\*\*\*\*\*\*\*\*\*\*\*\*\*\* Homework Week 1 Day 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Answer the following questions:

**1.** What are the major components in android?

They are four main components in Android: 1- Activities, 2- Services, 3- Broadcast Receiver, 4- Content Provider.

**2.** What are all the activity lifecycles and explain what each details?

1- onCreate(): perform basic application startup logic that should happen only once for the entire life of the activity.

2- onStart(): call makes the activity visible to the user, as the app prepares for the activity to enter the foreground and become interactive.

3- onResume(): This is where the lifecycle components can enable any functionality that needs to run while the component is visible and in the foreground, such as starting a camera preview.

4- onPause(): The system calls this method as the first indication that the user is leaving your activity (though it does not always mean the activity is being destroyed); it indicates that the activity is no longer in the foreground.

5- onStop(): When your activity is no longer visible to the user, it has entered the Stopped state, and the system invokes the onStop() callback. When the activity moves to the stopped state, any lifecycle-aware component tied to the activity's lifecycle will receive the ON\_STOP event.

6- onDestroy(): is called before the activity is destroyed. The system invokes this callback either because:

a- the activity is finishing (due to the user completely dismissing the activity or due to finish() being called on the activity), or

b- the system is temporarily destroying the activity due to a configuration change (such as device rotation or multi-window mode).

3. Explain each step of the sprint in agile.

The Manifesto for Agile Software Development is based on twelve principles:

1- Customer satisfaction by early and continuous delivery of valuable software.

2- Welcome changing requirements, even in late development.

3- Deliver working software frequently (weeks rather than months)

4- Close, daily cooperation between business people and developers

5- Projects are built around motivated individuals, who should be trusted

6- Face-to-face conversation is the best form of communication (co-location)

7- Working software is the primary measure of progress

8- Sustainable development, able to maintain a constant pace

9- Continuous attention to technical excellence and good design

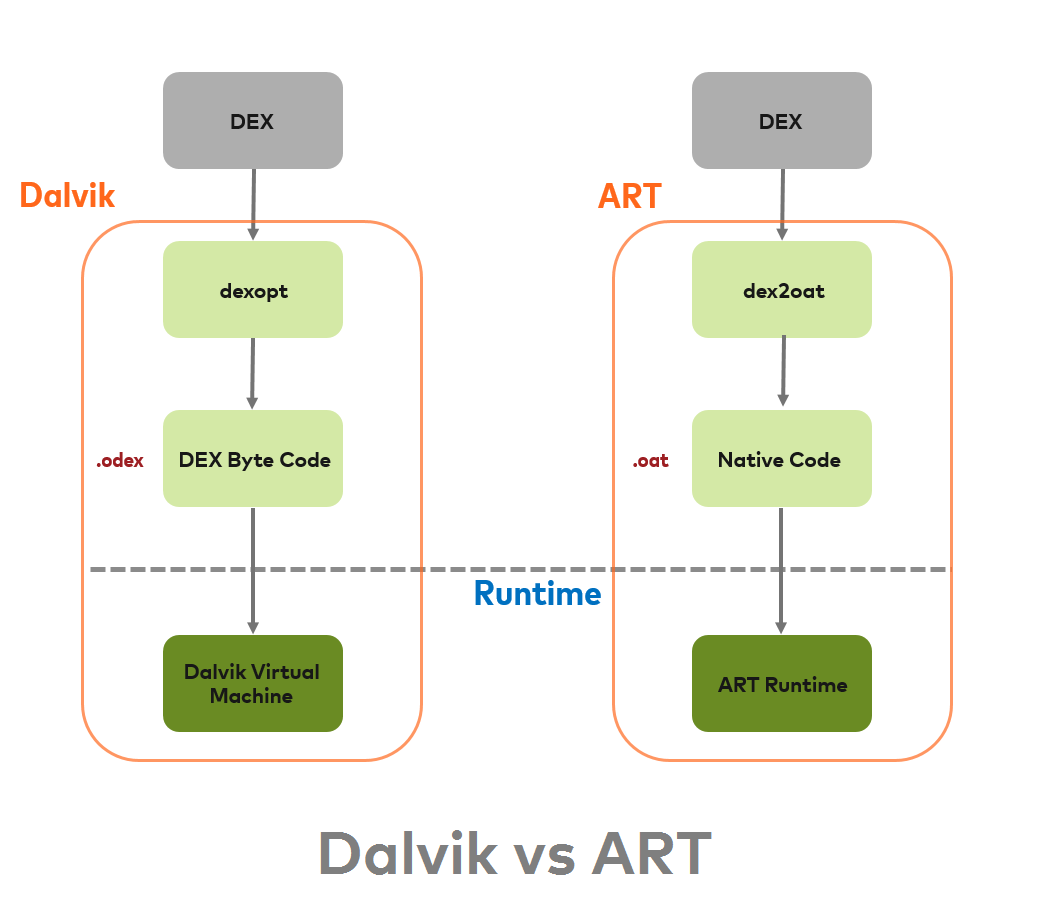
10- Simplicity—the art of maximizing the amount of work not done—is essential

