Location Based Services and Integration of Google Maps in Android

Pankti Doshi[1], Pooja Jain[2],Abhishek Shakwala[3]

[1]NMIMS University, Assistant Professor, Department of Computer Engineering,
V.L. Mehta Road, Vile Parle (W), Mumbai – 400 056
pankti.doshi@nmims.edu

[2]NMIMS University, Mukesh Patel School of Technology Management and Engineering,
V.L. Mehta Road, Vile Parle (W), Mumbai – 400 056
poojajain.nmims@gmail.com

[3]NMIMS University, Mukesh Patel School of Technology Management and Engineering,
V.L. Mehta Road, Vile Parle (W), Mumbai – 400 056
abhishekshakwala.nmims@gmail.com

Abstract:*In today's scenario, mobile computing has advanced to such an extent where the user has access to all the information on a single device. Today people are always moving with mobile devices like laptops, cell phones, tablets etc. Using the user's geographic location, a lot of information related to the user of the mobile device can be collected. The knowledge of mobile user's location can improve the class of services and applications that can be provided to the mobile device user. These classes of applications and services are termed as location based services. Location Based Service (LBS) is a kind of service that helps in getting the geographical location of the user and more useful information near to the user location. This location based information can be obtained in different terms such as position, vicinity, proximity, context, maps, routes, places etc. Our paper focuses on LBS in detail and identifies key components of LBS for providing this service to the user on the Android platform. It also explains the use and implementation of Google Maps and its APIs in getting various location based information on Android.*

Keywords: Location Based Services, Android, Location API, Google Maps

1. Introduction

Location based service (LBS) has emerged rapidly in mobile data services due to rapid enhancement in wireless communication and location positioning technologies. Users having location-aware wireless devices can know about their surroundings at any place, anytime. Location-based services (LBS) refer a set of applications that utilizes the knowledge of the geographical location of a location-aware mobile device and provides services based on that information. Such services can include navigating to a particular location, finding friends and family on social networking sites, tracking the location-aware mobile device, map assistance by locating points of interest etc.

Following are the types in which these services can be categorized [3]:

- **Public safety or emergency services:** In times of emergency the location of the mobile phone user can be easily traced.

- **Consumer services:**
  a. Navigation – navigating to a particular destination using route maps, routing real time traffic etc.
  b. Location based advertising – sending advertisements to a user that is within the vicinity of a store
  c. Family and friend finder – helps in keeping track of the location of family and friends.
  d. Location based search – allows users to query on the information about its surrounding like nearby restaurant, hospital, theatre etc.
  e. Location based mobile gaming

- **Enterprise services:**
  a. Firms in fleet and asset tracking
  b. Field service dispatching
  c. Route and delivery optimization
  d. Mobile workforce management.

There are two ways in which the location of a mobile device can be tracked:

1. **Communication Network:** Communication network is the mobile network which transfers service request from user to service provider, and requested information back to the user. Global System for Mobile communications (GSM) and Code Division Multiple Access (CDMA) are the most common standards for mobile network and are currently being used by majority of mobile phones globally. Mobile networks are usually controlled and maintained by service operators who provide connectivity for mobile users and charge them for data and voice transmission [2].

2. **GPS:** GPS is a satellite based navigation system which provides location and time information by communicating with the 24 satellites placed in the earth's orbit. GPS works in all weather conditions, anywhere on the earth, 24 hours a day. Mobile devices must have a GPS receiver in order to decode the GPS signals. [1] Assisted-GPS is the new technology for smart phones that greatly improves the startup performance or time-to-first-fix (TTFF) of a GPS satellite-based positioning system. It is used with a GPS enabled cellular phone, to make the location of a cell phone available, much faster and with better accuracy. It consumes less battery power and requires fewer satellites.

2. LBS Components and Service Process

LBS service is supported by some infrastructure components as listed below: [3].

Mobile Devices: Access to the LBS services is provided with the help of Mobile devices, to send requests and retrieve results for the users. This mobile device can be in the form of mobile phones, tablets, portable navigation devices (PNDs), and so on.

Applications: Applications is nothing but the software developed by an application developer or provider which is downloaded and installed on the user's mobile device, which also provides the interface to access LBS services.

Communication network: Communication network as explained above is the mobile network which transfers service request from user to service provider, and requested information back to the user [2].

Positioning component: To track user's current location, positioning component is needed in the LBS application, since it is not feasible for users to input location manually and then give it to LBS services. Instead some positioning technologies, such as satellite positioning, cellular network positioning can be used to obtain user's location.

