Privacy-Aware Urban Noise Mapping

by

Qian Wang, B.Sc.

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfillment

of the requirements

for the Degree of

Master of Science in Computer Science

University of Dublin, Trinity College

September 2013

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Qian Wang

August 27, 2013

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Qian Wang

August 27, 2013

Acknowledgments

I would like to express the deepest appreciation to my supervisor Melanie Bouroche for introducing me to the topic, the useful comments and advices through the process of the project implmentation and writing the dissertation.

Furthermore, I would like to express my gratitude to my co-supervisors Francesco Pilla and Atif Manzoor for their suggestions and encouragement.

Also, I like to thank my course director Meriel Huggard for helping me to coordinate my project and for her advices through the learning process of this master course.

A special thanks goes to the participants in my evaluation, who are responsible and willing to help to test my project.

Last but not least, many thanks to my parents and my friends for their endless love and support.

Qian Wang

University of Dublin, Trinity College

September 2013

Privacy-Aware Urban Noise Mapping

Qian Wang, M.Sc.
University of Dublin, Trinity College, 2013

Supervisor: Melanie Bouroche
Co-supervisors: Francesco Pilla, Atif Manzoor

Top end mobile phones include a wide variety of sensors and are now ubiquitously in our city, which enables the novel paradigm known as participatory sensing. This empowers ordinary citizens to use their mobile phones to collect and share data about themselves and their surroundings. However, such applications bring a few threats especially on personal privacy, which may hinder people's participation.

This dissertation investigates the privacy threats associated to participatory sensing for noise mapping. Indeed, the collection of sound data may expose private information, such as the identity of users or the context of their conversation. To solve the mentioned problems, the Noise Manager uses the device number as the user ID which is known by the server whereas keeps anonymous with others. Additionally, this project only stores the noise level instead of the original recording to avoid users' audio recordings to be misused by adversaries. In this way, the Noise Manager enables users to probe, view and share the noise level without privacy concern. The evaluation was carried on by recruiting a few participants to install and use this application for one week and finally proved the high level of privacy protection by summarizing the user questionnaire.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

The number of smart phone users has grown rapidly in recent years. The ubiquity and advanced technique of mobile devices make them capable to be new sensing equipment in a large scale. There are a few embedded sensors in the smart phones, such as: the GPS, microphone, camera, accelerometer and so on, which could provide different kinds of information for users.

The advantages of mobile phones mentioned above make contribution to the participatory sensing which encourages ordinary citizens to collect and share sensing data by using their mobile phones. Compared with traditional sensors, mobile phones are ubiquitous and flexible to implement participatory sensing. Currently, there are some existing projects of participatory sensing, for example, PetrolWatch [25] is for the cheapest petrol price around users and Skiscape [26] offers essential information for manager, staffs and skiers. Participatory sensing project on noise detection is also one of the many kinds. As the noise pollution has negative effect on human's behavior, sleeping and temper and many cities are facing this problem, it is necessary to develop such a project for citizens to probe and learn the sound level around them.

However, it may bring some problems during the process of participatory sensing. For example, the user's identity is revealed if he reports his name, address and other information in a project that will publish all the messages; the unknown third party intentionally steals and misuses sensing data for their own benefit; a participant exposes his social circle when he shares the picture on a party, and so on. All these threats may prevent individuals from engaging in the large scale sensing.

This project named Noise Manager aims to empower people to collect and share the noise level of their surrounding environment by using their smart phones without privacy concern. It protects the privacy of users by using device number as user ID, storing no audio recording, data encryption in reporting as well as storing on server side and anonymity when sharing with others. In order to achieve participatory sensing, all participants can review the noise levels with location pins on the map, which are contributed by themselves and others.

It is the research question for this project to investigate potential privacy threats and then seek an efficient privacy-protecting way to solve the problems when applying the concept of participatory sensing in noise mapping. Without privacy concern, citizens are willing to engage in the project to collect, share and view noise data in their daily life.

1.2 Road-map

The rest of the dissertation is organized as follows. Chapter 2 reviews the related work of participatory sensing, privacy threats, privacy protection and noise detection. The existing participatory sensing projects could be roughly divided as two types: people-centric applications and environment-centric applications. In addition, almost all existing projects are based on the three-part layout which includes mobile side, server and data consumer. As it will bring some privacy threats during the large scale sensing including noise detection, lots of literatures keep working on how to solve those problems and there already have been some practical methods.

Chapter 3 introduces the whole design of this project as well as the scenarios of using this application. The key challenge is how to protect privacy, which is solved by data encryption and anonymity. Additionally, the results will be shown on the map that enables participants to straightly view different color location pins representing corresponding noise levels.

Subsequently, the implementation procedure of the project and the final results will be described in chapter 4. It requires different technical support to complete many task such as: change audio data into noise level, remove user ID before sharing, data encryption during report and storage and so on, which will be discussed in detail in this chapter. What's more, the screenshots of functions provided by this mobile application are shown in the last section with corresponding explanation.

The evaluation of this project and its results are provided in chapter 5. Due to the limitation of time and devices, there are only seven participants engaging in the experiment for one week. They install this application and run it whenever they want to provide sensing data. At the end of the experiment, all participants give feedback by answering questions on the questionnaire, which helps to evaluate and improve this project.

The last chapter is the conclusion the dissertation as well as provides some possible work in the future. The research questions are discussed and summarized in this chapter, followed by the advantages and shortcomings of this project gotten from the evaluation stage. Finally, it describes some solutions to solve the problems happened during the test and new features that maybe added.

Chapter 2

State of Art

2.1 Introduction

With the development of technology, smart phones have changed from merely making calls to mobile devices with multifuctions that can be used to locate, sense, navigate and so on. To some extent, the inherent mobility of mobile phones enables them to perform better than traditional sensors, which makes them popular in more and more domains. For example: the embedded GPS could provide the real-time location information and the integrated microphone can be used to gather sounds.

Sensors included in the mobile phones help to create a people-centric society by encouraging ordinary citizens to collect, report, communicate with each other. Using participatory sensing for urban noise mapping is a case study of personal city project, which should address the privacy problem well to get better performance. Currently, there are some potential threats hindering people's participation. Therefore, this chapter reviews what has been done on other systems in order to help the development of this project.

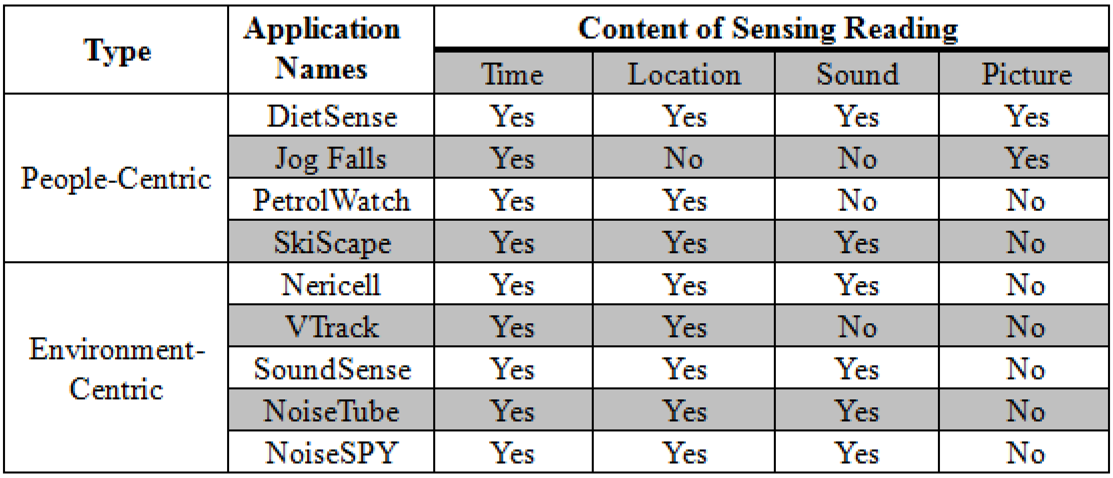
The investigation of participatory sensing is discussed in section 2.2 that mainly introduces the definination, system model of participatory sensing and what are the key challenges during the implementation. Section 2.3 reviews potential privacy threats in participatory sensing as well as the existing approaches to solve the problems. Section 2.4 covers a brief review in the noise area, including noise pollution, how to collect noise data and some technical challenges. The last section summarizes the literature review and proposes some future work.

2.2 Participatory Sensing

The combination of advanced technology of mobile phones with their connate ubiquity offers an opportunity to the implementation of participatory sensing, which empowers participants to collect and share information about themselves or their surroundings by using their own mobile phones.

2.2.1 Existing Applications

Currently, participatory sensing has been applied in different domains, such as: route plan, environment moniotring, price comparison and so on. The existing paradigms of the participatory sensing can be categorized as people-centric sensing and environment-centric sensing [1, 4]. The following is the summary of some applications reviewed by this dissertation and also lists the data type contained in their sensing reading.



Type	Application Names	Content of Sensing Reading			
		Time	Location	Sound	Picture
People-Centric	DietSense	Yes	Yes	Yes	Yes
	Jog Falls	Yes	No	No	Yes
	PetrolWatch	Yes	Yes	No	No
	SkiScape	Yes	Yes	Yes	No
Environment-Centric	Nericell	Yes	Yes	Yes	No
	VTrack	Yes	Yes	No	No
	SoundSense	Yes	Yes	Yes	No
	NoiseTube	Yes	Yes	Yes	No
	NoiseSPY	Yes	Yes	Yes	No

Figure 2.1: Existing Applications

People-centric sensing applications

People-centric sensing focus on human activities and behavior. It utilizes the embedded sensors of mobile phones to collect data about users and then shares the sensing data among ordinary citizens rather than professionals. People-centric sensing has given rise to many applications which can be generally classified as personal sensing and social sensing [18]. This dissertation discusses some examples as below.

Personal sensing applications are helpful for participants to record and archive their own personal information. The system named DietSense [23] gathers and saves the information about diet for participants who want to lose weight. Cameras embedded in mobile phones are used to take pictures of users' food selections. The system will assess the dietary intake and return the results. Participants could both review their own records and share the information with their health care professionals when necessary. Another example is Jog Falls project [24] that combines body-area sensors with an interface for calorie intake. It records the health data of patients and offers the analyzed results to physicians. In this way, patients are empowered to manage their diseases by monitoring their own activity and energy consumption and reviewing previous data to make essential changes. Personal sensing applications aid individuals to store and manage a flood of information and to process original data in order to make them more useful.

Social sensing puts the emphasis on the information or behaviors of individuals to be shared within trusted groups. For example, fuel price is automatically collected in Petrol Watch [25]. When a car approaches gas stations, cameras integrated in users' mobile phones will be automatically triggered to take pictures of prices. The interested data is uploaded to database after processed by some algorithms. Users are able to search the cheapest price of fuel around them by using this application. Social sensing also enables participants to locate and communicate with each other. SkiScape [26] system provides essential messages for all partakers. Skiers may want to know the trail conditions before they move to the top of the mountain. Rescue staffs would like to locate skiers in case of an accident. Manager is interested to be aware of customers' statistic and optimize maintenance. All of the users of the SkiScape are able to get their interested information. It is evident that a good social sensing application could bring much benefit to its stakeholders.

People-centric sensing focus on individuals ranging from analyzing personal information to communicating interested activities at a social scale which can be divided as personal sensing and social sensing. These applications gather sensing reading about users or their relevant things by lots of mobile phones. Finally, these systems would provide more useful information and better service after data integration.

Environment-centric sensing applications

Another type of participatory sensing is environment-centric sensing. The applications put more emphasis on surrounding environment gathered by integrated sensors in mobile phones and peripheral sensor devices when compared with people-centric sensing applications. What's more, the captured data is exploited at a community scale, such as road traffic report, environment monitoring and so on.

The costs on deployment and maintenance of traditional sensors for automatically monitoring traffic are usually expensive. Instead, it is can be realized by mobile phones to document various traffic conditions. The embedded accelerometer sensors, positioning sensors and microphones cooperate to localize and report road traffic conditions in [27]. In addition, such system makes the traffic data more valuable after processing and integrating. For example, the system in [9] could estimate driving time and plan best route for users based on the real time traffic conditions.

To monitor environment, mobile phones may collect less accurate parameters when compared with meteorological stations. However, their inherent mobility empowers them to capture data in a larger range as well as accidental pollution [1].

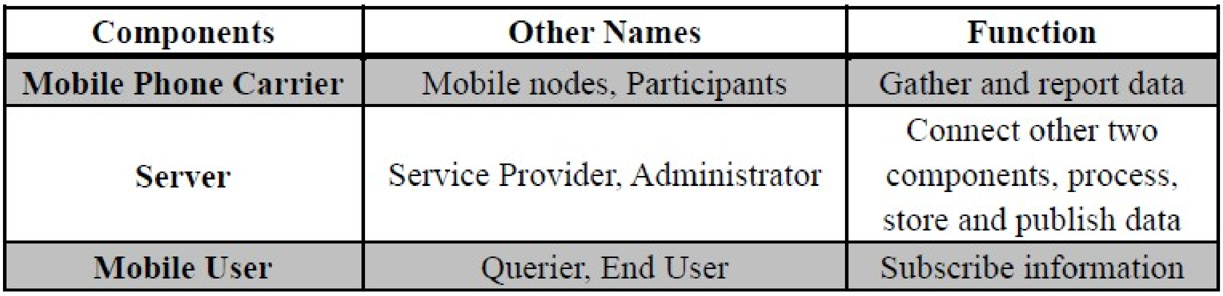
It is a trend that moving from traditional sensors to using sensors of mobile phones to probe sound levels. Authors in [14] propose an application named SoundSense to achieve the scalability of classification to a large population. It exploits the microphones-equipped mobile phones to recognize significant audio events of individual users' environment. In comparison of traditional audio recognition systems, SoundSense performs online and high accuracy with lower computational cost. NoiseSPY [11] is a real-time system for Symbian OS to explore and view noise level. It allows users to collect and view surrounding noise levels on the mobile phone platform, which is realized by the combination of microphones and GPS. Similarly, in NoiseTube [15] project, citizens are encouraged to measure their daily noise exposure by using GPS-equipped mobile phones. All sensing reading generated by users can be automatically shared online with oridinary citizens. At the same time, those data enables specialists to research behaviors and negative impact caused by noise pollution.

