

**CO3015 Computer Science Interim Report**

**Designing an Android Based Mobile Application for Sharing and Finding Local  
Events using Location Based Services**

**Danial Qamar**

**Thursday 5<sup>th</sup> December, 2017**

**Department of Informatics**

**University of Leicester**

## **Table of Contents**

Declaration.....	1
<b>1. Aims and Objectives.....</b>	<b>2</b>
1.1. Aims	
1.2. Objectives	
1.3. Basic Description of EventShare	
<b>2. Survey of Literature Review.....</b>	<b>3</b>
2.1. Introduction	
2.2. Introduction to Context	
2.3. Context Aware Applications and Examples	
2.4. Location Based Services	
2.5. Summary	
<b>3. Description of Prototype.....</b>	<b>5</b>
3.1. Features Completed So Far	
3.1.1.Developed	
3.1.2.Tested	
3.2. Upcoming Features	
<b>4. Software Architecture, Algorithms and Data Structures.....</b>	<b>8</b>
4.1. Software Architecture	
4.2. Algorithms	
4.2.1. Implemented	
4.2.2.Needs Implementing	
<b>5. Planning and Timescales.....</b>	<b>11</b>
5.1. Semester 1	
5.2. Semester 2	
<b>6. Bibliography.....</b>	<b>13</b>

## DECLARATION

---

All sentences or passages quoted in this report, or computer code of any form whatsoever used and/or submitted at any stages, which are taken from other people's work have been specifically acknowledged by clear citation of the source, specifying author, work, date and page(s).

Any part of my own written work, or software coding, which is substantially based upon other people's work, is duly accompanied by clear citation of the source, specifying author, work, date and page(s).

I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this module and the degree examination as a whole.

Name: DANIAL QAMAR

Signed: D 

Date: 05/12/2017

## **1. Aims and Objectives**

### **1.1 Aims**

The main aim of my project is to develop an application for finding and sharing local events using location based services. The application will be centred around context awareness features, specifically location based services. The name of my application is EventShare. EventShare will be android based and will enable users to find events near to them in their local community. EventShare will not be restricted by the proximity of the local community, as users will be able to find events anywhere in the world. Users will be able to view events which have been posted by other users. By default, the app would show all upcoming events, but this default setting can be configured manually. The target group is people between the ages of 13 to 25. Events could range from football tournaments for 12 to 14-year olds to educational events at universities.

### **1.2 Objectives**

The main objectives are:

- Gathering an initial set of requirements using feedback from a combination of interviews and online surveys from both event attendees and event organisers. Combined with a product review of existing events based applications from the Play Store.
- Design and develop EventShare using location based services considering the initial requirements.
- Engage in various forms of testing including Unit Testing, UI Testing and User Testing.
  - Build upon the application using the feedback from the User Testing.

### **1.3 Basic Description of EventShare**

Users will be able to find/post events in other cities, giving the application a wider geographical scope. Users will be able to sign up to events. Users who wish to post events, will fill out a form specialised for event details. If a user has posted an event, they will receive information such as how many people have signed up to come to the event.

## **2. Survey of Literature Review**

### **2.1 Introduction**

This literature review aims to help me understand how I can best incorporate context awareness into my application, specifically location, by first understanding what context is and how it has already been implemented in existing technologies. The focus will be to understand how, given the location of a user, my application will be able to use this information to provide events to the user tailored specifically to their location. The literature review will consist of four sub sections, Context, Context Aware Applications, Location Based Services and Summary. The first sub section will consist of an understanding of what context means with the aid of multiple definitions. The second sub section will consist of examples of applications where various forms of context awareness have been utilised to provide services to the user. The third sub section will provide examples with a focus on location based services specifically and the final sub section will summarise the review.

