A Safety Index for Smart Mobility using Real-Time Crowdsourced Data

Neale A. Smith
Dept. of Systems Engineering
The University of Texas at El Paso
El Paso, Texas, USA
nasmith2@miners.utep.edu

Adriana C. Camacho
Dept. of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
accamacho2@miners.utep.edu

Edgar J. Escobedo
Dept. of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
ejescobedo@miners.utep.edu

Jonatan M. Contreras
Dept. of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
jmcontreras2@miners.utep.edu

Oscar Mondragon
Dept. of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
oamondragon@utep.edu

Natalia Villanueva-Rosales
Dept. of Computer Science
The University of Texas at El Paso
El Paso, Texas, USA
nvillanuevarosales@utep.edu

Ruey Long Cheu
Dept. of Civil Engineering
The University of Texas at El Paso
El Paso, Texas, USA
rcheu@utep.edu

Victor M. Larios
Smart Cities Innovation Center
CUCEA, Universidad de Guadalajara
Guadalajara, Jalisco, Mexico
victor.m.lariosrosillo@ieee.org

***Abstract*— Smart Mobility is an important component of Smart Cities with most of the current approaches focusing on crash incidents for safety within Smart Mobility. The Safe Community System (SCS) aims to collect and provide information to city residents about events beyond crash incidents using mobile technology. The work reported in this manuscript aims to manage and provide information to residents in a meaningful way to support their decision making. This paper describes our efforts in extending an initial proof-of-concept of the SCS by establishing a Safety Index (SI)– a derived metric that aggregates the value of resident-submitted reports to generate real-time safety levels for streets within a city considering the lifespan and verification of these reports. The SCS mobile application has been refined to provide further information about a specific incident. The SCS updated design also proposes a Safe Path Algorithm (SPA) which is a modified Dijkstra's algorithm that uses the SI to compute a safe path for the resident. The goal of the SCS is to support resident's decision-making when choosing a route and thus fostering safety for Smart Mobility. Efforts like the SCS contribute to converting cities to Smart Cities.**

Keywords—Smart Cities, Smart Mobility, Smart City Metrics, Safety Index, Safe Path, ICT.

I. Introduction

This project aligns with Smart Cities' goals to create solutions that aim to improve the quality of life of city residents. Smart Mobility is one dimension of Smart Cities that focuses on efficient and effective mobility systems using technology [1]. Research on Smart Mobility has focused on traffic-related mobility [2]. Navigation tools, such as the one evaluated in [3], use crowdsourcing to collect and provide traffic incident reports but approaches to collect safety incidents are limited. The *Safe Community System (SCS)* aims to increase the community's mobility safety by addressing other types of incidents (i.e., criminal or suspicious activity) that enable residents to choose a path based on safety. Personal safety contributes to the quality of life of residents. Living and commuting in an area with a perception of low safety levels can affect the wellbeing of the residents. For example, according to a 2019 INEGI poll, 77.7% of residents in the state of Jalisco, Mexico, perceive a lack of safety [4]. This project was originally inspired by the need for real-time safety information around university campuses in Guadalajara, Jalisco with the engagement of residents, faculty, and industry in the design phase [5].

Currently, residents in Mexico may access public historical data such as the one provided by the "Instituto Nacional de Estadística, Geografia e Informatica" (INEGI) [6]. While historical data is useful, there is a need to support the decision making of smart mobility safety using real-time data. One way to collect real-time data is through crowdsourcing where residents provide information and share it with its community using resources such as mobile applications. Another advantage of real-time crowdsourced data over historical data is community engagement and building.

A binational interdisciplinary collaboration for Smart Cities research between faculty and students from Universidad de Guadalajara (UDG) and The University of Texas at El Paso (UTEP) has identified needs in the City of Guadalajara that can be addressed using Smart Cities solutions, with the additional goal of transferring these solutions to other cities, including El Paso. This international collaboration aims to train students from both universities, working in summer cohorts, with the knowledge and skills needed to convert cities to Smart Cities.

The initial design of the SCS and the proof-of-concept of its mobile application to enable the collection of crowdsourced data for incident reports were previously presented in [5]. This paper describes the new features of the SCS proposed by the research team of faculty and 2020 student cohort. The new features include the definition and calculation of the *Safety Index (SI)* using incident reports for measuring the safety of street segments and the lifespan of reports. In addition, the *Safe Path*

This material is based upon work supported by the National Science Foundation (NSF) under Grant No. 1658733 IRES: US-Mexico Interdisciplinary Collaboration for Smart Cities. This work used resources from Cyber-ShARE Center of Excellence, which is supported by NSF Grant number HRD-1242122. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

Algorithm (SPA) is proposed to use SI as a factor to compute a safe path for the resident. The SCS mobile application refined version enables residents to verify the accuracy of a report given by another resident.

The SCS integrates people and technology through crowdsourcing to provide meaningful information about safety for mobility purposes which aligns with the Smart Cities approach defined in [7]. In this project, residents are considered "sensors" since they are providing, verifying, and consuming the data generated in this system. The SCS leverages the concept of "participatory sensing" [8] since it relies on resident's engagement and participation. The research team assumes that residents will be incentivized to use the SCS application for self and community gain.