Although environment-centric sensing captures surrounding data, its final goal is also making the sensing data useful for the public. It seems less likely to be sole environment-centric sensing. More applications are the combination of people-centric with environment-centric.

2.2.2 Components and System Model

Participatory sensing is still at the developing stage, which aims to enable increasing number of mobile users to gather, share data and query information. In order to achieve the goal, the infrastructure of a typical participatory sensing usually includes at least the following three components shown in figure 2.2.

First is mobile phones carrier. These people are the data producers in the participatory sensing applications, which are also called as mobile nodes in [20] and participants in [1]. They are of great importance to form the basis of participatory sensing. As there are a variety of sensors integrated in today's smart phones, i.e. GPS, accelerometer sensor, camera, and microphone and so on, it enables mobile phone carriers consist of a large mobile nodes network, which could collect and share data. As illustrated in [10],



Components	Other Names	Function
Mobile Phone Carrier	Mobile nodes, Participants	Gather and report data
Server	Service Provider, Administrator	Connect other two components, process, store and publish data
Mobile User	Querier, End User	Subscribe information

Figure 2.2: Components of Different Systems

mobile phones play the important role of sensor data collection for a global sensor network (GSN) middleware. And mobile phones are used to gather interested parameters for an environmental monitoring application in [17]. The tasks of mobile phone carriers involve collecting, short term storage and reporting.

The second component is data consumers that consist of web users and mobile users, which are defined as queriers in [20] and end users in [1]. They could subscribe to desired information and get corresponding feedback within the participatory sensing network. Web users scan and reuse the information by using computers. While mobile users that include mobile phone carriers mentioned above, apparently get the information by using their phones.

Last but not least, the server, also called campaign administrator [1] and service provider [20], acts as the bridge between mobile phone carriers and data consumers. As there is no direct communication between the first two components, the server receives the reports form mobile nodes and then matches requests to corresponding end users. The main tasks of a server are matching, long term storage, processing and publishing. In addition, it is the responsibility of server to check the nodes' identification and to prevent malicious nodes to misuse data.

A basic and essential system model of participatory sensing application can be built with the three components: mobile phone carriers, server and data consumers. The operation sequence is summarized as: the mobile phone carriers collect interested data using their own mobile phones and then report to the server. It is necessary for the server to analyze and process received data before publishing. The server also needs to match results to corresponding queries. Finally, data consumers are able to access and reuse their interested information.

2.3 Privacy

As mentioned above, sensing data in all participatory sensing systems, which is related to the participants and their surrounding environment, may be used to extract or infer sensitive information about users. Dealing with the privacy problems well will encourage participatory sensing to be applied more widely.

2.3.1 Privacy Definition

Accompanied by the increasing number of the communication equipments and computing systems, the definition of information privacy was emerged and defined as ”an individual's claim to control the terms under which personal information information identifiable to the individual is acquired, disclosed, and used” [6]. To cater for the specific characteristics of participatory sensing systems, a notion is proposed by authors in [1]:
”Privacy in participatory sensing is the guarantee that participants maintain control over the release of their sensitive information. This includes the protection of information that can be inferred from both the sensor readings themselves as well as from the interaction of the users with the participatory sensing system”.

2.3.2 Privacy Threats and Analysis

People are sensitive to their privacy which means the protection for participants to control what personal information to share, with whom and how long, is the biggest challenge in participatory sensing. However, privacy threats are common problems in participatory sensing applications, which involve revealing and misusing participants' personal information. The consequences of privacy disclosure may be harmful for both individuals and society.

After reviewing the existing participatory sensing applications shown in figure 2.1 in section 2.2.1 and the types of sensing reading they capture, it helps to observe the potential privacy threats and to analyze the importance of privacy protection.

Location and Time

It is evident that all of the applications mentioned above require users to collect and share time information as well as location data, except the Jog Falls project. However, the large number of successive time and location data to be reported from participants makes it easy for third party to infer personal information about users. To some extent, deciding what to reveal to whom at what time is telling others who you are. For example, employers may be able to infer the medical condition of their employees from the frequency to hospitals, and similarly, attendance at a hip-hop concert may provide the music preferences of users. What is more, individuals' identity can be deduced by analyzing their residence location and workplace. Unauthenticated parties may sell these information to make profit or even threaten users directly, which results in negative consequences on individuals and raise panic in society.

Sound

More than half of applications shown in figure 2.1 collect sound data by using participants' mobile phones. When participants actively contribute data, they can choose to record only non-sensitive sound events which do not invade their privacy. When their mobile phones record automatically, users have no guarantees of the sound that will be recorded. In addition, collected sound data which is not relevant to the users themselves may reveal others' privacy information. Sound samples help adversaries to guess people's identity as well as their conversation context. Even worse, the sound data combined with the location and time information makes it more possible to consistently monitor participants, which will not be accepted by anyone.

Pictures

Although there are only two applications mentioned above requiring users to take pictures of events, it does not decrease the importance of the protection of picture information. Similarly to sound data, the content of pictures shared by participants could possibly reveal their identity and environment because the information represented by pictures is part of the participants self-presentation. Although sometimes the cameras embedded in participants' mobile phones are oriented away from themselves, faces of other people together with them may be taken in the photos. Frequently in some scenarios, participants would like to take pictures of their friends and themselves, and then share within the participatory sensing network. Thus, the participants' identity as well as their social circle can be inferred by analyzing the shared pictures together. Unauthenticated parties could get personal information, which endangers the privacy of individuals.

Participants may be reluctant to engage in the sensing activities when they are aware of possible results of privacy leak. Thus participatory sensing will lose the key components and is of little value. In order to attract more people to make contribution, the system should enable authorized parties to obtain interested information in an efficient way when necessary, e.g, users get noise level in NoiseSPY [11], and emergency staffs localize skiers by SkiScape [26], but it is never traceable for unauthorized parties to get the any information about users.

It may be one solution that the server in participatory sensing applications takes the responsibility to protect the users' information, such as: check a node's identification before allowing it to join in the system. There are a few other mechanisms to protect privacy and location privacy has been predominately addressed in the literature in comparison with the other sensing modalities. However, it is the fact that coming up with a solution for privacy protection is much more complicated in participatory sensing applications than other domains, as it involves multi-dimensional problems.

In conclusion, privacy threats especially coming from location and time disclosure create some barriers for the prevailing of participatory sensing. It really requires much effort to make the participants' privacy extremely safety. The less privacy threats are, the more successful the participatory sensing is.

2.3.3 Privacy Protection

Currently there are some existing approaches to address various threats to privacy in participatory sensing although it still has a long way to go. Based on the discussion in section 2.3.2, it is evident that all mentioned applications collect users' location and time information, which requires reliable protection mechanisms to guarantee security. What's more, the disclosure of the picture and sound information, when combined with location data, poses serious negative influence on users. Fortunately location protection has been predominately realized so far in current literatures.

User-centric privacy access control

Individuals are reluctant to expose the information about what they are doing at when and where to strangers or sometimes even to their friends. However, in location-sharing applications, the problem mentioned above cannot be avoided. For the purpose to make user to have more control of their privacy, authors in [16] present a method of user-centric privacy access control, which can be merged into existing location sharing systems.

For all participatory sensing applications, the server receives data from a flood of voluntary users and publishes the information for authenticated consumers. User-centric method allows users to divide their information into two parts that are standard privacy access policy and sensitive privacy policy. At the server side, it stores the standard privacy information that is not changed frequently by users and doesn't mind to release to others, which requires the end users to send a request to server before access. The server could send the decision of "approve" or "deny" to end users directly, but the request will forward to corresponding user if the server's decision is "ask the user". The sensitive privacy policy is stored at the user side and allows users to set, edit or delete at any time. Similarly, the user first receives the data consumers' request and then sends the feedback of "approve" or "deny" directly. If the user's decision is "ask user for real-time permission", it will alter the user to give real-time permission for every accessing.

This approach empowers the users in participatory sensing applications to set requirements in order to control their privacy. The server and users are responsible for part of the information, which reduces the task of both. User-centric privacy access control makes sense to meet privacy protection by allowing users to decide the tradeoff between high privacy and performance.

PEPSI

Privacy-Enhanced Participatory Sensing Infrastructure (PEPSI) [20] is a cryptographic tool consisting of the server, mobile nodes, querier and registration authority. Server is the intermediation between mobile nodes that collect data in the system and queriers that use the sensor reading. Registration authority sets the unified parameters for the application and requires both mobile phone carriers and data consumers to register before reporting and using data. In order to improve the match efficiency, all encrypted data will be tagged with some key words, which helps the server to identify the type of report without leaking any report information.

The operation of the PEPSI is initiated by the mobile nodes which want to report data to server. It is necessary for mobile nodes to register at registration authority and at the same time get the secret key for tagging. The encrypted data and corresponding tag are sent to server together by mobile nodes. The queriers register at registration authority and then obtain the decryption key of the interested data.

As the cryptographic key and decryption key are secret, the sensing data can be protected. In addition, the efficient match between request and reply is implemented by using tags.

Hot-Potato-Privacy-Protection

In order to protect the users' identity in participatory sensing, authors in [2] propose the Hot-Potato-Privacy-Protection (HP3) algorithm which is used to protect collected picture/video to be transferred through a few mobile users before arriving at server side. The initiated user generates a random value X between 0 and 1 as well as sends the sensing reading to next user. The value X decreases by 1/k along the travelling nodes k until the value X/k reaches the setting threshold T, which is the condition for the last user to upload the information to server. In addition, the initiator encrypts collected data using server's public key so that others would not know the data information during transferring.

Each user has a list of their trust friends to enable randomly select one to transfer the data. To avoid the malicious nodes pollute the collected picture or video, HP3 algorithm combined with the technique of fragmenting image into small segments with some redundancy. Thus the server could also reorganize the original picture or image based on good segments even though a few sections are polluted.

HP3 makes the server to be unaware of the users' identity privacy and enhance the protection for collected images. The location privacy protection of participatory sensing applications will attract more participants to make contributions. And the accurate sensing data is necessary as the aim of participatory sensing is to integrate various data to serve more people.

Anonymous

As location information of users in participatory sensing applications is so important that a lot of algorithms have been applied to make the users' location to be anonymous. A personalized anonymization model [8] allows each mobile user to set his or her minimum anonymity level and maximum temporal and spatial resolutions. The server in the model of k-anonymity processes a message according to the k value set by its mobile user in order to guarantee spatial tolerance or temporal tolerance. It decreases the location accuracy by enlarging the spatial area up to including k-1 other users. This approach is spatial cloaking. In addition, the approach named temporal cloaking makes a report from one location to be published with delay until k users visit the same area. As a result, the possibility of location disclosure becomes 1/k for adversaries.

Authors in [3] empower the protection of location privacy using microaggregation that replaces the value with the mean of Equivalence Class (EC) in which the value is found. In particular, the authors propose a new scheme called Variable-size Maximum Distance to Average Vector (V-MDAV), which could make location information anonymous by using EC generation and EC extension. The former one calculates the Euclidean distances within a certain range involving k users and then clusters the users. EC extension merges the users that close enough to an existing EC. Thus, the location of one mobile user will not be shown whenever he/she reports to server.

Various methods used to protect people's privacy away from different threats help to pave a wider way for the participatory sensing. As almost all participatory sensing systems require location information, it is the first step to make the location privacy safety. Advanced technology brings both opportunities and challenges, participatory sensing can develop faster based on excellent privacy protection and more participants engaging.

2.4 Noise

Sound exists everywhere and is essential for everybody in our daily lives. We cannot imagine the world without any sound. There are various kinds of sounds, such as: music from bars, voice from a conversation discussion, emergency alert from an office building and so on. Noise that no one likes is evidently a type of sounds. It is the fact that noise grows increasingly in our society, which is becoming a major pollution problem in urban environment affecting human activities. According to the green EU paper [15], "Environmental noise, caused by traffic, industrial and recreational activities is one of the main local environmental problems in Europe and the source of an increasing number of complaints from the public". So it is the first and basic step to be able to measure the noise level by gathering sound samples and implementing noise mapping in order to better understand the problems and coming up with solutions.

2.4.1 Noise Pollution

Noise pollution has become a major environmental problem for citizens in urban areas, besides air and light pollution. There are different sources to produce noise, i.e.: road traffic, construction, machine factory, bars and so on. Approximately 40% people in Europe Union are exposed to road traffic noise which account for 80% of the total noise pollution [11]. In addition, it is the fact that many people are suffered from stressful levels of noise at home and outside. As EU experts estimated in [15], unacceptable noise exists in the daily life of 80 million European people and even worse 170 million of them suffer from serious annoyance related to noise during daytime. Noise ranging from loud, annoying to harmful to people's ears can have negative effect on interrupting people's conversation, disturbing sleep and even causing physical damage.

