Describe the APK format.

The APK file is compressed the AndroidManifest.xml file, application code (.dex files), resource files, and other files. A project is compiled into a single .apk file.

What is an action?

A description of something that an Intent sender desires.

What is activity?

A single screen in an application, with supporting Java code.

What is intent?

A class (Intent) describes what a caller desires to do. The caller sends this intent to Android's intent resolver, which finds the most suitable activity for the intent. E.g. opening a PDF file is an intent, and the Adobe Reader is the suitable activity for this intent.

How is nine-patch image different from a regular bitmap?

It is a resizable bitmap resource that can be used for backgrounds or other images on the device. The NinePatch class permits drawing a bitmap in nine sections. The four corners are unscaled; the four edges are scaled in one axis, and the middle is scaled in both axes.

What languages does Android support for application development?

Android applications are written using the Java programming language.

What is a resource?

A user-supplied XML, bitmap, or other file, injected into the application build process, which can later be loaded from code.

Do add more questions below...

Top

admin Post subject: Re: Android Interview Questions & AnswersPosted: Tue Aug 17, 2010 8:00 pm

Site Admin

How will you record a phone call in Android? How to get a handle on Audio Stream for a call in Android?

Permissions.PROCESS\_OUTGOING\_CALLS: Allows an application to monitor, modify, or abort outgoing calls.

What's the difference between file, class and activity in android?

File - It is a block of arbitrary information, or resource for storing information. It can be of any type.

Class - Its a compiled form of .Java file . Android finally used this .class files to produce an executable apk

Activity - An activity is the equivalent of a Frame/Window in GUI toolkits. It is not a file or a file type it is just a class that can be extended in Android for loading UI elements on view.

What is a Sticky Intent?

sendStickyBroadcast() performs a sendBroadcast (Intent) that is "sticky," i.e. the Intent you are sending stays around after the broadcast is complete, so that others can quickly retrieve that data through the return value of registerReceiver (BroadcastReceiver, IntentFilter). In all other ways, this behaves the same as sendBroadcast(Intent).

One example of a sticky broadcast sent via the operating system is ACTION\_BATTERY\_CHANGED. When you call registerReceiver() for that action -- even with a null BroadcastReceiver -- you get the Intent that was last broadcast for that action. Hence, you can use this to find the state of the battery without necessarily registering for all future state changes in the battery.

Android is an open platform consisting of an operating system, middleware and powerful applications for all types of mobile devices. The revolutionary Android platform has been developed by leading technology and mobile companies like Google, T-Mobile, Intel, HTC, Qualcomm and others under the banner of Open Handset Alliance to provide a novel and reliable platform on which new mobile services can be quickly developed and provided to the end users.

Android is an open-source platform developed under the Open Handset Alliance to enable faster development of mobile applications and provision of services to the user. Google is the leading company to develop and promote Android, however there are other companies as well who are involved in the development of Android. Some of these include T-Mobile, HTC, Qualcomm and others. Android literally means a robot with human qualities and it seems it is just what the developers of this amazing mobile platform that combines an operating system, middleware and powerful application development, intended.

Android made its debut in 2007 on Google's G1 mobile phone developed in collaboration with T-Mobile. Although Google has claimed that Android would be open source, which means developers can twist and tweak Android's code to come out with their own Android versions, the reality is a little different. It turns out that Google has allowed access to only selective parts of Android's code while some other parts are still off bound.

Android Features:

Android has many amazing and unique features that are of significance to developers and users alike, some of which are:

•Application Framework that enables reuse and replacement of components

•Optimized Graphics that is powered by customized 2D graphics library and 3D graphics based on the OpenGL ES 1.0 specification

•Media Support for common video, audio, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)

•Provision of Bluetooth, EDGE, 3G and Wi-Fi

•SQLite for structured data storage

•Open source WebKit engine based integrated web browser

•Camera, GPS, Compass, and Accelerometer (dependent on hardware support)

•GSM Telephony

•Dalvik virtual machine optimised for mobile devices

•Rich Development environment that includes a device emulator, debugging tools, performance and memory profiling and a Eclipse IDE plugin.

Google Android will be available with a host of features that includes a web browser, email client, calendar, contacts, SMS feature, maps and others. All the Google Android applications are written in Java and run on Dalvik virtual machine, which itself functions on top of a Linux kernel.

Android's outstanding features are that developers can get complete access to the identical framework APIs as used by the core applications. Besides this the app architecture is so designed so as to make the reuse of components simpler and the capabilities of any application can be published and used by any other app subject to the relevant security constraints. The users can also replace the components using the same mechanism.

The fully integrated Android package comprising an OS, middleware, applications and user friendly interface is expected to considerably speed-up product development while lowering the cost of mobile services development.

Android Application Development

Android is the newest big-ticket entrant in the crowded mobile OS market, developed by Google and 30 other companies under the Open Handset Alliance (OHA). Android's unique feature is that it is an open source platform and Android-based apps can be freely distributed directly to the users instead of through a Google-only route.

Developers looking to develop great apps for the Android OS need to first download the Android SDK from Google's site and get started. Android application development is unique because Android is licensed under the Apache license that requires you to open source your portion of the code even while distributing, sharing or using the software for commercial purposes.

All Android apps are run on the Dalvik virtual machines that are optimised for parallel virtual machines execution on low memory systems. Setting up your Android Development Environment is a fairly straightforward process. First of all download the powerful Android SDK from the Google site. Next, download the free Eclipse IDE required for Java based programming on the Android. Finally, install the Android plugin and you are all set to develop great Android apps and programs.

The Android software application development versatility can be seen from the fact that you can create high quality Android apps using the same tools used for great Java application development. If you need a resourceful library for developing rich mobile applications you need no further than the exhaustive Android core libraries.

What's more the great Android dev tools simplify testing, running and debugging your Android apps a child's play, literally. What you need to know before starting your custom Android Development process is that you need to install the Android Development Tools (ADT) plugin only if you plan to use the Eclipse IDE as your development environment. If not then no need to install the ADT plugin. Simple, isn't it.

Android vs iPhone SDK:

Although this is a pretty difficult comparison to make given the fact that both the SDK's have something good going for them. While the Android SDK is available for use on any machine the iPhone SDK can only be used on Mac OS machines and that too, recent ones. Apple also requires you to sign a 10-page long restrictive agreement on developing and distributing the iPhone apps whereas Google has not put any such restrictions on the use of its Android SDK. Just download it and get started. The Apple iPhone SDK comes with their proprietary Xcode, Interface Builder and Instruments tools. The flip side is that if you aren't familiar with it you have to undergo a fairly steep learning curve. There are no such hassles with the Android SDK as the standard development environment used is the Eclipse IDE, which is kind of the industry standard for any platform except the Mac OS X and Windows. Also the Android SDK is more developer friendly and open source as compared to any other mobile OS development platform out there.