### **2.2 Introduction to Context**

Dey (2001) defines context as “any information that can be used to characterise the situation of an entity (Dey, 2001)”. He then defined an entity as a “person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves” (Dey, 2001). Shilit, Adams and Want (1994) expand the number of aspects which are included inside the definition of context, as they define context to be more than just the user’s location. “Context includes lighting, noise level, network connectivity, communication costs, communication bandwidth, and even the social situation” (Schilit et al., 1994). These two definitions are related to each other as the first is a subset of the second, as the second definition also includes aspects such as the social situation. This is significant as it implies that context is more than just what a device can gather in terms of raw data which concerns the interaction between a user and an application, but it is also the relationship between users.

### **2.3 Context Aware Applications and Examples**

According to Marmasse (1999) context awareness describes “the ability of the computer to sense and act upon information about its environment, such as location, time, temperature or user identity” (Marmasse, 1999). “A context aware application is one which adapts its behaviour to a changing environment” (Harter et al., 2002). Both definitions revolve around the ability of an application to change in accordance to its environment. This is significant as I can deduce that the state of context aware applications is constantly changing. Examples which utilise the context of a user include:

- The Pedometer application on the Play Store counts the number of steps a user makes, when started (I. T. Inc, 2017). The movement of the device is detected by the application and if the movement indicates a step has been made, then

the counter increments. The user can then review their steps in a daily, weekly and monthly format. The application uses the movement of each step to also calculate calories burned, distance travelled and the average speed within a session.

- Accurate Heart Rate Monitor is an application on the Play Store which reveals to the user, their heart rate ("Accurate Heart Rate Monitor – Android Apps on Google Play," n.d.). The user must hold their finger over the camera of the phone for a few seconds, while the application calculates the heart rate. Once calculated the user can remove their finger from the camera and their heart rate is revealed. While the heart rate is being calculated, a visual representation of the heart rate is shown to the user using a graph. The graph changes with the increase and decrease of the users' heart rate as it moves across the screen.
- Air Quality Index BreezoMeter is an android application which informs a user about the air quality of their location ("Air Quality Index BreezoMeter – Android Apps on Google Play," n.d.). In addition, a user can view the air quality of other areas. The application then presents the air quality data to the user as well as providing helpful tips regarding how to react to the information. Tips include: "There's no need to get worried, but keep tracking air quality in the next few hours" and "The air quality outside isn't the best... try to find a cleaner area".

## 2.4 Location Based Services

Junglas and Watson (2008) define location based services as "any service that takes into account the geographic location of an entity" (Junglas and Watson, 2008). This definition is brief and can be expanded to location based services being the "capability to find the geographical location of the mobile device and then provide services based on this location" (Liutkauskas et al., 2015). This definition differs from the first as it also takes it on account the ability for the service to find the location of the device whereas the first definition assumes the location has been provided. As for implementing location based services within an android application, android's network location provider can be used as it determines user location using cell tower and Wi-Fi signals, providing location information in a way that works indoor and outdoor (Storm, n.d.). The alternative to the method mentioned is to use the "GPS (Global Positioning System) which uses a constellation of 24 satellites orbiting the earth" (Storm, n.d.). Some examples where location based services have been used within existing applications, includes:

- Arriva Bus application on the Play Store (mbH, 2017). The application uses the location of the user as well as all buses and plots the real-time data coordinates on a google map view. This allows the user to view their own position on a map as well as the current location of all buses at a moment in time. The map and the positions update every time the user moves the map. The Arriva Bus

application uses location based services as a form of context awareness to fulfil the user needs.

- Waze has over 100 million downloads on the Play Store and the primary function of the application is, given the user allows access to the location and their location is enabled ("Free Community-based GPS, Maps & Traffic Navigation App | Waze," n.d.), Waze provides the user with a route from their location to a specified destination. As the user is travelling, the application will inform the user that there are road closures or any other incidents. These incidents are posted onto the application by other users. Therefore, each user can share and view incidents in real time posted by other users. A map view is central to the user experience of the application as the user is provided with turn-by-turn directions. In addition, the map view is used to visualise the real-time travel information.
- Uber allows a user to view all available Uber taxis in real time (U. T. Inc, 2017). The user enables their location and drops a pin to show their destination. Using the information and the available number of Uber drivers, the application calculates an estimate price range. If the user chooses to accept, then the application calculates the nearest available driver to the user's location and sends the driver a notification that the driver can choose to accept. The real-time tracking of the Uber drivers is the feature which makes the application so unique as the user can get to their destination in the shortest possible time. The main functionality is performed on a map view by a user, as the user can view the real-time movement of available Uber vehicles, their own location and routes.