Besides noise level, the duration of the exposure is another factor to threat people's health and daily life. Author in [11] also proposes that young children suffer decreasing hearing sensitivity year by year, as they usually wear earphones to listen to music with very loud volume. Exposing long time to noise including excessively amplified music causes the lack of sleep, indigestion and even worse disease for all individuals not only children. The severity of noise pollution has warned us to be away from noise as possible and reduce the time to expose in noisy environment in order to keep healthy.

2.4.2 Noise Measurement and Mapping

Based on the above discussion of negative consequences of noise pollution, it is encouraged for citizens to take responsibility to monitor and report the level and source of noise in urban environment. Traditional sensors could accurately measure the noise level within a certain range while the novel approach for gathering noise data by microphones embedded in mobile phones seems to be cheaper and more ubiquitous. Mobile phones act as the emerging sensors to collect and share noise information, which contribute to using participatory sensing for urban noise monitoring.

Using sensor nodes for assessment

As noise pollution is getting more and more attention, it really needs to take action to measure and manage the problem. In the initial stage, people use manual collection to obtain noise data by assigning officers to interested locations. The sound level meters or other microphone-equipped devices enable designated officers to gather noise samples with high accuracy within short time slot. However, the method cannot offer successive data and the range of measurement is also limited.

Moving to using wireless sensor for noise monitoring proposed in [5], authors could get fine-grained noise data by choosing the node location accurately. Wireless sensors no longer rely on the person's movement and they can collect accurate data no matter day or night once the sensors are put in the interested area with certain density. In this way, wireless sensor network overcomes the drawback mentioned above so that the more fine-grained and successive collection can be obtained to assess noise pollution level.

Using mobile phones for assessment

The advanced technology empowers noise measurement to move from traditional sensors towards microphones embedded into mobile phones. The new method could collect data in larger range compared with traditional sensors because of the movement of users. In addition, it costs much less in terms of installation. Mobile phones act as the assessment nodes making it possible and attracting for citizens engaging in noise monitoring in urban areas.

People carried with mobile phones are regarded as the sensing nodes and could send the collected data to server. It is a real time application named NoiseTube [15]. The mobile nodes gather the noise sample first and they are free to decide whether publishing their collection. The users are able to add comments for the collection, such as: noise tag and noise location. If they want to make the sensing reading public, they can send the collected data and their comment to the server that will centralize and process all the information. What's more, the server replies the noise level to the screen of mobile phones, which is represented by lines with different color indicating no harmful, be careful or risky. Once the software runs on the mobile phone, it keeps record sound every one second so that the information could be real time.

Similarly, NoiseSpy [11] has the basic client-server architecture. By default, it samples data every second but can be adjusted by users after running the program on mobile phones. Then the file which includes the graphic of noise data as well as its location is created with timestamp and stored in the phone's memory. The clients that are authentic by right user name and password can send the logs to server over HTTP using POST requests and the reply updates the state of the mobile applications. The server of NoiseSpy is a standard Apache web server with PHP and PostGreSql database. PHP script converts data from mobile phones into XML and KML formats to be visualized online.

Both two servers mentioned above will create noise map based on the information from mobile sensing nodes. In this way, users could see their own contribution or exposure on the website. Some trust third parties also could access the data for further study or reuse. In addition, it is another way for vehicles equipped with smartphones to collect noise data for creating noise map. Authors in [13] investigate pedestrians carried mobile phones to measure the noise level and report noise location.

2.5 Summary

Currently, there are various participatory sensing systems in different domains. Using mobile phones to monitor noise level in urban area is one of the emerging applications.

However, there are still many challenges existing in three aspects: participatory sensing, privacy as well as noise measurement by mobile phones.

As participatory sensing systems mainly rely on the users to voluntarily contribute to gather and report a flood of sensing data, it is associated to something troublesome, especially in protecting users' privacy. In addition, it is the fact that mobile phones are not exclusively to measure noise level. Thus the correctness of the sensor readings from phones becomes a concern. The following is a short summary of different problems that should be overcome in the future.

Privacy leak

Participatory sensing that enables all users to collect data and communicate with each other may face barriers if users do not trust it [18]. However, without suitable protection mechanism, the system may potentially reveal the privacy of users during the data collection as well as the data processing. Mobile phones which should be used to gather data will become miniature spies [4]. The possibly harmful consequences involve stalking or do other criminal activities or simply making people uncomfortable [7]. It will hinder people's participation when they are aware of the possible consequences. Thus, it will constrain the range of sensing and reduce the benefits of the system.

Data literacy

The risk of data gathering and collecting is not always self-evident. Sometimes, participants cannot be able to fully understand the benefits of managing data by themselves. So, data literacy [22] that means when and where to share what sort of data and what possible threats to publishing data should be acquired over time through various channels, such as: forum, traditional media, website and so on. Making people being aware of the potential risks helps them to self-protect, which is a huge task requiring much time and efforts.

Incomplete data

In terms of incomplete data samples, it includes two aspects: coverage [18] and content [4]. As we know, participatory sensing requires data in a large scale. In some cases, the number of participants is limited so that the incomplete coverage makes the correctness of information uncertain. If the server publishes the incorrect data coming from a small number of users, the participatory sensing application seems to be of little value. In addition, incomplete data content may occur. Usually, reported data supported by sensors embedded in participants' mobile phone, should tag with corresponding location and time. So sensing readings can only meet the requirements when the carriers are present there. Surrounding sounds which will captured by microphones integrated in mobile phones without any filtering may affect the actual level of interested noise. In this way, the data reported by users is less accurate so that the value of measurement needs to be considered before applying. Furthermore, volunteers may prioritize for other emergent tasks rather than reporting. As a result, data samples could be incomplete so that making it a challenge to recover the original one within crowdsourcing data.

Data trustworthiness

Encouraging lots of users to contribute to the participatory sensing, to some extent, results in the data trustworthiness [4] problem. It is the server's responsibility to evaluate the quality of data. Sometimes users may inadvertently gather the incorrect data and then send to server, which is with low trustworthiness. While in some caes, malicious users may join in the system, which report corrupted data or pollute sensor data deliberately. Additionally, it is possible for multiple malicious entities to launch collusion attacks [20] so that the participatory sensing system is of little use to a large scale of community. So, it is essential to create a criterion for server to do identity authentication and to evaluate the trustworthiness of information provided by users.

Energy consumption

It is the fact that mobile phones are not exclusively to be used for participatory sensing, whose main functions are making calls and accessing Internet. The measurement and sharing results for participatory sensing involve more than one interface such as: WIFI, GPS or camera. Thus, the battery will be consumed more quickly. Although most people charge their phones on a daily basis, it is better not to use up significant battery on participatory sensing so as to impede users from accessing their usual services. Or users are not able to gather data with insufficient power of mobile phones. Based on this, it requires participatory sensing applications to use sensors in a conservative way.

Data storage

Saving the collected data requires room in mobile phone, which is related to the size of memory. There will be a flood of data occupying lots of space of mobile phone after the user engaging in the participatory sensing for a long time. So how to solve the problem of memory and data logging is an essential factor for the final goal. For example, it is one way for the application developed in [11] to offer eight different options for its users to control how much data they want to store on their phone. All participatory sensing applications need to consider this question during development.

In conclusion, encouraging citizens to make use of their mobile phones for participatory sensing is opportunistic as well as challenging. It requires paying more attention when applying the emerging model to new domain so that to achieve the goal without bringing any other problems.

Chapter 3

Project Design

In this chapter, the whole design of the project is presented including the three parts organizing the system, the functional architecture to show the tasks of both mobile side and server side, how the privacy threats are solved by this application and who and in which case may use it. The introduction of system design helps to understand different components undertaking various responsibilities and explain how the key threats are eliminated.

3.1 System Components

Similar like other existing applications, there are three components to compose the system: mobile phone carrier, server and mobile user. This project is developed for Android smart phones which requires the embedded GPS sensor, embedded microphone sensor and the ability to connect network. Thus, users could use the mobile phones to collect, report sensing data, send requests and receive response from the server. The following figure describes the three components of this project and their works:

- Mobile Phone Carrier

This is the data producer of this project who installs this application on the mobile phone and then voluntarily contributes the sensing data. There are three tasks moving from

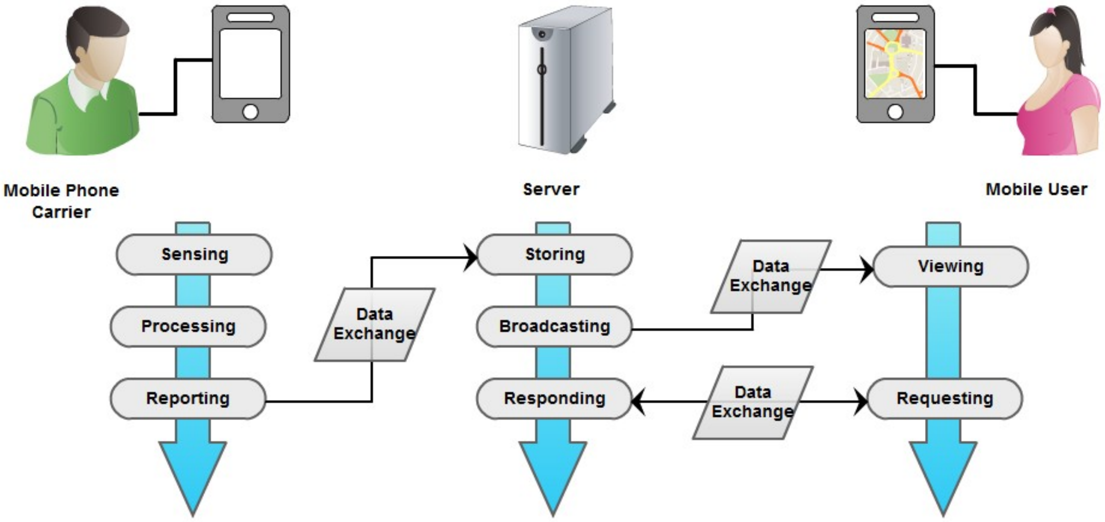


Figure 3.1: System Components

sensing to processing to reporting. These users will run the application when they want to collect the sound data. This application processes and stores the sensing data on the mobile phones before reporting, which means no original recordings will be stored in case to be misused or leaking privacy as well as enables users to share information whenever they want. These mobile phone carriers will complete the reporting task after they send the records to the server.

- Server

It plays as a bridge between other two parts and manages all reports and users. It also has three tasks including storing records, broadcasting and responding. Obviously, the server receives all records from mobile phone carriers and then it will do the further process before broadcasting in order to protect private information not to be leaked as well as to block the messages from malicious users. What's more, the server is capable to give appropriate response according to the users' requests.

- Mobile User

This component views the noise levels with location pins on the map, which are gathered by the mobile phone carriers. The reports in recent an hour are automatically shown when the application jumps to the map page. Additionally, mobile users are able to choose an interested period on the screen and get the corresponding results from server.

The following figure presents the detailed data types for different tasks of each component, which also shows the data exchange between each other.

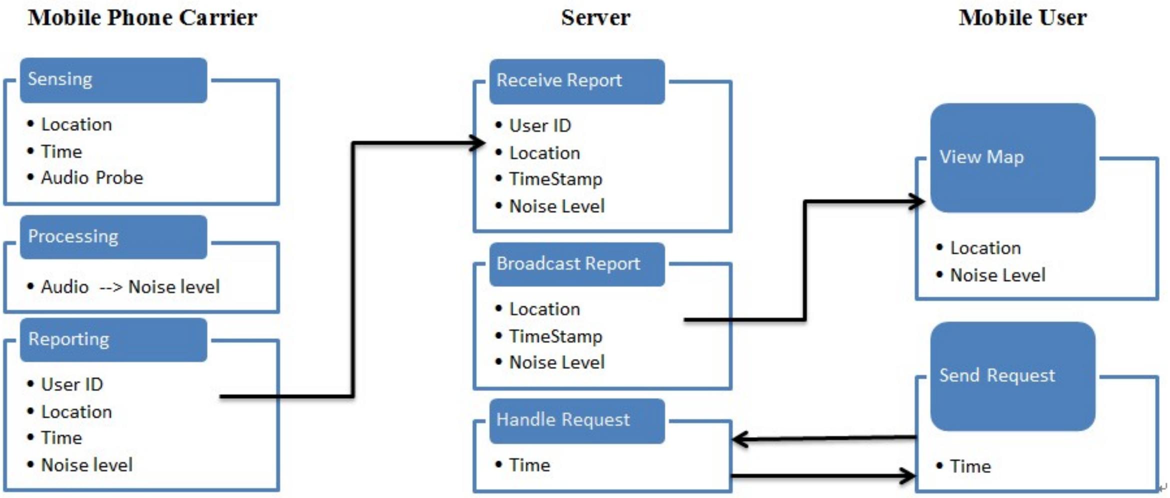


Figure 3.2: Probing Data Type

Based on the cooperation of the three components mentioned above, this application works well in noise detection and information sharing. All users act as both mobile phone carriers and mobile users, they collect as well as access the sensing data by the mobile phones they carried. The server communicates with both mobile phone carriers and mobile users by receiving sensing data and replying results respectively.

3.2 Functional Architecture

This project can be divided as mobile side and server side. The former has user interface on mobile phones that provides service for mobile phone carriers and mobile users while the server is the admin of this project that belongs to the server side.

