Data sensing and dissemination framework for smart cities

Atif Manzoor*, Constantinos Patsakis*, Jessica McCarthy*†, Gabriel Mullarkey†,
Siobhán Clarke*, Vinny Cahill*, and Mélanie Bouroche*
*School of Computer Science and Statistics, Trinity College Dublin, Ireland.
Email: firstname.lastname@scss.tcd.ie
†Energy and Sustainability Lab, Intel, Ireland
Email: firstname.lastname@intel.com

***Abstract*—Participatory sensing, which enables citizens to collect and share data, can be helpful to design and develop useful applications in the domain of environmental monitoring, transportation, and health-care. However, the data collection process for such applications involves dealing with a variety of data sources, ranging from fixed environmental sensors to mobile human sensors, generating and presenting data in different formats. The citizens' engagement in sensing campaigns creates additional requirements to take care of their privacy concerns and motivate them to share data. Applications also need to compute the trustworthiness of the source of information to ensure the high quality of the collected data. Consequently, successfully dealing with these issues and deploying an application in an urban environment becomes a challenging task.**

In this paper we present a framework that uses fixed and participatory sensing to collect data from heterogeneous sources and presents a uniform interface to disseminate data to concerned applications according to their data requirements. We used our framework to develop a participatory sensing based smart phone application enabling and motivating the citizens to report positive and negative urban environmental behaviours. We report the results of our ongoing field trial. One of our findings is that different users contribute information about different environmental issues with different intensity. Moreover, we find that the current user participation to collect and share data follows the power law.

I. Introduction

With the persistently-increasing rate of urbanization, it is expected that 6.3 billion people will be living in cities, towns, and conurbations in 2050 [1]. This unprecedented increase in urban population will pose new challenges to provide urban services in the domains of energy, infrastructure, mobility, and economy to provide a sustainable and liveable place for citizens [2], [3]. Concurrently, recent advancements in information and communication technology, such as the deployment of the fourth generation of the mobile phone communication standards, miniaturization of the computing devices, and ubiquitous computing have resulted in cities being instrumented with million of sensors accessible over the Internet [4]. These sensors may be controlled by individual citizens, such as sensors embedded in smart phones; owned by private companies, such as closed-circuit television cameras coupled with dedicated fixed sensors; deployed by city management authorities and utility companies, such as traffic detectors and smart energy and water meters; or publicly shared, as seen in social media [5]. Advancements in urban and community sensing techniques, such as participatory sensing and opportunistic sensing, can take advantage of the finely-grained instrumentation of our cities and capture big data [6], [7].

Data generated from the plethora of diverse sensors may be stored in data repositories or distributed in real-time. Data may also include short time concern data, such as data about disruptive events and sensor meta data. These data sources may generate data in different representations, such as multimedia contents, Web 2.0 data, and text, modelled in structured and semi structured data formats. Applications may also have to collect the context of the data sources and evaluate their reputation and trustworthiness to dynamically connect with them to collect high quality data. They also have to collect human social interactions and behavioural data to take care of users' data sharing preferences and privacy concerns. Taking care of all of the aforementioned issues and using urban sensing sources to their full potential is a formidable task for a common application. Some applications have been redundantly using the computation power, storage resources, and energy to each address these challenges. Indeed, resources spent on dealing with these issues may prevent the applications from concentrating on their main functionality. A framework that can collect data from diverse sources of information and provide a uniform interface to the applications to connect with it has the potential to really ease the rapid design and development of urban applications.

In this paper we present CityWatch - our framework enabling the collection and dissemination of urban data. Figure 1 shows some of the main features of our framework. Along with fixed sensing we also use opportunistic and participatory sensing to collect urban data. We incorporate open government data in our system. We transform the data collected in different formats and presentations to a uniform XML format. We also present a participatory sensing based application built on top of our framework, enabling citizens to collect and share good and bad environmental behaviors in the city. We incorporate gamification aspects in our mobile application to incentivize the citizens to share their data. We present the preliminary results of our ongoing user trial. Currently, more than fifty participants have been using our application to collect and share data. We found that their data sharing and interaction behavior follows the power law.

The rest of the paper is organized as follows. Section II provides an overview of the related participatory sensing frameworks. Section III describes our urban data sensing and

