Place-Tags, Discovering and Promoting Places Through Mobile Phones and Collaborative Filtering

Tomche Delev[1], Dejan Gjorgjevik[2], and Gjorgji Madzarov[3]
[123]Faculty of Electrical Engineering and Information Technologies – "Ruger Boskovik" Street, PO Box 574, 1000 Skopje, R. Macedonia
E-mail(s): tdelev@feit.ukim.edu.mk, dejan@feit.ukim.edu.mk, madzarovg@feit.ukim.edu.mk

Abstract. *This paper presents the design and implementation of a mobile application along with a web server for geo-tagging favorite and interesting places and sharing them with the community. The design and architecture shows some key aspects and issues concerning this kind of system. The mobile application is implemented in J2ME and tested on GPS enabled Nokia phones and the web server is implemented on cloud infrastructure implementation, the Google App Engine. The system was evaluated with real devices and a proof of concept was made that applications such as Place-Tags has its place in the mobile world.*

Keywords. Participatory sensing, mobile computing, context-aware, location based

1. Introduction

Billions of people are carrying and possessing mobile phones. These mobile phones are creating one of the largest ubiquitous processing environments. Equipped with various kinds of sensors such as integrated GPS, accelerometers and with WiFi network possibilities, these devices are increasingly capable of capturing images, voice, location and other context-aware data. Possessed by people, mobile phones are creating a new kind of intelligent and moving sensor networks as opposed to the already known wireless sensor networks. Networks are created from sensor nodes that are directly exposed to the physical environment, forming various topologies and aggregating large amounts of sensor data to central aggregators. This model is commonly successfully used in military, wildlife surveillance, habitat monitoring [1] or even volcano monitoring [2]. Given this successful model of sensing the environment by wireless sensor networks we are proposing using this model with a slightly different approach. Using people and their personal mobile phones as the intelligent sensors and central web server as data aggregator, we come with a new model of application. Using the enormous amount of ubiquitous processing power in the hands of intelligent people we create applications taking advantage of people's participation in collecting and processing of environmental data. This kind of participatory sensing [3] is bringing new level, in both the usage of mobile phones and usage of collective power for purpose of creating useful applications.

Place-Tags is a simple application taking advantage of this model of pervasive computing and participatory sensing. Given the great mass of people with mobile phones, this application tends to make them geo-tag the physical environment of any kind, and also rate by its popularity. By using this tagged data, one can link with visual map (Google Maps, Microsoft Live Map) and create more context-aware data for locations such as restaurants, shopping malls, hospitals, schools and any kind of public or private buildings or locations. On the other side having the people who search for this kind of nearby objects, they can use the other part of the system, the data retrieval, and use this data to easily find the searched location and contribute to the system by either retag or simply rate. Taking advantage of ratings from large groups, we can also recommend places using the advanced techniques of collaborative filtering. The rest of the paper is organized as follows. Section 2 and 3 are about the overall architecture, some related work and similar applications. The actual design of the system is described in Section 4, followed by presentation of its actual implementation in Section 5. Brief evaluation and proposing some ongoing work is written in Section 6. The idea of geo-tagging objects or locations and visualize spatially on Internet geo-maps is not new, but incorporating such applications in any variety of businesses and taking advantage of power of intelligent sensors (people possessing smart phones) gives it another dimension.

2. Overall architecture and application

Fig. 1 illustrates the overview architecture of Place-Tags mobile application with the complete web servers' infrastructure. Mobile clients are the main users and contributors of the system by geo-tagging their favorite places and also providing feedback to places they discovered with the application. The mobile application users take advantage of application by seamlessly creating placemark information for the places they like to mark, and later maybe share. This information consists of the place represented with textual description, some tags and rating (number from 1-5). Every place is associated to some placemark entities constructed with the actual geo-data, the latitude and longitude of the location. Then these places are synchronized (uploaded) with the web server, the central aggregator of all the places data. The web server serves this information to all web users, presenting it on web map such as Google Maps or Microsoft Maps. Places can also be searched or recommended.