## 2.5 Summary

In conclusion, location is the main type of context within the focus of my application. The reason being, location is relevant to the interaction between both the user and the application. I have also understood how current applications which incorporate location based context awareness function. In terms of user interface and experience, all applications display a map to convey their services. Therefore, I will also implement a map within my application and centre the main activity of finding events around the map. In addition, I have learnt how to implement location based services within my application using the two methods, GPS and Network Provider.

## **3. Description of Prototype**

An android application which allows users to share events as well as find events. My underlying software architecture in my prototype demonstration is the Amazon EC2 ("Amazon EC2," n.d.) Instance as well as the use of Amazon RDS ("Amazon Relational Database Service (RDS) – AWS," n.d.) and Firebase Storage ("Cloud Storage," n.d.). As EC2 and RDS are the components needed to use the application outside of localhost and so at the university this is the method I use. The purpose of Firebase Storage within my application which is to store any media content. Figures 1-5 are screenshots

depicting a few of the features I have developed. The screenshots show how the user finds events using a map, how the directions are provided to the user, posting, editing, deleting events and event details.



Figure 1 – Find and receive directions from current position to event location

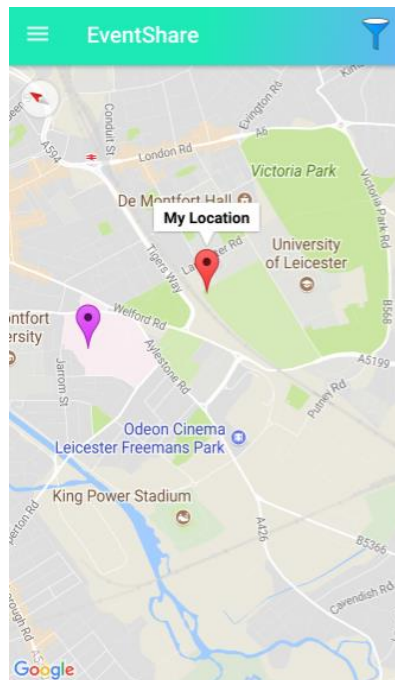


Figure 2 – Users Location

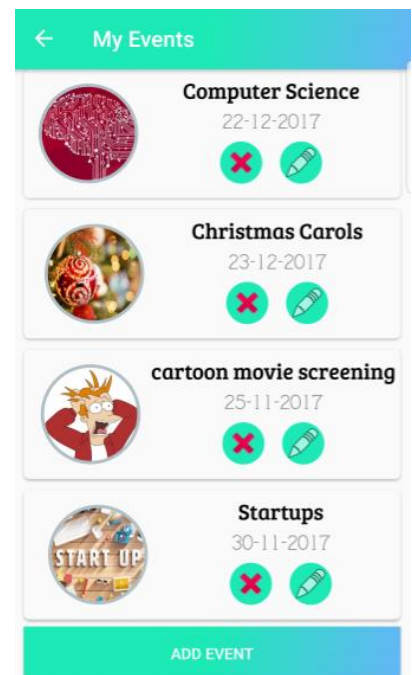


Figure 3 – Users' posted events. Choose to add, edit or remove events

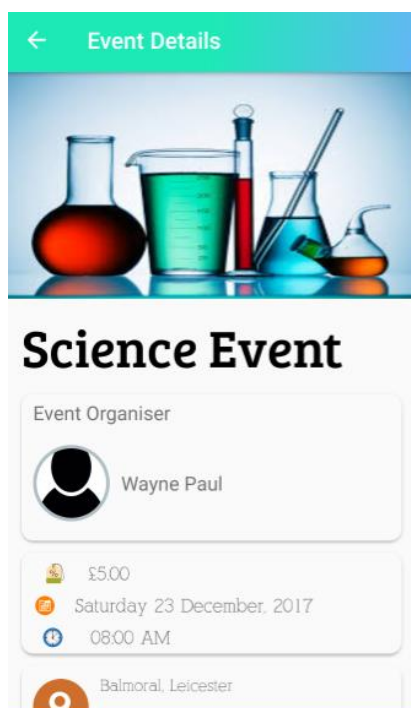


Figure 4 – Event Details (1)

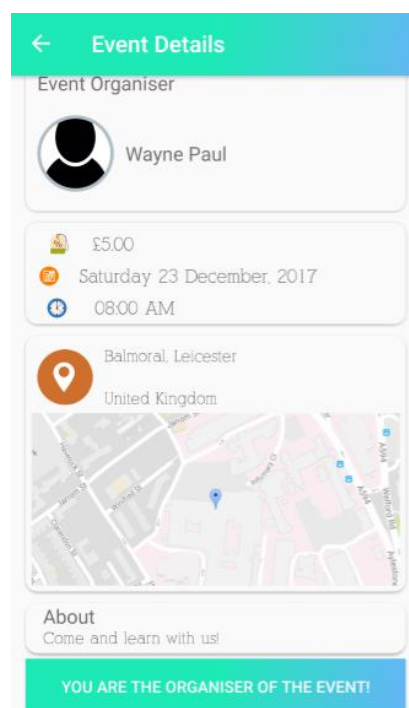


Figure 5 – Event Details (2)



### 3.1 Features Completed So Far

#### 3.1.1 Developed

I have developed the following features:

- Registration
  - User will be able to register on the application using their first name, last name, email address and password.
- Login
  - User will be able to login to the application using their registered email address and password.
  - User will not need to re-login once logged in if they leave the application.
  - User will be required to have access to internet as well as their location enabled to access the application.
- Posting events
  - Users will be able to post events by providing the following information: Name, Photo, Description, Price, Space Availability, Location, etc.
- Editing and deleting events