This mobile phone application needs to be installed on the Android smart phones offering the services shown in figure 3.3:

3.2.1 Mobile

On the mobile side, there are three functions offered to meet the requirements of both mobile phone carriers and mobile users.

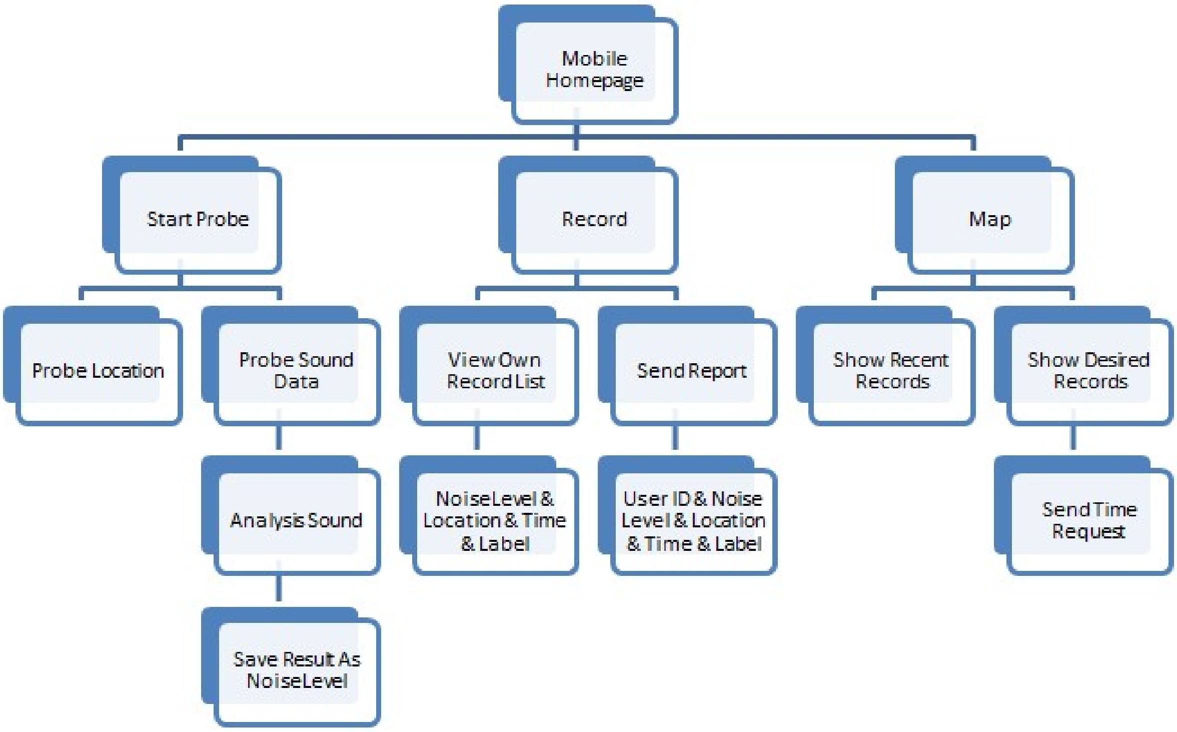


Figure 3.3: Mobile Side Functions

- Start Probe

This function enables the users to collect the audio data and location. At the same time, the application begins to count the total probing time that will be used during processing stage to calculate the average noise level. In order to protect the users' private information, original sound recordings will not be stored, but only analyzed and processed to noise level. Then the results of sound level will be store on the mobile phone.

- Record and Report

The record page shows the user's own history of sensing data and enables him/her to send data to the server. Every record shown on the mobile phone screen is consisted of the noise level, location, time and a label (”Reported” or ”Not Report yet”). The users can choose to send reports when they want and the label of every record will change from ”Not Report yet” to ”Report” after the server receives successfully. As shown in figure 3.3, the user ID will be created by the application automatically when the users choose to send reports, which is not visible from the screen.

Once the user clicks report button on the screen, the unreported records on the list will be transmitted to the server together. After that, these records will be marked as reported to avoid repeat sending same records.

- Map

It empowers all mobile users to view the shared information on the map. The application shows the records in recent one hour when the users first visit the map page.

Users can also view records in their desired time period by selecting the time on the map page. Because the minimum unite is one hour, so if a user selected a time manually, it will show the records within that hour. For example:

A user chooses: 2013, month: 01, day: 01, hour:11

It will display the records of ”2013-01-01 10 am to 11 am” on the map.

The records will show as heat points and different color represents different noise level. Generally there are four levels of noise, so the colors are:

1. Green (Mild): 40dB or lower
2. Yellow (Moderate): 40dB - 70dB
3. Orange (Severe): 70dB - 90 dB
4. Red (Profound): 90dB or greater

Both mobile phone carriers and mobile users belong to the mobile side and they need to be connected by the server.

3.2.2 Server

The functions of server are shown in the following figure:

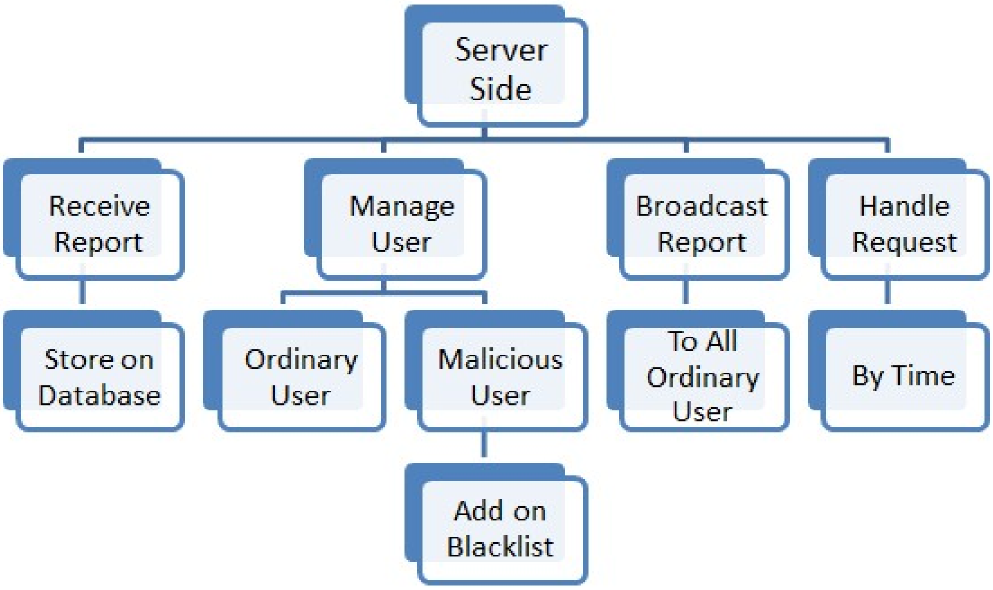


Figure 3.4: Server Side Functions

- Storage

The database on the server side stores the reports received from mobile phones. As described above, the user ID is included in every report so that the administer can view and check the source ID of every record before storing.
Every record will be checked by the server first. If the ID of one reporter has already contained in the blacklist, the reports from this user will be ignored.

- Managment

To recognize the unreliable users, the server may check data trustiness by comparing reports from different users. For example, a user may deliberately get high noise level by placing his phone near a radio. This data may be incorrect if other reports from the same location show the noise level is low. As a result, this user will be added into the blacklist by administer. The server will block all information from those on the blacklist and the previous records from the malicious users will never be shown on the map again. For the reports from ordinary users, the server stores them on the database.

- User ID Protection

The server removes the user ID before publicing reports to all ordinary users, which keeps the data anonymous to protect the identity of users.

- Respond

The server will make suitable responses after it receives requests from users.
1. Respond with records within last one hour
When the user initially loads the map page on the mobile phone, the server will receive a request that asks for recent records (within one hour). Then the server will check the database and send all matched records to the user.
2. Respond with records that user desired
Similarly, the server will also receive the request from users when they choose the time on the mobile phone screen. Then the server should check the user's required time and respond with corresponding records.

This application offers urban noise levels to the public relying on the working procedures of both mobile side and server side. The functions mentioned above enable the mobile phones to collect, process and view data as well as the server to manage, store and return reocrds.

3.3 Privacy Protection

The aim of participatory sensing is encouraging and enabling ordinary citizens to contribute and share sensing data in a large scale. However, there are different kinds of threats in existing participatory sensing applications developed for various areas. Certainly, this project also faces a few potential problems when encouraging user participation into noise detection. As shown in the following figure, it can be divided into four parts which may pose threats to users' privacy, involving collecting sound data, data transfer between two components, data storage on the server side and sharing sensing data with other users.

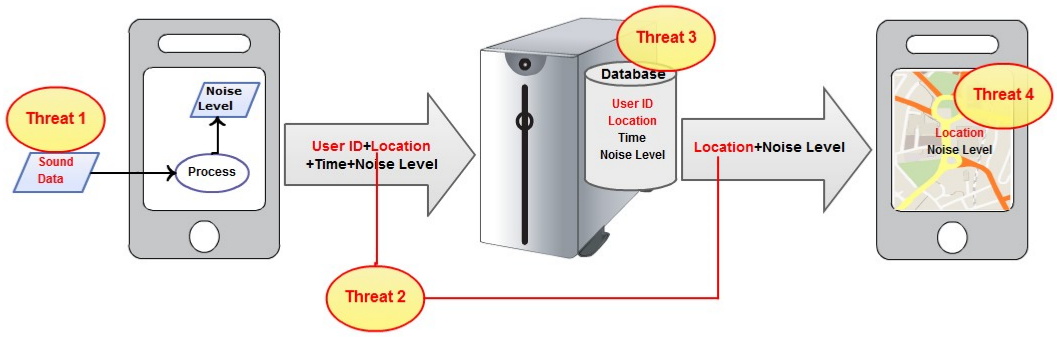


Figure 3.5: Potential Threats

After having outlined the potential privacy threats, this project makes the following efforts to address those problems:

3.3.1 Processing Sound Data Into Noise Level

As this application offers help to probe and share noise level, users need to collect the sound data first for further operation, which may cause people's concern if this application shares the original audio recordings with others including adversaries. For example, it is of great possibility to infer the gender, name, when and where of the user if he/she shares the sound sample of discussion on the work with other colleagues.

It may be a method to divid the whole sound data into several fragements and then reported to the server by different users, which has already described in chapter 2. For this project, it focus on the noise level rather than the content of sound, which is not necessary to restore original recordings. Additionally, the audio recordings increase the risk of revealing the context of users' conversation.

Considering the potential threats descried above and the requirment of this project, the

Noise Manager processes the sensing sound data into noise levels before storage instead of storing users' audio recordings on the database directly. There are four noise levels defined by this application: mild, moderate, server and profound, representing the sound volume from low to high. Both mobile phones and database on the server side will store the noise level in order to reporting and sharing respectively. Thus, all users are clearly aware of the noise level of one area without privacy concern.

3.3.2 Encryption During Data Transfer And Data Storage

The sensing data transmits in two directions between mobile side and server side and stores on both sides. As "Threat 2" and "Threat 3" shown in figure 3.5, the user ID, location and time information may be intercepted during data transfer and stolen from the server's database, which inevitably reflects users' private information when analyzing them together.

As we know, most applications prefer to user registration by using user name and password. This usually requires users to offer some personal information, such as: email, age and so on. To protect these data, it is common to use complex encryption method before storing.

The mobile phone carriers in this project are responsible for data collection and report. The server needs to identify different users in order to better managment. As user ID is related to users' personal information, the Noise Manager requires as less private information as possible. So it uses the device number as user ID and the mobile side will get it automatically when users choose to report data. In this way, this project requires and controls less users' information and discards the registration procedure.

Additionally, it is necessary to provide a solution covering report procedure and storage stage to guarantee data security. This project encrypts user ID before sending from mobile side as well as when storing on the database. As shown in the above figure, the user ID is a 16-digital combination of numbers and alphabets. So this project plans to use the key

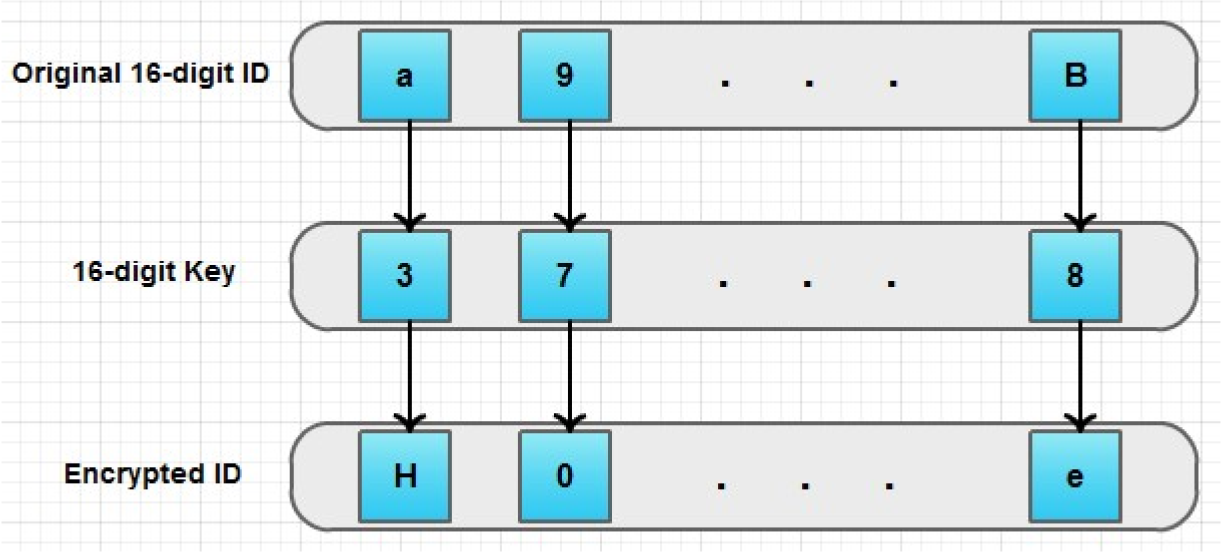


Figure 3.6: Encrypting Process

that also has 16 numbers in order to correspond to every digital of user ID. The processed result will be totally different from the original user ID, which can be only decrypted by the server. So, the encrypted user ID is meaningless for anyone without right decryption key. Although the location and other information can be correctly accessed, they also do not make any sense when the owner is anonymous.

