objects room and door from the CityGML model and the classes state and transition from the IndoorGML model are migrated to the database. CityGML objects are essential to provide the reference information of the location space, while the IndoorGML model classes provide the navigational network elements for routing. The migration of the CityGML and IndoorGML classes to the PostgreSQL database is performed using a workflow in FME; before performing the upload, the attributes were adjusted so that the tables created were compatible with the Dijkstra shortest path search algorithm (Dijkstra, 1959) available in pgRouting.

### 2.5 Indoor positioning

We use E2 Max Beacons produced by the company Minew for indoor location services and commercial advertising (Minew, 2021). The devices were configured to issue their UID using the BeaconSET+ application suggested by the manufacturer (Minew, 2021).

In order to test the accuracy of the indoor positioning methodology, the devices were installed following three scenarios. In the first one, the four devices were placed at the same height in a space without obstacles; therefore, there is a direct visual between the mobile device and the BLE devices. In the second one, the four devices were placed at the same height in a space with obstacles between them such as walls and doors; the devices had no visual between them. In the last one, the devices were installed at different heights with obstacles between them; there is no direct visual between the devices. It is important to point out that since there were only four BLE devices, it is not possible to have coverage of the entire building, so the devices were relocated according to the different scenarios at several rooms.

The information of the device location points is stored in the database. The table is composed of the following attributes: id, the point identifier; UID, MAC address of the BLE device; and the_geom, point geometry \((x, y, z)\) of the device location in the building according to the scenarios proposed. Each time the scenario is changed, the location of the devices in the database is updated.

Device detection is performed using the cordova-plugin-ble-central add-on, which enables communication between the smartphone and the BLE device (Coleman, 2014). The distance is calculated considering the signal loss model given by the equation 1 (Huh and Seo, 2017).

\[
RSSI = -10n \log_{10}(d) + A \tag{1}
\]

where \(d\) = Distance between the BLE and the mobile \(A\) = Reference RSSI value at one meter distance \(n\) = Signal propagation exponent

The reference RSSI value \(A\) corresponds to the median of 1000 measurements of the RSSI value at a distance of one meter. From the measurements made of the RSSI value at different distances in each scenario, the parameter \(n\) is calculated using the nonlinear least squares, Trust-region fit. Having the values of \(A\) and \(n\) it is possible to calculate the distance \(d\) by the equation 2 obtained from the equation 1.

\[
d = 10^{ \frac{-10}{n} (RSSI_{Ref} - A) } \tag{2}
\]

The positioning technique used is based on the Weighted Path Loss - WPL algorithm proposed by (Zou et al., 2013a) and used in WiFi and RFID based systems. The system detected the RSSI signal from the sensors, in this case four (4) BLE devices, and calculate the distance between them and the mobile device. The distances \(d\) assigned to each device \(N\) at a time step \(t\) are expressed as a vector \(\{d_1^t, d_2^t, d_3^t\}\), from which a weight is assigned to each device using the equation 3.

\[
w_i^t = \frac{1}{N} \frac{1}{d_i^t} \sum_{i=1}^{N} \frac{1}{d_i^t} \tag{3}
\]

Thereafter, the location of the mobile \(\{x, y, z\}\), is calculated with the equation 4:

\[
(x, y, z) = \sum_{i=1}^{N} w_i^t(x_i, y_i, z_i) \tag{4}
\]

where \(\{x_i, y_i, z_i\}\) = Location of each BLE device.
2.6 Interior Navigation System

On the Amazon Web Services (AWS) cloud computing platform an EC2 instance is created with Ubuntu 18.04 (Amazon Web Services, Inc., 2022), where the web server and database were configured. The web service is composed of a PostREST web server, which focuses exclusively on database mapping, so a reverse proxy is required to bring the system into production. The reverse proxy is configured on NGINX, a free and open web server for HTTP, TCP and UDP traffic (Ngixn, 2021). The SSL certificate is generated using the OpenSSL library. PostgreSQL is mounted in a Docker container using a Postgres 11 image. PostGIS 2.5 and pgRouting 2.6.3. Once PostgreSQL is installed, a schema for the application named api is created in the database. The required tables of the CityGML and IndoorGML model and the table with the sensor information, mentioned above, were created within this schema. Additionally, the following functions were created to manage the queries:

- A function to provide information from the network of sensors belonging to the navigation system.
- A function to query the optimal path from a initial space to a destination space within the building. This has as parameters the corresponding name of both spaces. The computation of the route is performed using Dijkstra’s algorithm.
- A function provides the context information of the user’s location. It receives the coordinates (x, y, z) corresponding to the user’s location and selects which space contains it which can be a door or a room.
- A function to provide the geometry to display the start and end points of the path in the application. is named space returns the geometry of the class state.
- A function to provide the geometry to display the estimated location point of the user. The function receives the coordinates (x, y, z) in EPSG:3857 and converts them to the EPSG:4326 coordinate system.

The graphical user interface is created with a simple design, with high contrast colors that allow easy differentiation between functionalities, facilitating its use for people with reduced visual capacity and users in general. The 3D visualization is created using CesiumJS (Cesium GS, Inc, 2022b), version 1.80 released on April 1, 2021. The CityGML model is deployed using the Cesium ion platform (Cesium GS, Inc, 2022a). This platform makes the data available online and makes use of tiles, providing a high performance. In addition, the 3D Tiles format allows a faster deployment of the model and it does not lose geometric and semantic information (Buyukdemircioğlu and Kocaman, 2020). Users press the location button and the location space is highlighted on the screen, the result is accompanied by audios to provide contextual information of the user’s location, which is beneficial for visually impaired users.

The indoor navigation system application is functional prototype developed for a mobile environment which allowed testing the generic functionalities of the system. It is created in the open source Apache Cordova development environment which is used to create cross-platform applications through a combination of standard (HTML5, CSS3 and JavaScript) and native web application technologies. HTML cannot access the native components of the mobile device, therefore, plugins were installed to be able to access them (The Apache Software Foundation, 2016).

To search for the route, a text box must be used to enter the origin and destination spaces, an aspect that should be improved to facilitate the use by visually impaired users. Figure 1 shows the user interface of the application with the different functionalities to which the user has access.

The data used, the workflows and source code of the application is available on the GitHub link: https://github.com/ymontilla/IndoorNavigationSystem

3. RESULTS

3.1 Models of representation

Figure 2 shows the result of the CityGML model of the building. The appearance is limited to the assignment of similar colors and textures or other shapes were omitted to produce a realistic appearance. For example, on the facade only a similar color is assigned to the bricks.

Figure 3 shows the IndoorGML model derived from the CityGML model. The different spaces with their respective nodes and the transition class that connects the spaces horizontally and vertically can be observed. It is important to highlight that this model is automatically derived from the CityGML model by applying the FME workbench. The state and transition classes of the IndoorGML model constitute the navigation network and are therefore essential for routing queries.

3.2 Indoor positioning

The first step determines the parameters of the equation to calculate the distance, assuming the logarithmic decrease of the
signal strength as the distance increases. The parameter $A$ is taken as the median of the RSSI values at a distance of 1m. The parameter $n$ is determined by nonlinear least squares, Trust-region fitting. For each scenario, different reference points were stipulated at a given distance from the location of the BLE devices, as shown in the figure 4, taking 1000 readings for each point. The device involved in the experiments is the moto g(8) plus smartphone with Android 10. To perform the measurements an Android application is developed, considering the same parameters with which the indoor navigation system application works.

In the first scenario, with direct visual between the BLE devices and the smartphone, the RSSI value was read at a distance of 1m, 2m, 3m, 4m and 5m. According to these measurements the parameter $A$ is set at $-56$ RSSI and the parameter $n$ is determined at 1.4. According to the literature this value corresponds to a free space (Naghdil and O’Keefe, 2019), it is assumed that the obtained value responds to the fact that the measurement is performed without any obstacle present. Figure 5 shows the RSSI measurements at each reference point, in red the RSSI value measurements and in blue the theoretical values calculated using the distance equation and the parameters $A$ and $n$ defined above.

For the second and third scenarios, with obstacles such as walls and doors between the BLE devices and the smartphone, the RSSI value is read at a distance of 4m, 6m, 8m and 10m, which allowed setting $n$ at 3.0. This value increased with respect to scenario one due to the presence of obstacles in this scenario. Figure 6 shows in red the RSSI value measurements and in blue the theoretical values calculated using the distance equation. It shows that as the distance increases, the signal loss value increases, reflected in the RSSI value.

To determine the precision and accuracy of the location calculation, reference points, considered as actual locations, were compared with estimates of the location of each point. The reference points were drawn on the building plan, then the corresponding coordinates $(x, y, z)$ were extracted using ArcGIS (Esri, 2022). Subsequently, the points were measured and marked on the floor of the respective space, taking measurements from the walls. To perform the measurements, the mobile device is placed over each mark at a height of approximately 1.10 m. In each scenario, 30 measurements were taken for each reference point, then different comparisons were made to determine their precision and accuracy, in this process the differences in each dimension $(x, y, z)$ and the distance differences were considered. The precision of the estimated locations is determined by the standard deviation $(\sigma)$ and the coefficient of variation $(CV)$. The maximum error $(e_{max})$ and the RMSE were used as criteria to analyze the accuracy the latter is calculated by the equation 5 (Yang et al., 2020).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}[(\hat{x}_i-x_i)^2+(\hat{y}_i-y_i)^2+(\hat{z}_i-z_i)^2]}{N}}$$

where $(\hat{x}_i, \hat{y}_i, \hat{z}_i) =$ Estimated coordinates of the point. $(x_i, y_i, z_i) =$ True point coordinates $N =$ Total number of measurements

The results of the test to determine the location are summarized in table 4. Accuracy values lower than at 1m were achieved in all three scenarios, however, in a few cases, 2m error were measured as in the case of scenario one, which indicates a high variability of the data, a statement that is confirmed by the coefficient of variation.
The results do not show a significant difference in accuracy between a space with or without obstacles between the devices; therefore, it can be suggested that in terms of costs a system where only one BLE device per space is located is more viable. The results of scenario three indicate that the height can also be calculated from the WPL technique; however, it is important to determine how much the result is affected by the location of the BLE sensors.

On the other hand, the number of hits (correct positioning) of the location space is also evaluated. In scenario one (figure 7) all estimates hit the location space. In scenario two (figure 8) all the estimates made for points 1 and 3 got the location spaces completely right, while at point 2, located in the corridor, only 33% got the location space correctly. In scenario three (figure 9), 37% of the estimates were correct with location sub-spaces in the stairs. The results suggest that the smaller the space or sub-space the lower the probability of hitting the location space.

### 3.3 Indoor Navigation System

A tangible result of this research is a working prototype of the application which integrates all the components of the navigation system. The operation of the application is illustrated in the video available at the following link https://youtu.be/L11Y8j_zKWA. The evaluation of the prototype focused on measuring the response times of the main functions and their respective REST services. The process consisted of executing each function 20 times, measuring the response time when the function starts until a result is obtained and also measuring the response time of the REST services involved; finally, the mean of the measurements for each test is calculated. All measurements were performed using google Chrome DevTools tool (Chrome Developers, 2022).

<table>
<thead>
<tr>
<th>Function</th>
<th>Time(ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Startup</td>
<td>8423.85</td>
</tr>
<tr>
<td>Routing</td>
<td>6357.25</td>
</tr>
<tr>
<td>Location in unobstructed spaces</td>
<td>9705.45</td>
</tr>
<tr>
<td>Location in spaces with obstacles</td>
<td>16358.44</td>
</tr>
</tbody>
</table>

The average startup time of the application is 8423.85ms of which approximately 4544.99ms are used for the loading of the CesiumJS library elements. Additionally, it is detected that in an unobstructed space the signal detection and weighted location calculation is approximately 3400.65ms while in an obstructed space the time increases to 10082.16ms. In the above two scenarios the process involved in requesting the REST services and displaying the results both audio and visually takes approximately 6.2s. In conclusion, having the BLE devices in a space with direct visual with the mobile device offer better performance in terms of response time than a system with the BLE devices distributed in a way that there are obstacles with the mobile device.

### 4. DISCUSSION

Regarding the functionality of the IndoorGML and CityGML models in an indoor navigation system oriented to visually impaired users, we can state that those standards are relevant and complement each other for this study case. It was possible to determine from the IndoorGML model, the shortest route between different points of the building, no matter their location inside the building; on the other hand, the CityGML model made possible to search for routes based on the name of the...
spaces as well as provides context information of the user’s location space, such as its physical characteristics; this information is especially useful for visually impaired users because it helps them to notice possible obstacles. Further work is needed to assess the relevance of the created models to determine the ideal route according to the preferences and particular conditions of each user.

Regarding the positioning technique, in previous researches using the WPL algorithm in a two-dimensional space \((x, y)\), with RFID technology obtained an average error of 1.651m (Zou et al., 2013a), with WiFi technology (Poulose and Han, 2019) obtained an RMSE of 1.7m. We obtained an RMSE of 0.88m in a two-dimensional scenario \((x, y)\) with obstacles. Another aspect to note is that in previous researches the WPL algorithm had been used in a two-dimensional \((x, y)\) with obstacles. Another aspect to note is that in previous researches the WPL algorithm had been used in a two-dimensional \((x, y)\) with obstacles. Another aspect to note is that in previous researches the WPL algorithm had been used in a two-dimensional \((x, y)\) with obstacles.

The above comparison suggests that using the WPL algorithm together with the BLE technology yields better results. However, we obtained better results by the fusion of the WPL algorithm with the extreme learning machine (ELM) algorithm with an average accuracy of 0.799m (Zou et al., 2013a). (Mahida et al., 2017) suggest that to carry out an ideal indoor navigation, an accuracy of less than 2m is required; our results comply with this premise, however, it could also be evidenced that for small spaces such as corridors and stairs an approximate accuracy of 0.5m is required to signal the correct location space to the users.

The proposed navigation system is designed to provide the user’s location in real time, or in the shortest possible time. In the tests performed, the user’s location is obtained in 9.7s on average, considering a space with direct visual between the BLE devices and the mobile device. Although the times are acceptable, in a navigation system oriented to people with disabilities the location estimation should be provided milliseconds in advance to notify of changes in space especially when the user is on the move (Murata et al., 2019), this is an aspect that should be improved of our current implementation. On the other hand, the effectiveness of the system should be evaluated by the end users, since omission of relevant aspects is possible; so far, only the aspects mentioned in previous studies have been emphasised.

5. CONCLUSION

In this research we propose an indoor navigation system aimed for people with disabilities. The proposal integrates CityGML and IndoorGML standards as the basic data providers for context information delivery and route planning, with user positioning based on BLE technology by a weighted path loss algorithm - WPL and a user interface that allows the user to interact with the system. From the work developed we obtained the following conclusions:

- The IndoorGML model supports the entire topological structure of the navigation network for routing, however, as it does not provide context information, it is necessary to integrate the CityGML model, not only as a visualization model but also as a supplier of context information.
- From the integration of BLE technology and the positioning technique, a position accuracy of 1.02m is achieved in a scanner without obstacles, an average error of 0.86m 1.06m for a 3D scenario with obstacles. Although the results are better than those achieved in other investigations, they do not allow the correct location the user in small size spaces such as stairs or corridors.
- From this research it could prove that it is possible to build a navigation system based on an architecture consisting of four basic components. (1) A network of sensors that emit the RSSI signal every 100m. (2) A database created in Postgres which stores parts of the CityGML and IndoorGML model classes from which the optimal route is queried and context information is provided. (3) An Android app that is responsible for calculating the location of the device from the received RSSI signal, receiving the user’s instructions, visualising the results in a 3D interface by means of the CesiumJS library and informing the user’s position and the characteristics of the location space by means of audios, an aspect mainly beneficial for visually impaired people. (4) A web server that allows communication between the application and the database.

REFERENCES


Kim, J. S., Yoo, S. J., Li, K. J., 2014. Integrating IndoorGML and CityGML for indoor space. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8470 LNCS, Springer Verlag, 184–196.


