Location Based Services

Department of Electrical Engineering
California State University, Los Angeles
jilluri@calstatela.edu

California State University, Los Angeles

By
Jagadeesh Iluri
Cin: 304373269

Abstract:

Location based services have been at front and center of almost every major mobile service these days. As a part of this project, I intend to demonstrate the location based services and how they play well together with web services to create an efficient mobile application. The mobile platform chosen is ANDROID and I will be using Yahoo's weather API that is free to use for registered users. This application will essentially locate the user using GPS, send that data to Yahoo servers to determine the current and forecast weather for that particular location, parse the data received and present it in a user-friendly mobile UX design.

Introduction:

Usage of smart phones increasing rapidly, Services providing using location based plays vital role in our day to day life. Providing information at right place and right time is very important, using location based services we can achieve this goal.

Few Examples of Applications using LBS:

Emergency:

Location services plays important role during natural calamities, such as alerting people who are situated around the danger location and informing them way ahead, can avoid loss of life and sending medical services to a location is also way simple using this services.

Commercial:

LBS can provide latest information on sales to consumer using his/her location, LBS knows the consumer location and whenever consumer passes by store or close to the location the they give an updates on sales, which helps boosts the sales.

Household:

Safety is very important and using LBS we can track our loved once time to time, like kids in school, elderly people goes for walk etc., which provides location updates time to time and it is very safe.

Gamers:

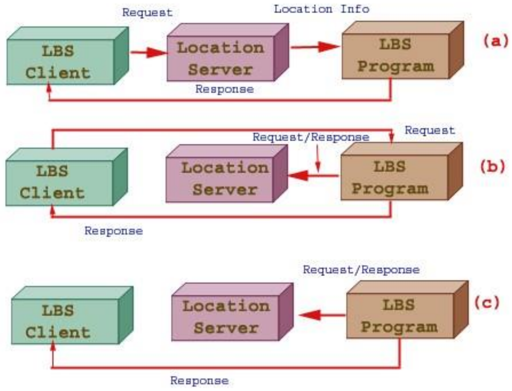
By using LBS people can find new people and challenge them, LBS provides information using their current location and gives you information of gamers situates at that location which improves the gaming quality.

LBS Services:

Location based services has mainly 3 stages they are

- LBS Client
- LBS Server
- LBS Program

LBS client sends request to Location server and location server responds with location information which is processed in LBS program and sends information to LBs client. Complete process runs on request/response. Response sent back to LBS client from LBS program/LBS Server.



Three Models of LBS Services

LBS PROGRAMMING IN ANDROID

Android provides access to the following components to

Facilitate the implementation of LBS services;

1. Location Manager
2. Location Provider
3. Geocoding
4. Google-Map

Location Manager:

Location Manager Class of android is present to manage all other components needed to establish a LBS system.

Location Provider:

Location provider represents the technology to determine the physical location i.e. to handle GIS. *LocationProvider* component of Android application is a present to facilitate the determination of available provider and selection of suitable one.

Geocoding:

Reverse geocoding provides a way to convert geographical coordinates (longitude, latitude) into street address and forward geocoding provides a mean to get geographical coordinated from Android provides a number of objects to handle maps in LBS system like MapView which displays the map.

Google-Map:

Google Map is an open source which provides human readable date to binary digits address. To handle this a MapActivity class is there. To annotate map it provides the overlays class. Even it provides canvas by which one can easily create and display multiple layers over the map.

HTTP Request:

An HTTP client sends an HTTP request to a server in the form specific format, the format should be same in order to interpret correctly.

Parts of request:

1st part:

There are critical and optional parts in client request to server.

- Request Method

  The request method indicates the method to be performed on the resource identified by the given Request-URI. The method is case-sensitive and should always be mentioned in uppercase.

- Recourses Line (path: To take action on it)

  The Request-Line begins with a method token, followed by the Request-URI and the protocol version, and ending with CRLF. The elements are separated by space SP characters.

2nd part:

Headers: Extra information to help servers.

We will study General-header and Entity-header in a separate chapter when we will learn HTTP header fields. For now, let's check what Request header fields are.

The request-header fields allow the client to pass additional information about the request, and about the client itself, to the server.

- Languages (Response to come back)
- Character set (Response)
- Content type (What we are sending back)
- Cookies (Small data of contents)

Final part of information:

Body (Optional):

HTTP Message Body is the data bytes transmitted in an HTTP transaction message immediately following the headers.

Location Manager in Android:

android.location.LocationManager

java.lang.Object

This class provides access to the system location services. These services allow applications to obtain periodic updates of the device's geographical location, or to fire an application-specified Intent when the device enters the proximity of a given geographical location.

Context.getSystemService(Context.LOCATION_SERVICE).

Location API methods require

ACCESS_COARSE_LOCATION or ACCESS_FINE_LOCATION

If your application only has the coarse permission then it will not have access to the GPS or passive location providers. Other providers will still return location results, but the update rate will be throttled and exact location will be obfuscated to a course level of accuracy.

JSON: (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

JSON is built on two structures:

- A collection of name/value pairs. In various languages, this is realized as an *object*, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an *array*, vector, list, or sequence.

REFERENCES

1. Lee, Wei-Meng, “Beginning Android Application Development: Provides information on creating Android applications for mobile devices”.
2. Brimicombe, Allan Chichester, UK ; Hoboken, NJ: Wiley-Blackwell, 2009, “Location-based services and Geo-information Engineering”.

3. Murphy, Joe Chicago: ALA Tech Source, an imprint of the American Library Association, 2012, “Location-aware services and QR codes”.

4. Bartlett, David Cambridge: Cambridge University Press, 2013, “Essentials of positioning and location technology”.

5. Michel Kareis, “Geographic locator codes”.