MOBILWARE 2013, November 11-12, Bologna, Italy

DOI 10.4108/icst.mobilware.2013.254260

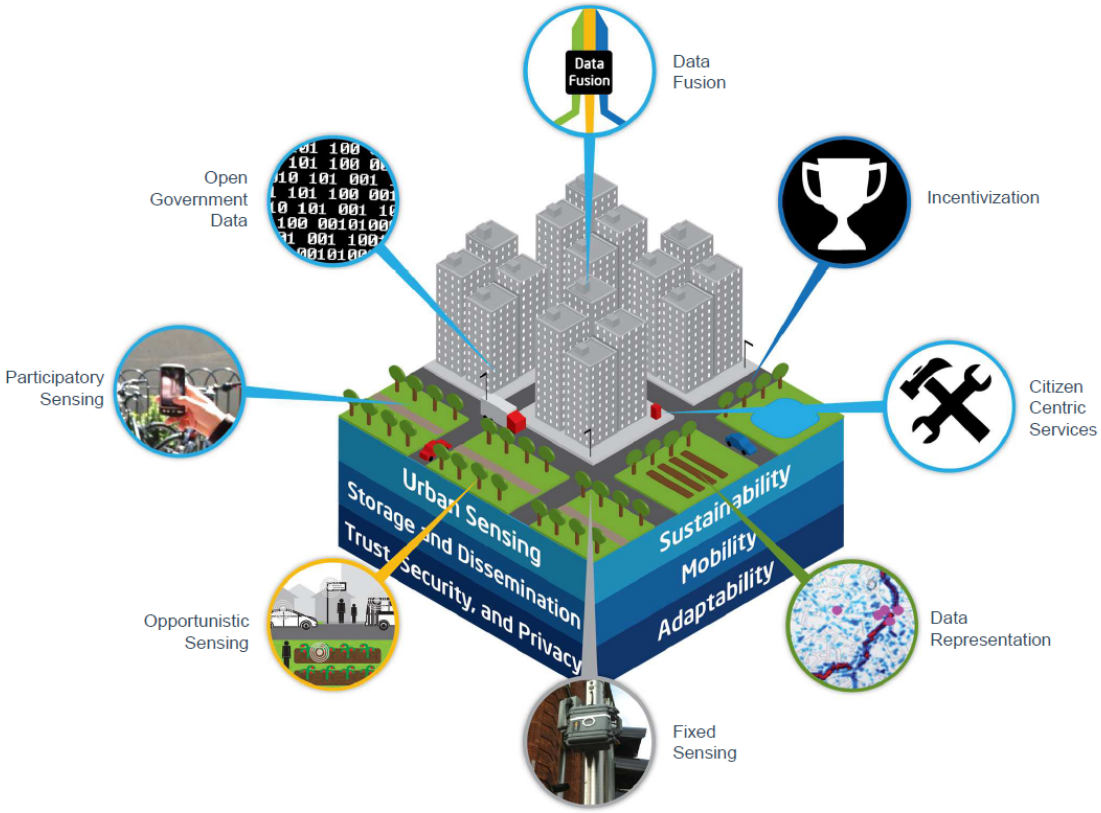


Fig. 1: Main features of the CityWatch system

collection framework for smart city - CityWatch. Section IV presents the details of our implementation. Section V presents the user trial and our results. Finally, we conclude the paper in Section VI.

II. RELATEDWORK

The availability of a number of sensing systems on smart phones had made them an important tool to collect data from social and urban environments [8]. Recently many applications have been developed to take advantage of these ubiquitously-available sensing systems to collect data and provide useful services in the domains of urban life [9], [10], environment sustainability [11], [12], air quality and noise monitoring [13], [14], and health care [15], [16]. However these applications directly connect to the smart phone sensors to collect data for their own use [17]. It is a challenging task for application developers to deal with different kind of communication protocols and sensing systems [18]. They may also have to learn different languages to develop software for different platforms. Consequently, a framework to perform those tasks for the application developers [19] is expected to be highly invaluable. Few research efforts have targeted the provision of a generic data collection and dissemination framework for rapid application development. In this section, we present the comparative analysis of those research efforts with our proposed system.

Trossen et al. [20] presented a platform, NORS, that allows mobile devices to be the part of sensing frameworks and work as gateways to forward sensor data. NORS is based on the publish-subscribe communication paradigm. However, unlike our work they have not provided the details of their interface to enable sensors to send data and applications to query data from their middleware. They have also been using only fixed sensors, temperature, humidity, pressure, and dew point in their test bed implementation of their platform, whereas, along with fixed sensing, our platform also enables users to contribute data.

Reddy et al. [21] propose a network service architecture for participatory sensing. They emphasize that participatory sensing participants selection, incentivisation, and tasking for effective data collections are the main challenges of a participatory sensing framework. They stress that a sensing framework should pay special attention to the evaluation of the trustworthiness of the data sources and the protection of the privacy of sensing participants. However, they have not implemented

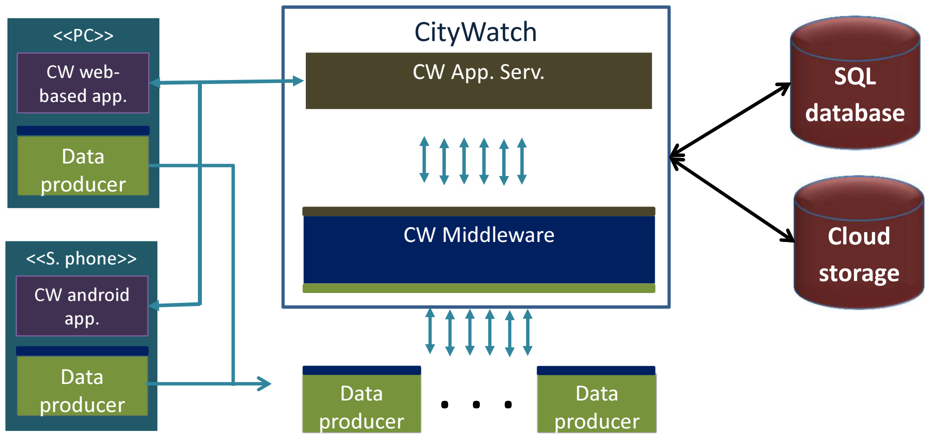


Fig. 2: The CityWatch architecture

their proposal.

Tuncay et al. [22] presented a distributed framework to select sensing participants and collect data. They used profile-cast [23], a behaviour-oriented protocol that aims to send messages to the nodes matching a certain target profile, to select sensing participants suitable for a given data collection campaign. They propose to use data sinks in a particular area to collect data. Subsequently, the data is conveyed to the organizers of the sensing campaign. However, their framework targets a direct sensing campaign for each data consumer. They did not make provision for the collected data to be distributed to multiple applications in their framework.

Sheng et al. [24] targeted the provision of sensing services using mobile phones via a cloud computing system. They aimed to support different mobile phone based sensing applications through their service. They also discussed different design choices for such service and emphasized that energy efficiency and an effective incentivisation mechanism for sensing participants are the main requirements of such a service. They have also discussed their hypothetical models to achieve those capabilities. However, they have not presented any information about the design and implementation of their system.

Joki et al. [25] presented a framework for participatory data collection on smart phones - Campaignr. The framework enables individuals and groups of people interested in collecting urban data to initiate a data collection process. However, the framework was specifically developed for Symbian S60 3rd Edition phones. Consequently, only the owners of that particular smart phone can take advantage of their framework. The framework did not allow a generic data consumer or smart phone application to collect data according to their own requirements.

Rather than proposing a complete data collection and dissemination framework, some research efforts have also targeted addressing a particular issue in participatory sensing. A special emphasis has been put on preserving the privacy of sensing participants [26], [27] and evaluating their trustworthiness [28], [29]. Other work has also been undertaken to characterize the sensing systems and ensure the collection of high quality data [30]. Although these solutions can be used to improve the performance of a generic data collection and distribution framework, they are not directly related to our work. Existing research literature lacks a generic data collection and dissemination system.