The SCS is a Smart City solution that uses *Information and Communication Technology (ICT)* to support the decision-making of city residents with the use of crowdsourced information. An enduring challenge to manage crowdsourced data is the verification of the data [9]. The SI and SPA employ resident-submitted reports for their calculations. Thus, the reports must be verified. To address this, the SCS design proposes a crowdsourcing approach to allow residents that are close to the location where the report has been created to confirm or deny the accuracy of the reports. These confirmations and denials of the accuracy of the reports are also included in the information provided to the residents. Each resident can confirm or deny the report's accuracy once. The SI and SPA aim to provide the most recent safety information to the resident. Thus, the lifespan of the crowdsourced data is also considered in this research. The overall **objective** of this project is to provide city residents with information about safety that supports their mobility through the city using mobile technology. The levels of safety are obtained by classifying crowdsourced data using the SI.

Through this research, the following three questions are answered: *How can crowdsourcing data contribute to a mobility safety index? How can safety paths be generated using a SI? and How can a safe path and SI be effectively communicated to the residents*?

II. BACKGROUND

This section reviews relevant concepts such as smart mobility, safety indices, and routing algorithms. In addition, this section provides a summary of existing mobile applications that provide similar functionalities.

A. Smart Mobility Approaches

As previously mentioned, the focus of current Smart Mobility approaches has been on traffic-related incidents [2]. Research regarding smart mobility and non-traffic related personal safety is sparse. Research on how to mitigate problems in mobility caused by many tourists in certain areas is provided in [10]. Work focusing on the connection between smart mobility and increasing accessibility for the elderly is provided in [11]. Other research provides measures for Smart Mobility; however, it does not include a measure for personal safety [12]. A bibliometric analysis confirms that Smart Mobility research mainly focuses on improving current transportation's environmental friendliness as well as mitigating issues such as congestion and traffic accidents; nonetheless, personal safety beyond traffic activities is not mentioned [13].

B. Safety Metrics

This section provides examples of mobility metrics. Metrics and indices are important for understanding the performance of a Smart City. Standard metrics for smart cities, however, have not yet been fully defined and adopted [14]. Few research articles examine measurement techniques for Smart Cities. There have been some attempts in measuring the performance of Smart Cities. For example, the United Nations initiative "United for Smart Cities" developed over 91 Key Performance Indicators (KPI's) for the evaluation of a smart city in the areas of Economy, Environment, Society, and Culture [15]. The only KPI that is directly related to personal safety is a violent crime rate. There are also some socioeconomic status indicators such as the Gini Coefficient and Poverty but these KPIs cannot directly inform residents of the safety levels of an area at any given time [15].

Smart Mobility indices for understanding or quantifying the safety of an area in a city have yet to be clearly defined and standardized. As mentioned earlier, past work has established measures for Smart Mobility but excluded measures for personal safety [12]. Park et.al. proposed detection of dangerous events by using data from different types of sources (i.e., camera, sensors, smartphones) where residents' mobile devices are used to pinpoint crime and alert authorities [16]. Moreover, Machine Learning techniques have been employed to analyze reports submitted by residents on a website designated for them to report problems in the city they have witnessed [17]. Approaches [16], [17] neither provide a measure for dangerous events nor addresses the problem from a mobility perspective. A system for detecting and predicting vehicle accidents based on historical data as well as sensor data has been proposed, even though this approach provides a mobility perspective, it does not consider personal safety [18]. Also, a system for detecting dangerous events in underground transport systems like a subway was proposed [19], this approach does consider personal safety and mobility but it is limited to a single mode of transportation and does not provide a safety indicator.

All these systems proposed methods for detecting unsafe situations by using sensor data, crowdsourced data, or video data, but they all work within specific domains and none contribute a metric that can serve as a SI of any given area of a city. To the best of our knowledge, these metrics cannot be used to create a SPA due to their domain specificity.

C. Existing ICT Approaches

During the literature review for the SCS, multiple applications that had similar features or ideas as the SCS were found. One important aspect of SCS is the use of real-time resident data. Similar applications that take this approach are Retio and Avisora. Retio is a website that classifies Tweets related to public safety by Artificial Intelligence algorithms [20]. The use of data provided by Retio was not considered because of its lack of geographical coordinates [20]. Avisora uses crowdsourced data to identify infrastructure issues in the city and alert users about them. The application verifies users and prioritizes reports to meet the user's needs. Though their number of users has declined over time and it has not been updated since 2017 [21].

An important proposed feature in SCS is the ability to create safe paths – paths that consider safety information. An application that uses crowdsourced data for the creation of safe paths is Chi Safe Path. This application is a web-based tool for the residents of the city of Chicago to identify hazardous zones in sidewalks from the accessibility perspective. It also provides a feature that creates paths that avoid zones with reported incidents [22]. The Chi Safe Path does not provide information on the specific incident reported and its current use is limited to Chicago.

The study of these applications and the need for comprehensive safety information identified by the binational group of researchers [5] facilitated identifying features needed to refine the SCS. The SCS includes some of the features provided by the current approaches and adds the definition and proof-of-concept of the SI, which considers several parameters of safety data, to better inform and guide residents while transiting their city. Table 1 provides a comparison of the features offered by SCS and the applications reviewed in this section.