3.3.3 Anonymity For Data Sharing

The final presentation of this application is on the map including the noise levels with corresponding location pins. "Threat 4" labeled on figure 3.5 points to the possibility that unknown third party could infer the users' identity from the location information. As user ID is not necessary information for urban noise mapping. In this project, data sharing keeps anonymous among all users, which is implemented by the process on the server side. It means all sensing data coming from the mobile phone carriers is analyzed and handled and then the server removes the user ID before broadcasting to the mobile users. In consequence, all users have no worry about leaking their privacy as well as the separate location information does not make any sense for adversaries.

3.4 Scenarios Of Using This Application

This project targets the noise detection and noise information sharing in urban areas, which mainly serves for ordinary people using Android smart phones. As all users can access and query the data gathered by themselves and others, it is helpful for ordinary citizen to reuse the data for their purpose as well as for some government officials to get data for their work.

3.4.1 For Ordinary Citizens

Nowadays, it is convenient for people to do lots of things by installing corresponding applications on their smart phones, such as: locate nearest gas station, preview the road traffic to plan their journey and so on. This project aims to empower ordinary citizens to know and share the noise level around them as well as view the noise information in certain period of some areas on the map according to their interests.

- Find a quiet route

  There are a few routes from the house of a student in Trinity to the college. He could find a quiet route by using this application. The heat map of this project shows the noise data collected by users before and allows all users to review so that the quiet route can be easily chosen.

- Rent an apartment in quiet area

  An office lady cannot decide which apartment she should rent between two nice ones. She can seek help from this application by reviewing the heat map of noise level around the apartments. According to the previous data, it is better for the office lady to choose the apartment in the quieter area.

- Compare the surrounding noise level of kindergartens

  Parents may use this application to compare the better kindergarten for their children. By collecting and reviewing the surrounding noise level for a period, parents could roughly know the average sound level of kindergartens before they make any decisions.

- When my house has the lowest noise level during the day
  I just wonder whether my house is quiet or not and when it is the quietest during the day. I can run the application once an hour and report it so that the corresponding result will show on the map. In this way, I could review all the results in the evening and learn the noise level of my house to get the answer.

- Check the surrounding noise level of a hotel
  Two friends will travel together and they need to book a hotel for three days. The introduction of a hotel is very nice and they want to know whether it is quiet at night before their booking. They can open this application and review the surrounding noise level on the heat map. They can also run the application to measure the noise level by themselves when they live in order to give recommendation for others in the future.

- Probe the sound level of the emergency alert
  It is common for everyone to hear the emergency alerts of Garda or ambulance when walking on the street. The sound level of one location can be affected obviously if the emergency vehicles often appear. People can use this application to probe the sound level and share with others.

Besides the examples listed above, there are many other scenarios for individuals to use this application in their daily life. It is the goal of this project to encourage more and more people to involve as well as to provide convenience for them.

3.4.2 For Government Officials

Noise Manager is part of the Personal City Project, which helps to supervise and manage the resource in the city. Government officials could view the noise level of different areas in various periods so that they may make some regulations or modification to deal with the noise pollution. This project can be one source of the required sound data as it is large scale sensing and also categories all records according to every hour.

- Rank the noise level of the different areas in Dublin

  As the noise pollution attracts more and more attention, a government official may collect and record the noise level of different areas in Dublin for every month in order to rank the quietest block for people to live. The government official could use this application to measure the noise level when he is in the area or view the map on this application at certain frequency so that he can record and summarize the data to rank.

- Record the sound level of the O'Connell Street of a day

  A government official wants to know how the noise level of O'Connell Street varies of a day. He could view on this application and record corresponding data while assigning the patrolling police officers carried with this application and run it once an hour.

- Launch an activity for noise monitoring

  To control and manage urban noise levels, government officials can encourage citizens to install this application to report real time noise level anywhere they may be. This activity helps to save the human resources and money investment to master more and new noise information of our city. At the same time, it can be regarded as a measurement for developing people-centric city as citizens make their own contributions.

Although the quality of the sensing data cannot achieve exact accuracy, this application is still capable for assisting officers' work as it has large scale sensing data to guarantee the quantity.

For a short summary, the chapter first introduces of the system structure including three components and functions of different components with the aid of diagrams. Then it discusses the common methods of privacy protection and then propose its own way to solve potential threats. The scenarios are only a few examples for the target customers to use this application. This chapter draws a basic picture of the project as well as sets the final goal for implementation.

Chapter 4

Implementation

The implementation procedure of Noise Manager will be described in detail in chapter 4. This application employs standard client-server architecture, which is mainly written in Java language. The development process starts from the mobile user side including graphic user interface, collecting and processing data, then works on the database, broadcasting and replying of the server and finally connecting both sides to communicate.

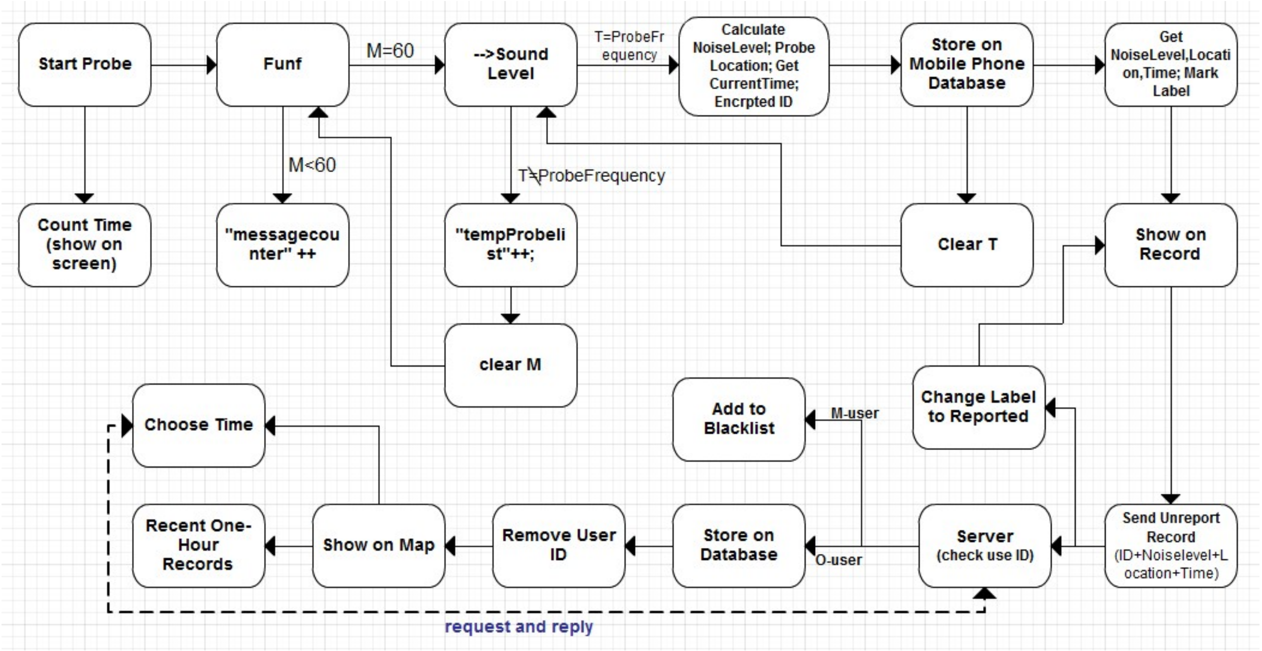


Figure 4.1: Data Flow

The figure above is the data flow of this project, from probing, processing, storage to report and showing.

4.1 Collecting Data

This application gets data from Funf open sensing framework that is exclusively sensing and processing data for mobile phones. There are lots of sensing functions provided by Funf while this project only relies on two types of data (location and audio) and gets the time and user ID by itself, as shown in the following figure.

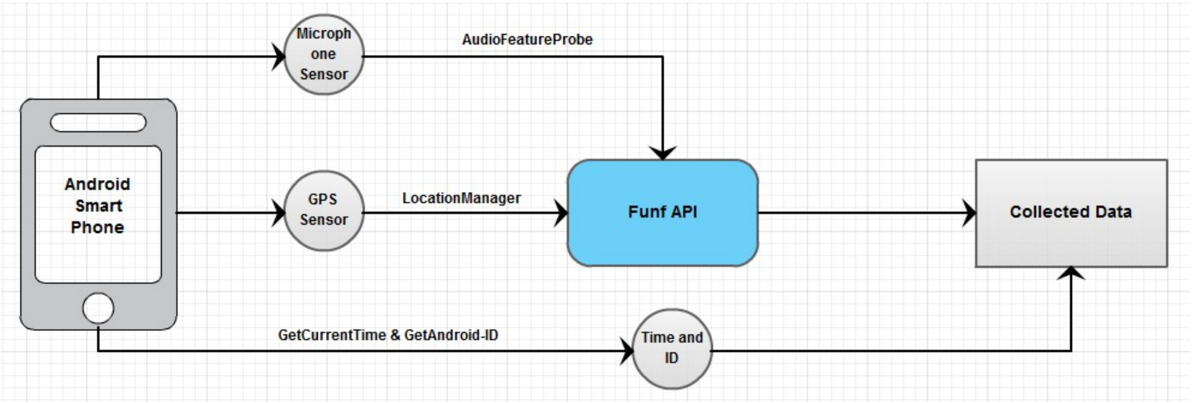


Figure 4.2: Collected Data

When the users run the application by clicking "Start Probe" button, it enables the Funf API to work, which probes audio data every second using the microphone sensor embedded in the mobile phone. The location information will be gathered only when this project stores data. The two sensing data can be collected by the methods offered by the Funf:

```
funfManager = ((FunfManager.LocalBinder)service).getManager();
afp = gson.fromJson(new JsonObject(), AudioFeaturesProbe.class);
locationManager = (LocationManager) getSystemService(Context.LOCATION-SERVICE);
```

However, this application has its own frequency to get and process data. Thus this project creates the ”messagecounter” variable to control the quantity of sensing data provided by Funf. As shown in figure 4.1, when the message counter equals to 60, which means one minute, this application will get the sensing data and then temporarily stores in a list called ”tempProbelist”. Then the ”messagecounter” will be cleared to start counting for the next time to get audio data. The location data is probed by the embedded GPS sensor only according to the storage frequency that is five minutes. The mobile phones also automatically get the current time and device number (as user ID) every five-minute when it stores data on the mobile side database. The data is gathered by the following methods and then the user ID will be encrypted:

```
Timestamp stamp = new Timestamp(System.currentTimeMillis());
String deviceID = Secure.getString(this.getContentResolver(),Secure.ANDROID-ID);
public String encryption(String input);
```

Two sensing data (sound level and location), time and user ID can be collected at different frequency, which is the first step of this project and executed completely by the mobile side.

4.2 Processing Data

It moves to this step after data collection. Processing data in this project can be divided into two parts: mobile phone and server. The former is mainly responsible for making original data meaningful and protecting privacy of mobile phone carriers. The server executes this step in order to manage all users and keep records anonymous. Privacy protection relies on the data processing of the two parts.

4.2.1 Mobile Phone

After importing the Funf Manager Classes and related libraries, it implements the basic audio probing function showing six fields pieces in numbers:

- diffsecs: passing time
- PsdAcrossFrequencyBands: power spectral density at different frequency
- mfcs: Mel-frequency cepstral coefficients
- L1: l1Norm
- L2: l2Norm
- Timestamp: current time

These original data that reflects the sound volume in different aspects needs further process in order to make sense for this project. L1Norm is used to calculate the sound level by applying the following formula provided by Funf:

SoundLevel (dB) = 10*log10(l1Norm*l1Norm)

As the figure 4.1 shows, "M=60" means every minute this application gets data from Funf API, which will be processed by the above formula to achieve a sound level in dB. When it is the time to store the sensing data, all sound levels that temporarily put in the "tempProbelist" will be added together and then divided by the size of the list, in order to get a more accurate result.

For the user ID, this application uses device number to identify different users instead of asking any personal information, such as: email. To make it safer, the user ID will be encrypted before stored on the database of the mobile phone.

A 16-digital number key is corresponding to 16-digital user ID that is a mixture of numbers and alphabets. The current key of this project is "7183669393867201" and can be reset in the code.

Figure 4.3 is an example to illustrate the encryption procedure. Firstly, every digit of user ID should be changed to the corresponding character value that will be shifted by the 16-digit key. Then the temporary result is a string that stores the processed character values. Finally the encrypted user ID is consisted of 16 characters changing from the value of every digit in the temporary result, which may be symbols, alphabets and numbers.