III. CityWatch: System description

This section presents an overview of the CityWatch urban data collection and dissemination system and the CityWatch application. Figure 1 illustrates some of the salient features of CityWatch (CW). CW uses modern urban sensing techniques to collect city data. Along with traditional fixed sensing, those techniques also include participatory sensing and opportunistic sensing. Participatory sensing uses the proliferation of smart phones and recent advancements in communication technologies to involve citizens to share their data. It is an important tool to collect data at a low level of granularity, for example in the case of unforeseen disruptive events, such as pluvial flooding. Instrumenting all parts of the cities to gather data about such unpredictable events would be very costly. Opportunistic sensing also allows CW to dynamically connect to mobile sensors to collect data. Along with these sensing techniques, CW is also connected to open government data sources to gather archived city information, e.g., the location of recycling centres in the city.

While collecting and sharing urban data, citizens may also

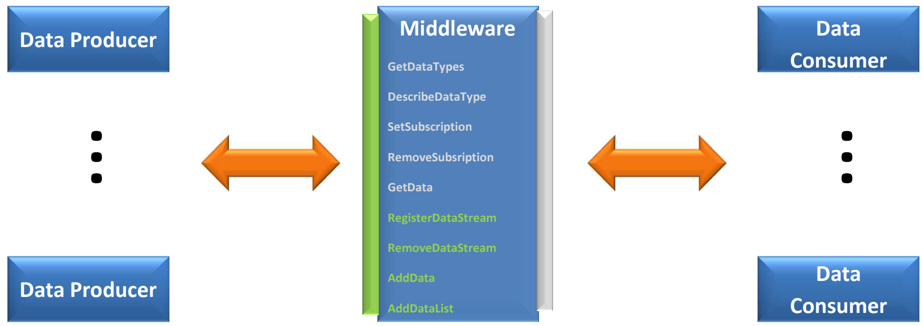


Fig. 3: The CityWatch middleware interface for data producers (such as sensors) and data consumers (such as applications)

expose a lot of private information, such as their location and that of others, images, and voices. This private information can compromise users security and privacy. Furthermore, the awareness of these circumstances might prevent users from sharing useful urban data for the public benefit. CW along with enabling users to have full control of their private information, also incentivises them to share information by gamifying the data collection campaign. CW also creates awareness among its participants about the public benefits of sharing their information. Once urban data is collected using different sensing techniques, it is fused together to make sense of that data. Finally, the data is presented to application users.

Figure 2 shows the CW system architecture. The CW middleware and the CW application server are the two main components of the CW framework. The CW middleware works as an intermediary layer between the data producers and the data consumers and provides an interface for data collection and dissemination. The CW application server manages the CW application deployed on the WWW[1] and smart phones, and adds an abstraction layer between the application and the middleware. The CW WWW version of the application can only be used to view data, whereas the smart phone version can be used to both view and share data. In the remainder of this section we present more details about these components.

A. Middleware

The CW middleware is a layer of interoperable and scalable software components that enable smart city applications to discover and access urban sensor resources. Figure 4 shows the details of these components.The CW middleware interface for smart city applications or data consumers allows them to interact with sensor resources in a well-defined, systematic, way. Similarly the CW middleware interface for data producers or sensor resources enables them to share data resources. Figure 3 shows the CW middleware interface for data producers and data consumers. In the subsequent paragraphs, we explain the CW middleware components and the CW middleware interfaces in details.

The *SensorManager* is responsible for managing data producers in CW middleware. Data producers include the urban entities that produce data, such as fixed sensors and mobile sensors. The CW middleware allows data producers to

[1]http://www.citywatch.ie/

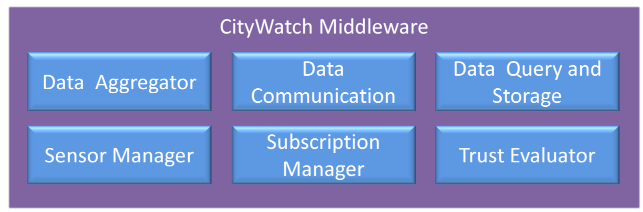


Fig. 4: CityWatch middleware components

share urban data through a well-defined interface as shown in Figure 3. A data producer that wants to connect to the CW middleware to share data will call *RegisterDataStream* to let the middleware know its intention to share data. It also sends information about the data types it can provide, their representation format, the frequency of updates, and the measurement unit, as well as a call back function to the middleware. The CW middleware calls this function to pull data from the data producer when required. Data producers can also use the *AddData* and *AddDataList* interfaces to push data to the middleware. If a data producer decides to stop sharing data, it can call *RemoveDataStream* to un-register the data stream.

The *SubscriptionManager* keeps track of the consumers' data subscriptions and sends data to them. Data consumers include the entities that consume data to perform different functionalities. These entities can be smart city applications or other data services. The CW middleware provides an interface for data consumers to get information about available data in the middleware and set subscriptions for that data. Figure 3 shows the details of that interface. Data consumers can call *GetDataType* to get information about available data types in the middleware. Data consumers can further call *DescribeDataType* to query meta-data, such as data models. Finally, data consumers can use the *SetSubscription* interface to set data subscriptions. Data consumers will specify the requested data type, update frequency, and a call back function to receive data. Data consumers can also call *GetData* to receive data.

Once data is collected from different sensing systems, the *Trust Evaluator* quantifies the trustworthiness of the data producers as the belief in their capabilities to collect high-quality data as described by Manzoor et al. [29]. The *Data Aggregator* detects and removes duplicate and conflicting information. The CW middleware supports push- and pull-based data communication techniques. Data producers can push data to middleware or set a call back function to allow the middleware to pull data when needed. Similarly data consumers can also use the CW middleware interfaces to pull data or set subscriptions to allow the middleware to push data. The *Data Communication* module is responsible for providing this functionality. The *Data Query and Storage* module is responsible for storing data in the SQL data store and to query data.

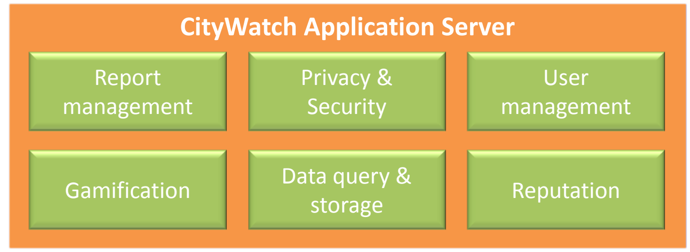


Fig. 5: The components of CityWatch application server.