11- Best architectures, requirements, and designs emerge from self-organizing teams

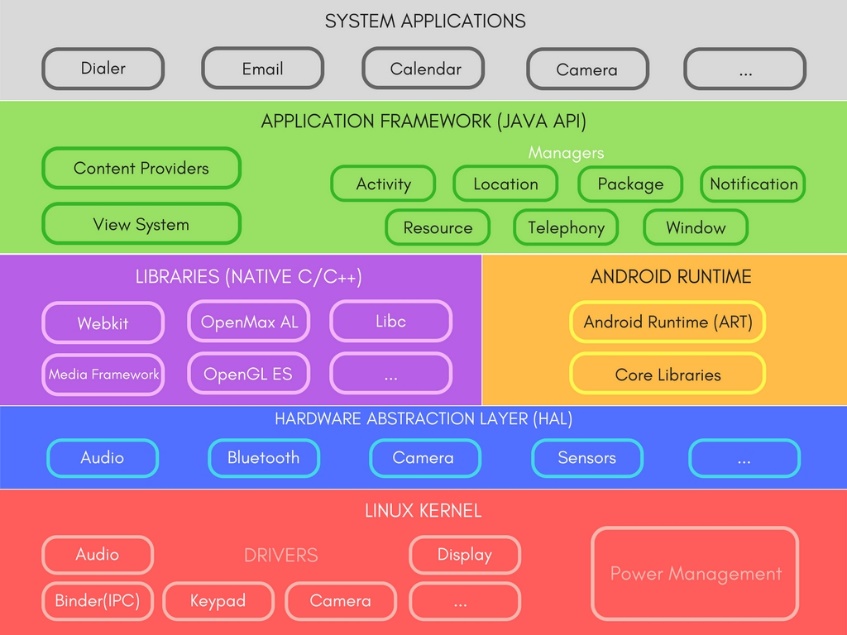
12- Regularly, the team reflects on how to become more effective, and adjusts accordingly

4. What is dalvik and ART?

Dalvik is a discontinued process virtual machine (VM) in Google's Android operating system that executes applications written for Android.

The Agile Release Train is a long lived, self-organizing team of Agile Teams, a virtual organization (5 to 12 teams) that plans, commits, and executes together. ARTs are organized around the enterprise’s significant Value Streams and live solely to realize the promise of that value by building solutions that deliver benefit to the end user.

5. How is the android platform architecture designed?

1- Linux Kernel: This layer is the foundation of the Android Platform.

Contains all low level drivers for various hardware components support.

Android Runtime relies on Linux Kernel for core system services like,

a- Memory, process management, threading etc.

b- Network stack

c- Driver model

d- Security and more.

2- Hardware Abstraction Layer (HAL): Provides Abstraction between hardware and rest of the software stack.

3- Android Runtime (ART): Designed to run apps in a constrained environment that has limited muscle power in terms of battery, processing and memory.

4- Libraries: Exposed to developers through Android Application Framework.

Contains C/C++ libraries used by components of Android Systems.

5- Application Framework: It is a collection of APIs written in Java, which gives developers access to the complete feature set of Android OS.

Developers have full access to the same framework APIs used by the core applications, so that they can enhance more in terms of functionalities of their application.

Enables and simplify the reuse of core components and services

5- Applications: Top of the Android Application Stack, is occupied by the System apps and tonnes of other Apps that users can download from Android's Official Play Store, also known as Google Play Store.

A set of Core applications are pre-packed in the handset like Email Client, SMS Program, Calendar, Maps, Browser, Contacts and few more.

This layer uses all the layers below it for proper functioning of these mobile apps.