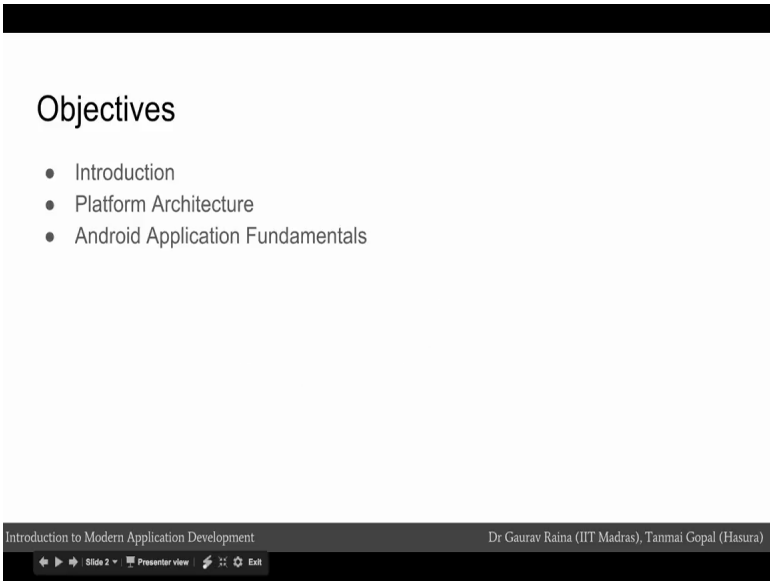
Introduction to Morden Application Development

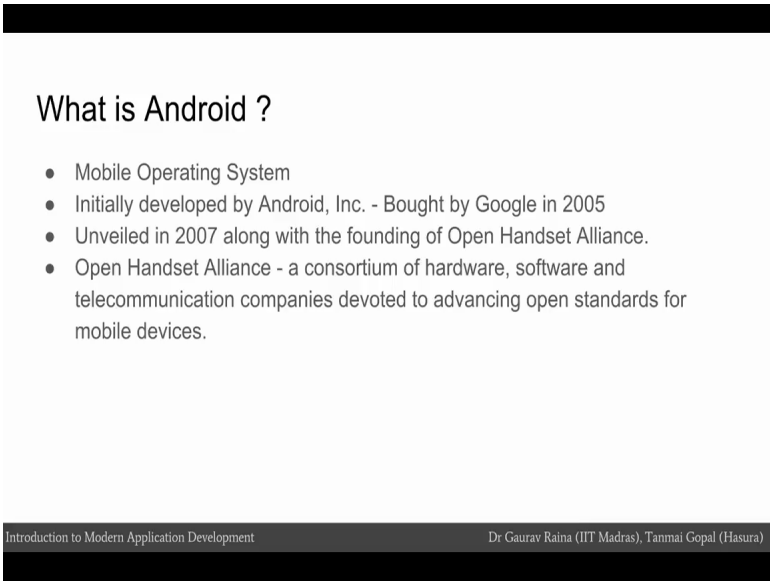
Dr. Gaurav Raina
Prof. Tanmai Gopal
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - 17
Lecture - 29
Introduction To Android

(Refer Slide Time: 00:02)

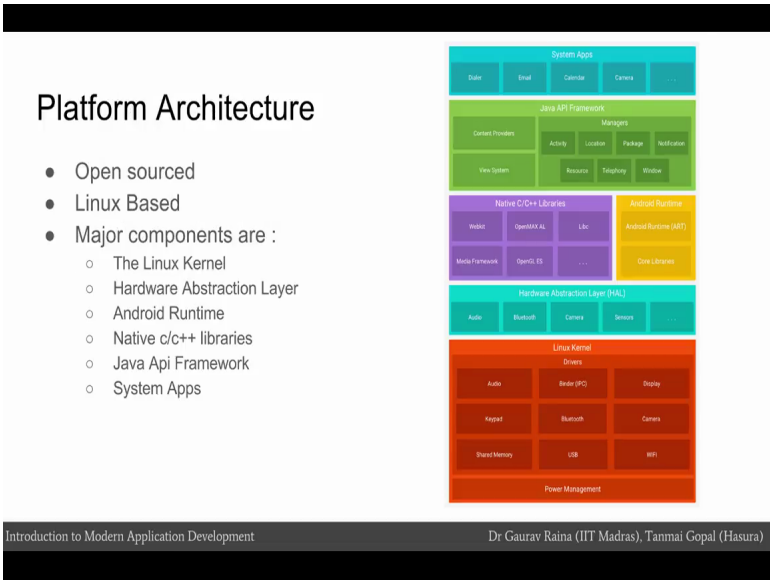


(Refer Slide Time: 00:07)



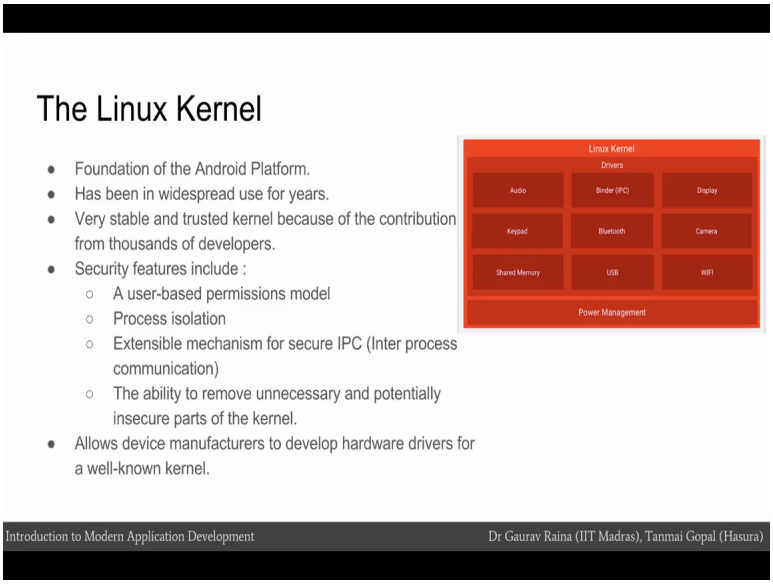
In this module we are going to get into the brief details of the Android eco system. What is Android? Android is a mobile operating system and was initially developed by Android Inc. In the august of 2005 Google bought Android.

(Refer Slide Time: 00:34)



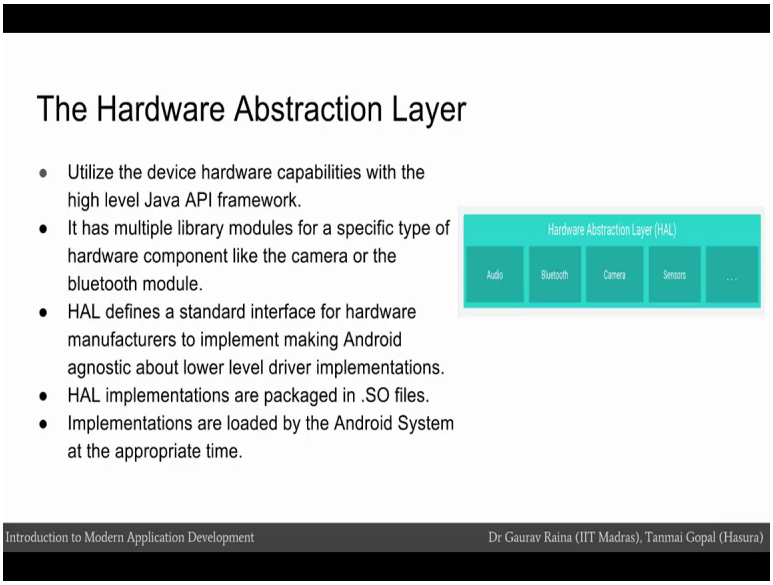
And in 2007, the open handset alliance was formed and Android was unveiled to the public the open handset alliance is an alliance as mobile and technology companies is developed Android. It is a consortium of 84 firms to develop open standards for mob mobiles devices.