B. Application Server

The CityWatch middleware provides the data collection and dissemination functionality for multiple applications. One of the applications that we have developed on top of this middleware is the CW application. The CW application server provides additional functionality on top of the CW middleware to the CW application. This functionality includes the user management, group management, report management, gamification and information anonymization. Figure 5 shows the main components of the CW application server.

The *Report Management* contains the necessary methods for storing, processing and retrieving the user submitted reports. The *Privacy and Security* module provides secrecy to the exchanged data, along with image anonymization either by removing the traced faces or by stripping the image metadata. The *User Management* component provides general management of the user data, groups and areas. The *Gamification* component provides the main functions to construct a game to incentivise users. There is also the *Data query and Storage* component, which manages the procedures for storing, retrieving and processing information. Moreover, the *Reputation* component rates the credibility of each user's feedback. Finally, the *Notification* component provides location-aware user notifications within specific time frames.

A very important feature of the Application Server is the gamification. In order to motivate citizens to provide more feedback, we considered that rather than a reporting tool, users should consider CW as a game, which rewards them according to their activity. In this sense, users gain points according to how many tags they report. However, something like this could only encourage people to take photos, not to act. Therefore, we designed the reward system so that users will get more points by fixing issues rather than just reporting them. Users can also gain points by voting or by reporting abuse. This way, users not only contribute data to the platform, but additionally provide the required feedback to "rate" the trust of each user in the game. In this context, by reporting a user gains some points, depending on the issue category these points range from 5 to 50. By voting, a user gets 10 points and the user who submitted the report gets 5 points if the vote was positive, or 3 points are removed from him, if it was a negative vote. By fixing a reported problem in his area, a user gets 30 points, and 50 points if the report was outside his area.

To encourage competition between groups of users, they are also associated to one area (that they choose at registration), and can join groups. Points are also awarded to areas and groups as follows. Each group's score is the sum of the scores of its users. In order to trigger more competition between different areas, the area scores are more complex, as users can affect the points of both their areas and others. Areas gain points whenever a positive report is submitted, while they lose points for each negative report. However, the areas are able to regain their points if the reported problems are fixed. To encourage the members of each area to take action on their area as soon as possible, extra points are given if a user from another area fixes the problem. This way, users are likely to encourage other users in their areas to be more active to gain more points, not only by reporting, which can be considered a passive action, but by fixing problems. In more details, for positive reports, each area gets 10 points, while for negative reports 10 or 20 points are removed, depending on the origin of the user who reported it. If the report is fixed by someone from the same area, then 20 points are added, but in the other case only 10 points, while 30 points are added to the area of the user who fixed the problem. Figures 6 and 7 illustrate the process of increase in the user, group, and area scores.

C. Mobile Application

Users are using the CW mobile application to interact with the middleware and the application server. Its role is to authenticate users and provide a intuitive front-end, so that they can send data and their feedback, and receive feedback from others. Generally speaking, users can upload positive or negative reports related to Green actions. Therefore, the application provides them with an interface that allows them to type some introductory and explanatory text about their report, and append. Apart from that, users may choose to publish events or send kudos for specific things that they notice in their city.

Using the same interface, users can see the reports that others have submitted and interact with them. For example, if a user has submitted a report about litter that has been left out, then another user may verify it, report abuse or even better, fix the problem and upload a photo where he shows that the problem has been solved. A special part of the application is devoted to Community Gardens. In order to promote Community Garden related actions, users may choose to see all the available gardens, for which they may upload photos or see live data from local sensors. Finally, the application provides users with access to several urban environmental metrics in real-time, from noise sensors to temperature, and from humidity to CO. This way, users have

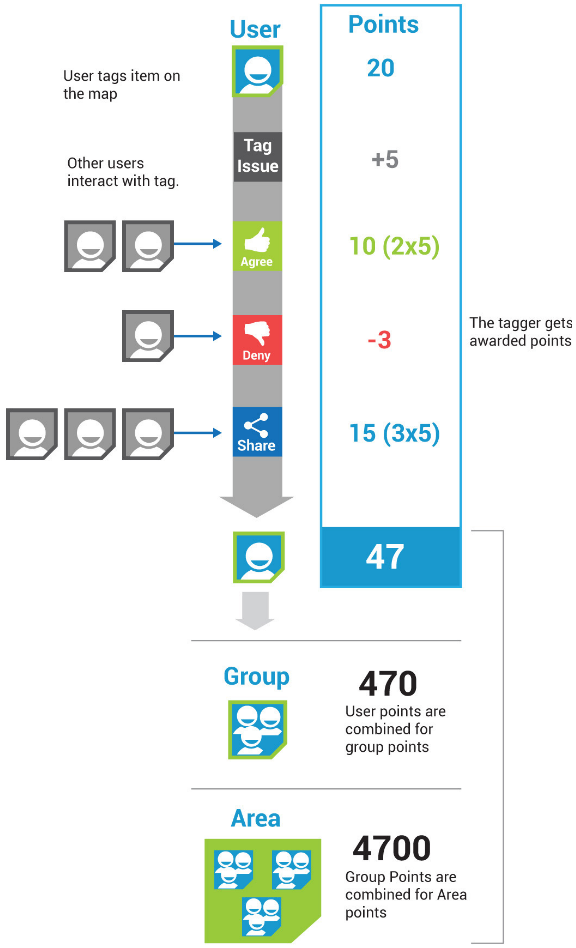


Fig. 6: Gamification

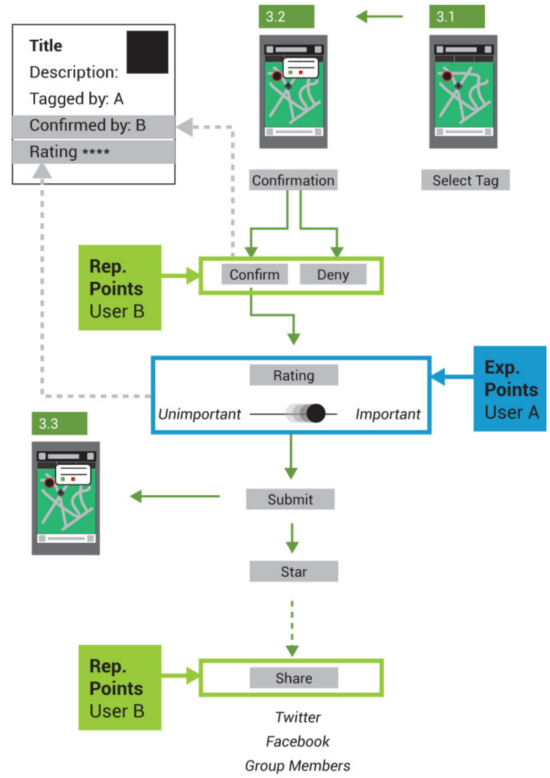


Fig. 7: User points flow

a real-time crowd-sourced view of their city, gathered from static and mobile sensors and even other citizens.

