KING'S College LONDON

University of London

6CCS3PRJ Final Year
LOST

Final Project Report

Author: Joshua Simpson

Supervisor: Andrew Coles

Student ID: 1225043

April 15, 2015

Abstract

Navigational tools have been developing rapidly over the past couple of decades - at the moment, capable of providing us with a location within a few meters of where we are. Unfortunately, they still have not progressed to a point where a user can use location-based services from within a building to an accurate degree - an advance that would open new avenues of development in contextually aware services

This report showcases an extension of mostly theoretical work that uses wireless access point data to provide contextually aware location based services at a building level with some precision - the end result in this being a mobile application package alongside a backend to collect contextual data. The main aim of this entire package being to demonstrate the value of contextually aware systems to different types of users.

Originality Avowal

I verify that I am the sole author of this report, except where explicitly stated to the contrary. I grant the right to King's College London to make paper and electronic copies of the submitted work for purposes of marking, plagiarism detection and archival, and to upload a copy of the work to Turnitin or another trusted plagiarism detection service. I confirm this report does not exceed 25,000 words.

Joshua Simpson
April 15, 2015

Acknowledgements

Dr Andrew Coles - for his patience, expertise, and lulzy memes

The Hackathon Community - for their amazing support through difficult times

Contents

Chapter 1

Introduction

Location based services have been developing over the past few decades, and are prevalent in several aspects of every day life - location based services range from the route-planning software that is used to figure out a morning commute, to live time-tabling services that use bus or train positions to give accurate estimates. It's safe to say that in some way or another, location services are prevalent in modern day life.

However, GPS[1] services only work to a certain level; specifically, they only work to within a few feet at street level[3]. Whilst this is still a feat, it leaves a large market gap for contextually aware services inside large buildings.

Contextually aware services are another rapidly developing technology, providing users with information or activities specific to their current location or situation, determining the suitability for such events by using sensor data[2].

There are previous studies on using local area wireless access points to provide this sensor data at a level more precise than GPS can currently allow, but they are usually run in controlled environments, deriving a single mathematical equation in order to determine distance from the access point with high precision[2]. Unfortunately, this is impractical in a lot of real world applications - such as university campuses.

[1] Global Positioning System - a system using satellites to provide precise location information

[2] Data such as location, temperature or time

1.1 Project Aims

The main goal of this project is to provide a proof-of-concept mobile application that can provide location and route-planning services whilst also delivering contextually aware services in a specific building - all without using GPS - demonstrating the possibilities that this area of application development holds for developers. In extension, the project will aim to create software suite allowing for the efficent setup of a 'contextually aware building', as well as provide practical functionality to a variety of user groups (by using the application to crowdsource data), to demonstrate the potential in this field.

To demonstrate a successful project, a large building with many potential locations is useful, as it shows the accuracy of of the location-finding and route-planning algorithms. To this end, the building that I am using to test this application is King's College London's Strand Campus[3]. With roughly 17,000m2 of space and 9 floors to utilise, the application would not only be thoroughly tested but also most likely welcomed by King's 25,000 students[5].

1.2 Report Structure

This report will begin with a background on the project - taking a look at location based services and contextually aware services and the applications of both that are already available, along with an analysis of projects or papers that have attempted to solve the problem of location services at building level. This data will then be collated to analyse key problem areas during implementation.

Following this there will be an outline of the requirements and specification for the project - showing both the project itself and the chosen extensions - complete with justifications on decisions made in the specification.

[3] It has not escaped my attention that this will also help all the new first years find Waterfront

Chapter 2

Background

2.1 Location Services

Defined as *"services that integrate a mobile device's location or position with other information so as to provide added value to a user"*[8], the origins of location information services date back to 1973, when the US Department of Defence developed GPS to overcome the limitations imposed by navigation systems in use at the time[6]. This original system has since been developed into an integral part of many mobile applications, including mapping and route planning applications, as well as forming the 'sensor' in a lot of contextually aware applications. There is a well established method that is used in order to ascertain location, called trilateration[1]. Triangulation[2] is another method used to determine location, but is mostly inapplicable in the context of mobile devices[3].