TABLE 1 – COMPARISION OF FEATURES OF RELATED APPLICATIONS

	User Report	Safe Path Generation	Report Visualization (Map)	Safety Index	Report Verification
SCS	✓	✓	✓	✓	✓
Retio	✓	X	X	X	X
Avisora	✓	X	✓	X	✓
Chi Safe Path	✓	✓	✓	X	X

III. THE SCS DESIGN

The main purpose of the SCS is to provide real-time mobility safety information to residents as they transit in the city. This is achieved by collecting and providing information about safety incidents reported by residents. This information aims to support residents' decision making about paths. The previous version of the SCS reported in [5] included the proof-of-concept of a mobile application to create and display reports. A detailed description of new SCS features follows.

A. *Incident Reports*

The visualization of incident reports has been enhanced by sharing the report information at the street level to address privacy concerns. The exact location of a report is not shown on the map, but rather the street segment where the incident happened using the generalizations and suppressions privacy concepts [23]. The current categories for reporting include criminal activity, suspicious activity, infrastructure, and perceived risk. The SCS categories were refined to cover the most common reported incidents in Jalisco, Mexico [24] where the first scenario of this work is located. The SCS system allows incidents to be further refined into subcategories, e.g., the criminal activity may be classified into auto theft, and assault. Reports in the SCS map are displayed using color-coded markers for usability purposes. Each report now contains a *Severity Weight* (SW). The current version of the SCS mobile application displays the SW or the report on the map using the report's category (Fig. 1). The greater the SW of the report, the less safety of the affected area.

The following fictional reports were generated as SCS test scenarios to illustrate the management of reports and other SCS features such as the calculation of the safety index. Fig. 1 shows their visualization in the current SCS mobile interface. These testing reports were manually generated in the middle of the street to illustrate how privacy will be addressed in a future version of the system. The functionality to automatically place the marker of reports to the middle of the street currently is under development in the SCS.

The SW associated with each testing report is shown in {} brackets at the end of each report description:

1. Perception of risk, reported on 07-17-2020 10:15 a.m. {5}
2. Infrastructure-graffiti, reported on 07-17-2020 4:18 a.m. {5}
3. Criminal activity-theft, reported on 07-17-2020 10:10 a.m. {20}
4. Suspicious activity- unusual attention to facilities, reported on 07-16-2020 2:00 a.m. {10}
5. Infrastructure-sewage, reported on 07-17-2020 9:37 a.m. {5}
6. Perception of risk, reported on 07-16-2020 6:24 p.m. {5}
7. Infrastructure-electric system reported on 07-15-2020 3:08 p.m. {15}

B. *Verification of Reports*

The SCS's main source of data is the crowdsourced resident-generated reports. Residents create reports as indicated in Section III-A. Reports have different weights based on their category. In the current implementation, incidents are classified in the following categories with the corresponding weights: criminal activity {20}, infrastructure {5}, suspicious activity {5}, and perceived danger {10}. The report is then displayed on the map at the respective street segment and not using the street number.

The accuracy of reports is addressed in the SCS current version to foster trust in residents. A resident may input an erroneous report by choosing the wrong type of incident, misunderstanding the incident, misplacing the location, or even maliciously generating a false report. The SCS has new functionality in the mobile application to enable residents to verify the accuracy of reports by confirming or denying a report

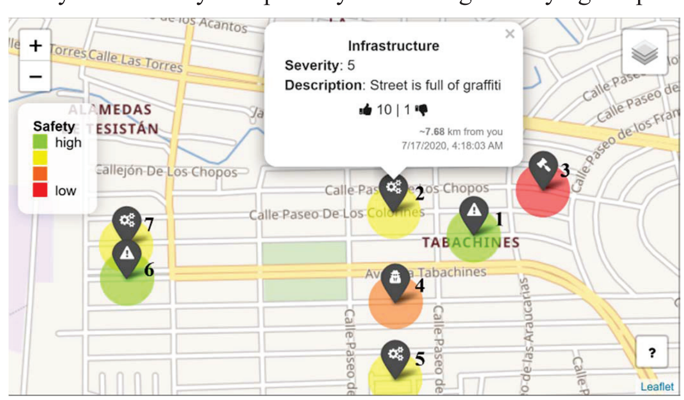

Fig. 1. Visualization of testing reports in the SCS mobile application with the numbers identifying the reports on the map and a popup

(i.e., pressing thumb-up/down). The SCS mobile application displays the number of confirmations and denials when clicking on a report map marker in addition to the category, SW, and description (See Fig. 1). For example, when the resident clicks on report #2, they will see that the report has ten confirmations and one denial. In a future version of the SCS, this verification information will be used to affect the weight of the SW by considering, for example, the number of confirmations/denials of a report within a fixed threshold. Current considerations include using the verification status to determine whether or not an incident report will be used in the SI calculations and/or displayed in the SCS mobile application. To preserve the integrity of this verification process, a feature under development will implement the restriction that only residents that are near the incident location will be able to confirm or deny the reports.