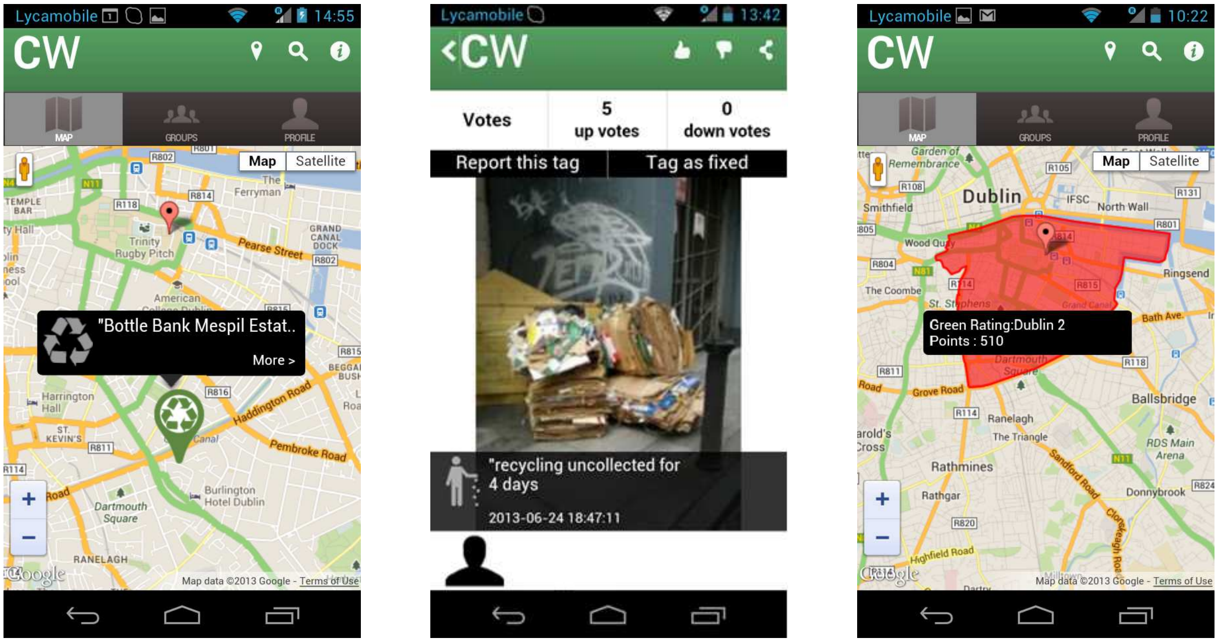
The mobile application enables users to become aware of the green environment activities in Dublin and report related problems. Moreover, users can share information about green initiatives in the city and motivate friends to participate in them. The application provides users with a map that displays all the requested information so that they can access directly, depending on their search, an event, a reported issue or environmental sensors. Additionally, it provides them with an overview of the group activities, how many points each group has gained in the past week, how many reports have been sent, which users lead, motivating them to interact. Users can change groups and declare their area, so that their groups and respective areas get the according points.

The application provides additionally an insight into city resources, such as bottle banks, recycling centres, parks and playgrounds in each area. Given that these resources are not advertised, many people are unaware of their existence, so the application is facilitating users by making people aware of their location. Moreover, people who are engaged in community gardens actions which are scattered throughout Dublin, are able to promote their activities by publishing them and providing photos of the work that is done, urging people to participate.

IV. Implementation

A general overview of the implementation is illustrated in Figure 9. As shown we used Java EE to implement the CW middleware functionality described in Section III-A and deployed it on the Google App Engine (GAE). The Java API for XML-Based Web Services (JAX-WS) JSR 224 is used to implement SOAP-based Web services providing the middleware interface to data producers and data consumers as described in Figure 3. WSDL files describing CW middleware interface details and schema files describing the data types used in WSDL file can be accessed at the GAE. Data consumers and data producers share data with the middleware in XML format. A common XML schema describing the data representations is shared among different components. Data producers generate data objects according to the XML schema. Once the middleware receives an XML data object, it generates a unique identifier for that measurement and adds it to the data object. We use the Java API for XML-Based RPC (JAX-RPC) JSR 101 for data validation and XML-Java object binding. Subsequently the data is stored in a Google SQL database. We use the Hibernate and JAVA Persistence API 2.0 for object relation mapping and interaction with the SQL database.

The CW application server sits on top of the CW Middleware, providing a RESTful API to the mobile application. It



(a) Current location (b) User group (c) User profile

Fig. 8: Screen shots of CityWatch mobile application

is implemented in Python 2.7 and deployed on the Google App Engine. For storing user information, the Application Server is using a Google SQL database, while images, which have a significantly bigger size, are uploaded to the Google Cloud Storage. Since these two services provide automatic scaling and very good performance, handling the data and load balancing are forwarded to Google's infrastructure, which scales them smoothly.

Generally, all input calls and output data are XML-formatted, to provide interoperability and easy parsing, independently of the used client. All the calls are made through HTTPS to provide secrecy of the exchanged data, while only calls from users that have submitted their credentials through OAuth 2.0 are permitted. Since the vast majority of Android users has a Google account, this account and its credentials are used to authenticate the user to the application server.

To provide some anonymity to the reports, each uploaded image is processed using OpenCV[2] to remove detected faces. Afterwards, any metadata is removed from the photos and they are uploaded to the Google Cloud Storage. To minimize bandwidth costs, the application server returns only links to images in Google Cloud Storage. This way, the exchanged messages are short and the programmer chooses when to receive and display pictures.