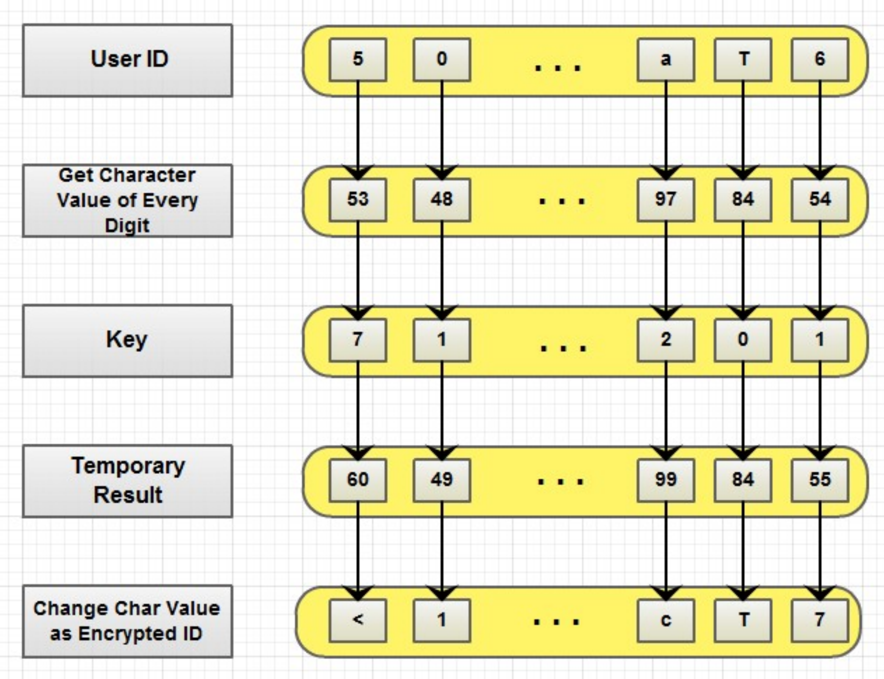


Figure 4.3: Encryption

Another process of the mobile side is ”Change Label to Reported” that happens after the records were sent to the server. Users are able to view their previous sensing data on the ”Record” page. As shown in figure 4.1, every record includes three types of data gotten from database and a label, which marks as ”Not Report yet!” if it has not been sent to the server, otherwise the label changes to ”Reported!” . Every time the users choose to send report, all unreported records will be sent.

The mobile phone side is mainly responsible for the process of audio data and user ID, which aims to eliminate the threats to users' private information when they use this application.

4.2.2 Server

This project is Tomcat web service that using REST technique particularly POST and GET actions to implement the communication between the mobile phone and the server. The "Send Report" function of mobile side only involves posting data to server without any getting response process. Based on successfully received data from mobile side, the server needs to do some process before broadcasting to all users. Firstly, the user's identity will be checked in order to distinguish reliable users and malicious users.

There is a blacklist on the server side recording those who had bad behaviors previously. The server compares the received user ID with those on the blacklist. As shown in figure 4.1, the "M-user" means malicious user that will be added on the blacklist and the "O-use" is ordinary users whose reports will forward to the next step. For messages offered by malicious users, the server will block them forever and never show their previous reports on the map.

The next process on the server side is removing the user ID of ordinary users. The identity of users is shared with the server while keeps anonymous with other users. The server only shares the noise level and location without user ID, which protects the users' privacy.

4.3 Noise Level

As can be seen in figure 4.1, the items stored on the mobile phone database are not exactly the same as those shown on the screen. Besides user ID will not be seen, the sound levels are in different formats: they are represented in numbers on database of both mobile side and server side while shown in words on the screen.

Sound is necessary in people's day to day life while noise is not. The influence of noise on human emotion and health varies according to the level as well as the exposure time. According to the research of noise on hearing impairment in [28], this application classifies noise into four levels: mild, moderate, severe and profound. It gets the noise level in dB after processing (section 4.2.1), which stores on the database first. When showing on the record list for users, the numbers representing noised level will be judged to get the corresponding noise category, see table 4.1.

Table 4.1: Noise Category

Category	dB	Colour on Map
Mild	40 or Lower	Green
Moderate	40-70	Yellow
Severe	70-90	Orange
Profound	90 or Greater	Red

This application also allows viewing the noise level on the map that attempts to show the result more straightly. So, different noise categories have corresponding colors to mark on the reported location when jumping to the map page.

4.4 Data Storage

There are two databases of this project, one is on the mobile phone and the other is on the server. Figure 4.1 reveals when data storage happens on both sides and what items are saved, which will be explained below in details.

4.4.1 On Mobile Phone

The storage frequency of Noise Manager is every five minutes set by the ”probeFrequency” variable and at that time the ”tempProbelist” will be cleared for next use. The average noise level, probed location, encrypted user ID, current time and report label consist of a record to store on the mobile phone for further showing on the screen as well as sending to the server. Data storage on mobile phone is forever meaning users cannot delete any items unless they uninstall this application. The table structure shows in table 4.2.
There are four functions offered by the Database Class on the Android Project supporting senior actions of this application:

- Add A Record: to save a new record at storage frequency
- Get A Record: to process all records one by one
- Update A Record: to change the label of every record after reported
- Get All Records: to get all records for further process to show on screen and send to the server

Table 4.2: Table on Mobile Side Database

Field	Type
id	INT
soundLevel	INT
lat	DOUBLE
lng	DOUBLE
time	LONG
report	INT
deviceID	STRING

4.4.2 On Server

There are two tables on the database of server side named as record-list and blacklist to store information of ordinary users and malicious users respectively. It is clearly seen in figure 4.1 that the server checks every user ID to decide where to store the reports. Reports from ordinary users will be saved in detail in the record-list table while the blacklist table only saves the user ID of unreliable users; the following are the structure of two tables.

Table 4.3: Blacklist Table

Field	Type
id	INT
deviceID	STRING

Table 4.4: Record-list Table

Field	Type
id	INT
soundLevel	INT
lat	DOUBLE
lng	DOUBLE
time	LONG
deviceID	STRING

For those users on blacklist, their reports will be blocked by the server and their previous records cannot be shown on the map anymore. However, reports from malicious users that have already been stored on the database will not be deleted, which can be regarded as the evidence.

The Database Control Class on Java EE provides three functions to support senior actions of the server:

- Get All Records On Record-list: get all reports from ordinary users to process for showing on the map
- Get All Records On Blacklist: identity the user ID to block their reports
- Add Report: store every record from ordinary users

4.5 Result Map

As both sending report and viewing map are the messages from the mobile phone to the server while the former needs to be stored but the latter should be given response without storage. To distinguish the two types of communication, this project sets the ”checktime” variable as ”0” to represent the action of sending report; otherwise it means the viewing map requests from mobile users.

The functions mentioned above are implemented by the RESTful web service that uses standard HTTP, POST and GET actions to submit and retrieve data that is transmitted by the JSON object.

All users of this application are able to view the noise level gathered by others and themselves on the map. Different color circles vary from the lowest noise level to the highest, which locate at the sensing place. This application shows records in recent one hour on the map by clicking the "Map" button on the mobile phone screen while users can also view their interested records after choosing the time on the map page.

4.5.1 Show Recent Records

After the server receives the request to show the records in last one hour, it will get system current time that should be transformed from long type to the "yyyyMMddHH" format. As this project divides all records into every hour, the minute and second have no effect to the final result. There are two other variables helping to set the range of the time, one is "start" that is the number of the hour of the current time and the other is "finish" which equals to the number plus one. Then the server will get all records from the record-list table and compare the time of every record with two variables above.

For example, imagine the number of the hour of one record is 19, the "start" will be 19 and the "finish" will be 20. The time of every record should be larger or equal to the "start" and also smaller than the "finish", then this record meets the requirement and will be shown on the map. Otherwise the record is not reported within the recent one hour. The server will return all required records after processing one by one.

4.5.2 Show Records In Interested Time

On the top of the map page, users are able to choose their interested time and then view corresponding records by clicking "search" button. The selected time will be sent from the mobile phone, which is set as the time instead of getting the current time by the server. It is the same as the process procedure described above so that users can view previous data whenever they want.

4.6 User Interface Overview

Based on the implementation procedure introduced before, this project successfully provides the functions of noise collection and mapping for Android smart phones. All users can use this mobile phone application after installing without registration. This application enables users to measure noise level of their surroundings, to view their previous sensing records and to share data on map, which can be illustrated with the aid of screenshots below.

- Homepage: it shows this page when users open the application, presenting all functions as a list.

- Probing page: this application shows figure 4.4 when users first jump to the probing page and then it shows figure 4.5 when users run this application to collect noise data by clicking the button at the bottom of this page.

- Record page: all previous sensing data of the user will be shown here.

- Map page: it shares the noise level as well as the location here. All users not only can see recent records but also their interested ones by choosing the time on the top of the screen.

- Help page: "Readme" file helps users to be familiar with this application and also explains the basic information of how this application works.

In conclusion, Chapter 4 presents how this application looks like and how it works. The Noise Manager is implemented based on the communication between the mobile phone and the server. The whole procedure requires lots of technical support in data collecting, processing, exchanging, sharing and storage and so on. To achieve the goal of privacy protection, the implementation put an emphasis on the process of user ID and audio data, which finally gets a good result. This chapter briefly describes the development of this project as well as completes the tasks of the design.

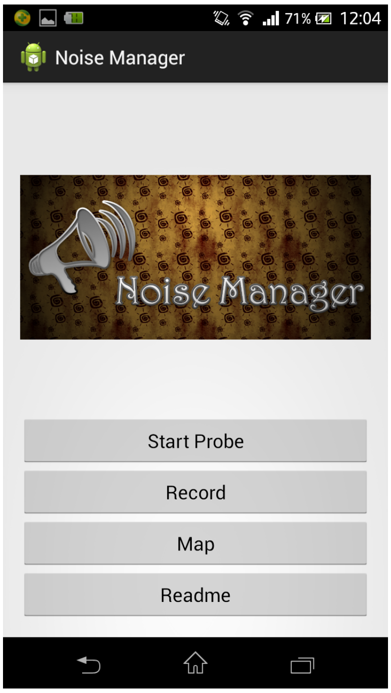
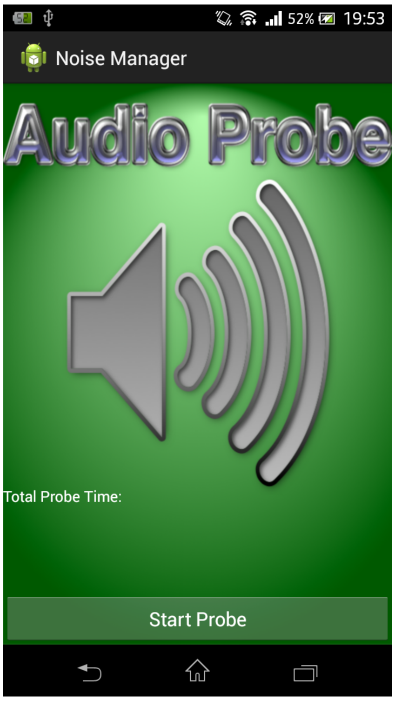
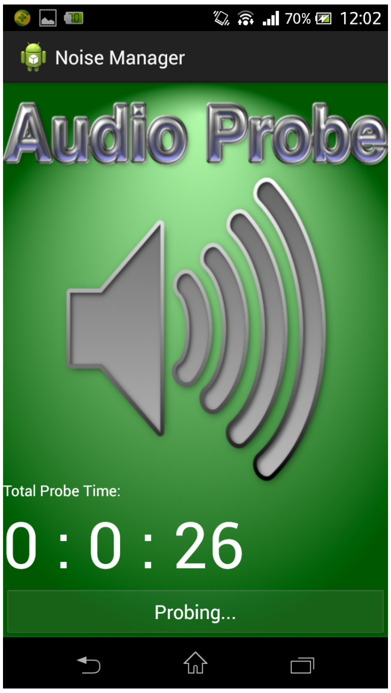


Figure 4.4: Homepage



Total Probe Time:

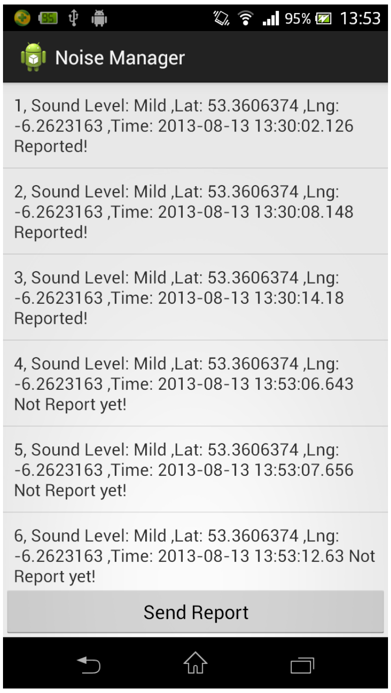
Figure 4.5: Probe page



Total Probe Time:

0 : 0 : 26

Figure 4.6: Start Probing



1, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:30:02.126 Reported!

2, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:30:08.148 Reported!

3, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:30:14.18 Reported!

4, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:53:06.643 Not Report yet!

5, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:53:07.656 Not Report yet!

6, Sound Level: Mild ,Lat: 53.3606374 ,Lng: -6.2623163 ,Time: 2013-08-13 13:53:12.63 Not Report yet!