(Refer Slide Time: 00:52)



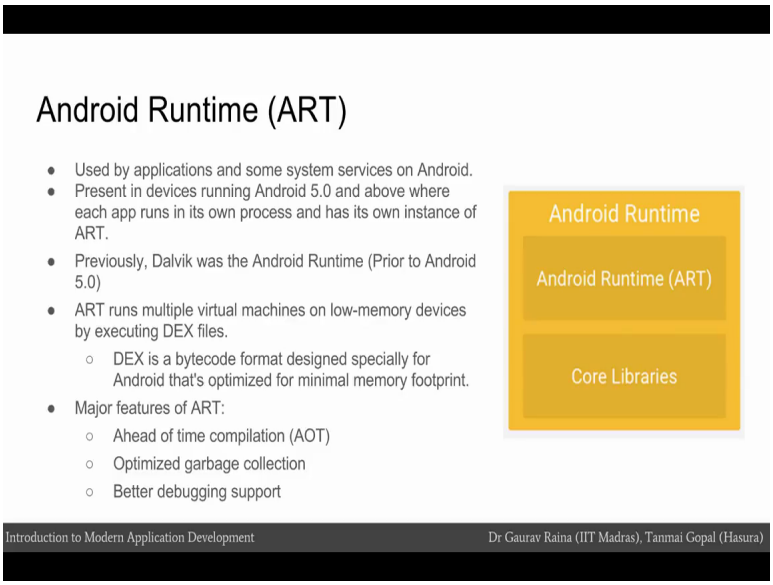
Let us now discuss the platform architecture that Android follows. Android is an open source Linux based software stark the diagram on the right shows the major components of the Android platform the Linux Kernel is a foundation of the Android platform. Linux has been in widespread use for years and because of the contribution from thousands of developers it has now become a very stable and secure Kernel. The Linux Kernel provides Android with several security features like a user based permission model, process isolation extensible mechanism for secure IPC, it stands for inter process communication and the ability to remove unnecessary and potentially insecure parts of the Kernel using Linux also allows device manufacturers to develop hardware drivers for a well known Kernel.

(Refer Slide Time: 01:32)



Next comes the hardware abstraction layer HAL. Let us utilize the device hardware capabilities it has multiple library modules for a specific hardware components you would you would come across camera module or Bluetooth module this is the thing that help us interact with the mobile device. HAL defines a standard interface for hardware vendors to implement by doing this Android becomes agnostic about lower level driver implementations. HAL implementations are packaged into dot SO files and loaded by the Android system at the appropriate time.

(Refer Slide Time: 02:13)