In order to display the functionality that our infrastructure is providing, a mobile application has been developed using HTML5 and JavaScript, using the PhoneGap[3] framework. The application has been packaged for Android OS, but, as it is in HTML5, can be easily ported to any other widely-used mobile OS, such as iOS, Blackberry, or Windows Phone. We choose Android to pilot our application, as it currently has the biggest market share[4]. The interface is quite intuitive to allow users to easily report positive or negative events and access city sensors as evident from some of screenshots shown in Figure 8. The mobile application is authenticating users using their credentials from their Google account through OAuth 2.0 and forwards all their requests encrypted to the Application Server, using a HTTPS connection and the provided RESTful API. The application requires minimal access to the user account, only name and email. Moreover, the application is using a user's location only when he submit a report, in keeping with the principle of data minimization.

V. User Trial and Results

In this section we report the preliminary results of our on-going user trial of the CityWatch application built on top of our framework. We organised two workshops in Dublin City Council and Trinity College Dublin to invite the volunteers to participate in our trial. Currently fifty five users have volunteered to take part in our trials. They have installed the application on their own phones and are collecting and reporting the data during their everyday life. They are also using the application to view sensor data as described in Section III-C.

Figure 10 shows the distribution of the number of of CW users' interaction with the application and subsequently the back-end servers against the number of users performing those interactions. The CW users' interactions are counted as the total number of their reports capturing positive and negative urban practices, their actions to resolve reported problems, votes up or down of reports, sharing of reports on social media,

[2]http://opencv.org/
[3]www.phonegap.com
[4]http://www.idc.com/getdoc.jsp?containerId=prUS24108913

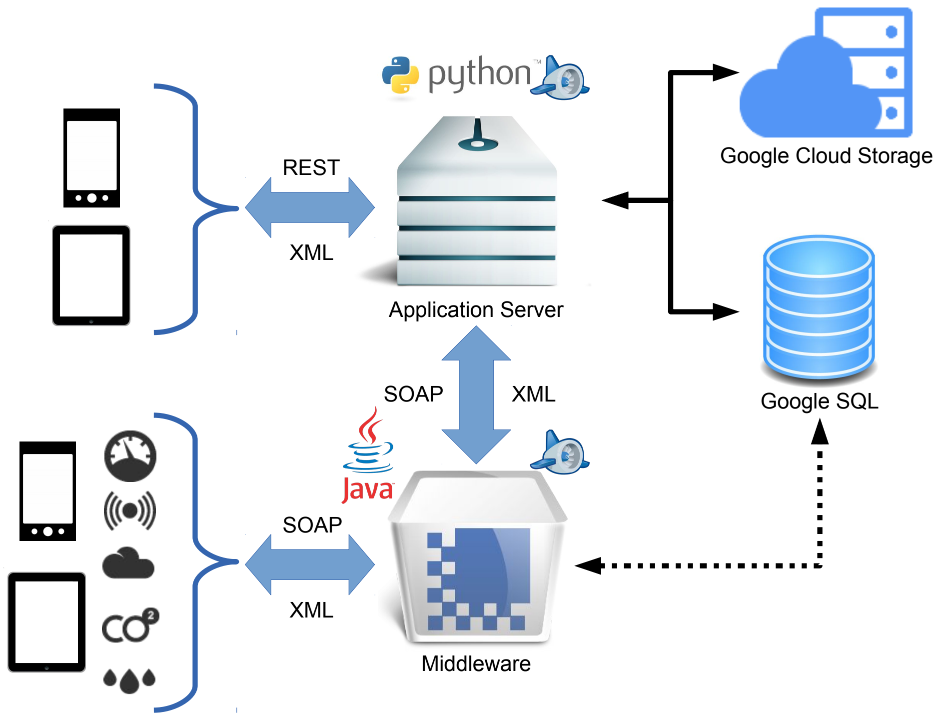


Fig. 9: Implementation diagram.

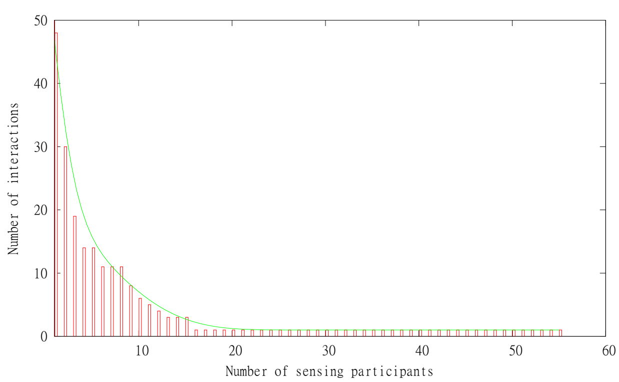


Fig. 10: User participation cumulative distribution in terms of number of interactions

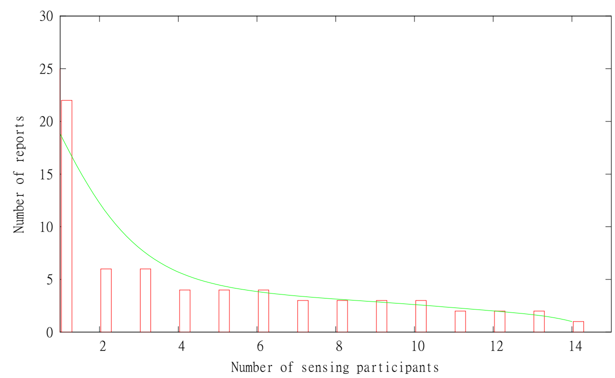


Fig. 11: User participation cumulative distribution in terms of number of reports

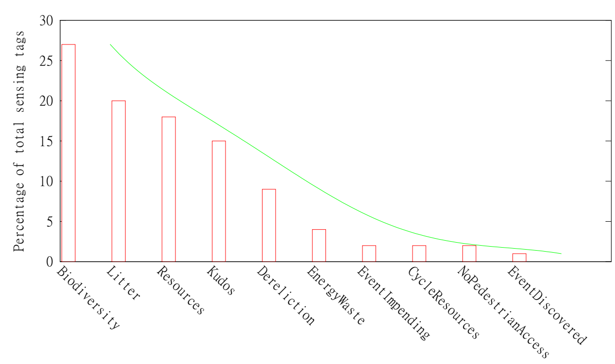


Fig. 12: User participation for different issue categories

queries on sensors data, and queries of the green score of an area.