Service servers: Service servers are responsible for processing service requests and sending back request results, and these service servers are maintained by service providers. Basically these servers provide functionalities such as calculating positions, search for a route, or search specific information based on user's position. Service providers usually do not store and maintain all the information requested by users. Instead, content providers are responsible for collecting and storing geographic data, location-based information, and other related data. These data will be requested and processed by service servers and then returned to users.
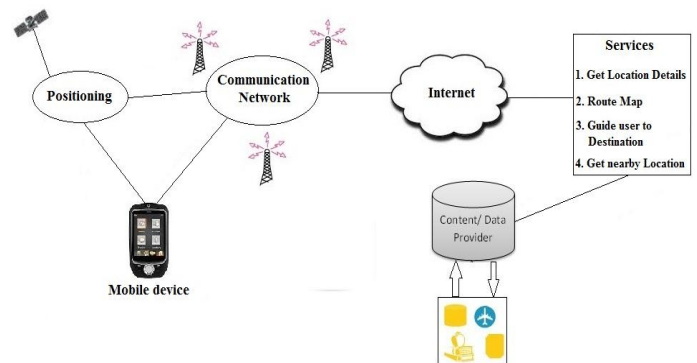

Figure 1:LBS Process

Following are the steps for LBS Service process as shown in Figure 1:

Step1: User sends a service request to service server using the LBS application running on the mobile device.

Step2: Along with the service request, user's current location information is retrieved from positioning component and sent to the service server using communication network.

Step3: Service server now analyses the geo-specific information requested by the user and then asks for relevant data from content or data provider (for example, geographic databases).

Step4: Finally, the requested information is sent back to the user's mobile device through mobile communication network.

3. Architecture of LBS

The architecture of LBS consists of three layers as follows and is shown in figure 2:

1. LBS Application
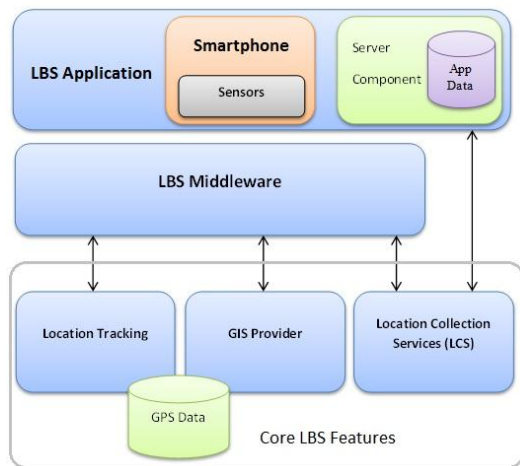2. LBS Middleware
3. Core LBS Features

Figure 2: LBS Architecture [2]

3.1 LBS Application

LBS Application layer comprises of applications such as Locating people on a map, getting nearby places, Find my friends, and so on. This layer also consists of Smart phones with number of sensors and server components such as application server and spatial databases. Application server, which is the processing center for a LBS platform that handles user interface functions and communicates with the spatial database [4].

3.2 LBS Middleware

LBS middleware bridges protocols and network technology with wireless and Internet technology. It is either deployed within the network operator's network or hosted by an application service provider [4]. LBS middleware gives access to core features which include:

a. Location tracking
b. GIS Provider
c. Location Collection services

3.3 Core LBS Features

a. Location tracking

Location tracking is a service for Global positioning System (GPS) phones that can be used by users to monitor the movement of the person caring the mobile device. This component stores traced location of users and it is the fundamental component in LBS as it contains the data that allows a user's route to be determined and potentially predicted.

b. GIS Provider

GIS Provider facilitates with geo-specific functionality for LBS application including map information, map visualization and directory services. For example, Google maps with its API can be considered as a GIS Provider [2].

c. Location Collection Service

This component basically focuses on getting latitude and longitude for a specific user. Location Collection Service can be accessed with the help of LBS middleware, For example, mobile network triangulation via a service provider or directly via GPS receiver in the smartphone.

4. Android Location API

Android contains the *android.location* package which provides the API to determine the current geo position. This package contains various classes as shown in table below[5]:

Table 1: Classes in android location package [5]

Address	It is a class representing an Address, which is a set of Strings describing a location.
Criteria	It is a class indicating the application criteria for selecting a location provider.
Geocoder	It is a class for handling geocoding and reverse geocoding.
GpsSatellite	This class represents the current state of a GPS satellite.
GpsStatus	This class represents the current state of the GPS engine.
Location	This is a data class representing a geographic location.
LocationManager	It is a class which provides access to the system location services.
LocationProvider	It is an abstract superclass for location providers.