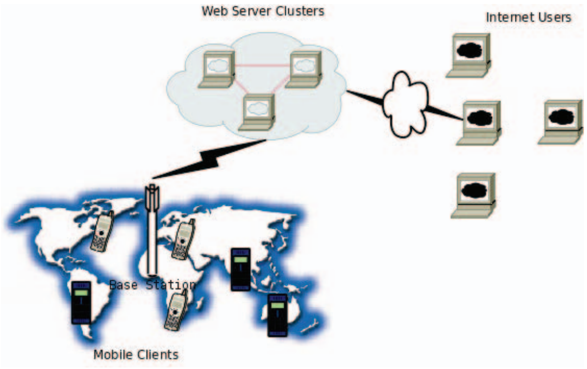

Figure 1. Very high level view of the architecture of the system composed from the mobile client application and the web server infrastructure. Two types of users exists, mobile clients who use the application to geo-tag or just find some interesting places and normal Internet users who can browse the content uploaded by the mobile users.

2.1. Mobile client application

In this section we describe the mobile client application and all its potential usages.

a) Navigation and tourism: Sometimes we all have been in situations visiting some special places that we like and we would like to visit sometime in the future again, but we have also always facing the problem of remembering the exact location of these places. Some users of the application, for example, say Peter is getting his mobile phone and tagging the wonderful place of church St. Sofia in Ohrid with some tags and description meaningful to him and associating some placemarks near by the exact location. Then he synchronize his place with the web server and now whenever he or some other user of the application is coming somewhere near Ohrid and query the web server by some relevant tags, or query for recommendation he can get the exact location of this wonderful place and maybe take his time to visit. All this can be achieved taking advantages of the GPS enabled mobile phone and using only GPRS or WiFi connection.

b) Promotion: World is full of wonderful and useful places, but most of them lack of good and effective promotion. Place-Tags is a new way of seamlessly promoting places by tagging them and introducing and sharing with the community.

Every place on the earth can be interesting and attractive with the right promotion and this application makes it very easy for simple users to promote a lot of different places using their GPS enabled mobile phones.

c) Collaborative exploration: Exploring new places is a new adventure and by using this kind of application it can become even more interesting. Collaboratively exploring new places can be achieved simultaneously by group of people attending at same location and tagging and querying for new and interesting places. People easy can find the places of their own interest by collaborative filtering, advanced technique of artificial intelligence which can recommend some potentially interesting places by comparing ratings from all users contributing with tagging and rating places.

3. Related work

Recently, a whole new wave of research is focusing on location-based application and some kind of participatory sensing in a way of developing people centric application. Most of the technologies and application involving context and location awareness are summarized in [4]. In this section is presented some of the related work in this direction.

AURA [5] is a system that is used to link on line content to physical objects implemented with commercially available pocket computers using integrated bar code scanners, wireless networks, and web services. This is implemented by adding tag reading devices to wireless enabled devices, which shortens the gap between the physical objects and places, and the digital information related to them.

GeoLife2.0 [6] is bringing together the social networking aspect and location awareness. This GPS-data-driven social networking service enables people to share life experiences and connect each other with their location histories. By mining people's location history, GeoLife can measure the similarity between users and perform personalized friend recommendation for any individual. In terms of mining the geo-location data [7] is trying to learn the transportation methods from raw GPS data. In [8] are explored various ways of mining the users similarity based on the location histories, and [9] are going even one step further by trying to predict the users next location.

4. Architecture and design

Fig. 2 presents the overall software architecture of the Place-Tags application. It's a client/server architecture with clients of the mobile application and clients of the web service. Mobile application consists of three important modules: location, persistence and network. The location module asynchronously acquires the user's location and when succeed allows him to create places with tags and then save in the mobile phone persistent storage.

The places then can be synchronized with the web server using the mobile network which creates HTTP connections over GPRS or WiFi, depending on the current mobile phone connected networks. To find place, user enters some tags or he may leave blank the text field. When the system tries to find places for the user query first it filter places by tags, and then applies second filter using the user location to give results in his interest. If the user doesn't enter search keywords the system responds with best matches, queried using only by the user's location and recommendation algorithms.