The graph shows that a few users have a very high number of interactions with the application, such as one user who has interacted with the application forty eight times whereas a large number of users rarely interact with the application. We can also find the same pattern in the graphs shown in Figure 11 and Figure 12. Figure 11 shows the cumulative distribution of the total number of reports of negative and positive urban behaviours. We can observe that about fourteen sensing participants contributed at least one report. The number of sensing participants decreases as the number of sensing reports increases, and we can find that very few sensing participants contributed over twenty reports.

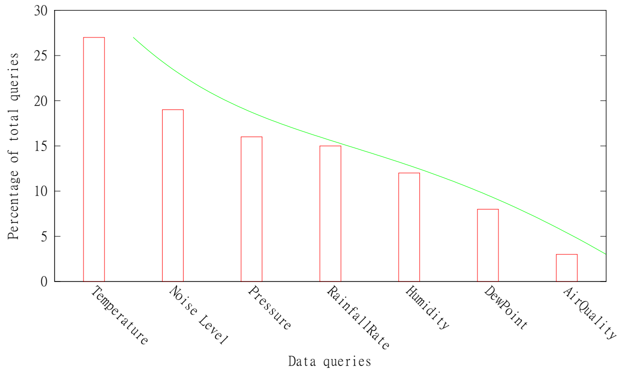


Fig. 13: Users' queries to view data from different sensors

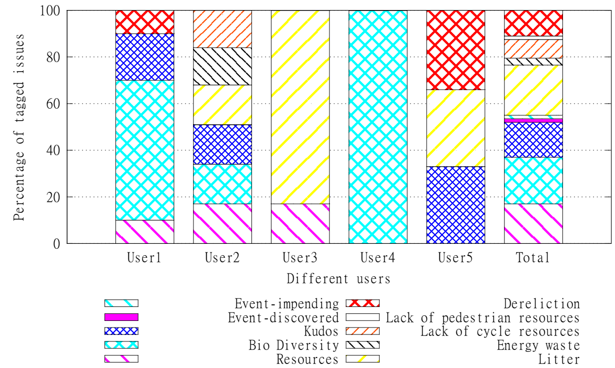


Fig. 15: Proportion of different issues tagged by the top five CityWatch users

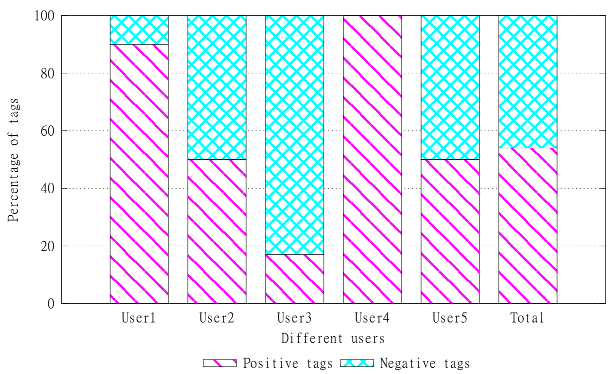


Fig. 14: Proportion of positive and negative issues tagged by the top five CityWatch users

Figure 12 shows the distribution of the issues by CW users' in the different issue categories. (Note, the tag categories are pre-defined in the application, and users can only to send information in one of those categories). We can observe that users took different level of interest in tagging those issues. CW users showed more interest in sharing the reports about Biodiversity and Litter issues in the city than the reports about the availability or lack thereof of cycling resources or the pedestrian-accessibility of different areas of the city.

We can observe that the graphs plotted in Figure 11-13 follow the power law. Power law phenomena are also evident in the social interactions and social networks. Observing the presence of a power law based distribution in the collected data, we can establish that all the sensing participants are not sharing data at the same rate. Few participants have a high interaction with our participatory sensing based application and are sharing a lot of information about different aspects of urban life. Whereas a large number of people rarely interact with the application and share any information. Sitting between these two extremes, a number of people interact occasionally with the application. Active users can motivate the moderately-active user group to increase their participation and engage the less active users.

Observing the presence of a power law pattern in Figure 12, we also found that our application is not receiving the same number of reports about every aspect of the urban life. We got every a high number of reports shared about the presence of Biodiversity and Litter in the city and there were very low numbers of reports contributed about green events in the city. We identified two possible reasons for such sensing behaviour: firstly, Biodiversity and Litter are the top options to share positive and negative reports (respectively) in the CW application user interface, whereas NoPedestrianAccess and EventDiscovered lie at the bottom of the CW user interface. Secondly, sensing participants may have different level of interests in sharing data about different topics. Considering these observations, it is highly recommended that special consideration should be paid to the user interface of participatory sensing based applications and interests of a particular community to effectively collect urban data.

Figure 14 and Figure 15 show data contributed by the five most active CW users. Figure 14 shows the proportion of positive and negative reports whereas Figure 15 shows the proportion of detailed tags. We can observe that different users contributed data in different patterns. We can observe in Figure 14 that User 3 mostly contributed negative events reports whereas User 4 only contributed positive reports. However, if we look at the total number of reports contributed by all users, we can find that positive and negative reports have been shared in equal proportion. We find the same pattern in Figure 15. Although different users have shared reports about different issues with different intensity, most of the issues have been shared with almost the equal proportion. This observation proves that although different users may share data about different issues according to their own interests in that particular issue, if there are good number of users sharing data with an application, overall all the issues will be shared with similar proportions.

VI. Conclusions

In this paper we presented CityWatch (CW) - a data collection and dissemination framework for smart cities. Our framework uses fixed sensing, participatory sensing and opportunistic sensing to collect data and provides a uniform interface to disseminate data. We also presented the CW application that was developed using our CW framework. We reported preliminary results of our on-going user trial. We showed that different users have shared data about different issues with a different intensity. We find that overall user participation follows the power law and although there is a group of very active users, but a large number of users rarely interact with the application. Active users can play their role in motivating the moderately-active user group to increase their participation and engage the less active users. For future work, we plan to enhance the capabilities of our framework by including more features, such as opportunistic task assignment by dynamically finding out the most suitable group of sensing participants to gather information about a specific issue.