2.1.1 Application: Google Maps

Google Maps is easily the most widely used mapping tool on the planet, with over 54% of smartphone owners using the app on a regular basis in 2013 (for comparison, this was 10% more than Facebook at the time)[7]. Google Maps are a chief provider of location information services, with advanced route planning and street-view functionality. They serve this data through their own API[4]. The most interesting aspect of their service (in relation to this paper, at least) is their recent addition of the use of Wireless Access Points for their location services. Whilst they don't rely solely on it for location, they do use the data to increase accuracy

[1] The process of finding a location by measuring distances, using geometry

[2] The process of finding a location by estimating the direction in which signals are coming from

[3] It is also commonly mistaken for trilateration

[4] https://developers.google.com/maps/

and speed through their Maps application[4]. Interestingly, Google collect this WiFi data by crowdsourcing through an ’opt out’ scheme in the Google Maps application[9] - this may not be possible in the case of this project (as Google have used WiFi in a more assistive manner), but it provides a good basis from which to approach the problem.

2.1.2 Paper: Indoor Location Using Trilateration Characteristics

[3]

This paper is especially relevant to the problem, considering the use of Wireless Local Area Networks in order to achieve trilateration. This is achieved by measuring the signal strength for at least 3 access points, and using the signal strength to determine the distance from the access point. Once you have three accurate distances you can essentially create a ’Venn Diagram’ of where the user is located.

The paper goes on to state that there are inherent inaccuracies with this positioning technique, due to the variable nature of radio signals[3].

To circumvent these inaccuracies, an elegant averaging method was developed, which started returning an accurate ’average signal strength’ after 40 readings.

Issues with the trilateration approach

- As stated above, the trilateration approach has difficulties providing accuracies:

”Radio signals are extremely variable, particularly indoors, due to being reflected by obstacles or refracted round corners, known as multipath reflection. Environmental changes can also affect the signals, such as the number of people around. This means that the positioning technique is inherently inaccurate, with positions from raw Wi-Fi signals being in excess of 10m out. ”[3]

- Whilst there is a solution designed to solve this problem in the case of the test environment, an application in a building such as Kings would require a number of different formulae in order to provide accurate signal strength averaging for different areas of King’s[5]

- Finally, in cases where the averaging solution is easily applicable, there *should* be concerns regarding the impact on a mobile device’s battery life when performing repeated scans and operations on the results of those scans.

[5] Because of the variables such as distance between routers, building material, and the number of users in any area at a given time

2.1.3 Paper: More Stuff on Location - Fuzzy Location?

2.2 Context-Aware Services

Contextually aware services are services that use data that pertains to the user or application's current situation to provide actions or functionality specific to that scenario. It is defined by Abowd as

"any information that can be used to characterize the situation of entities (i.e., whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves." [1]

Contextually aware services use 'sensors' to fetch this information - examples include location data, temperature, even the camera on your phone could be used as a sensor for contextual services[6]

2.2.1 Estimote

A rapidly accelerating tech start up, Estimote provides 'beacons' using low-power Bluetooth in order to send an ID[7] to an Estimote enabled application - these can then be programmatically associated with locations, events, etc. and calculates its distance from the phone using RSSI (the received signal strength), meaning that it can be used very effectively to provide location-based services. This is currently one of the most prominent devices currently using its base technology (iBeacon).

Currently the major drawback of the Estimote is the price, which limits Estimote beacons / stickers to small buildings âĂŞ covering an area such as KingâĂŹs would be incredibly costly[8], and any structural changes regarding the beacons would require significant effort to represent in an application developed to use Estimote. Another issue is that using RSSI to develop an idea of location can become very complex in a large building with lots of people in it (which would cause signal attenuation at different levels throughout the day).

[6] Consider how recent smartphones have software that stops the screen coming out of standby if it thinks it's in your pocket

[7] `http://estimote.com/api/getting-started/intro-to-beacons.html`

[8] `http://estimote.com/#jump-to-products`

Chapter 3

Report Body

The central part of the report usually consists of three or four chapters detailing the technical work undertaken during the project. **The structure of these chapters is highly project dependent**. They can reflect the chronological development of the project, e.g. design, implementation, experimentation, optimisation, evaluation, etc (although this is not always the best approach). However you choose to structure this part of the report, you should make it clear how you arrived at your chosen approach in preference to other alternatives. In terms of the software that you produce, you should describe and justify the design of your programs at some high level, e.g. using OMT, Z, VDL, etc., and you should document any interesting problems with, or features of, your implementation. Integration and testing are also important to discuss in some cases. You may include fragments of your source code in the main body of the report to illustrate points; the full source code is included in an appendix to your written report.