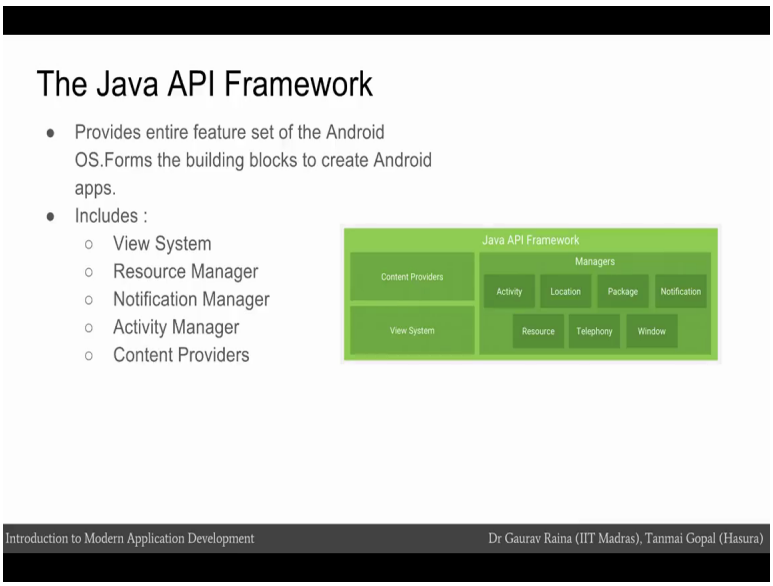


Next comes the Android runtime also known as the ART which is a managed runtime and on Android it is used by applications and some systems services. For devices running Android versions 5.0 are higher each app runs in its own process and with its own instance of Android runtime, prior to the Android version 5.0 Dalvik was the Android runtime. So, if your app runs well on ART then it should work on Dalvik as well, but the reverse may not be true ART is written to run multiple virtual machines on low memory devices by executing Dex files which is a byte code format designed specifically for Android and which is optimized for minimal memory footprint. ART also has some major features like ahead of time compilation, optimized garbage collection, better debugging support it provides better debugging support by including a dedicated sampling profiler, detailed diagnostic exceptions and crash reporting or it also provides the ability to set specific watch points to monitor specific fields.

(Refer Slide Time: 03:23)

Another important part of the Android platform are the native C and C plus libraries many core Android system components and services such as the ART and HAL are built from native code that require native libraries written in C and C++. The Java framework APIs provide a wrapper over these native liabilities to be used while developing Android apps, if you are developing an app that requires C or C++ code you can use the Android native development kit also known as the NDK to access some of these native platform liabilities directly from a Java code. Next comes the Java API framework.

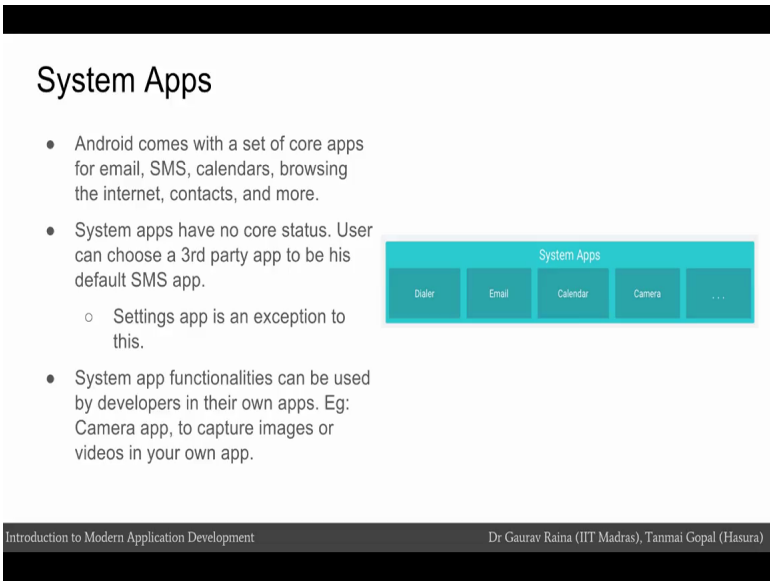
(Refer Slide Time: 04:02)



The Java APIs provide you with the entire set of the Android OS these APIs from the building blocks will need to create Android apps by simplifying the reuse of code modulus system components and services. The Java APIs includes a view system which can be used to build the UI these include list grids text boxes buttons etcetera.

It also includes a resource manager which provides access to noncore resources such as low class frames graphics and layout files we will see this more in action when we develop these apps. It also provides the notification manager that enables all apps to display custom alerts in the status bar these enable notification that you receive on your mobile phone from different apps and activity manager that manages the life cycle of the apps and provides the navigation back stack. This is more in the lines of when you navigate 2 different activities or different views on a mobile app and when you click the hardware or the back button provided by app you pop back to the previous page this is also handled by the Java API framework; content providers that enable apps to access data from other apps such as contacts for to share their own data.

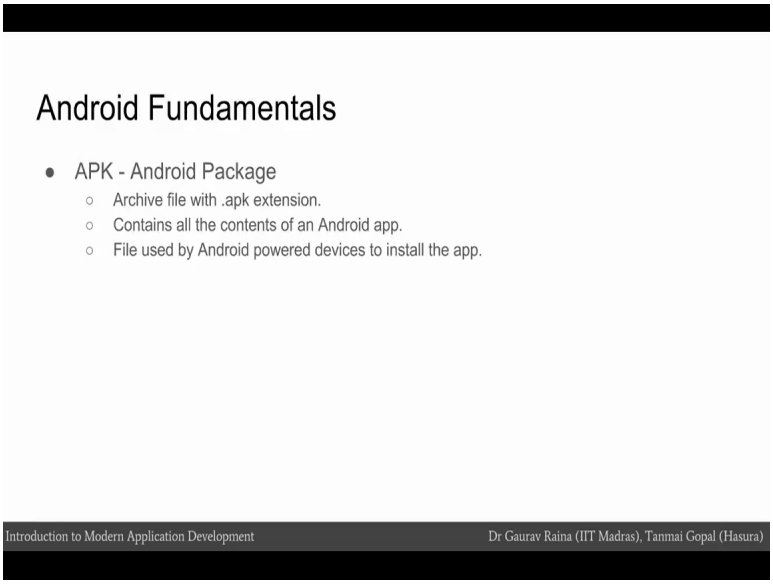
(Refer Slide Time: 05:11)



What are system apps? System apps are basically those apps come preinstalled on your phone when you just buy them these are apps for functionalities like email, SMS, phone call, camera etcetera system apps have no core status this means that the user can choose any third party app to replace the functionality of a system app. For example, you can have a third party default SMS messenger app the settings app is an exceptional case

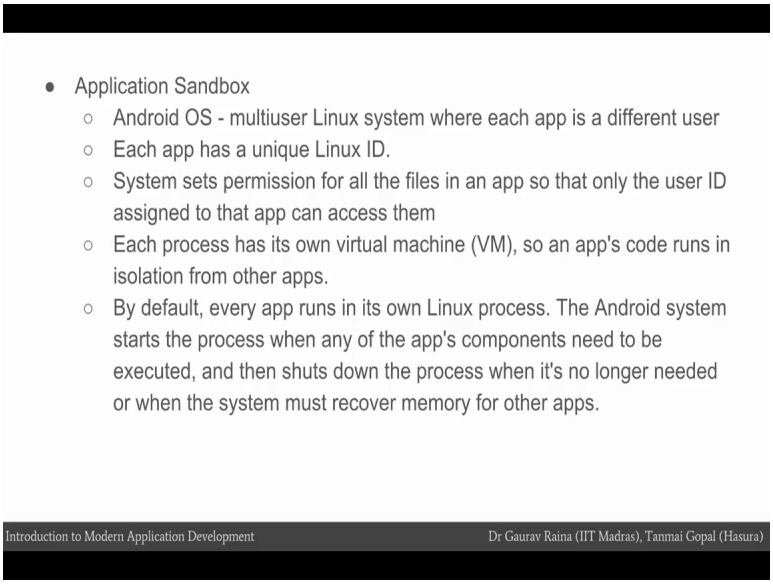
system app functionalities can be used by developers in their own apps this means that if you are developing an app that involves a feature of image capture or video capture you can leverage the functionality of the camera app.

(Refer Slide Time: 05:53)



Let us talk Android fundamentals next the Android SDK tools compile your code with any data and resource files into an APK which is an Android package. An APK is an archive file with an dot APK suffix one APK file contains all the contents of an Android app and it is the file that Android powered devices use to install the app.

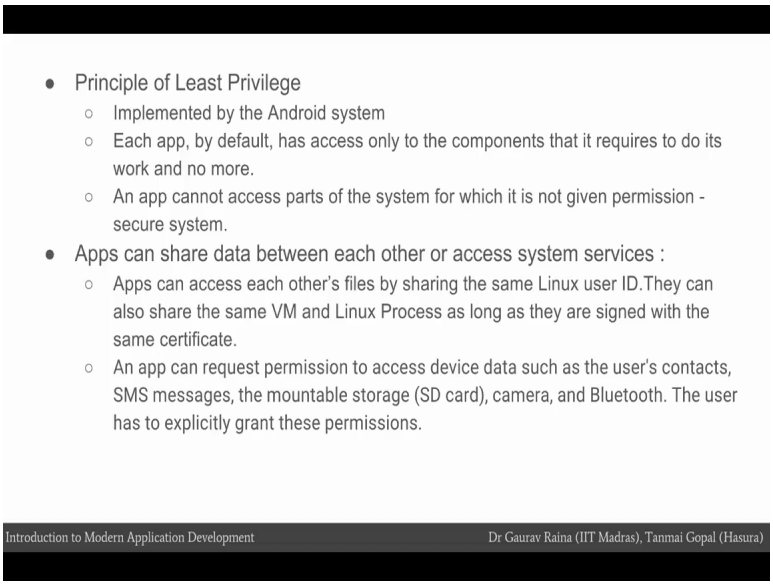
(Refer Slide Time: 06:16)



You must have come across the term application sandbox this is very common term among mobile apps what it means is that every Android app lives in its own security sandbox and every app runs in isolation from the device and the apps and the device. Each app is given permission to access only the files that it needs and nothing else. This happens due to key Android security features like the fact that the Android operating system is a multiuser Linux system in which each app is a different user. By default this system assigns every app a unique Linux user id and this is only remember the system and unknown to the app the system then sets permissions for all the files in an app. So, that only the user id assigned to that app can assign to that app can access these files.

Furthermore each access has its own virtual machine. So, an apps code runs in complete isolation from other apps, by default every app runs in its own Linux process the Android system starts the process when any of the apps components are needed and then shuts it down when it is no longer needed or when it has to clear some memory or recover some memory for some other apps.

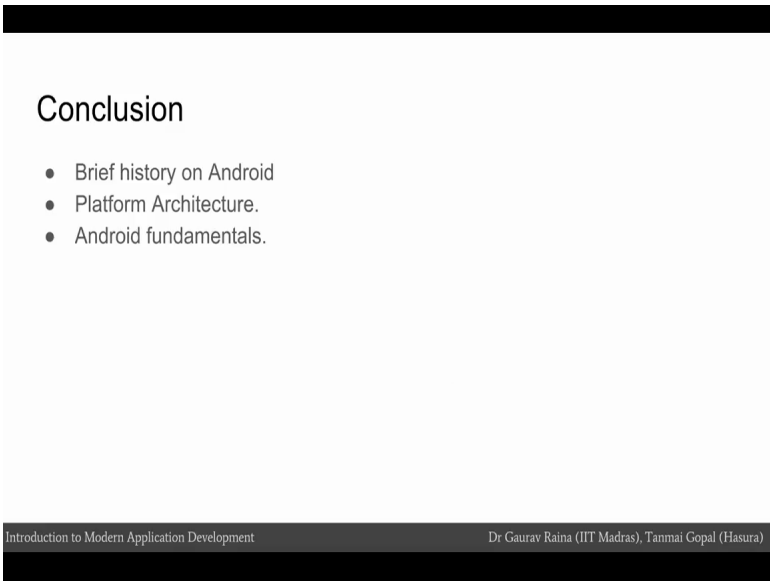
(Refer Slide Time: 07:28)



The Android system implements the principle of least privilege this is part of the application sandbox feature wherein every app can only access the component it requires this ensures security because no app can have unauthorized access to any parts of the system. In case different apps wants want to share data they can share the same Linux user id in which case they can share the other files apps with the same Linux use the id

that are assigned to the same certificate can also run the same Linux process and share the same VM, an app can also request permission to access device data such as the users contact SMSs, images, gallery images etcetera the user has to explicitly grant these permissions.

(Refer Slide Time: 08:14)



With this we come to the end of this module we have got a brief understanding of the Android platform and how it has come about, and we have also spoken in brief about the Android fundamentals. We have spoken about things like application sandbox, the APK file, the principle of least privilege and how different apps can share files and data among each other. In the coming module we are going to start building an Android application.