

JokeOfTheDay

Aniket Pansare, Sagar Abhang

September 14, 2012

Abstract

JokeOfTheDay is an application developed in Android which allows the user to get his current location along with a joke for that day. The application makes use of the Android's Location Manager Services for fetching the user location based on the GPS provider and Network Provider. It selects the best provider among the available providers on the basis of criteria specified. The location information is retrieved using the latitude and longitude. The Application makes a TCP connection to the `impact.asu.edu` server to fetch the joke of the day. It performs a proper handshake with the server in a predefined message format and retrieves the Joke of the Day. After receiving the Joke, the application displays the joke on the screen and gracefully closes the connection with the server.

Contents

0.1	Purpose of the Assignment:	2
0.2	Software Requirement and Specifications	2
0.2.1	Major Constraints:	2
0.2.2	Hardware Resources used:	2
0.2.3	Software Resources used:	2
0.3	Technical Details:	2
0.4	Design and Implementation:	3
0.5	Limitations	4
0.6	Future work	4
0.7	Conclusion	5
0.8	Application Run	5

0.1 Purpose of the Assignment:

To design an Android application, that makes use of location services and client-server programming. The main focus of this assignment is for students to get familiar with the android framework.

0.2 Software Requirement and Specifications

0.2.1 Major Constraints:

1. The Application will run only on android powered devices.
2. The device must have GPS and Internet access for location information to be displayed along with the joke.
3. The <http://impact.asu.edu:8000> server must be up and running to display the Joke of the Day.

0.2.2 Hardware Resources used:

- Android powered Mobile device or Tablet.
- RAM 256MB
- Processor: 600Mhz, MSM7227, Qualcomm

0.2.3 Software Resources used:

- Android 1.6 SDK, Release 1
- Platform: Android 2.2,Froyo
- Eclipse SDK: Juno
- JDK 1.7.0_07

0.3 Technical Details:

- User-permissions: INTERNET and ACCESS_FINE_LOCATION permissions given in AndroidManifest.xml
- Package: com.example.jokeoftheday
- Source files:
 1. MainActivity.java
 2. ConnectGetThread.java
- xml files: activity_main.xml
- Executable File: JokeOfTheDay.apk

0.4 Design and Implementation:

The JokeOfTheDay application displays a joke on the user's request. It also makes use of the Location services provided on the Android platform to determine the current location of the user and displays it on the screen.

The Joke server is hosted at impact.asu.edu and port 8000. The application acts as client and connects to Joke server via TCP connection. The application follows a predefined syntax to communicate with the server and fetch the joke of the day.

We used the request and response format, so that messages sent and received are parsed properly on both client and server side. Multiple contiguous spaces to separate fields not allowed, we used just one space character to separate the fields in message.

Request and Response format:

Format: <AA1 MSGTYPE WHO WHAT >

- with MSGTYPE being HELLO THIS IS, HI, GOODBYE, REQUEST, RESPONSE;
- with WHO being the server name or client name;
- with WHAT being what is being communicated with the MSGTYPE. (WHAT may be empty)

Message Parsing:

The application establishes a socket connection with the server and waits for server to respond. The server responds in pre-defined syntax, such as <aa1 msgtype who what>. The request from the server is parsed till the end of the message i.e >is received. To do message parsing java class StringBuffer is used as it is more efficient than class String.

1. Server: <AA1 HELLO THIS IS servername WHO ARE YOU?>
2. Client: <AA1 HELLO THIS IS clientname>
3. Server: <AA1 HI clientname PLEASE SEND YOUR REQUEST>
4. Client: <AA1 REQUEST servername JOTD>
5. Server: <AA1 RESPONSE clientname JOTD joke_of_the_day_text>
6. Client: <AA1 GOODBYE servername>
7. Server: <AA1 GOODBYE clientname HAVE A NICE DAY>

Criteria:

We used the method `getBestProvider(Criteria criteria, boolean enabledOnly)` to fetch the location provider that best meets the given criteria. If several providers meet the criteria, then one with the best accuracy is returned.

Following criteria is set for getting the Best Provider.