Android Location API methods require ACCESS_COARSE_LOCATION and ACCESS_FINE_LOCATION permissions to be included in the manifest file which provides essential information about an application to the android system. ACCESS_COARSE_LOCATION allows an app to access approximate location derived from network location sources such as cell towers and Wi-Fi. ACCESS_FINE_LOCATION allows an app to access precise location from location sources such as GPS, cell towers, and Wi-Fi [5].

4.1 Location Manager

The LocationManager helps to manage all the components needed in a LBS system. It provides access to the Android location service. This is done by instantiating an object of the LocationManager class through *getSystemService(Context.LOCATION_SERVICE)*[5].

4.2 Location Provider

The *LocationProvider* class is the superclass of the different location providers which deliver the information about the current location in an LBS system. This information is stored in the Location class. The Android device can have several *LocationProvider*. In most cases we have the following*LocationProvider*available:

Table 2: Location Providers

Network	It uses the mobile network or Wi-Fi to determine the best location. May have a higher precision in closed rooms then GPS.
GPS	It uses the GPS receiver in the Android device to determine the best location via satellites. Usually it has better precision than network in detecting the user location.
Passive	It allows to participate in location of updates of other components to save energy.

To get the list of all the enabled location providers available on the device, call getProviders function in android:

List providers = locationManagerObject.getProviders(true);

The parameter true indicates that only enabled providers are to be returned.

Selecting Location Provider via Criteria

For a suitable selection of the best location provider, a Criteria object can be used which defines how the provider should be selected.

Criteria criteria = new Criteria();

criteria.setAccuracy(….);

criteria.setPowerRequirement(….);

/ more criteria can be set here*/*

bestProvider= locationManagerObject.getBestProvider(criteria, true);

If more than one provider is available fulfilling the given criteria then the one with best performance is returned and if no provider is found for the defined criteria then criteria are loosened in order Power use, Accuracy, Ability to return bearing, speed, and altitude[2].

4.3 Location Listener:

LocationListener class provides periodic updates about the geo-position. It provides callback methods which are called when location gets changed. The listener object needs to be registered with the location manager to receive location updates.

4.4 Geocoder

The Geocoder class helps in determining the geo-coordinates (latitude, longitude) for a given address which

is called as forward geocoding and possible addresses for a given set of geo-coordinates which is called as reverse geocoding. In android, forward geocoding is done by using *getLatitude()* and *getLongitude()* method and reverse geocoding is done by using getFromLocation method.

5. Google Maps

Android allows us to integrate google maps with our application and provides users with functionalities such as show any location on the map, or show different routes on the map, etc. Google provides this facility using google play services library which can be downloaded externally from the android developers' site or from the android SDK manager. A Google Map API key is required to integrate the maps in android application. Google Maps can be integrated by performing the following steps:

1. Installing and configuring the Google play services library from android SDK manager.
2. Adding Google play services to the application library.
3. Retrieving Android certificate i.e. the SHA1 fingerprint of the keystore that is used to sign the application.
4. Create a project in the Google APIs console.
5. Obtaining the Google Maps API key.
6. Add the unique generated API key to the application.

To handle maps in LBS system android provides a number of objects like *MapView* which displays the map, *MapFragment* which provides the life cycle management and services for displaying a google maps widget, Markers and so on.

5.1 Directions API

The Google Directions API [6] is a service that calculates directions between locations using an HTTP request. Directions can be searched for several modes of transportation, including transit, driving, walking or cycling [6]. To use the Directions API in an android application we need to obtain a Directions API key in the same way as we obtain the Google Maps API key.

5.1.1 Directions Request

A Directions API is an HTTP URL defined in the following way [6]:

http://maps.googleapis.com/maps/api/directions/output?parameters
Where output may be JavaScript Object Notation (JSON) or XML

Table 3: Direction Request Parameters [6]

Arguments	Description
Origin (required)	The address or textual latitude/longitude value from which we wish to calculate directions. If

Parameter	Description
	address is passed as a string, the Directions service will geocode the string and convert it to a latitude/longitude coordinate to calculate directions. If coordinates are passed, it must be ensured that no space exists between the latitude and longitude values.
Destination (required)	The address or textual latitude/longitude value upto which we wish to calculate directions. If address is passed as a string, the Directions service will geocode the string and convert it to a latitude/longitude coordinate to calculate directions. If coordinates are passed, it must be ensured that no space exists between the latitude and longitude values.
Sensor (required)	Indicates whether or not the directions request comes from a device with a location sensor. The value of this argument must be either `true` or `false`.
Mode (optional)	Specifies the mode of transport to use when calculating directions.
`Language` (optional)	The language in which to return results
Key (required)	Application's API key. The key determines your application's identity.
`units` (optional)	Specifies the unit system to use when displaying results

5.1.2 Direction Response

The direction response is returned in the format indicated by the `output` flag within the URL request's path [6].