The SCS database has been designed to be updated every time a new report is submitted, confirmed, or denied in the system. This is an important feature in an application that promotes real-time access to safety information. In a future version of the SCS, when the database is updated, a new calculation of the SI will occur for updating the corresponding street-segment safety level.

There are further considerations when processing crowdsourced data. Multiple residents may generate a report that describes the same incident, instead of confirming an existing report which would create inaccurate indices. The design of the SCS has been refined to manage this scenario in a future implementation as follows. When the resident submits a report, the system checks the surrounding area for similar reports. If a similar report has been submitted by another resident, the system displays the previously generated report and asks the resident to determine if that report is the same as the new report. If the resident determines that the new report is the same, the new report is not saved and is managed as a verification of the previous report adjusting its weight. If the resident determines the new report is different, the new report is submitted to the database.

C. *Safety Index*

The refined SCS design defines a new derived metric called SI to represent the safety of a street segment by aggregating the SWs of the reports submitted by residents for that specific street segment. The SI of a street segment is space- and time-dependent and will be used in the SCS to calculate a safety path. The SI calculation considers the SW of the submitted reports, the frequency of reports, and the decay (i.e., life span) of reports within a street segment.

The SI calculation of a street segment is affected by its proximity to a report. Currently, if a street segment is within 40 meters radius of a report, its corresponding SI will be changed based on the SW of the report. A report may affect more than one street segment within the affected area. A street segment may be affected by many reports based on the affected area of these reports.

The SI calculation also considers the decay of reports by classifying reported incidents into *current reports* and *historical reports* to reflect the relevance of reports over time. The current reports include reports submitted within the last twenty-four hours. The historical reports include reports submitted after twenty-four hours and up to one year. The decay impacts the weight of the report by decreasing its value through time. Each report has an initial SW based on the type of incident reported, as defined in section III. The report's SW value is affected by a decay coefficient through time. of a report is affected by a decay coefficient. From now on, when describing formulas and algorithms, a street segment is referred to as a *link*.

The equation (1) defines how multiple reports, both current and historical, aggregate to a *Link Weight*. Each link has an initial weight (w_o) that represents the distance to transverse the link. The *Link Weight* also aggregates the SW values of current reports which are defined in equation (2). In addition, the *Historical Weight (HW),* described in equation (3), aggregates the weight of historical reports over time. The equation that defines how reports are aggregated into a *Link Weight* is:

$$Link\ Weight = w_0 + SW_1(t) + \cdots + SW_n(t) + HW(t) \quad (1)$$

$$SW_i(t) = a_i \times e^{-b_i(t-t_i)} \quad (2)$$

In equation (2), $SW_i(t)$ represents a report's decayed SW where i=1, …, n. The variable a_i represents the initial SW of the report, b_i represents the decay coefficient. t_i represents the time at which the new report affecting the link is submitted. An exponential decay equation is used because it is a common numerical model to represent continuous decay in many domains such as radioactive half-lives and continuous failure rates [25]. Using the testing scenarios, reports #6 and #7 displayed in Fig. 1 are aggregated into the link representing the street segment where report #6 is generated since this link is in the 40-meter threshold defined. Since the severity weights are 5 and 15, the total *Link Weight* is 20.

The decay coefficient can change depending on the nature of the incident. For example, in the test scenario, report #5 (infrastructure incident) may have a slower decay than report #3 (criminal activity) therefore their coefficient decays should be different. Decay coefficient values for each incident's categories have been initially defined for the decay formula. To demonstrate the decay with the testing scenarios, report #4 was reported at 2:00 a.m. Consider a new resident is accessing the map at noon. The initial SW of report #4 was 10, but through decay, its SW value at 12:00 p.m. is 0.5. This shows that report #4 has less relevance to the SI and the generation of safe paths due to its decay.

The historical reports have a much longer time span, currently predetermined in the SCS as one year. The HW represents the frequency and consistency of reports which have affected a link, even if no current reports are affecting the link at that time. The greater the frequency of reports in an area, the greater the HW. The HW also decays but much more slowly compared to the SW_i. The calculated HW is updated when there is a new report affecting the link. The equation for the *HW(t)* is defined as:

$$HW(t) = c \times wf \times e^{-d \times DF \times (t - t_0)} \qquad (3)$$

$$c = \frac{\sum reports}{Timespan} \qquad (4)$$

$$d = \sum reports \qquad (5)$$

The weight factor (*wf*) and decay factor (*DF*) are variables to control the HW. Further research is needed on how to calibrate these variables and determine the frequency to recalculate SW and HW to have real-time information used to produce safe paths that align with the residents' needs.

The equation to calculate the SI converts *Link Weight* values to values between 100 and 0 to be better understood by the residents. A value of 100 represents a safe street segment, and values closer to 0 represent less safe street segments. The equation to convert the values generated from Eq. (1) is:

$$SI = \ 100 \times e^{-0.1*(Link\ Weight - w_0)} \qquad (6)$$

If there are no reports in a street segment (according to the information provided to the SCS), the segment is considered safe with an SI value of 100. In a street segment with incident reports with a high SW, such as report #3 in Fig. 1, the SI will have a low value of 13.5.