Figure 4.7: Record page

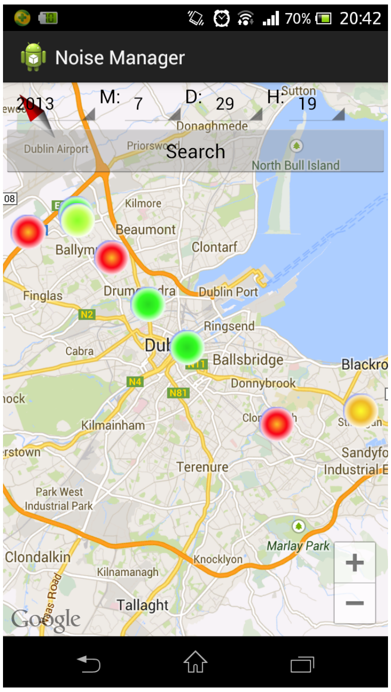


Figure 4.8: Map page: show results

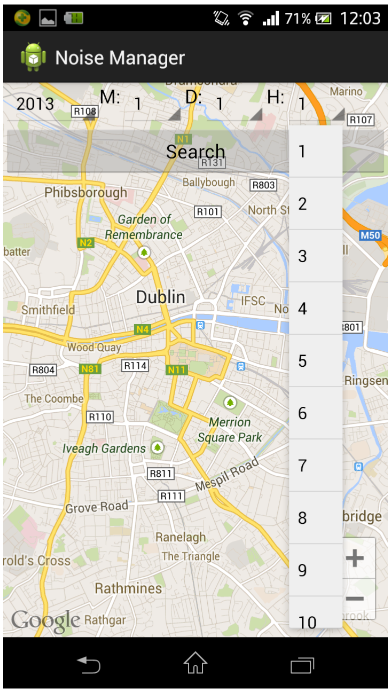
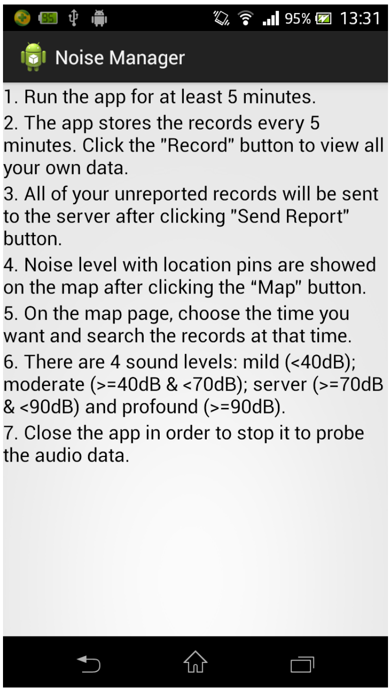


Figure 4.9: Map page: choose time



1. Run the app for at least 5 minutes.

2. The app stores the records every 5 minutes. Click the "Record" button to view all your own data.

3. All of your unreported records will be sent to the server after clicking "Send Report" button.

4. Noise level with location pins are showed on the map after clicking the “Map” button.

5. On the map page, choose the time you want and search the records at that time.

6. There are 4 sound levels: mild (<40dB); moderate (>=40dB & <70dB); server (>=70dB & <90dB) and profound (>=90dB).

7. Close the app in order to stop it to probe the audio data.

Figure 4.10: Help page

Chapter 5

Evaluation

This chapter is the final test of the Noise Manager after the implementation. Volunteers are recruited to use this application and give feedback. The whole evaluation procedure and its results are described in the following sections. According to the information of the experiment, it discusses strengths and limitations of the current work. This is done to evaluate the usability and reliability of this project as well as to help improvement in the future.

5.1 Evaluation Procedure

The first section is the basic information and operation of this experiment:

- Goal: test whether the privacy of users is protected when they use this application
- Number of participants: 7
- Recruitment method: Email
- Duration: one week
- Procedure

All of the participants have Android smart phones and before starting the experiment, they attend to a meeting about how this application works and what they need to do. Then participants should download the application package file (APK) of the Noise Manager and install on their phones so that they are able to use the application at any locations whenever they want.

This project uses Tomcat server and MYSQL database. The IP address of the server has been set in the code so that every mobile phone can connect to the same server by the network after installing this application. As the limited size of the users, they are expected to run the application at least 3 times a day.

It is necessary to contact the participants if there is no data in the database sending from them all the day. Certainly, all participants can also report any problems they meet during the test. Additionally, participants have the right to quit at any time when they feel their privacy threatened.

At the end of the test, it is grateful for all participants to do a questionnaire to give the feedback. The questions are showing as below.

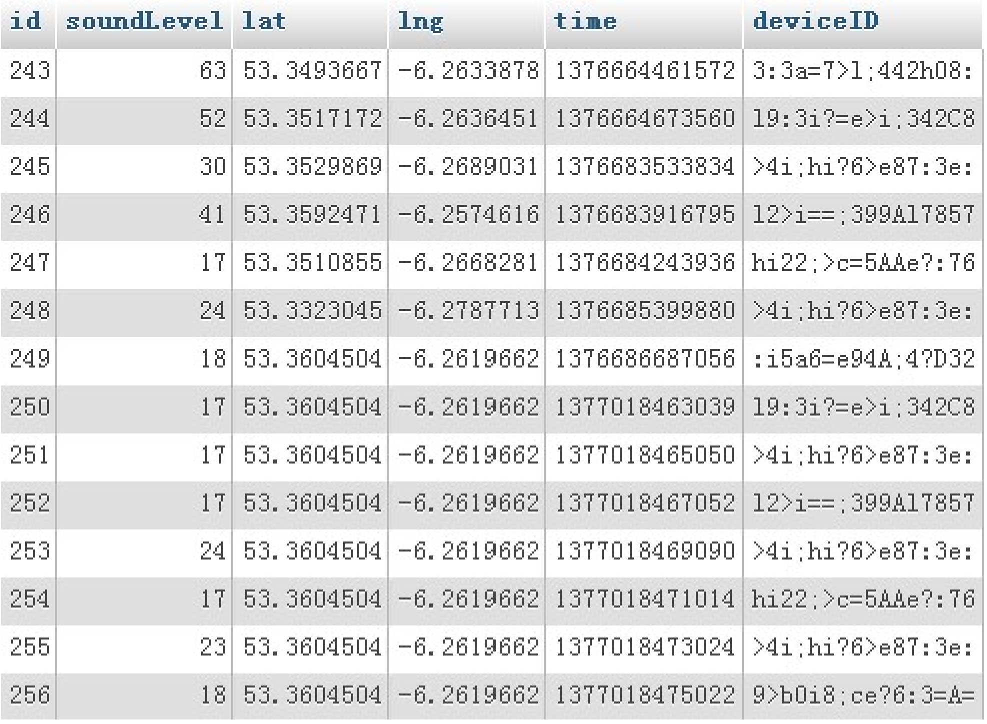
- How often did you submit reports per day?
- Which level of privacy protection do you think this project offers? (low, medium, high)
- Have you ever known any other users' identity during the experiment? If so, how?
- Do you think your identity or private information which you do not want to share was revealed during the experiment? If so, how?
- Do you think this project implement privacy-aware urban noise mapping?
- What would you like to suggest for this project to be improved?

The questionnaire is mainly about the privacy protection and suggestions for improvement, which aims to analyze the final results of the evaluation and identify potential issues to be solved.

The above roughly describes the goal of the evaluation and how it proceeds. Following the steps, the evaluation of the Noise Manager did well for getting desired results and finding a few problems, which is going to be discussed in the next section.

5.2 Evaluation Analysis

The evaluation lasts one week and totally there are 256 records (Figure 5.1) received and stored on the database of the server side. Seven participants are able to collect and share data during the test, which basically achieves the goal of applying participatory sensing concept into noise mapping. The following presents the final results of the evaluation and discusses strengths and limitations of current work.



id	soundLevel	lat	lng	time	deviceID
243	63	53.3493667	-6.2633878	1376664461572	3:3a=7>l;442h08:
244	52	53.3517172	-6.2636451	1376664673560	19:3i?=e>i;342C8
245	30	53.3529869	-6.2689031	1376683533834	>4i;hi?6>e87:3e:
246	41	53.3592471	-6.2574616	1376683916795	12>i==;399A17857
247	17	53.3510855	-6.2668281	1376684243936	hi22;>c=5AAe?:76
248	24	53.3323045	-6.2787713	1376685399880	>4i;hi?6>e87:3e:
249	18	53.3604504	-6.2619662	1376686687056	:i5a6=e94A;4?D32
250	17	53.3604504	-6.2619662	1377018463039	19:3i?=e>i;342C8
251	17	53.3604504	-6.2619662	1377018465050	>4i;hi?6>e87:3e:
252	17	53.3604504	-6.2619662	1377018467052	12>i==;399A17857
253	24	53.3604504	-6.2619662	1377018469090	>4i;hi?6>e87:3e:
254	17	53.3604504	-6.2619662	1377018471014	hi22;>c=5AAe?:76
255	23	53.3604504	-6.2619662	1377018473024	>4i;hi?6>e87:3e:
256	18	53.3604504	-6.2619662	1377018475022	9>b0i8;ce?6:3=A=

Figure 5.1: Records on Database

5.2.1 Result Discussion

The figure above presents part of the database storing records from the participants during the test. As introduced the data storage on server side in chapter 4, it can be seen clearly that there are six fields to save the reports from ordinary users in the ”record-list” table. The ”deviceID” is the user ID of this project, which have been encrypted before storing. Although the device number cannot indirectly reflect any private information of users, data encryption makes the information totally no sense for adversaries without decrypted method.

As there are no malicious users of the seven participants, we add the device number of one test phone on the blacklist to test the function of blocking reports from unreliable users. As a result, it is proved to be successful because no participants can view the records from this phone.

During the evaluation, there is no participant withdrawing, which reflects they never feel extra burden and their private information are not exposed when using this application. Here is the summary of the feedback from participants:

Table 5.1: Feedback from Participants

Participants	No.1	No.2	No.3	No.4	No.5	No.6	No.7
Average Report Frequency	3/day	3/day	2/day	5/day	3/day	2/day	3/day
Reveal Own Privacy	No	No	No	No	No	No	No
Know Others' Identity	No	No	No	No	No	No	No
Privacy-Protecting Level	High	Medium	High	High	Medium	High	High

From the questionnaire collected from the seven participants, the advantages of this project and suggestions can be organized as the following points:

- Implement privacy-aware urban noise mapping
- Never know others' personal information

- Never reveal own privacy
- Hide context of users' conversation
- Two of the participants prefer to registration
- Show all results instead of records in recent one hour

This project is a case study of the Personal Cities project attempting to investigate the privacy threats associated to using participatory sensing for noise mapping. All users should be empowered to collect and share the noise data and then it is possible to discuss the privacy protection. The evaluation procedure exactly proves that all the participants are able to use this application to probe noise data and then share. And the privacy protection offered by this application makes users to engage in urban noise mapping without privacy concern.

The question of report frequency helps to investigate whether privacy exposure is related to the report frequency. It is of great possibility to reveal one's identity if he shares his user name and location in high frequency. Due to data anonymity when sharing noise level in this project, the identity of participants will not be revealed even they report up to five times a day. The final result shows the Noise Manager reveals no privacy of users no matter how frequent they report.

For the most important test goal, the result of the evaluation points that all participants have no idea of the owner of the sharing data and they never feel their private information is exposed. Certainly, this conclusion has limitation as there are too few participants engaged in the evaluation. While the privacy protection can be achieved currently with small number of users, it means this project is successful so far and may also perform excellent in the future.

As discussed in chapter 3, the sound data may reveal the context of users' conversation so that it requires to process well to solve the threat. This project saves and shares the noise levels instead of original recordings to protect users' conversation context. The final result proves this project works well as no one can access the audio recordings.

For some suggestions, it should mention the registration feature. Two participants think it is necessary to protect own information. Without correct user name and password, no one can access my records. While other participants held the view that it is convenient without registration. At the initial stage of the project design, the Noise Manager plans to add the registration feature. However, it gives up the idea in order to protect the privacy better. This project requires and saves as less personal information as possible. Additionally, the stored data format of the Noise Manager is of littel possibility to reveal neither user identity nor their conversation context even others view previous records gathered by the users.

Another good suggestion is to show all noise levels stored on the database when first jump to the map page. It may be more useful and convenient for users who want to know the noise levels of some areas. Currently, the Noise Manager shows the records in recent one hour, which may not enough to reflect the noise level changing trend and overall level of one location. So, this suggestion is useful and important for the improvement of this project.

So far, according to the feedback from participants, it is satisfied for privacy protection method provided by this application. Users can collect and share noise data in various scenarios without privacy concern. The Noise Manager successfully implements privacy-aware urban noise mapping.

5.2.2 Limitations

The evaluation results verify the strengths of this project as mentioned above while there are also some shortcomings found during the test. Summarizing the feedback from participants, the following four points are the main problems:

- Report failed and cannot query
- One location has multiple noise levels at the same time

- Different users share the same user ID
- Limited size of test users

Currently, the server of this project is not online so that it creates a local-host network running all the day to simulate. All participants are able to transferring data with the server by accessing the IP address of the server's network. In this way, the server may stop working due to no network signals and it cannot restart automatically. Reports from participants will not be received and stored by the server whereas the labels of every record on the mobile side have been changed from "unreported" to "reported". As a result, those records reported when the server is not working are wasted, which cannot be sent again after the label of records have been marked as "reported".

Similarly, the disconnection with server may cause this case that the participants get no results when they want to query the records in interested time. After choosing the time on the mobile phone screen, the request needs to be sent to the server which returns corresponding results. It is impossible to transmit data when the server is not running. Certainly, the participants will not get any results for their query if there are no reports at that period.

The Noise Manager enables all users to search based on every hour while the storage frequency of sensing data is every five minutes. It may show multiple noise levels or present incorrect noise level at the same location when the user runs this application for a long time at that position. Especially when the noise level of one location varies violently within one hour, all users possibly see different noise levels every time after they click the location pin.

As mentioned before, this application sets the device number as user ID that is not shown on the screen for the user. This method definitely protects users' privacy as it involves on personal information. However, it also brings a few problems; for example, users do not know their own ID and different users can share the same ID. If someone deliberately send malicious information by using one's mobile phone, the device number of this phone

will be add on the blacklist so that any reports or requests from this mobile phone will be blocked, which means its owner cannot use this application unless he changes a new phone.
Last but not least, one obvious shortcoming of the evaluation is the limited size of the participants, which results in the small number of the sensing data such as: no results in some time. From the feedback of seven participants, this application achieves the goal of privacy protection that is the emphasis of this project when applying participatory sensing into noise mapping. However, there are still some problems found even tested by a few users. So, to be better test and improve this project, the evaluation requires to be done in a larger scale.

In short, the evaluation of this project proves the achievements of current work as well as finds some issues. Due to the time and device limitation, there are only seven participants installing and using this application for one week, which helps to test in a very small size. To get more convinced result and identify potential issues, it is necessary to do the evaluation with more participants for a longer time.

Chapter 6

Conclusion

This dissertation presents the experience for design, implementation and evaluation of the Noise Manager that aims to offer privacy protection when using participatory sensing for urban noise mapping. There are two research questions of this dissertation:

- Investigate privacy threats associated to using participatory sensing for noise mapping
- Seek an efficient method to enable people to engage in urban noise mapping without privacy concern

In order to get answers for the research questions, the whole procedure of this dissertation starts from the state of the art to have a basic idea of participatory sensing and common privacy problems. Then it moves to the detailed analysis of using participatory sensing for urban noise mapping and constructs the system architecture of this project. According to the system design, this project implements from the mobile side to the server side and finally connects the two parts to communicate successfully. Last but not least, it is necessary to do the evaluation work after implementation, which is processed by recruiting participants to engage for one week. The results summarized from the user questionnaire prove this project achieving its goals and also identify some problems.

By analyzing the potential privacy threats associated to using participatory sensing for noise mapping, it can be categorized into four threats existing in the noise data collection, exchange, storage and sharing. This project proposes corresponding solutions to solve the problems mentioned above:

- Threat in sound data collection: the mobile side of this project processes the sound data into noise level, which eliminates the potential threat of revealing the context of user' conversation.

- Threat in data exchange and storage: to solve this problem, this project requires as less personal information as possible to save and transmit. The mobile side automatically gets the device number as user ID and also encrypts before exchanging and storing.

- Threat in data sharing: the server removes the user ID before broadcasting the data to all users, which means the user ID is only shared with the serve while keeps anonymous with others.

Based on the essential privacy protection methods, this project is implemented step by step to achieve all features proposed in the system design. Finally, the mobile side has the functions including probing noise data, viewing own records and sharing sensing data with others, while the server is responsible for data storage, broadcasting and responding. This application works well relying on the cooperation and communication the both mobile side and server side.

Due to the time and device limitations, the evaluation of this project lasts one week by recruiting only seven participants. The final result is satisfied as this application never reveals any users' privacy no matter the report frequency is high or low. Additionally, all participants accept the privacy protection method offered by this project and they think the privacy-protecting level is fine. However, there are a few problems found during the test, such as: report failed because of unstable network connection and no results of queries as there is no corresponding data stored, which has been analyzed before.

One potential shortcoming of this project is the server can know the common places of users. This problem cannot be avoided as the Noise Manager enables all users to report directly to the server. Although all user ID have been encrypted, the server is able to decrypt the data when necessary. According to this, it can be improved by raising the reliability of the server so as to never leaking users' privacy.

In conclusion, the Noise Manager is standard client-server architecture, which implements privacy-aware urban noise mapping for Android smart phones users. From the current evaluation, it proves that this project provides an efficient way to enable citizens to engage in participatory sensing for urban noise mapping without privacy concern.

In future work, it should solve the problems found during the evaluation and also adds some new features to make this application stronger, as shown below:

- Test by more users for longer time

As mentioned before, the current evaluation has done in very small scale. To get more convinced results and identify potential problems, it is necessary and essential to do the evaluation with more participants for a longer time. This will also alleviate or solve the problem of no results for query as there are lots of data provided by more users.

- Show all results on the map

It may be better to show all records stored on the database when users first jump to the map page, instead of showing the records in recent one hour. To get desired result, it should calculate the average noise level not just showing all data once. This improvement helps people to know the noise changing trend and overall noise level of one area, which could provide more information and be more convenient.

- Assigning tasks by server

This is a new feature to make the server stronger and also helps to solve the problem of no results of querying. The server will forward the requests from one user to others once at the location if there is on data stored on the database currently. The server may also assign tasks when the noise levels of some areas are not updated for a long time. Thus, this application will be more useful in people's daily life.

Appendix A

Abbreviations

Short Term	**Expanded Term**
ID	Identity
GPS	Global Positioning System
OS	Operating System
GSN	Global Sensor Network
PEPSI	Privacy-Enhanced Participatory Sensing Infrastructure
HP3	Hot-Potato-Privacy-Protection
EC	Equivalence Class
V-MDAV	Variable-size Maximum Distance to Average Vector
EU	European Union
HTTP	Hypertext Transfer Protocol
PHP	Hypertext Preprocessor
XML	Extensible Markup Language
KML	Keyhole Markup Language
JSON	JavaScript Object Notation
IP	Internet Protocol
APK	Application Package File

Appendix B

User Questionnaire

At the end of the experiment, this questionnaire aims to help the researcher to evaluate and improve this project.
Each question is optional. Feel free to omit a response to any question; however the researcher would be grateful if all questions are responded to.

1. How often did you submit reports per day?

2. Which level of privacy protection do you think this project offers? (low, medium, high)

3. Have you ever known any other users' identity during the experiment? If so, how?

4. Do you think your identity or private information which you do not want to share was revealed during the experiment? If so, how?

5. Do you think this project implement privacy-aware urban noise mapping?

6. What would you like to suggest for this project to be improved?

Bibliography

[1] Delphine Christin, Andreas Reinhardt, Salil S. Kanhere and Matthias Hollick,"A survey on privacy in mobile participatory sensing applications", The Journal of Systems and Software 84, 2011, pp.1928-1946.

[2] Ling Hu and Cyrus Shahabi, "Privacy Assurance in Mobile Sensing Networks: Go Beyond Trusted Servers", IEEE, 2010.

[3] Kuan Lun Huang, Salil S. Kanhere and Wen Hu, "Towards Privacy-Sensitive Participatory Sensing", IEEE, 2009.

[4] Salil S. Kanhere, "Participatory Sensing: Crowdsourcing Data from Mobile Smart Phones in Urban Spaces", In Proceedings of the 12th IEEE International Conference on Mobile Data Management, Sweden, 2011, pp.3-6.

[5] Silvia Santini, Benedikt Ostermaier and Andrea Vitaletti, "First Experiences Using Wireless Sensor Networks for Noise Pollution Monitoring", In Proceedings of the 3rd ACM Workshop on Real-World Wireless Sensor Networks (REALWSN'08), Glasgow, United Kingdom, April 2008.

[6] Jerry Kang, "Information Privacy in Cyberspace", Social Science Research Network, Stanford Law Review, Vol. 50, 2004, pp.1193-1998.

[7] Predrag Klasnja, Sunny Consolvo, Tanzeem Choudhury, Richard Beckwith, and Jeffrey Hightower, "Exploring Privacy Concerns about Personal Sensing", Pervasive 2009, pp. 176-183.

[8] Bugra Gedik and Ling Liu, ”Location Privacy in Mobile Systems: A Personalized Anonymization Model”, In Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICSCS05), Columbus, Ohio, USA, pp.620-629.

[9] Arvind Thiagarajan, Lenin Ravindranath, Katrina LaCurts, Samuel Madden, Hari Balakrishnan, Sivan Toledo and Jakob Eriksson, ”VTrack: accurate, energy-aware road traffic delay estimation using mobile phones”, In Proceedings of the 7th ACM conference on Embedded network sensor systems, New York, USA, 2009, pp.85-98.

[10] Charith Pereral, Arkady Zaslavsky, Peter Christen, Ali Salehi and Dimitrios Georgakopoulos, ”Capturing Sensor Data from Mobile Phones using Global Sensor Network Middleware”, In Proceedings of the 23rd Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Sydney, Australis, 2012, pp.24-29.

[11] Eiman Kanjo, ”NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping”, Springer Science and Business Media, LLC 2009,pp.562-574

[12] Silvia Santini, Benedikt Ostermaier and Robert Adelmann, ”On the Use of Sensor Nodes and Mobile Phones for the Assessment of Noise Pollution”, In Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICSCS05)evels in Urban Environments, Columbus, Ohio, USA, 2009.

[13] G. Bennett, E.A. King, J. Curn, V Cahill, F. Bustamante and H. J. Rice, ”Environmental noise mapping using measurements in transit”, In Proceedings of International Conference on Noise and Vibration Engineering including International Conference on Uncertainty in Structural Dynamics, Leuven, 20-22 September 2010, pp.1795-1809.

[14] Hong Lu, Wei Pan, Nicholas D. Lane, Tanzeem Choudhury and Andrew T. Campbell, "SoundSense: Scalable Sound Sensing for People-Centric Applications on Mobile Phones", 2009 ACM, pp.558-566.

[15] Nicolas Maisonneuve, Matthias Stevens, Maria E. Niessen, Peter Hanappe and Luc Steels, "Citizen Noise Pollution Monitoring", In Proceedings of the 10th International Digital Government Research Conference, Puebla Mexico, 2009, pp.96-103.

[16] Fei Xu, Jingsha He, Matthew Wright and Jing Xu, "Privacy protection in Location-Sharing Services", In Proceedings of the 2010 International Conference on Computer Application and System Modeling, Taiyuan, 22-24 Oct 2010, pp.V8-488-V8-491.

[17] Ville Kotovirta, Timo Toivanen, Renne Tergujeff and Markku Huttune, "Participatory Sensing in Environmental Monitoring Experiences", In Proceedings of the Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2012, Italy, pp.155-162.

[18] Andrew T. Campbell, Nicholas D. Lane, Emiliano Miluzzo, Ronald A. Peterson, Hong Lu, Xiao Zheng, Mrico Musolesi, Kristof Fodor, Shane B. Eisenman and Gahng-Seop Ahn, "The Rise of People-Centric sensing", In Proceedings of the 2nd ACM/IEEE Annual International Wireless Internet Conference (WICON), Boston, MA, August 2006, pp.2-5.

[19] Rong Tan, Junzhong Gu, Jing Yang and Peng Chen, "Designs of privacy protection in location-aware mobile social networking applications", In Proceedings of the 5th International Conference on Pervasive Computing and Applications (ICPCA), Maribor Slovenia, 2010,pp.62-68.

[20] Emiliano De Cristofaro, Claudio Soriente, "Participatory Privacy: Enabling Privacy in Participatory Sensing", IEEE Network, January-February 2013, pp.32-36.

[21] E. Miluzzo, N. Lane, K. Fodor, R. Peterson, S. Eisenman, H. Lu, M. Musolesi, X. Zheng, A. Campbell, "Sensing Meets Mobile Social Networks: The Design, Implementation and Evaluation of the CenceMe Application", in Proceedings of ACM SenSys, Raleigh, NC, USA, November 2008.

[22] Katie Shilton, "Four Billion Little Brothers? Privacy, mobile phones, and ubiquitous data collection", Communication of the ACM, November 2009, pp.48-53.

[23] Sasank Reddy, Andrew Parker, Josh Hyman, Jeff Burke, Deborah Estrin and Mark Hansen, "Image Browsing, Processing, and Clustering for Participatory Sensing: Lessons From a DietSense Prototype", In Proceedings of the 4th workshop on Embedded networked sensors, New York, USA, 2007, pp.13-17.

[24] Lama Nachman, Amit Baxi, Sangeeta Bhattacharya, Vivek Darera, Piyush Deshpande, Nagaraju Kodalapura, Vincent Mageshkumar, Satish Rath, Junaith Shahabdeen and Raviraja Acharya, "Jog Falls: A Pervasive Healthcare Platform for Diabetes Management", In Proceedings of the 8th International Conference, Pervasive 2010, Helsinki, Finland, May 17-20, 2010, pp. 94-111.

[25] Y. F. Dong, S. Kanhere, C. T. Chou, N. Bulusu, "Automatic Collection of Fuel Prices from a Network of Mobile Cameras", In Proceedings of the 4th IEEE International Conference on Distributed Computing in Sensor Systems, Santorini Island, Greece, June 11-14, 2008, pp.140-156.

[26] Shane B. Eisenman and Andrew T. Campbell, "SkiScape sensing", In Proceedings of the 4th ACM International Conference on Embedded Network Sensor Systems, New York, USA, 2006, pp.401-402.

[27] Prashanth Mohan, Venkata N. Padmanabhan and Ramachandran Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones", In Proceedings of the 6th ACM conference on Embedded network sensor systems, New York, USA, 2008, pp.323-336.

[28] Kristina Levak, Marko Horvat, Hrvoje Domitrovic, ”Effects of Noise on Humans”, In Proceedings of the 50th International Symposium ELMAR-2008, Zadar, Croatia, September 10-12, 2008, pp.333-336.