  - User will be able to edit an event they have posted.
  - User will be able to remove an event posted.
- Finding events using a map
  - User will be able to find events by accessing a map to view which events are closest to their location.
- Filtering events based on price, categories and distance from current location
  - User will be able to find events by providing a price range.
  - User will be able to choose categories for which they would like to view events for.
  - User will be able to enter a distance in miles, and view all events within the provided radius.
- Payments for tickets
  - User will be able to pay for a ticket using their credit/debit card information, if the ticket is not free.
- Event forum
  - User will be able to post any queries about an event, on the events' forum.

#### 3.1.2 Tested

I have tested the following features:

- Registration
- Login
- Filtering events based on price.

The reason why I have not tested all my developed features is because some features such as posting events are subject to change. When I implement my optional and additional features such as finding an optimal location for an event,

this will change the posting and editing event features. Furthermore, the testing I have completed is a combination of UI Testing and Unit Testing.

### 3.2 Upcoming Features

- Features which concern extending the prototype include:
  - Liking, rating and reviewing feature.
  - Event organisers being able to post jobs under events.
  - Users being able to apply for jobs.
  - Event organisers being able to track users who have signed up to come to their events, on the day of the event. The tracking will only be available 30 mins before the event starts till 30 mins after the event is finished. Then the feature will be disabled.
- Features extend all functionality mentioned above as well as incorporating new aspects to my application which use location based services include:
  - Consume existing events from sources such as Facebook and Eventbrite and displaying these events on my application.
  - The option for a user who is posting an event to choose a boundary on the map to show that the intended event must be within this vicinity and the application will calculate the optimum location within the given boundary as to where it should be for best success. This feature does rely on the ability of the application to be able to consume existing data regarding events as I would need this data to calculate the optimum location.
  - Context awareness which encompasses location as well as the users' current activity to recommend events tailored to the user.