ACKNOWLEDGEMENT

This work was supported by Science Foundation Ireland under the Principal Investigator research program 10/IN.1/I2980 "Self-organizing Architectures for Autonomic Management of Smart Cities".

REFERENCES

[1] "Un. world urbanization prospects: the 2011 revision." http://esa.un.org/unpd/wup/index.htm, 2011, [Online; accessed Aug-2013].

[2] J. M. Shapiro, "Smart cities: quality of life, productivity, and the growth effects of human capital," *The review of economics and statistics*, vol. 88, no. 2, pp. 324–335, 2006.

[3] D. Toppeta, "The smart city vision: How innovation and ict can build smart,livable, sustainable cities," 2010.

[4] C. Aggarwal and T. Abdelzaher, "Social sensing," in *Managing and Mining Sensor Data*, C. C. Aggarwal, Ed. Springer US, 2013, pp. 237–297. [Online]. Available: http://dx.doi.org/10.1007/978-1-4614-6309-2_9

[5] D. J. Cook and S. K. Das, "Pervasive computing at scale: Transforming the state of the art," *Pervasive and Mobile Computing*, vol. 8, no. 1, pp. 22–35, 2012.

[6] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *Communications Magazine, IEEE*, vol. 48, no. 9, pp. 140–150, 2010.

[7] J. A. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory sensing," 2006.

[8] E. Miluzzo, "Smartphone sensing," Ph.D. dissertation, DARTMOUTH COLLEGE Hanover, New Hampshire, 2011.

[9] T. Horanont and R. Shibasaki, "An implementation of mobile sensing for large-scale urban monitoring," *Proc. of Urbansense08*, 2008.

[10] J. Froehlich, J. Neumann, and N. Oliver, "Measuring the pulse of the city through shared bicycle programs," *Proc. of UrbanSense08*, pp. 16–20, 2008.

[11] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda, "Peir, the personal environmental impact report, as a platform for participatory sensing systems research," in *Proceedings of the 7th international conference on Mobile systems, applications, and services*, ser. MobiSys '09, 2009, pp. 55–68.

[12] R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu, "Ear-phone: an end-to-end participatory urban noise mapping system," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*. ACM, 2010, pp. 105–116.

[13] D. Mendez, A. J. Perez, M. A. Labrador, and J. J. Marron, "P-sense: A participatory sensing system for air pollution monitoring and control," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 344–347.

[14] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, "Noisetube: Measuring and mapping noise pollution with mobile phones," in *Information Technologies in Environmental Engineering*. Springer, 2009, pp. 215–228.

[15] S. Reddy, A. Parker, J. Hyman, J. Burke, D. Estrin, and M. Hansen, "Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype," in *Proceedings of the 4th workshop on Embedded networked sensors*. ACM, 2007, pp. 13–17.

[16] M. N. Boulos, S. Wheeler, C. Tavares, and R. Jones, "How smartphones are changing the face of mobile and participatory healthcare: an overview, with example from ecaalyx," *Biomedical engineering online*, vol. 10, no. 1, p. 24, 2011.

[17] P. Dutta, P. M. Aoki, N. Kumar, A. Mainwaring, C. Myers, W. Willett, and A. Woodruff, "Common sense: participatory urban sensing using a network of handheld air quality monitors," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09, 2009, pp. 349–350.

[18] S. Martin, G. Diaz, I. Plaza, E. Ruiz, M. Castro, and J. Peire, "State of the art of frameworks and middleware for facilitating mobile and ubiquitous learning development," *Journal of Systems and Software*, vol. 84, no. 11, pp. 1883–1891, 2011.

[19] J. Payton and C. Julien, "Integrating participatory sensing in application development practices," in *Proceedings of the FSE/SDP workshop on Future of software engineering research*. ACM, 2010, pp. 277–282.

[20] D. Trossen and D. Pavel, "Nors: An open source platform to facilitate participatory sensing with mobile phones," in *Mobile and Ubiquitous Systems: Networking Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, 2007, pp. 1–8.

[21] S. Reddy, V. Samanta, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Mobisense - mobile network services for coordinated participatory sensing," in *Autonomous Decentralized Systems, 2009. ISADS '09. International Symposium on*, 2009, pp. 1–6.

[22] G. S. Tuncay, G. Benincasa, and A. Helmy, "Autonomous and distributed recruitment and data collection framework for opportunistic sensing," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 16, no. 4, pp. 50–53, Feb. 2013. [Online]. Available: http://doi.acm.org/10.1145/2436196.2436219

[23] N. D. Lane, S. B. Eisenman, M. Musolesi, E. Miluzzo, and A. T. Campbell, "Urban sensing systems: opportunistic or participatory?" in *Proceedings of the 9th workshop on Mobile computing systems and applications*, ser. HotMobile '08. New York, NY, USA: ACM, 2008, pp. 11–16. [Online]. Available: http://doi.acm.org/10.1145/1411759.1411763

[24] X. Sheng, X. Xiao, J. Tang, and G. Xue, "Sensing as a service: A cloud computing system for mobile phone sensing," in *Sensors, 2012 IEEE*. IEEE, 2012, pp. 1–4.

[25] A. Joki, J. A. Burke, and D. Estrin, "Campaignr: a framework for participatory data collection on mobile phones," 2007.

[26] K. L. Huang, S. S. Kanhere, and W. Hu, "Preserving privacy in participatory sensing systems," *Computer Communications*, vol. 33, no. 11, pp. 1266–1280, 2010.

[27] E. De Cristofaro and C. Soriente, "Short paper: Pepsi—privacy-enhanced participatory sensing infrastructure," in *Proceedings of the fourth ACM conference on Wireless network security*. ACM, 2011, pp. 23–28.

[28] A. Dua, N. Bulusu, W.-C. Feng, and W. Hu, "Towards trustworthy participatory sensing," in *HotSec09: Proceedings of the Usenix Workshop on Hot Topics in Security*, 2009.

[29] A. Manzoor, M. Asplund, M. Bouroche, S. Clarke, and V. Cahill, "Trust evaluation for participatory sensing," in *Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Springer, 2013, pp. 176–187.

[30] A. Manzoor, H.-L. Truong, and S. Dustdar, "Quality of context: Models and applications for context-aware systems in pervasive environments," *The Knowledge Engineering Review, Special Issue on Web and Mobile Information Services*, 2011.