Table 4: Direction response elements [6]

Elements	Description		
Status	Contains status of the request. May contain debugging information to help track down why the Directions service failed.		
Routes	Contains an array of routes from the origin to the destination. Routes consist of nested Legs and Steps.		
Legs	Each element in the `legs` array specifies a single leg of the journey from the origin to the destination in the calculated route.		
	steps[]	An array of steps denoting information about each separate step of the leg of the journey	
	distance	Total distance covered by this leg	
	duration	Total duration of this leg	
	duration_in_traffic	Total duration of this leg (taking into account current traffic conditions)	
	arrival_time	Estimated time of arrival for this leg (only returned for transit directions).	
	`departure_time`	Estimated time of departure for this leg. The `departure_time` is only available for transit directions	
	start_location	The latitude/longitude coordinates of the origin of this leg	
	end_location	The latitude/longitude coordinates of the given destination of this leg.	
	start_address	Human-readable address reflecting the `start_location` of this leg.	
	`end_address`	Human-readable address reflecting the `end_location` of this leg.	
Steps	Each element in the `steps` array defines a single step of the calculated directions. Step is the most atomic unit of a direction's route.		
	html_instructions	Formatted instructions for this step	
	Distance	Distance covered by this step until the next step.	
	Duration	Time required to perform the step until the next step	
	start_location	Location of the starting point of this step (single set of `lat` and `lng` fields).	
	end_location	Location of the last point of this step (single set of `lat` and `lng` fields).	
	sub_steps	Detailed directions for walking or driving steps	

		in transit directions (only available when `travel_mode` is set to "transit")
	`transit_details`	Transit specific information

5.2 Places API

The Google Places API is a service that returns information about places defined within the API as establishments, geographic locations, or prominent points of interest using HTTP requests [7]. The location within theplace requests are specified as latitude/longitude coordinates.

The following Place requests are available [7]:

a. **Place Searches:** It returns a list of Places based on user's location or search string along with summary information about each Place. Places Search consists of:
   i. Nearby Search Requests
   ii. Text Search Requests
   iii. Radar Search Requests
   iv. Search Responses

The HTTP URL of a place search request is defined in the following way [7]:

https://maps.googleapis.com/maps/api/place/search/output?arguments

Where output may be JavaScript Object Notation (JSON) or XML

b. **Place Details:** This request return more details about the indicated place such as address, phone number, etc, along with user reviews. When the Places service returns results from a details request, it places them within a single result which consists of many fields [7].

The HTTP URL of aplace details request is defined in the following form [7]:
https://maps.googleapis.com/maps/api/place/details/output?rguements

Where output may be JavaScript Object Notation (JSON) or XML

c. **Place Actions:** It allows you to supplement the data in Google's Places Database with data from your application. It includes scheduling of Events, add and remove Places, or weight Place rankings from user activity with Place Bumps. These data added using Place Actions is application specific, which means it will be available to applications that share the same API Key.

6. Conclusion

In this paper we probe into Location Based Services, which utilizes the knowledge of the geo-specific location of a location-aware mobile device and facilitates services based on that information. We also discussed about LBS Components and LBS Architecture. Location Based Services uses technologies like GPS, cellular network, Wi-Fi to provide various services. Android provides various services to implement location based applications. Our paper discusses APIs like Location API, Google Maps, Direction API and Places API which helps in making location-aware applications under android platform.

Today android is the most used mobile operation system in the market. Hence developing LBS applications on android platform can help maximum population to get the benefits of this technology. Hence Location Based Services can help the users in a variety of aspects and has a greater scope of development in various mobile operating system technologies like Android.

References

[1] ManavSinghal, Anupam Shukla, "Implementation of Location based Services in Android using GPS and Web Services ", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, January 2012.

[2] Amit Kushwaha, VineetKushwaha, "Location Based Services using Android Mobile Operating System", International Journal of Advances in Engineering & Technology, Vol. 1, Issue 1, pp.14-20, Mar 2011.

[3] Schwinger, W., Grin, C., Prll1, B., and Retschitzegger, W. A light-weight framework for location-based services. In Lecture Notes in Computer Science (Berlin, 2005), Springer, pp. 206_210

[4] Location-based services: architecture overview, Costas Pontikakos, Thomas Glezakos and Theodore Tsiligiridis

[5] Android Developers http://developer.android.com/reference/android/

[6] Google Directions API

https://developers.google.com/maps/documentation/directions/

[7] Google Places API https://developers.google.com/places/documentation/