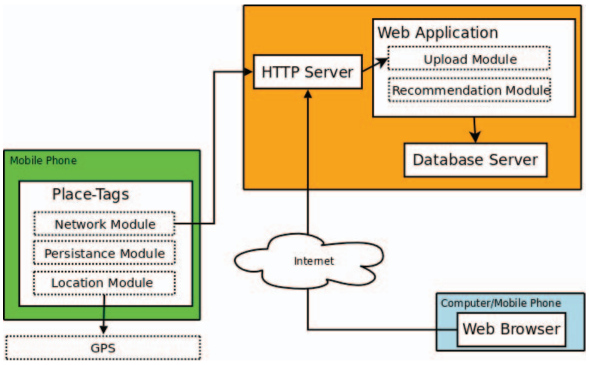

Figure 2. Block diagram of the software architecture of Place-Tags mobile application

4.1. Mobile client design consideration

Smart phones, a special category of mobile phones are relatively powerful mobile computing platforms, offering reasonable data-processing and storage capabilities, incorporate various communication technologies, and often include embedded devices such as cameras or sensors. The most useful feature of smart phones as enablers of pervasive computing is the possibility of installing new applications, which in many cases anyone, not just the manufacturer or operator, can write. But they come with some constraints such as limited input (no keyboard), small display sizes, restricted storage memory and heap size, and slow CPUs. With this kind of limitations, mobile software design is inherently challenging task and every aspect of design should be considered carefully.

a) Limited processing power**:** Commonly equipped with ARM architecture CPUs due to its power efficiency mobile phones are very limited and constrained in processing power. Clock frequency in modern models is usually around 200-250 MHz, but recent high-end models exceed 300-400 or even 600 MHz. Total RAM is normally between 64 and 128 Mbytes, but applications can use only a fraction of this, typically 10-20 Mbytes. External storage is provided by flash memory cards with capacity up to 8 Gbytes, but because smart phones operating systems does not support virtual memory, this cannot automatically be used as an extension to RAM.

b) Energy consumption: Every mobile application executes on a mobile phone, and mobile phones primarily run on battery power, and each device has a "battery budget". Most of the smart phones use lithium-ion batteries due to their small form factor. Charge ranges between

780 and 1,800 milliamp hours (mAh), which in most cases is sufficient for several days of standby operation or a few hours of phone calls. Fig. 3 shows where it goes all that battery energy from these power full smart phones.

5. System implementation

Place-Tags mobile client was implemented in J2ME [10] in approximately 4,500 lines of code using NetBeans IDE, the web service was implemented in Python using the Django framework [11] and Google App Engine [12] as a back-end. Here are some of the details of the implementation.

5.1. Mobile client

The mobile client was developed using the Sun Java Wireless Toolkit integrated in to NetBeans IDE in Ubuntu distribution of Linux environment. The application requires access to location services through Location API for J2ME (JSR 179) and also to networking services required for communication with the web service. The J2ME platform was chosen mainly because the application doesn't require some special access to low level information such as WiFi MAC IDs, signal strength information or cell tower IDs which can be only achieved writing native software like C++ for Symbian OS. Also J2ME allows lot easier deployment with no special signing procedure and very important it's supported on most mobile phones that have implementation of KVM[1].

The application is designed with very simple user interface with the purpose of providing great usability to users. Starting the application for first time alerts the user to enter his email which will be his only unique piece of information for identification and communication with the web service. The main menu list is the normal entry point of the application and provides access to

- My Places - saving, editing and synchronizing places.
- Find Place - searching and finding or recommending places.
- My Email - changing settings such as your email identification.

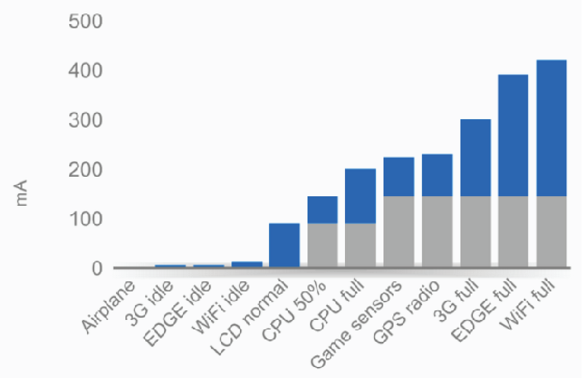

Figure 3. Usage of battery by different modes. Source: Values measured using power monitor at 5kHz sampling rate, and taking average power with lowest standard deviation.

