

HW 1

1. Question 1

- a. Explain the need for a compatibility definition document (CDD) for devices supporting the Android framework.
    - ➔ Android framework runs on various hardware and devices. Manufacturers building devices for android must ensure that all requirements are met in order for android framework and applications run correctly. CDD document is helpful in the sense that it prevents possible ambiguity. For instance, as documented in section 2.1 of CDD, android handheld device must include touchscreen and microphone components. If a device does not meet those requirements, then it cannot say that it supports android framework. If android framework is installed on device that does not support android framework, android applications that running on the device will have unexpected behavior. It's obvious that most, if not all, of android applications requires touch inputs. If touchscreen component is not included or not configured according to CDD, it would be difficult for any user to interact with android apps.
  - b. Identify the major device types described in the CDD. What are the MUST have features for each device type (at least 2 features)?
    - ➔ These requirements are taken from CDD section 2.1 "Device Configurations".
      1. Handheld devices:  
MUST: Touchscreen, Microphone, Speaker and/or Audio output ports.
      2. Television devices:  
MUST: D-pad, Wi-Fi, Bluetooth, Bluetooth Low Energy, Speaker and/or Audio output ports.
      3. Watch:  
MUST: Touchscreen, Microphone, Bluetooth.
      4. Automotive devices:  
MUST: Microphone, Bluetooth, Speaker and/or Audio output ports.
      5. Other devices:  
MUST: Speaker and/or Audio output ports.
  - c. Activity component can use Android's Intent mechanism to request the framework to find other components which may offer services or functionality needed by the current activity. What constraint(s) does the CDD place on device manufacturers for handling Intents? How does the constraint requirement fulfil Android's design philosophy?
    - ➔ As noted in CDD section 3.2.3, device manufacturers must provide "an Android activity or service that specifies a matching intent filter that binds to and implements correct behavior for each specified intent pattern". Android's defined intent pattern are implemented by "core applications". Thus, if a third party application wanted to open calendar, or adjust brightness in device settings, android framework can successfully find an activity that can handle that intent.
2. An Android application is a loose collection of one or more of the following components – Activity, Content provider, Broadcast receiver, Service. Components of the same or other application can be invoked using Intents. Activity is one of the most important component of an application since it presents a conduit to interact with Android framework, Widgets and other app components.
- We discussed some reasons for the component based architecture of Android apps in class. Justify the component based architecture from a user interaction view point. Why does such an architecture for apps is important?

➔ Every component is triggered by intents. Component architecture allows for components from multiple applications to bind dynamically. From a user interaction point of view, it reduces number of actions to perform in order to complete a task. For instance, say a user received an email and email contains some text, images and links to a website. If user clicks on the website link, user expects the website to open in a browser. Component architecture allows developers to make an intent with some additional parameters such that it triggers appropriate activity and the website appears in browser. This prevents user from having to open up a browser, copy the URL, paste it in a new tab of the browser. Component architecture allows application developer to simplify the whole process.

3. How does Android framework maintain isolation of apps? What feature of Linux kernel is utilized to achieve isolation?

➔ Each application is started in a new Linux process with a unique process ID. The application will then execute in security sandbox. Every application is hosted in a forked virtual machine providing the isolation with different applications.

4. Briefly explain the function(s) of each of the four app components – Activity, Content provider, Broadcast receiver, Service

➔ Activity: This component is mainly focused on UI. Layout and different UI components are defined in Activities. Users are able to directly interact with activity to provide input.

Services: Services do not contain any UI in their implementation. They are mainly used to process data. Such functions include update server data, push notification on user device, and send intent for appropriate activities. They often run in the background.

Content provider: Content provider is a mechanism provided by android framework to manage and share data. Android manages how data is shared between applications. Content providers also handle security of the shared data, in that for each request, it checks whether application requesting data has permission to view it.

Broadcast receiver: These are “system wide intents”. They are expected to perform their task quickly and return. It is basically an event handler that is registered with android system. When the event occurs, all registered receivers get notified in normal or ordered fashion.

5. What is the purpose of empty process in the application life cycle? How does an empty process help with mobile application performance?

➔ Empty process is used in the memory management aspect of mobile application. Android framework keeps an empty process running in the background. If user wants to open an application but memory usage at the time is very high, android framework would kill the empty process to gain memory and perform task requested by user. It helps keep the foreground process smooth and responsive.