## **4. Software Architecture, Algorithms and Data Structures**

### 4.1 Software Architecture

This section will discuss, with the use of diagrams, the underlying software architecture of the application. Furthermore, the algorithms which will allow me to implement features such as recommending events to the user based on the context, activity, location etc. In addition, the feature which provides an option to an event organiser to find the optimal location, within a given boundary, for their event they intend to share.

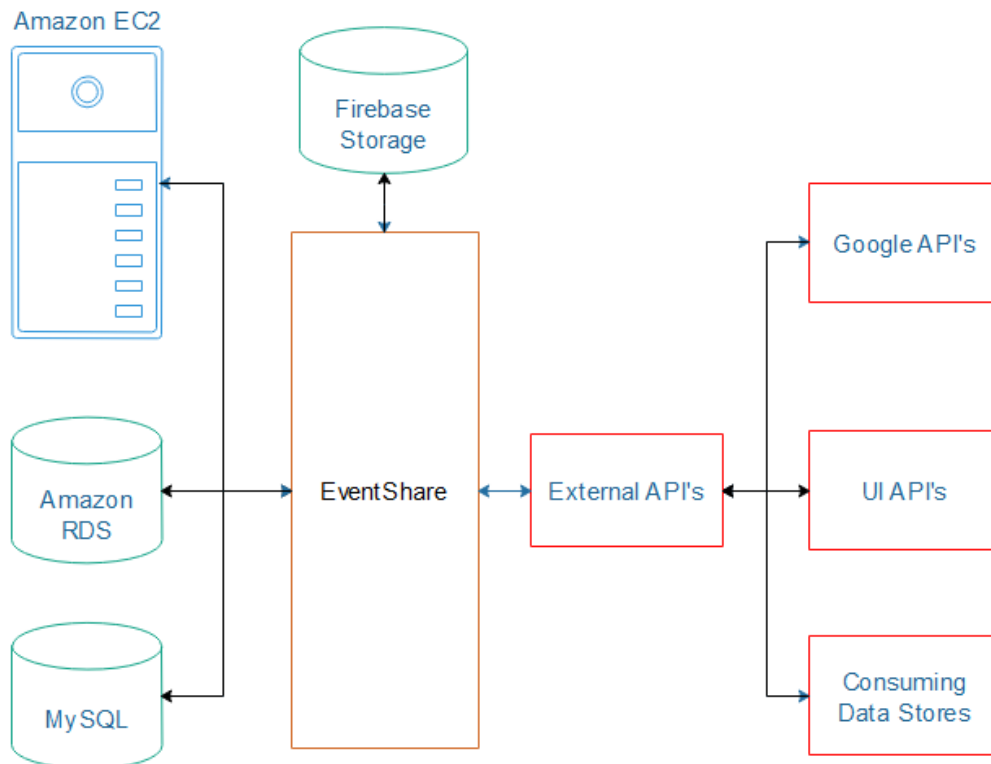


Figure 6 – High Level Architecture

Figure 6 illustrates a high-level view of my applications' underlying architecture. The EventShare component of the diagram represents the android device. This is the centre of the diagram as it central to all surrounding components. Containing all Java files, such as activities, domains, XML files and all other necessary files needed for building the application such as the Gradle files.

The Amazon EC2 component depicts the virtual server setup which provides the application with a public DNS. The reason for setting this up is because at the university, the application cannot connect to the localhost using the eduroam IP address. In addition, I wanted my application to be accessible anywhere. The Amazon EC2 instance provides a virtual server where all PHP files can be stored. This leads me to the next two components, Amazon RDS and MySQL. When the application wishes to access the PHP files in the EC2 instance, it uses the public DNS Amazon provides. However, the PHP files stored on the server cannot be used to access my Localhost database. Thus, I created an Amazon RDS in the cloud which provides a host. This host is used to create a connection in MySQL Workbench and create and utilise my database for the application here. I then query the database within the PHP files in the server using the host name provided.