The My Places part of the application is concerned with tagging your favorite places and sharing them with the community or just keeping for personal tracking of all the great places you visited. The process of tagging a place requires to enter name (required), URL, rating some tags for the place and also associate some placemarks (latitude, longitude and timestamp) for describing the geo location of the place. When users are adding placemarks to some place, the application requires their location by using location services. This process takes some time and is done asynchronously in separate thread, so the user interface is not blocked. After the first location fix, the location is updated on short intervals (3 seconds) and user can add placemarks to places almost instantly with the latest location. After associating some placemarks with his favorite place, the user is prompted to save these placemarks to persistent storage and also to synchronize with the web service. These few steps are all it's needed for simple storing, tagging and sharing places. Achieving simple usage scenario is critical in mobile application since users can be easy distracted or interrupted by phone call or any other activity that is common in mobile world. The core implementation is separated in three distinct modules. The GUI module is implementing the user interface and it consists of standard J2ME UI controls such as TextFields, Labels and Lists. Achieving responsive UI while doing tasks such as storing data to persistence storage or sending to web server is done by using separate threads in the task management module. The task classes in this module are implemented using the strategy pattern [13]. The last important

[1] A small Java virtual machine for J2ME. Sun's K Virtual Machine gives embedded developers a virtual kick in the pants. KVM makes it possible to, for the first time, run Java programs on any 16-bit processor with 128KB of available memory.

module is the one concerned with the persistence of the data. Persistence in J2ME is usually implemented using RMS (Record Management System) and that is the case in this application, although file system storage is also supported with additional user privileges.

5.2. Web server

The web application is implemented around the new concept of cloud computing, precisely using the cloud infrastructure [14] provided by Google App Engine [12]. Google App Engine lets you run your application on Google's infrastructure using Python or Java programming language. From the mobile client perspective the web server has the responsibilities of serving the upload and search requests. All the communication is done using the HTTP over TCP connections. All messages are in JSON format which is very suitable taking its very compactness and small size. For all other users browsing the web site it provides view of latest, most popular or best rated places. Using collaborative filtering algorithms web server implements module for making recommendation of places for every mobile user. A collaborative filtering algorithm usually works by searching a large group of people and finding a smaller set with tastes similar to yours. It looks at other things they like and combines them to create a ranked list of suggestions [15]. In Place-Tags implementation of the recommendation module first step is calculation of the similarity between the user and all other users who contributed with rating. Similarity is computed using the Pearson correlation coefficient which is a measure of how well two sets of data fit in a straight line. The formula for this is

$$\boldsymbol{r} = \frac{\sum \boldsymbol{XY}^{\frac{\sum \boldsymbol{X} \sum \boldsymbol{Y}}{\boldsymbol{N}}}}{\sqrt{\left(\sum \boldsymbol{X}^2 - \frac{\sum \boldsymbol{X}^2}{\boldsymbol{N}}\right)\left(\sum \boldsymbol{Y}^2 - \frac{\sum \boldsymbol{Y}^2}{\boldsymbol{N}}\right)}} \quad (1)$$

and is more complicated than Euclidean distance score but tends to give better results when data isn't well normalized. Second and final step is to score places by weighted score that ranks the places. This is calculated by summing the product of similarity and rating for every place that is ranked and then this value is divided by the total similarity sum. The result list of place, score pair is sorted by the weighted score in descending order and returned as result for recommendations.

6. Evaluation and future work

Place-Tags was evaluated using off-the-shelf Nokia N95 and Nokia E71 phones. Primary goal of the evaluation was to create some content and to evaluate interruption of the normal usage of the phone. The application was starting fast and about 30 seconds was required to initiate the assisted GPS, so the application is ready to record placemarks for any outdoor places. Also was evaluated the battery life of the phone while using the application and the GPS locations accuracy. Using the application without using the GPS location services, for instance when saving or editing places or synchronizing with the web server, has no substantial effects on normal battery consumption of these 3G mobile phones and the battery life can be measured in days. While using the GPS service constantly the battery life spans between 6-8 hours, but since this application is not intended to be used continuously for so long time, the final conclusion can be that this kind of application with short usage scenario doesn't have any special requirements on battery consumption. So for, it can be normally used in parallel with every other phone service such as phone calls, or even listening music. Evaluation of the accuracy of location service provided by the assisted GPS technology of tested phones was done by comparison of the placemarks uploaded by application and their location relative to some referent place on Google Maps. A-GPS has documented accuracy of 5-10 m, and this method of comparison showed results that generally fit in this assumption, although can be seen some landmarks with extremely accurate position within 1-2 meters.