The implementation of calculating the SI in the SCS server is in progress. The current SCS mobile application shows a graphical representation of the initial SW using different colored circles as shown in the legend in Fig. 1 and Fig. 2 (left).

D. Safe Path Algorithm and Index

The SCS has been designed to generate safe paths from the resident's location to the resident's desired destination using the SPA. The paths generated by the SPA use safety as the routing criteria. The SPA generates safe paths by avoiding links with low SI values. The more safe a path link is between the origin to the destination, the more the system is incentivized to use that link. When a link does not have reported incidents, the system defaults to use the link travel distance in the SPA.

The SPA modifies the Dijkstra's algorithm provided in [26] by adding *Link Weight* values in addition to the link distance values. The SPA designed for the SCS contains four data lists (storage structure): *visited nodes*, *unvisited node*, *distance from the source,* and *previous node*. Each list is a vector of dimension equal to the total number of nodes that can be visited to create a path from the origin to the destination.

Dijkstra's algorithm contains the following steps: (1) The algorithm starts from the source node and ends when it reaches the destination node. Since the distance from the source node to itself is 0, the algorithm places a 0 in the source node element in the list *distance from the source* and leave its list *previous node* empty. Given that no other information is available at the first step, the algorithm sets all other elements in the *distance from the source* list to infinity; (2) The algorithm visits the node with the smallest distance in the list *unvisited list*, by setting the source node as the *distance from source* list to the distance from the source node to each of them, following the current node path; (3) The algorithm selects the node with the smallest *distance from the source* list and sets this node as the current node; (4) The steps 2-4 are repeated until the destination node is visited.

Dijkstra's algorithm produces a total path length of the shortest path. Similarly, the SPA produces a Safe Path Index (SPI) which measures the safety of the safest path. This is done with the following equation:

$$SPI = 100 \times e^{-\frac{-\sum(Link\ Weight_i - w_{0i})}{20}} \qquad (7)$$

The summation represents the sum of the links that the path takes to get to the destination from the source.

As mentioned before, the SPA considers safety as the main criteria to create the path from Point A to Point B. If there are no incident reports in the area or if different paths have the same SPI, then the SPA selects the shortest path.

The selection of the safe path is explained with an example. Consider that a resident named Joe Doe wants to travel from Point A to Point B (Fig. 2, left) making use of the Safe Path feature of the SCS. From the multiple possible paths the SCS can offer, there are two shortest paths. The first shortest path (with street segments colored in red) is omitted because of report

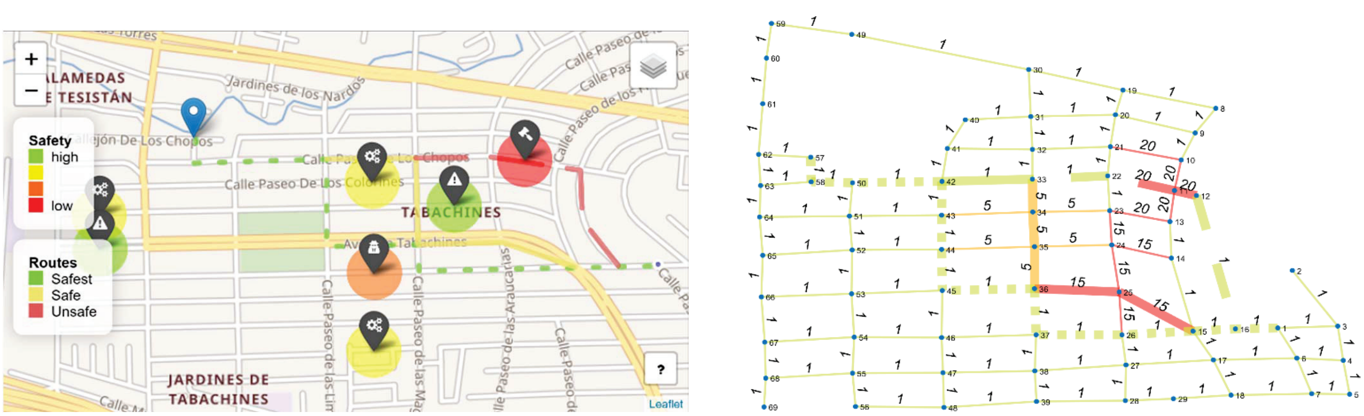

Fig. 2. The reports added to the graph with its initial SW and decay coefficient affect the link weights. The affected link weights influence the fastest paths in the simulation (right) and are shown as a mockup (left) using the colors red (unsafe), yellow (safe), and green (safest).

#3 (identified in Fig. 1) with an SW value of 20. The second shortest path is avoided given the recent reports #1 and #2 (also identified in Fig. 1) that are close to the path, with segments depicted in yellow lines in Fig.2. The third shortest path depicted as the green line in Fig. 2, is offered to Joe Doe since it is not affected by any incident reports. Please note that other current reports in the SCS (i.e., reports #4, #5, #6, and #7 in Fig. 1) are not relevant to this scenario because the possible paths are not crossing the area affected by these reports. The total distance in this scenario increases as the system prioritizes safety over the travel distance. In particular, the suggested path visits three more segments than the shortest path. However, the suggested path has a composed SPI of 100 compared with the shortest route that has a composed path safety index of 13.5.

