CORE ANDROID & SOFTWARE ENGINEERING QUESTIONS

This document is strictly private, confidential and personal to its recipients and should not be copied, distributed or reproduced in whole or in part, nor passed to anyone.

Instructions

This is the hiring assignment for the Software Engineer Team at Neurafarm. It consists of 23 essay questions related to Android and Software Engineering. You are permitted to search the web for information. Fill in the available boxes and answer the questions carefully. Answer as much as you can, do not hesitate to include images/GIF if you feel it's important to explain your answer.

After you finish, compress the answer to PDF format.

Note: [AD] means it is an Android related question, and **[SE]** is a code for Software Engineering question.

No.	Question
1.	**[AD] Which application from your portfolio are you most proud of? Describe it.**
	Answer: Notes App. The reason is because this project is a personal project based on my desire to learn android. In addition, this application can also be used for me, my family, and friends. In contrast to my other projects based on organizational needs.
2.	**[AD] Mention the Android frameworks you have used most.**
	Answer: - I haven't use any Android frameworks because I just learned the development of Android recently
3.	**[AD] What do you think about current Dr. Tania application? How could you improve it?**

	Answer: Dr. Tania is an application to diagnose plant diseases through photographs. This application uses Artificial Intelligence technology (artificial intelligence) so that it can identify accurately and quickly. How to improve it ? Besides being available in English, this application can also be available in Bahasa. The reason is because Indonesia still has many farmers and this will be the target user of this application. I hope this could facilitate Indonesian farmers in using this application.
4.	**[AD] Describe MVC, MVP, and MVVM and what is the architecture of your last app?**
	Answer: **The Model-View-Controller (MVC)** is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects. A **minimum viable product (MVP)** is a development technique in which a new product or website is developed with sufficient features to satisfy early adopters. The final, complete set of features is only designed and developed after considering feedback from the product’s initial users. **Model-View-ViewModel (MVVM)** is a structural design pattern that separates object into three distinct groups : - **Model** hold application data. They’re usually structs or simple classes. - **View** display visual elements and controls on the screen. They’re typically subclasses of UIView. - **View models** transform model information into values that can be displayed on a view. They’re usually classes, so they can be passed around **My last app architecture** : MVVM
5.	**[AD] Tell all the Android application components.**
	Answer: **Activities** : An activity is a class that is considered as an entry point for users that represents a single screen. **Services** : a component that runs in the background, it acts as an invisible worker of our application. It keeps updating data sources and activities. It also broadcasts intents and performs tasks when applications are not active.

	Content Provider : Content Provider is a component that allows applications to share data among multiple applications. It hides the details of the database and can be used to read and write private data of the application which is not shared. **Broadcast Receiver** : Broadcast Receiver is a component that responds to broadcast messages from another application or the same system. It can also deliver broadcasts to applications that are not running. Additional Component : - Intent - Widgets - Views - Notifications - Fragments - Layout XML Files - App APK Files - Resources
6.	**[AD] What is `Context`? How is it used?**
	Answer: A Context is a handle to the system; it provides services like resolving resources, obtaining access to databases and preferences, and so on. An Android app has activities. Context is like a handle to the environment your application is currently running in. The activity object inherits the Context object. How is it used ? by invoking getApplicationContext(), getContext(), getBaseContext() or this (when in a class that extends from Context, such as the Application, Activity, Service and IntentService classes).
7.	**[AD] What is the difference between a `Fragment` and an `Activity`? Explain the relationship between the two.**
	Answer: **The difference** : 1. Activity is an application component that gives a user interface where the user can interact. The fragment is a part of an activity, which contributes its own UI to that activity. 2. For Tablet or if mobile is in landscape then Using fragment we can show two lists like the only list for show the state name and other lists will show the state description in single activity but using Activity we can't do the same thing. 3. Activity is not dependent on fragment. But The fragment is dependent on Activity, it can't exist independently. 4. without using fragment in Activity we can't create multi-pane UI. but using multiple fragments in a single activity we can create multi-pane