There are many ways this system and its application can be upgraded and there are features that can be developed to make the system in whole more ubiquitous. When the system will have critical amount of data, modules for auto-tagging and clustering places can be developed. Also API for exposing the basic functionalists can be developed and allow users developers to exploit it, developing different types of client applications that can upload or query content from the system. Other types of localization services that acquire location faster or more accurate can be considered. Many of the features lay down in the end-user experience with the system, so providing a good way of expressing feedback is a key in achieving system that is representative and also useful.

7. Conclusion

In this paper was presented the design and implementation of a participatory sensing application called Place-Tags, which allows users equipped with GPS enabled mobile phones to generate, share and contribute geo-tagged information about their favorite places. Every place can be browsed or queried through the mobile application or simple web browser and presented with its location placemarks on Google Maps. Also the places can be rated with numerical ratings, which later are used to provide recommendation of places to users. Web site also keeps track of popularity of places by counting the views and ranks them by this attribute, along with rating and date.

Some of the technical challenges involved in this work include substantial work on mobile application to design seamless and easy to use application for small and limited devices such as mobile phones. Using and incorporating new technologies such as cloud infrastructure service by Google App Engine which is storing all the data in non-relational database, the Big Table[16] distributed data file system, is requiring different views of the design process.

Evaluation of this kind of systems and applications requires contribution from large amount of users equipped with expensive devices, so for an independent student project, evaluating with real devices, and providing some real data can be concluded as a success.

Usage and manufacturing of mobile phones is going in to new era, so this kind of mobile applications and systems are required for the billion mobile users to find their killer application and take advantage of their smart phones. Place-Tags is one such attempt that tends to encourage users to participate in exploring and sensing the environment, and share that data with unlimited set of users.

8. References

[1] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson. Wireless sensor networks for habitat monitoring.

[2] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh. Deploying a wireless sensor network on an active volcano. IEEE Internet Computing, 10(2):18–25, 2006.

[3] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M.B. Srivastava. Participatory sensing. In World Sensor Web Workshop, pages 1–5, 2006.

[4] M. Hazas, J. Scott, and J. Krumm. Location-aware computing comes of age. Computer, 37(2):95–97, 2004.

[5] M. Smith, D. Davenport, and H. Hwa. AURA: A mobile platform for object and location annotation. In Fifth International Conference on Ubiquitous Computing-UbiComp 2003, 2003.

[6] Y. Zheng, X. Xie, and W.Y. Ma. GeoLife2. 0: A Location-Based Social Networking Service.

[7] Y. Zheng, L. Liu, L. Wang, and X. Xie. Learning transportation mode from raw GPS data for geographic applications on the Web. 2008.

[8] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.Y. Ma. Mining user similarity based on location history. In Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographicinformation systems. ACM New York, NY, USA, 2008.

[9] J. Krumm and E. Horvitz. Predestination: Where Do You Want to Go Today? Computer, 40(4):105–107, 2007.

[10] M.J. Yuan. Enterprise J2ME: Developing Mobile Java Applications. Prentice Hall PTR Upper Saddle River, NJ, USA, 2003.

[11] Django Software Foundation. The Django Framework,2005-2009. http://www.djangoproject.com.

[12] Google App Hosting Service. Google App Engine,2008-2009. http://code.google.com/appengine/.

[13] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design patterns: Elements of reusable object-oriented design, 1995.

[14] B. Hayes. Cloud computing. 2008.

[15] T. Segaran. Programming collective intelligence: building smart web 2.0 applications. O'Reilly Media, Inc., 2007.

[16] F. Chang, J. Dean, S. Ghemawat, W.C. Hsieh, D.A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R.E. Gruber. Bigtable: A distributed storage system for structured data. In Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI'06), 2006.