An agent-based simulation was conducted in MATLAB. The goal was to verify the behavior of the SI by using parameters for agent locations on the map, types of reports, the frequency of report generation, and the report verification. By modifying the frequency and number of the reports, the simulation assisted in understanding the behavior of the SI values and the effect of report decays on the SI values. Moreover, another goal of the simulation was to verify the use of the SI to calculate the SPA and identify the safest path. The safest path changed based on the number of reports added, the report's SW, and report decay. The simulation was successful in assigning safer links with a higher SI and providing the safest paths to a destination. In addition, in the simulation, the SI can be color mapped to the links directly where green segments represent safe links and red segments represent less safe links (Fig. 2, right).

IV. The SCS Implementation

A. System Architecture

The SCS is divided into 3 main components (Fig. 3). The *first component* is the SCS mobile application that serves as the interface between the SCS and the resident. The SCS functionality to enable the submission and visualization of reports was implemented in the previous SCS version described in [5]. The current version of the SCS has been further developed to include the following new features: defining a report's SW which is automatically assigned when reports submitted, verification of reports, the addition of user-friendly guidelines in the map, filtering of reports based on category, visual clustering

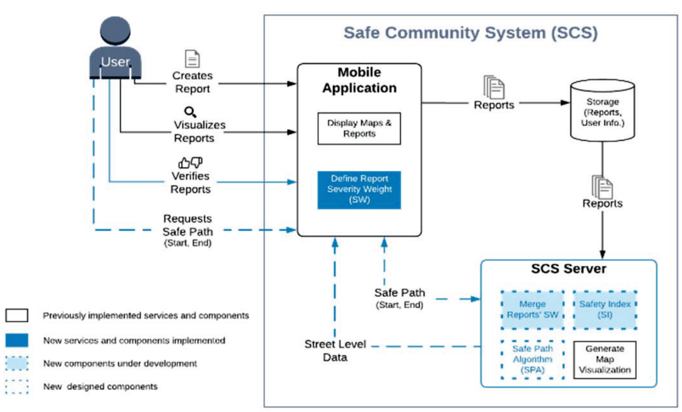

Fig. 3. SCS Main Services and Components

of reports when zooming in/out of the map, and displaying of report's additional information such as SW, number of reports confirmations and denials, the distance of the incident from the resident, and report time. The *second component* of the SCS is the storage of incident reports and resident credentials using MongoDB [27], which was not modified from the previous version. The *third component* is the SCS server, which in the current version handles the generation of map visualizations including the aggregation reports using Leaflet API [28]. The implementation of the SW for aggregating reports, the SI at the street-level, and the SPA for the generation of safe paths is under development at the time of writing this paper.

B. Safe Path Algorithm Component

The SPA component is currently under development and is evaluating different tools such as QGIS [29] and LeaftLet API [28] for the management of street-level data. The SCS server will request street-level information of areas of interest, initially UTEP and UDG campuses. Said information could come from digital maps (i.e., shapefiles or manually created street-level data) to be loaded into the database.

The SCS was designed to generate the *Link Weight* of street segments using the proposed algorithm and update it into the SCS database. In a future SCS version, this feature will be implemented to enable a resident to request a safe path using the SPA algorithm. The suggested safe path will be sent to the SCS Safe Path display generator which is currently envisioned to use the Leaflet Routing Machine [30] provided by the LeafLet API, but other tools may be tested for this purpose.

V. Discussion

The SCS aligns with the principles of Resiliency, Security, Scalability, Interoperability, and Modularity defined for Smart Cities solutions [31]–[33] as follows.

- *Resiliency*: The ability to recover from failure quickly is an important characteristic of systems [31]. The SCS promotes resiliency as future implementations will enable the creation of reports even when there no active connection to the server is available, once a connection is regained, the report is submitted.

- *Security*: This characteristic aims to prevent unauthorized access to data and services in the system [33]. The SCS implements security features by using the NodeJS Crypto module [32] for the encryption and decryption of the user information such as a password and report coordinates. Salt Hash [34] is used as an extra precaution for the safe management of sensitive residents and report information. This is done by the mixture of said information with a random string, making it harder to be deciphered in case of an attack.

- *Scalability*: In this work, this concept is interpreted as the ability of the SCS system to be functional with a large number of residents. The use of third-party tools such as the LeafLet API and MongoDB and the processing of services on the server-side allows the SCS to scale with a larger number of residents.

- *Interoperability*: This characteristic refers to the ability of a system to exchange data across devices [31]. The

SCS complies with this principle by enabling the exchange of data between related applications such as QGIS [29].

- *Modularity*: In this work, this characteristic is interpreted as the ability to allocate similar services into modules that can work independently. SCS aligns with this principle by having low-coupled modules for the front-end services, which are the SCS mobile application, and back-end services, which are MongoDB, NodeJS [32], and the SCS server.

The new SCS version focused on generating a metric to better inform residents about safety levels of areas they transit to support Smart Mobility, thus the SI and SPA were proposed. For these metrics, an important factor is the weights assigned to each report. These weights are currently assigned based on the category of the incident being reported. These categories will be refined once a beta version of the system is tested.

Given that the SCS relies on crowdsourcing data, verifying reports is an important feature of the system to foster the trust of residents on the information being provided. One proposed approach is to consider all reports accurate and having other residents to report as inaccurate. The other option is to consider all reports inaccurate and enabling other residents to assess them as accurate. The current SCS mobile implementation allows the verification of reports as accurate or inaccurate and the manipulation of report verification within the SCS is currently under development.

An initial approach for the aggregation of reports was to use the individual report value for the SI. Introducing a lifespan to reports captured better real scenarios where a recent incident may be more important than a past incident.

The last major decision for the SCS was how to obtain the street-level information required for the generation of paths with the SPA. The Leaflet API did not provide such information. The first option is the use of shapefiles and a system such as QGIS to manipulate these files. A second option involves the manual creation or importing of street-level information to be stored in the SCS database.

The privacy of data was considered in the SCS design and implementation. Thus far, the SCS includes the mobile application's current location in the incident report, encrypts them in the database, and only uses them for internal processes - the exact location of incidents is not shown on the map. Ongoing work to address privacy is the use of generalizations and suppressions [23] to display reports at the street level in a fixed position depending on the category of the report (e.g., middle of the street, left corner of the street). This technique aims to balance the tradeoff between privacy and utility by keeping the report's location private but usable for finding street segments affected by a report.

The SCS was designed to be used in both Guadalajara and El Paso, and thus considered the different needs, data sources, and services provided in these cities. The collaboration between the two universities was key in the design and development of a system that can potentially be used in these two cities.

VI. Conclusions

The SCS aims to provide city residents with real-time information about reported safety incidents to support their decision making. The creation of a SI and safe paths illustrates how this information can support Smart Mobility in urban areas. A decay model was proposed to aggregate data from multiple reports and capture how the relevance of reports decreases through time. SCS makes use of crowdsourced data and allows residents to verify the accuracy of the report's information to build trust in the use of the application. Applications such as the SCS that provide real-time data have the potential to provide better services to city residents and ultimately improve their quality of life and contribute to converting cities to Smart Cities.

VII. Future Work

An important factor that affects the generation of safe paths is the mode of transportation. Some incidents (e.g., a pothole) may have less impact on a resident using a bike than another using a car. In a future iteration, the SCS aims to customize the mobile application and calculations to recommend residents' paths depending on their mode of transportation, which impacts the SW of reports. Considerations for data privacy included in the discussions will be incorporated in future versions of the SCS.

The testing of the SCS beta version is planned to take place after key features such as calculation of SI and safe path generation are fully implemented in the system. University campuses in the cities of Guadalajara and El Paso Texas will be used to evaluate the use of SCS functionality using different languages, metric systems, infrastructure, and cultures once social distancing regulations are lifted.

A future research direction includes analyzing the use of pictures provided by residents. While images uploaded to the SCS by residents can increase the reliability in reports, images can be misused or disclose unintended information. Ongoing research on the use of images for the SCS includes the use of algorithms for blurring faces or car plates for privacy purposes.

The ongoing work on simulating the behavior of the SCS provides insights on the expected results once the system is deployed, including the function of a decaying weight, assigning SW to links, and display of SI on the street segments. Further research in the simulation includes testing of parameters such as decay coefficients, severity weights, and weight factors to find a set of parameters that will assist in calibrating the SCS.

Acknowledgments

The authors would like to thank the previous two cohorts of the IRES (US-Mexico Interdisciplinary Collaboration for Smart Cities) program, in particular, Ubaldo Castro for his technical advice. The authors would also like to thank Judith Garcia, Fernie Briones, Elisa Jimenez-Todd, and Nour Moraveg for their contributions to the current version of the SCS. The authors would also like to thank UDG faculty Dr. Rocio Maciel Arellano, Dr. Jose Antonio Orizaga Trejo, and Dr. Jesus Raul Beltran Ramirez for their insights and guidance for this project.

REFERENCES

[1] C. Benevolo, R. P. Dameri, and B. D'auria, "Smart mobility in smart city," in *Empowering Organizations*, Springer, 2016, pp. 13–28.

[2] G. Cledou, E. Estevez, and L. S. Barbosa, "A taxonomy for planning and designing smart mobility services," *Gov. Inf. Q.*, vol. 35, no. 1, pp. 61–76, 2018.

[3] M. Amin-Naseri, P. Chakraborty, A. Sharma, S. B. Gilbert, and M. Hong, "Evaluating the reliability, coverage, and added value of crowdsourced traffic incident reports from Waze," *Transp. Res. Rec.*, vol. 2672, no. 43, pp. 34–43, 2018.

[4] "Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE) 2019," *INEGI*. https://www.inegi.org.mx/programas/envipe/2019/ (accessed Jun. 17, 2020).

[5] C. V. Chavez *et al.*, "Towards Improving Safety in Urban Mobility Using Crowdsourcing Incident Data Collection," in *Proceedings of the fifth IEEE International Smart Cities Conference (ISC2)*, 2019, pp. 626–631, doi: 10.1109/ISC246665.2019.9071714.

[6] N. I. of S. and G. INEGI, "National Institute of Statistics and Geography. INEGI." https://en.www.inegi.org.mx/ (accessed Jun. 17, 2020).

[7] N. Villanueva-Rosales *et al.*, "A collaborative, interdisciplinary initiative for a smart cities innovation network," in *2015 IEEE First International Smart Cities Conference (ISC2)*, Oct. 2015, pp. 1–2, doi: 10.1109/ISC2.2015.7366179.

[8] M. N. K. Boulos *et al.*, "Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, OGC standards and application examples," *Int. J. Health Geogr.*, vol. 10, no. 1, pp. 1–29, 2011.

[9] S. Meek, M. J. Jackson, and D. G. Leibovici, *A flexible framework for assessing the quality of crowdsourced data*. AGILE Digital Editions, 2014.

[10] P. Mrazovic *et al.*, "CIGO! Mobility management platform for growing efficient and balanced smart city ecosystem," 2016, pp. 1–4.

[11] R. Battarra, F. Zucaro, and M. R. Tremiterra, "Smart Mobility and Elderly People. Can ICT Make the City More Accessible for Everybody?," *TeMA-J. Land Use Mobil. Environ.*, pp. 23–42, 2018.

[12] N. N. Aini and H. Amani, "Indicators to measure smart mobility: An Indonesian perspective," 2017, pp. 81–85.

[13] E. J. Tomaszewska and A. Florea, "Urban smart mobility in the scientific literature—bibliometric analysis," *Eng. Manag. Prod. Serv.*, vol. 10, no. 2, pp. 41–56, 2018.

[14] R. W. S. Ruhlandt, "The governance of smart cities: A systematic literature review," *Cities*, vol. 81, pp. 1–23, 2018.

[15] "United 4 Smart Cities," *ITU*. https://www.itu.int/en/ITU-T/ssc/united/Pages/default.aspx Future Generation Computer Systems, 107, 871-882. (accessed Jun. 17, 2020).

[16] H. Park, E. Kwon, E.-S. Jung, H.-W. Lee, and Y.-T. Lee, "Heterogeneous data based danger detection for public safety," in *Proceedings of the 36th IEEE International Conference on Consumer Electronics*, 2018, pp. 1–2.

[17] Y. Lin, T. Mine, K. Yamaguchi, and S. Hirokawa, "Estimation of Danger Signs in Regional Complaint Data," *JOIV Int. J. Inform. Vis.*, vol. 2, no. 4–2, pp. 342–348, 2018.

[18] H. Park, E. Kwon, S. Byon, E.-S. Jung, Y.-T. Lee, and G.-Y. Kim, "Multi-log analysis of vehicle accidents for public safety services," in *Proceedings in International Conference on Information and Communication Technology Convergence (ICTC)*, 2018, pp. 1040–1042.

[19] M. Spirito, C. S. Regazzoni, and L. Marcenaro, "Automatic detection of dangerous events for underground surveillance," in *Proceedings in IEEE Conference on Advanced Video and Signal Based Surveillance*, 2005, pp. 195–200.

[20] "Como Funciona," *Retio*, Jun. 17, 2020. https://ret.io/docs/#como (accessed Jun. 17, 2020).

[21] "App Scholars," *Avisora*. https://www.appscholars.com/app/com.agavelab.avisoraapp/Avisora (accessed Jun. 17, 2020).

[22] "Chi Safe Path," *Chi Safe Path*. http://www.chisafepath.com/about (accessed Jun. 17, 2020).

[23] V. S. Iyengar, "Transforming data to satisfy privacy constraints," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, Jul. 2002, pp. 279–288, doi: 10.1145/775047.775089.

[24] "Instituto de Informacion Estadistica y Geografica de Jalisco." https://iieg.gob.mx/ns/ (accessed Jul. 24, 2020).

[25] [25] M. Finkelstein, *Failure rate modelling for reliability and risk*. Springer Science & Business Media, 2008.

[26] J.-C. Chen, "Dijkstra's shortest path algorithm," *J. Formaliz. Math.*, vol. 15, no. 9, pp. 237–247, 2003.

[27] "The most popular database for modern apps," *MongoDB*. https://www.mongodb.com (accessed Jun. 17, 2020).

[28] "Leaflet API reference," *Leaflet*, Jun. 17, 2020. https://leafletjs.com/reference-1.6.0.html (accessed Jun. 17, 2020).

[29] "QGIS Documentation." https://www.qgis.org/en/docs/index.html (accessed Sep. 10, 2020).

[30] "Leaflet Routing Machine," Jul. 27, 2020. http://www.liedman.net/leaflet-routing-machine/#about.

[31] V. M. Larios, L. Gomez, O. B. Mora, R. Maciel, and N. Villanueva-Rosales, "Living labs for smart cities: A use case in Guadalajara city to foster innovation and develop citizen-centered solutions," 2016, pp. 1–6.

[32] "Node.js v14.6.0 Documentation," *Node.JS*. https://nodejs.org/api/crypto.html (accessed Jul. 27, 2020).

[33] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Addison-Wesley Professional, 2003.

[34] D. Arias, "Adding Salt to Hashing: A Better Way to Store Passwords." https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/ (accessed Jul. 24, 2020).