	UI. 5. If we create a project using only Activity then its difficult to manage but if we use fragments then the project structure will be good and we can handle it easily. 6. An activity may contain 0 or multiple numbers of fragments. A fragment can be reused in multiple activities, so it acts like a reusable component in activities. 7. The activity has own life cycle but fragment has there own life cycle. 8. For Activity, we just need to mention in Manifest but for fragment its not required. 9. Activity a lot of memory used and the fragment is non-memory used. 10. Activity is not lite weight. The fragment is the lite weight. **Relationship** : Activity and fragments operate under the one-to-many relationship. That is, there can be one activity and host many fragments on top of it. The opposite case does not hold true. In fact, a fragment can't exist without an activity to be the host. A fragment must always be hosted in an activity and a fragment's lifecycle is directly affected by the host activity's lifecycle. For example, when an activity resumes so will all of its fragments and when the activity pauses so will all of the fragments. However, while an activity is running (resumed lifecycle state), you can freely manipulate the state of a fragment, such as add or remove them. Another way to look at it is that when a fragment changes state it doesn't affect the activity, but when the activity changes state it does affect the fragment.
8.	**[AD] How would you communicate between two Fragments?**
	Answer: Define an interface in first fragment, for example Fragment A, let Activity implement that Interface. Now call the interface method in the Fragment, and the Activity will receive the event. Now in the activity, we can call our second Fragment to update with the received value
9.	**[AD] What are ViewGroups and how they are different from the Views?**
	Answer: **What are ViewGroups** : A ViewGroup is a special view that can contain other views. The ViewGroup is the base class for Layouts in android, like LinearLayout, RelativeLayout, FrameLayout etc. In other words, ViewGroup is generally used to define the layout in which views(widgets) will be set/arranged/listed on the android screen. **How they are different from the Views?**

`View` is a basic building block of `UI` (User Interface) in android. A view is a small rectangular box which responds to user inputs. Eg: `EditText`, `Button`, `CheckBox`, etc. While `ViewGroup` is a invisible container of other views (child views) and other viewgroups. Eg: `LinearLayout` is a viewgroup which can contain other views in it.

10. **[AD] Under what condition could the code sample below crash your application? How would you modify the code to avoid this potential problem? Explain your answer.**

```
Intent sendIntent = new Intent();
sendIntent.setAction(Intent.ACTION_SEND);
sendIntent.putExtra(Intent.EXTRA_TEXT, textMessage);
sendIntent.setType(HTTP.PLAIN_TEXT_TYPE); // "text/plain" MIME type
startActivity(sendIntent);
```

Answer:
It will be crash if there are no application that can handle our intent. To avoid this, we should first verify that there is at least one application registered in the system that can handle the intent before calling startActivity(). To do this use resolveActivity() :

```
if (sendIntent.resolveActivity(getPackageManager()) != null) {
    startActivity(sendIntent);
  }
```

11. **[AD] What is the relationship between the life cycle of an AsyncTask and an Activity? What problems can this result in? How can these problems be avoided?**

Answer:
An AsyncTask is not tied to the life cycle of the Activity that contains it. So, for example, if you start an AsyncTask inside an Activity and the user rotates the device, the Activity will be destroyed (and a new Activity instance will be created) but the AsyncTask will not die but instead goes on living until it completes. Then, when the AsyncTask does complete, rather than updating the UI of the new Activity, it updates the former instance of the Activity (i.e., the one in which it was created but that is not displayed anymore!). This can lead to an Exception (of the type java.lang.IllegalArgumentException: View not attached to window manager if you use, for instance, findViewById to retrieve a view inside the Activity).

How can these problems be avoided ?
By not using AsyncTasks for long-running background tasks. Rather, for long-running background tasks, a different mechanism (such as a service) should be employed.