The Firebase Storage component is used for storing all media content such as images for events, profile images etc. An external API called Picasso ("Picasso," n.d.) is used to retrieve and display images from the cloud (Firebase Storage). This leads me onto my

set of components within the architecture which are all under the header component, External API's. I have three sub sections within the External API's component.

- The first is, Google API's, these API's include Places API, Static Maps API, Maps API, Directions API etc. Each API has its own independent purpose within the application. For example, Directions API is used to provide data regarding the route from one position (latitude, longitude) to another (latitude, longitude). Whereas the Places API can be used to carry out various tasks including searching for places by name.
- The second is, UI API's, these API's include Picasso as mentioned above, is used for retrieving images using the URL of the images. It can also be used to display the image onto the view.
- The third is, Consuming Data API's, these API's include Facebook's and Eventbrite's API's. The API's will be used to retrieve data regarding existing events from their data stores and display these events onto my application.

## 4.2 Algorithms

### 4.2.1 Implemented

I have implemented an algorithm for the Haversine Formula ("Haversine formula," 2011). The formula calculates the distance between two coordinates. I have implemented this formula for the radius based filtering feature. A user can input 10 miles and the formula is used to calculate the distance between the users' location and every event. Then all events within 10 miles of the users' location are presented to the user on a map. Figure 7 shows the Haversine formula with a brief description of the notations used.

## **The Haversine Formula**

The haversine formula is used to find the distance  $d$  between two points with longitude and latitude  $(\psi, \phi)$ .

$$d = 2r \arcsin \left( \sqrt{\sin^2 \left( \frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\psi_2 - \psi_1}{2} \right)} \right)$$

Where  $r$  is the radius of the Earth.

Figure 7 - Haversine Formula

#### 4.2.2 Need Implementing

The first algorithm I will implement will allow those who have posted an event, the ability to track those who have signed up to come to their event. The tracking feature can be enabled 30 mins before the event begins all the way through to 30 mins after the event ends. After this, the feature will no longer be available.

The second algorithm I will implement will access the users' context, such as location, movement, etc, to recommend events. Once the application retrieves the users' activity, this information combined can be used to indicate which categories of events would be the most suitable to recommend. Then using the location, the application can consume data regarding events which are within a certain radius of the users' location and recommend these events.

The final algorithm I will implement will find the optimal location for an event the user wishes to post. The user will place multiple markers on a map, these markers will represent the boundary. This boundary will indicate that the location of the event cannot be outside this boundary. As this option will be available to the user only if the details they have entered for posting an event are all valid. The algorithm will then use the data regarding categories the user has chosen as well as the location boundary to retrieve all events which are in accordance to these factors from the database first, and then due to the database having very few events, events will be consumed from data stores such as Facebook and Eventbrite. Once the application has the events, it will then check the ratings of the events and the events which have the highest rating, their location is displayed to the user on a map. If more than one event has the highest rating, then all their locations are displayed to the user on a map.

### **5. Planning and Timescales**

#### Semester 1

<u>Task</u>	<u>Start Date</u>	<u>End Date</u>	<u>Status</u>
Learn Android development through online courses.	01/06/2017	01/07/2017	Completed
Create questionnaires for both event attendees and organisers. Then assign the questionnaires to end users to obtain feedback.	02/07/2017	25/07/2017	Completed
Choose and analyse 7 applications and analyse the feedback from the questionnaires for the dissertation. The result of the analysis will be a set of initial features, with priorities, and requirements.	26/07/2017	31/08/2017	Completed

Literature review on location based services within mobile applications for dissertation.	02/10/2017	15/01/2017	In Progress
Begin my design section of the project towards the end of September.	1/09/2017	10/01/2017	In Progress
Start the implementation and develop a mandatory feature every week, whilst also testing each feature after development.	15/09/2017	27/11/2017	In Progress
Preparation for interview and prototype demo.	20/11/2017	26/11/2017	Completed
Start the interim report and career plan after my prototype demonstration.	28/11/2017	05/12/2017	Completed
Continuing implementation, developing and testing an optional feature each week.	27/11/2017	12/01/2018	In Progress
Implement and test an additional feature each week.	13/01/2018	23/02/2018	Not Started

The changes I have made to my semester 1 timetable include the literature review was initially intended to be complete by 23/10/2017, however I have extended the completion date to 15/01/2017. Another change is the completion date of the design was initially 15/10/2017, however this has been extended to 10/01/2017. My preparation for prototype demo has been completed. Furthermore, my interim report and career plan will be completed upon hand in.

## Semester 2

<u>Task</u>	<u>Start Date</u>	<u>End Date</u>	<u>Status</u>
Complete dissertation, ready for submission.	26/07/2018	30/04/2018	In Progress
Review and submit software system.	30/04/2018	02/05/2018	Not Started
I will be preparing for my viva and presentation	03/05/2018	08/06/2018	Not Started

## 6. Bibliography and Citations

Accurate Heart Rate Monitor – Android Apps on Google Play [WWW Document], n.d. URL [https://play.google.com/store/apps/details?id=com.repsi.heartrate&hl=en\\_GB](https://play.google.com/store/apps/details?id=com.repsi.heartrate&hl=en_GB) (accessed 11.29.17).

Air Quality Index BreezoMeter – Android Apps on Google Play [WWW Document], n.d. URL [https://play.google.com/store/apps/details?id=app.breezometer&hl=en\\_GB](https://play.google.com/store/apps/details?id=app.breezometer&hl=en_GB) (accessed 11.29.17).

Amazon EC2 [WWW Document], n.d. . Amaz. Web Serv. Inc. URL <https://aws.amazon.com/ec2/> (accessed 11.29.17).

Amazon Relational Database Service (RDS) – AWS [WWW Document], n.d. URL <https://aws.amazon.com/rds/> (accessed 11.29.17).

Cloud Storage [WWW Document], n.d. . Firebase. URL <https://firebase.google.com/docs/storage/> (accessed 11.29.17).

Dey, A.K., 2001. Understanding and Using Context. Pers. Ubiquitous Comput 5, 4–7. <https://doi.org/10.1007/s007790170019>

Free Community-based GPS, Maps & Traffic Navigation App | Waze [WWW Document], n.d. URL <https://www.waze.com/> (accessed 11.29.17).

Harter, A., Hopper, A., Steggles, P., Ward, A., Webster, P., 2002. The Anatomy of a Context-aware Application. Wirel Netw 8, 187–197. <https://doi.org/10.1023/A:1013767926256>

Haversine formula, 2011. . MrReid.org.

Inc, I.T., 2017. Pedometer. ITO Technologies, Inc.

Inc, U.T., 2017. Uber. Uber Technologies, Inc.

Junglas, I., Watson, R., 2008. Location-based services. Commun ACM 51, 65–69. <https://doi.org/10.1145/1325555.1325568>

Liutkauskas, V., Matulis, D., Plėštys, R., 2015. Location Based Services. Elektron. Ir Elektrotechnika 52. <https://doi.org/10.5755/j01.eee.52.3.10935>

Marmasse, N., 1999. comMotion: A Context-aware Communication System, in: CHI '99 Extended Abstracts on Human Factors in Computing Systems, CHI EA '99. ACM, New York, NY, USA, pp. 320–321. <https://doi.org/10.1145/632716.632910>

mbH, H.I., 2017. Arriva UK Bus App. HaCon Ingenieures. mbH.

Picasso [WWW Document], n.d. URL <http://square.github.io/picasso/> (accessed 11.29.17).

Schilit, B., Adams, N., Want, R., 1994. Context-Aware Computing Applications, in: 1994 First Workshop on Mobile Computing Systems and Applications. Presented at the 1994 First Workshop on Mobile Computing Systems and Applications, pp. 85–90.  
<https://doi.org/10.1109/WMCSA.1994.16>

Storm, S., n.d. Implementation of Location based Services in Android using GPS and Web Services.