- Accuracy: ACCURACY_FINE (To get finer location accuracy)
- Altitude Required: True (Provider with altitude information)
- Bearing Required: True (Provider with bearing information)
- Cost Allowed: True (Provider allowed to incur monetary cost)
- Power Required: POWER_LOW (To get provider with low power requirement)

Threading:

Initially we tried implementing the joke fetching code in the main thread. But the Honeycomb SDK and above versions does not allow networking operations in main thread. It recommends the use of a separate thread[1]. So we created a separate thread to fetch the joke from Joke Server. Threading also helps to capture the on location change event updates and display it on the screen.

Scrollbar:

TextView widget is used to display the joke received from server. If the joke has multiple lines, then only few lines of the joke were getting displayed. To address this issue, we used the scrollbar to display the Joke of the day. This allows the user to scroll in case the joke has multiple lines or the application is used on a device having smaller screen.

Log:

To keep track of what is happening in the background when application is running, we used android.util.Log API to log the output.

0.5 Limitations

1. The client does not time out even if server does not send anything for certain time. The client needs to specify timeout period for which client waits for response from server.
2. Device must have GPS enabled to get location.

0.6 Future work

1. Google maps can be used to display location of the user.
2. Timeout functionality is to be implemented so that client does not wait for indefinite time for servers response.

0.7 Conclusion

JokeOfTheDay application is implemented on Android SDK. On the technical perspective, using of Android API to get location of the user is very efficient than writing new one. Use of Java Multithreading concept improves the responsiveness of the application[2]. Future work can be done using Google Map API and project results can be verified on different locations.

0.8 Application Run

Figure 1: JokeOfTheDay in Applications.

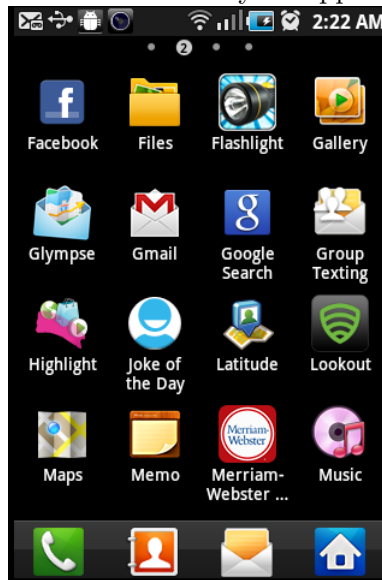


Figure 2: JokeOfTheDay Application On Launch

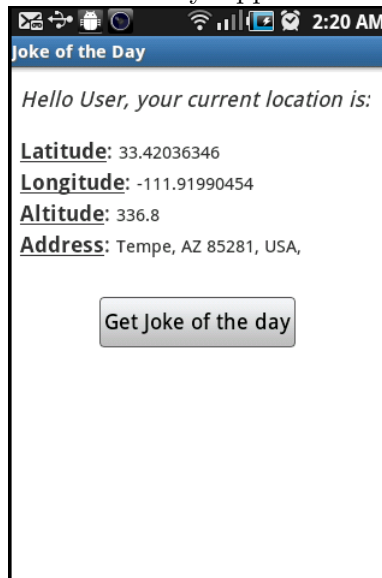
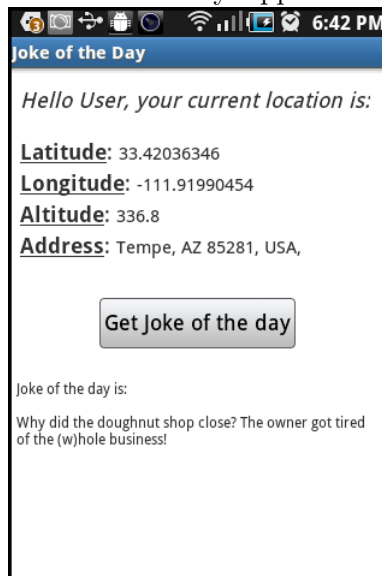


Figure 3: JokeOfTheDay Application Output



Bibliography

- [1] Developers: NetworkOnMainThreadException:

`http://developer.android.com/reference/android/os/NetworkOnMainThreadException.html`, September 2012.

- [2] Android Developers: Designing for Responsiveness:

`http://developer.android.com/guide/practices/responsiveness.html`, September 2012.