3.1 Section Heading

3.1.1 Subsection Heading

Chapter 4

Design & Specification

4.1 Section Heading

Chapter 5

Implementation

5.1 Section Heading

Chapter 6

Professional and Ethical Issues

Either in a seperate section or throughout the report demonstrate that you are aware of the **Code of Conduct & Code of Good Practice** issued by the British Computer Society and have applied their principles, where appropriate, as you carried out your project.

6.1 Section Heading

Chapter 7

Results/Evaluation

7.1 Software Testing

7.2 Section Heading

Chapter 8

Conclusion and Future Work

The project's conclusions should list the key things that have been learnt as a consequence of engaging in your project work. For example, "The use of overloading in C++ provides a very elegant mechanism for transparent parallelisation of sequential programs", or "The overheads of linear-time n-body algorithms makes them computationally less efficient than $O(n \log n)$ algorithms for systems with less than 100000 particles". Avoid tedious personal reflections like "I learned a lot about C++ programming...", or "Simulating colliding galaxies can be real fun...". It is common to finish the report by listing ways in which the project can be taken further. This might, for example, be a plan for turning a piece of software or hardware into a marketable product, or a set of ideas for possibly turning your project into an MPhil or PhD.

References

[1] Gregory D Abowd, Anind K Dey, Peter J Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing*, pages 304–307. Springer Berlin Heidelberg, 1999.

[2] Yongguang Chen and Hisashi Kobayashi. Signal strength based indoor geolocation. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 1, pages 436–439, 2002.

[3] B Cook, G Buckberry, I Scowcroft, J Mitchell, and T Allen. Indoor location using trilateration characteristics. In *Proc. London Communications Symposium*, pages 147–150, 2005.

[4] Google. Configure access points with google location service. `https://support.google.com/nexus/answer/1725632?hl=en`.

[5] King's College London. King's college london headcount summary 2012-2013. `http://www.kcl.ac.uk/aboutkings/facts/ida/stats/HeadcountSummary2012.pdf`, 2012.

[6] National Research Council (U.S.). Committee on the Future of the Global Positioning System, National Academy of Public Administration, Commission on Engineering, and Technical Systems. *The Global Positioning System: A Shared National Asset.* Romantic Reassessment; 118. National Academies Press, 1995.

[7] Felix Richter. 15 years of google. `http://www.statista.com/chart/1502/15-years-of-google/`, 2013.

[8] J. Schiller and A. Voisard. *Location-Based Services.* The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2004.

[9] Steven J. Vaughan-Nichols. How google–and everyone else–gets wi-fi location data. http://www.zdnet.com/article/how-google-and-everyone-else-gets-wi-fi-location-data/, 2011.

Appendix A

Extra Information

A.1 Tables, proofs, graphs, test cases, ...

The appendices contain information that is peripheral to the main body of the report. Information typically included in the Appendix are things like tables, proofs, graphs, test cases or any other material that would break up the theme of the text if it appeared in the body of the report. It is necessary to include your source code listings in an appendix that is separate from the body of your written report (see the information on Program Listings below).

Appendix B

User Guide

B.1 Instructions

You must provide an adequate user guide for your software. The guide should provide easily understood instructions on how to use your software. A particularly useful approach is to treat the user guide as a walk-through of a typical session, or set of sessions, which collectively display all of the features of your package. Technical details of how the package works are rarely required. Keep the guide concise and simple. The extensive use of diagrams, illustrating the package in action, can often be particularly helpful. The user guide is sometimes included as a chapter in the main body of the report, but is often better included in an appendix to the main report.

Appendix C

Source Code

C.1 Instructions

Complete source code listings must be submitted as an appendix to the report. The project source codes are usually spread out over several files/units. You should try to help the reader to navigate through your source code by providing a “table of contents” (titles of these files/units and one line descriptions). The first page of the program listings folder must contain the following statement certifying the work as your own: “I verify that I am the sole author of the programs contained in this folder, except where explicitly stated to the contrary”. Your (typed) signature and the date should follow this statement.

All work on programs must stop once the code is submitted. You are required to keep safely several copies of this version of the program - one copy must be kept on the departmental disk space - and you must use one of these copies in the project examination. Your examiners may ask to see the last-modified dates of your program files, and may ask you to demonstrate that the program files you use in the project examination are identical to the program files you had stored on the departmental disk space before you submitted the project. Any attempt to demonstrate code that is not included in your submitted source listings is an attempt to cheat; any such attempt will be reported to the KCL Misconduct Committee.

You may find it easier to firstly generate a PDF of your source code using a text editor and then merge it to the end of your report. There are many free tools available that allow you to merge PDF files.