12. **[AD] What is an intent? What is the difference between implicit vs explicit vs sticky intent? Can intent be used to provide data to a ContentProvider? Why or why not?**

Answer:

	Intents are messages that can be used to pass information to the various components of android. For instance, launch an activity, open a webview etc. **Difference between implicit vs explicit vs sticky intent?** - Implicit: Implicit intent is when you call system default intent like send email, send SMS, dial number. - Explicit: Explicit intent is when you call an application activity from another activity of the same application. - Sticky intent : allows communication between a function and a service. sendStickyBroadcast() performs a sendBroadcast(Intent) known as 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). **Can intent be used to provide data to a ContentProvider? Why or why not?** We cannot start a ContentProvider using an Intent. When you want to access data in a ContentProvider, you must instead use the ContentResolver object in your application’s Context to communicate with the provider as a client. The ContentResolver object communicates with the provider object, an instance of a class that implements ContentProvider. The provider object receives data requests from clients, performs the requested action, and returns the results.
13.	**[AD] How would you perform a long-running operation in an application?**
	Answer: By using android services. An Android service is defined as an application component that is generally used to perform long tasks in the background.
14.	**[AD] How to persist data in an Android app?**
	Answer: We can use SharedPreferences to store our data, or SQLite, or even Firebase.
15.	**[AD] How do you find memory leaks in Android applications?**
	Answer: By using android profiler. Android Profiler help us to identify memory leaks and memory churn that can lead to stutter, freezes, and even app crashes. It shows a realtime graph of your app's memory use and lets you capture a heap dump, force garbage collections, and track memory allocations.
16.	**[AD] How to reduce battery usage in an android application?**

	Answer: By not running sevices all the time.
17.	**[AD] What is Espresso? What is the difference between Roboelectric and UI-Automator?**
	Answer: *Espresso* is a testing framework for Android to make it easy to write reliable user interface tests. Espresso automatically synchronizes our test actions with the user interface of your application. The framework also ensures that our activity is started before the tests run. It also let the test wait until all observed background activities have finished. The difference between Roboelectric and UI-Automator is Roboelectric validate how our code interacts with other parts of the system but without the added complexity of a UI framework. Roboelectric is a tools for integration test, while UI-Automator is a tools for UI test.
18.	**[AD] Is it possible to create an activity in Android without a user interface? Explain.**
	Answer: Yes, Abstract activities is an activities that can be created without any user interface.
19.	**[SE] Have you done unit testing or automated testing? Explain your experience.**
	Answer: Yes, I've done unit testing. I use Junit for my java project. It's faster and more reliable than manual testing.
20.	**[SE] Describe how REST APIs work.**
	Answer: REST is a way for two computer systems to communicate over HTTP in a similar way to web browsers and servers. REST utilizes four methods in HTTP 1. GET - requests a representation of the specified resource. 2. POST - used to submit an entity to the specified resource. 3. PUT - replaces all current representations of the target resource with the request payload. 4. DELETE - deletes the specified resource.
21.	**[SE] What is MongoDB? How is it different with MySQL? Tell us what**

	makes MongoDB best? Explain.
	Answer: MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schema. There is a difference in the representation of data in the two databases. In MongoDB, data represents in a collection of JSON documents while in MySQL, data is in tables and rows. One of the best things about MongoDB is that you are not responsible for defining the schema. MongoDB's performance is better than that of MySQL and other relational DBs. This is because MongoDB sacrifices JOINS and other things and has excellent performance analysis tools.
22.	**[SE] What is Gitflow Workflow? Explain.**
	Answer: The Gitflow workflow is a strict branching model that provides a robust framework for managing large projects and teams. This workflow assigns specific responsibilities to named branches in the repository and defines how and when they should interact.
23.	**[SE] How would you design a chess game? What classes and objects would you use? What methods would they have? Write the classes in Node.js or Python.**
	Answer: