MOBILE APPLICATION DEVELOPMENT LAB

Lab Course Material

Prepared By

K. Ravi Chythanya
Assistant Professor

D. Ramesh
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SR ENGINEERING COLLEGE
Approved by AICTE, New Delhi | Affiliated to JNTUH | Accredited by NAAC 'A' Grade
Ananthasagar (V), Hasanparthy (M), WARANGAL - 506 371
(An Autonomous Institution)

SUMMARY

Course Title	**:** Mobile Application Development
Course Code	**:** CS146
Course Type	**:** Practical
Regulation	**:** RA15
Academic Year	**:** 2019-2020
Year/Semester	**:** IV/I
Course Coordinator	**:** K. Ravi Chythanya Assistant Professor Department of CSE +919000188956 ravi_chythanya@srecwarangal.ac.in
Course Co-Faculty Member	**:** D. Ramesh Assistant Professor Department of CSE +919848142720 ramesh_d@srecwarangal.ac.in
Objective of the Course	**:** Mobile Application Development course is designed to quickly get you up to write applications for Android devices. The student will learn the basics of Android platform, get to understand the application frontiers and able to design his own applications.
Prerequisites	**:** Basics of Java and XML
Learning Resources	**:** https://bit.ly/rcandapp

SYLLABUS

Week -1

1. Java Android Program to Build a Simple Android Application
2. Java Android Program to Demonstrate Activity Life Cycle

Week-2

1. Java Android Program to Change the Background of your Activity
2. Java Android Program to Demonstrate Action Button by Implementing on Click Listener

Week-3

1. Java Android Program to perform all Operations using Calculators
2. Java Android Program to Change the Image Displayed on the Screen

Week-4

1. Java Android Program to Create Multiple Activities within an Application
2. Java Android Program to Demonstrate the Sound Button Application

Week-5

1. Java Android Program to demonstrate the use of Scroll View
2. Java Android Program to Demonstrate Radio Group Application.

Week-6

1. Java Android Program to Demonstrate the Menu Application
2. Java Android Program to Demonstrate List View Activity

Week-7

1. Java Android Program to Demonstrate an Advanced Xml Layout
2. Java Android Program to Demonstrate Layouts in an Activity and Nesting of Layouts

Week-8

1. Java Android Program to Demonstrate Motion Event on Android screen with the help of an image
2. Java Android Program to Demonstrate a Simple Video View

Week-9

1. Java Android Program to Demonstrate Explicit Intent
2. Java Android Program to Demonstrate Implicit Intent

Week-10

1. Java Android Program to Demonstrate Broadcast Receiver
2. Java Android Program to Demonstrate Broadcast Receiver to Intercept Custom intent.

Week-11

1. Java Android Program to Demonstrate Reading and Writing data to the Internal Memory
2. Java Android Program to Demonstrate Reading and Writing data to the External Memory

Week-12

1. Java Android Program to Read and Write to a SQLite Database in Android
2. Java Android Program to Read Write and Delete to a SQLite Database in Android

Week-13

1. Java Android Program to register for the application using SQLite Database.
2. Java Android Program to Login to the application using SQLite Database.

Week-14

1. Java Android Program to register for the application using MySQL Database.
2. Java Android Program to login for the application using MySQL Database.

ASSESSMENT

Internal Assessment:

Total Marks: 30 Marks

Internal Exam : 10 Marks (Twice per semester)
Record : 3 Marks (Every Week)
Observation : 3 Marks (Every Week)
Activity Assessment : 14 Marks (Every Week)

External Assessment:

Total Marks: 70 Marks

External Exam : 35 Marks
Course Project : 35 Marks

Course Project:

Week-1	Identify problem statement and design a logo	5 Marks
Week-2	Finding Requirements and Designing Complete App (prototype)	10 Marks
Week-3, 4, 5and 6	Implementing	15 Marks
Week 7	Testing and Deploying	5 Marks

Technology Requirements:

- Laptops with minimum 4GB RAM and 160GB HDD
- Laptops should be installed with Android Studio Latest Version along with Latest Android SDK
- Smart Phones with Android Operating System
- USB Cable.

Course Outcomes (COs):

At the end of the course the student should be able to:

1. Apply OOPC to develop Mobile Applications.
2. Apply Layout Management and Multi layout definition techniques to create adaptable User Interface
3. Develop user interface for mobile Application using widgets with event handling.
4. Design push notifications for incoming messages
5. Deploy applications to the Android marketplace for distribution.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs)

Course Outcomes (COs) / Program Outcomes (POs)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**
Apply OOPC to develop Mobile Applications.	3		2		3									
Apply Layout Management and Multi layout definition techniques to create adaptable User Interface				3		3								
Develop user interface for mobile Application using widgets with event handling.		3		3				3					1	
Design push notifications for incoming messages	2					2			3		3			2
Deploy applications to the Android marketplace for distribution.	3		3				2			1		1		

Lecture Schedule:

S. No	Topic of the Lecture	Name of the Activity & Instructional Aids	Tentative Date
	MODULE I(Simple Android app)		
1	Recall OOPC		
2	Introduction of Android Studio		
3	Developing Simple Application		
	MODULE II (VIEWS)		
1	Event Handling	Develop a calculator	
2	Introduction of Views	Design login page	
3	Developing UI Using views	Implement a Counter	
4	Creating Menu, Scroll View, Image View	Develop Hotel Menu	
5	Creating View Group		
	MODULE III (ACTIVITIES)		
1	Introduction of Activity		
2	Creating Multiple Activities		
3	Developing Activity life cycle		
	MODULE IV (Layouts)		
1	Introduction about layouts		
2	Creating App on Relative layout	Design a Registration page	
3	Develop an app on Constraint Layout		
	MODULE V (Services, Broadcast Receivers, Intents)		
1	Demonstrate explicit Intent /Filter	Develop an app to share data from one activity to another activity	
2	Demonstrate implicit Intent		
3	Demonstrate Service for Alarm Manager		
4	Demonstrate Broadcast Receiver		
	MODULE VI (Data Base Connection)		
1	Demonstrate Reading and writing to a file in internal storage		
2	Demonstrate Reading and writing to a file in external storage		
3	Write SQLite database in android	Develop login page and store the information in SQLite database	
4	Read and write to a SQLite database in android		
	MODULE VII (MySQL Data Base Connection)		
1	Demonstrate AsyncTask Class	Develop login page and store the information in MySQL database	
2	Create MySQL database and tables.		
3	Read and write to a MySQL database in android		

Activities: (Group Problem Solving)

1. Develop an application to display message (size, font, color, etc)
2. Develop a counter App
3. Develop an application to change background color and image
4. Develop an application to add 5 images and change the image on the button click
5. Design Canteen menu (list view, menu)
6. Design Registration page
7. Develop an application with multiple Activities
8. Develop an app to share data from one activity to another activity
9. Develop an application with login and registration page connecting to database

ANDROID OVERVIEW

What is Android?

Android is an open source and Linux-based **Operating System** for mobile devices such as smartphones. Android was developed by the *Open Handset Alliance*, led by Google, and other companies.

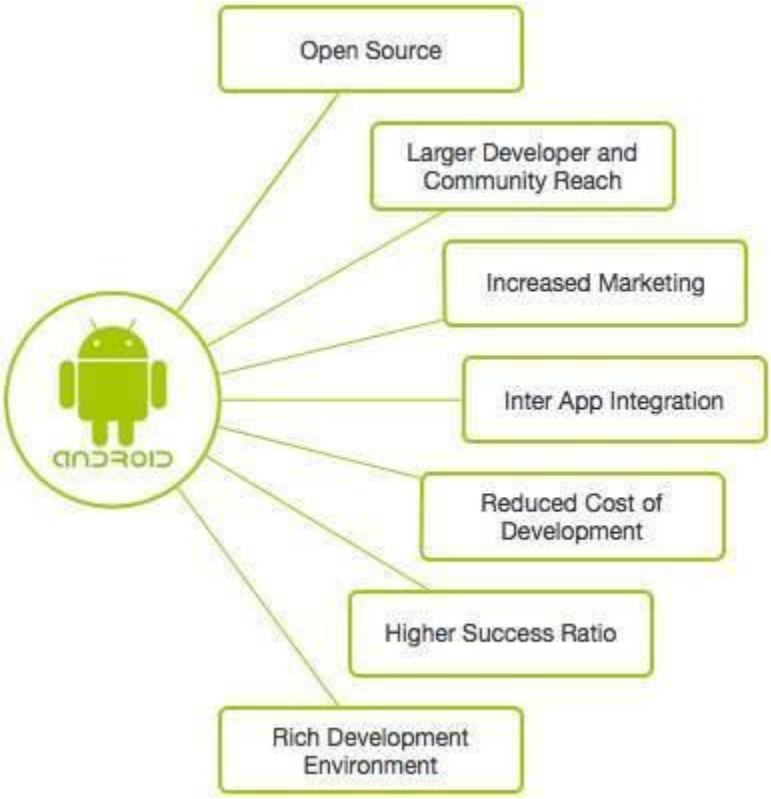
Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.

The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008. On June 27, 2012, at the Google I/O conference, Google announced the next Android version, 4.1 **Jelly Bean**. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance.

What is Open Handset Alliance (OHA)?

It's a consortium of 84 companies such as google, samsung, AKM, synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc. It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Platform.

Why Android:



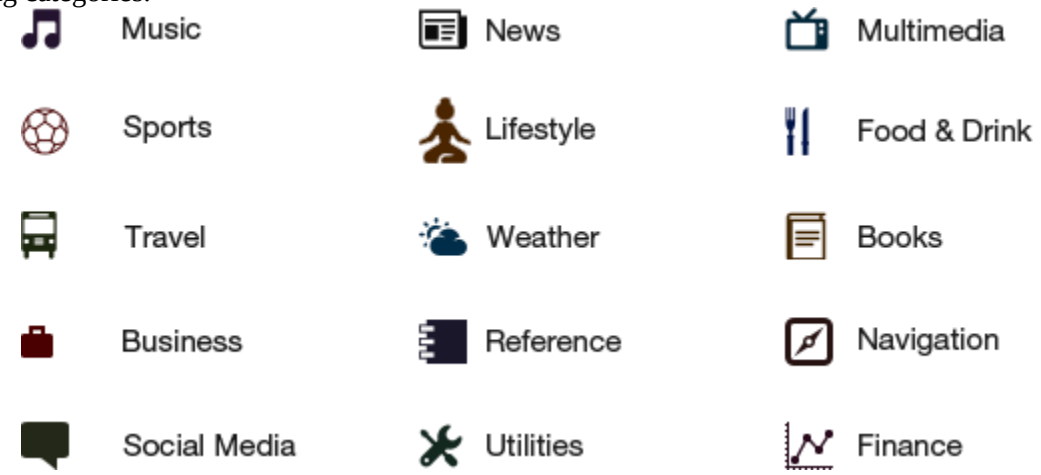
Features of Android:

The important features of android are given below:

1. It is free and open-source.
2. Anyone can customize the Android Platform.
3. There are a lot of mobile applications that can be chosen by customers.
4. It provides many features like weather details, opening screen etc.,
5. It provides supports for messaging (SMS, MMS), web browser, storage (SQLite), Connectivity (GSM, CDMA, Blue Tooth, Wi-Fi etc.,) media, handset layout etc.,
6. It supports multi language.

Categories of Applications:

There are many android applications are available in the market. They are categorized in to the following categories:



Android Versions:

Version	Code name	API Level
1.5	Cupcake	3
1.6	Donut	4
2.1	Éclair	7
2.2	Froyo	8
2.3	Gingerbread	9 and 10
3.1 and 3.3	Honeycomb	12 and 13
4.0	Ice Cream Sandwitch	15
4.1, 4.2 and 4.3	Jelly Bean	16, 17 and 18
4.4	KitKat	19
5.0	Lollipop	21
6.0	Marshmallow	23
7.0	Nougat	24-25
8.0	Oreo	26-27

Android Architecture:

Android architecture or android software stack is categorized into five parts:

1. Linux Kernel
2. Native Libraries (Middleware)
3. Android Runtime
4. Application Framework
5. Applications

1. **Linux Kernel:** This layer provides a level of abstraction between the device hardware and the rest of the architecture and it contains all the essential hardware drivers like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.
2. **Native Libraries:** On top of Linux kernel there is a set of libraries including open-source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.,

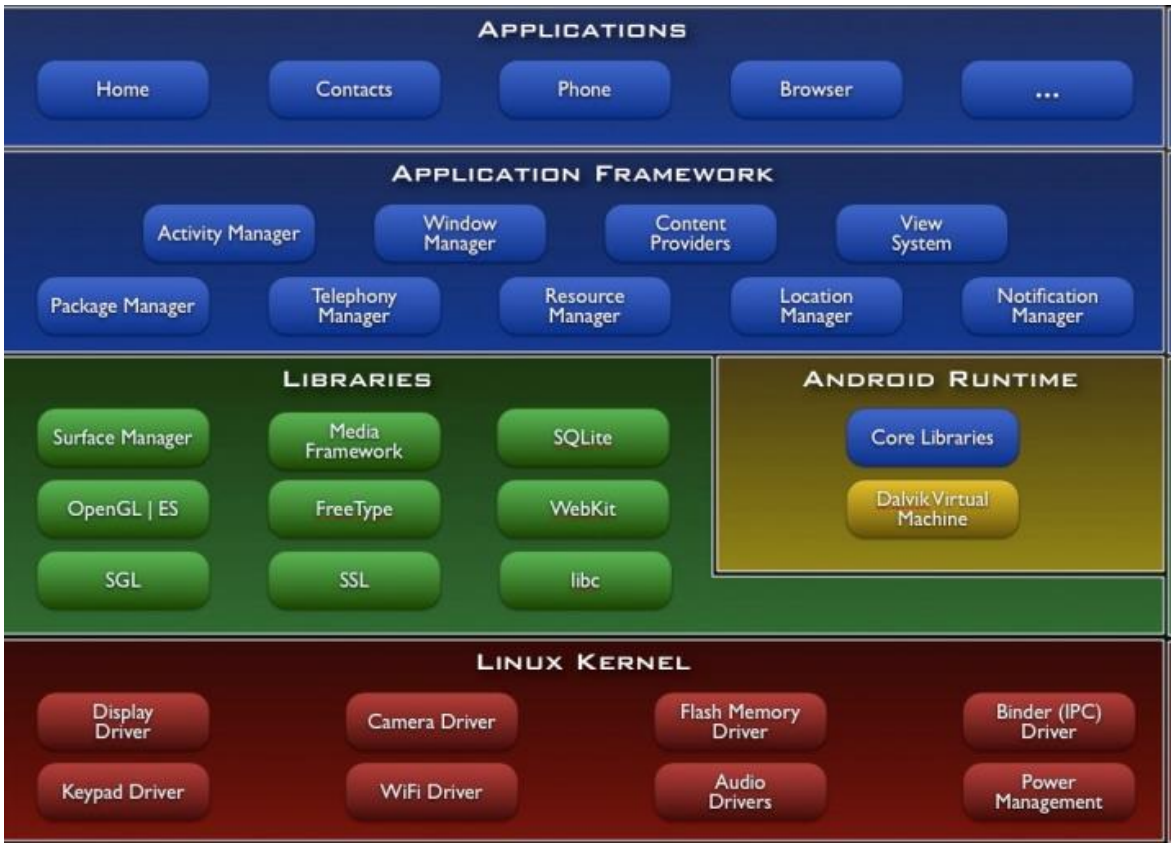
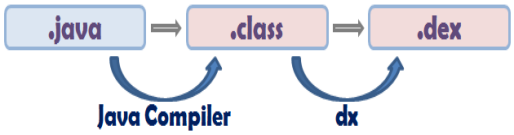


Fig: Android Architecture

3. **Android Runtime:** In android runtime, there are core libraries and DVM (Dalvik Virtual Machine) which is responsible to run android application. DVM is like JVM but it is optimized for mobile devices. It consumes less memory and provides fast performance.
   Dalvik Virtual Machine: It executes the Dalvik executable Format. The .dex format is optimized for minimal memory footprint and it compiles the .java files to .dex files.



4. **Application Framework:** On the top of Native libraries and android runtime, there is android framework. Android framework includes **Android API's** such as UI (User Interface), telephony, resources, locations, Content Providers (data) and package managers. It provides a lot of classes and interfaces for android application development.

Feature	Role
View System	Used to build an application, including lists, grids, text boxes, buttons, and embedded web browser
Content Provider	Enabling applications to access data from other applications or to share their own data
Resource Manager	Providing access to non-code resources (localized strings, graphics, and

	layout files)
Notification Manager	Enabling all applications to display customer alerts in the status bar
Activity Manager	Managing the lifecycle of applications and providing a common navigation back stack

5. **Applications:** On the top of android framework, there are applications. All applications such as home, contact, settings, games, browsers are using android framework that uses android runtime and libraries. Android runtime and native libraries are using Linux kernel.

Android Core Building Blocks:

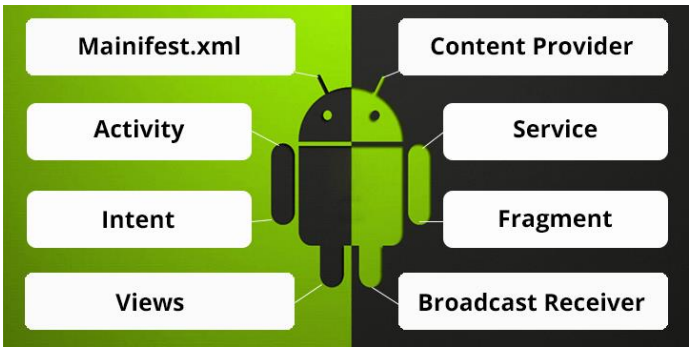
The core building blocks of the android are the core components of the application which we are developing. These components are loosely coupled by the manifest file of that application (AndroidManifest.xml) which describes the characteristics of the application as well as the components of the application.

There are four android core building blocks. They are:

1. Activities
2. Services
3. Broadcast Receivers
4. Content Providers

Along with these there are some other components which are important for the development of the applications. They are described as follows:

Component	Description
Activities	They dictate the UI and handle the user interaction to the smart phone screen.
Views	A view is the UI element such as button, label, text field etc. Anything that you see is a view.
Services	They handle background processing associated with an application.
Fragment	Fragments are like parts of activity. An activity can display one or more fragments on the screen at the same time.
Broadcast Receivers	They handle communication between Android OS and applications.
Content Providers	They handle data and database management issues.
AndroidManifest.xml	It contains information about activities, content providers, permissions etc. It is like the web.xml file in Java EE.



Android Environment Setup:

Step 1 – System Requirements:

Android application development can be done on either of the following operating systems:

i) Microsoft® Windows® 10/8/7/Vista/2003 (32 or 64-bit)
ii) Mac® OS X® 10.8.5 or higher, up to 10.9 (Mavericks)
iii) Linux or UNIX operating system

Next, you need the following software's before you start your android application programming:

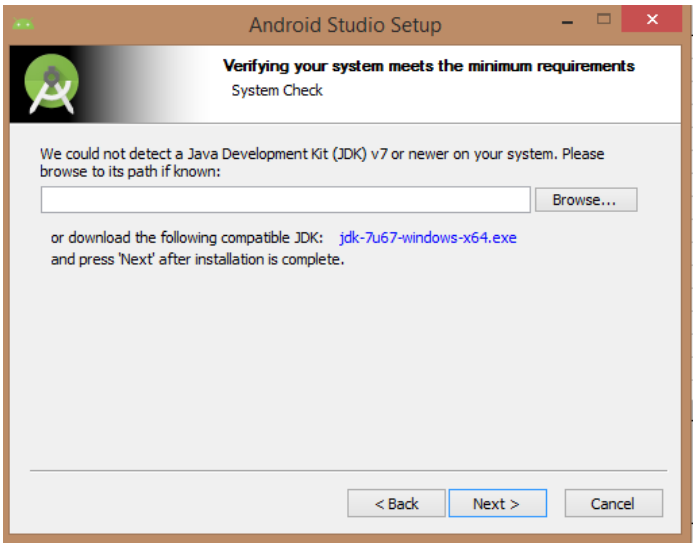
i) JDK 5 or later
ii) Java Runtime Environment (JRE) 6 or later
iii) Android Studio

Step 2 – Setup Android Studio:

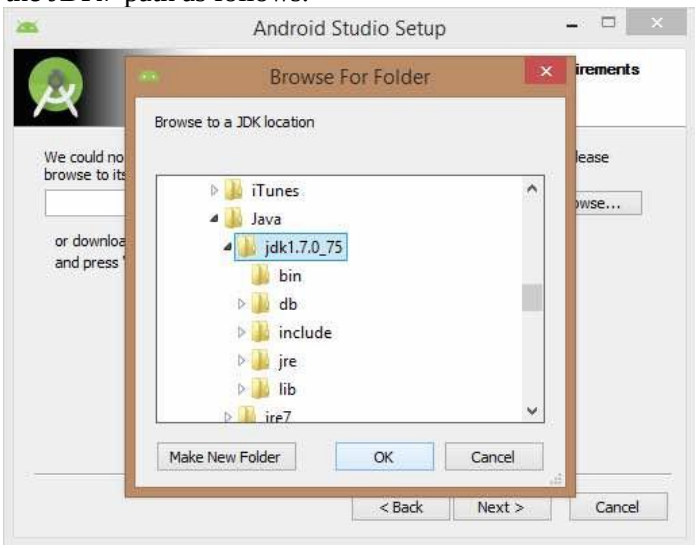
Android Studio is the official IDE for android application development. It works based on **IntelliJ IDEA**, we can download the latest version of android studio from Android Studio Download, if we are new to installing Android Studio on windows, and you will find a file, which is named as ***android-studio-bundle-xx.xxxxxxx-windows.exe***. So just download and run on windows machine according to android studio wizard guideline.

Click on the .exe

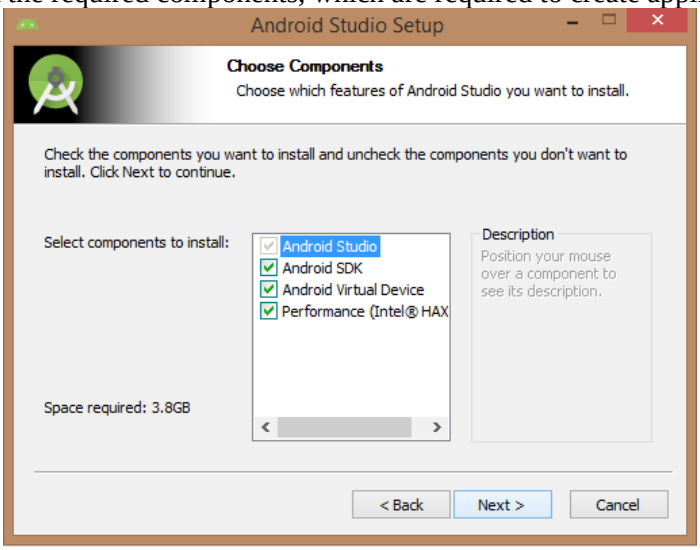
Once you launched Android Studio, it's time to mention JDK7 path or later version in android studio installer.



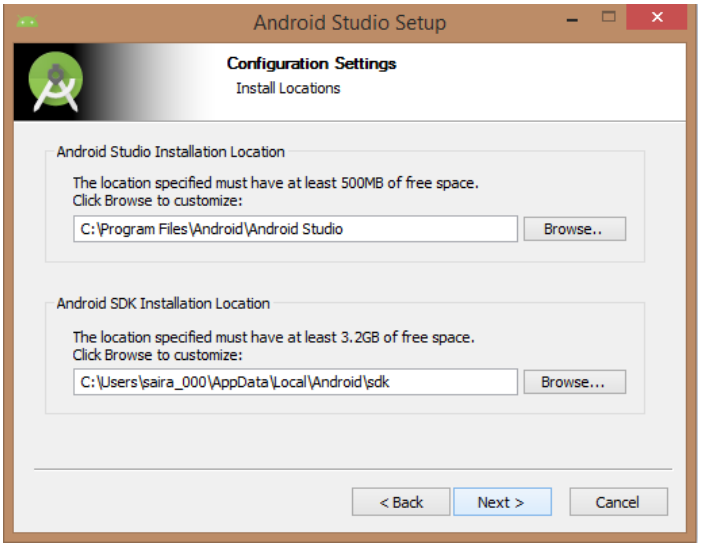
Next, we need to select the JDK7 path as follows.



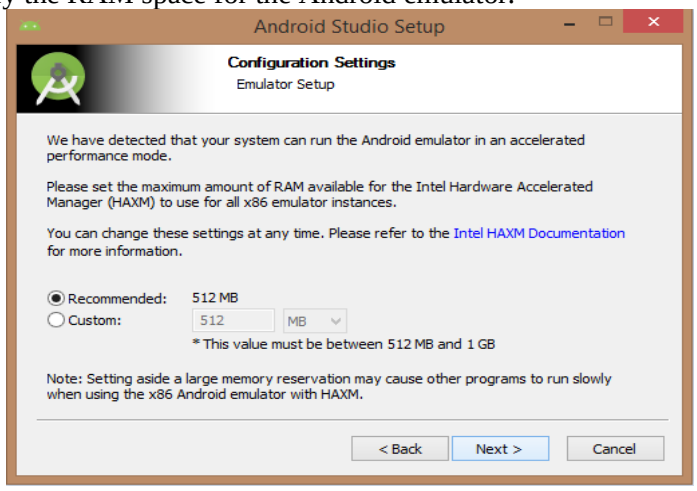
Next, we need to check the required components, which are required to create applications.



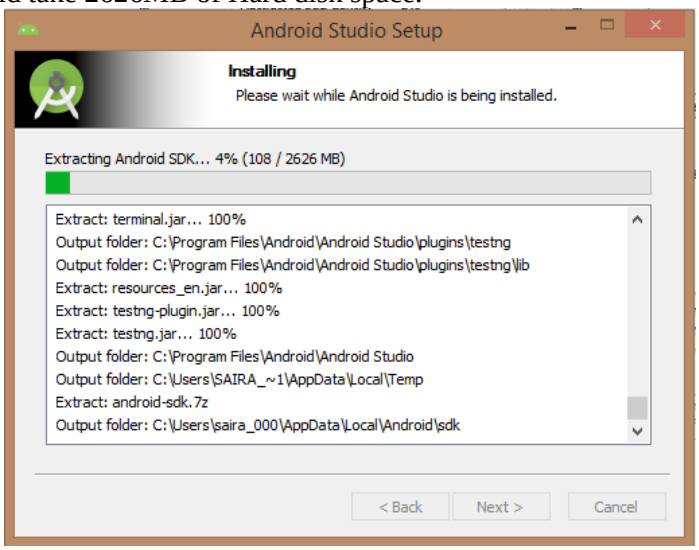
Next, we need to specify the location of the local machine path for the Android Studio and Android SDK tools.



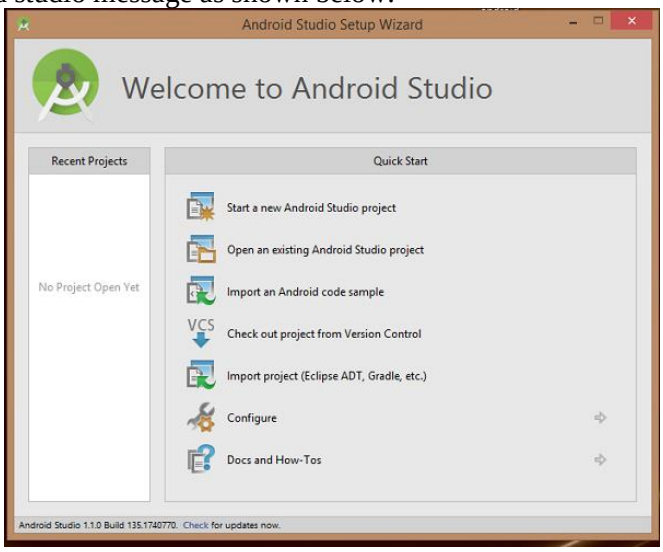
Next, we need to specify the RAM space for the Android emulator.



At the final stage, it would extract SDK packages into our local machine, it would take a while time to finish the task and would take 2626MB of Hard disk space.



After done all above steps perfectly, you must get finish button and it will open android studio project with Welcome to android studio message as shown below:

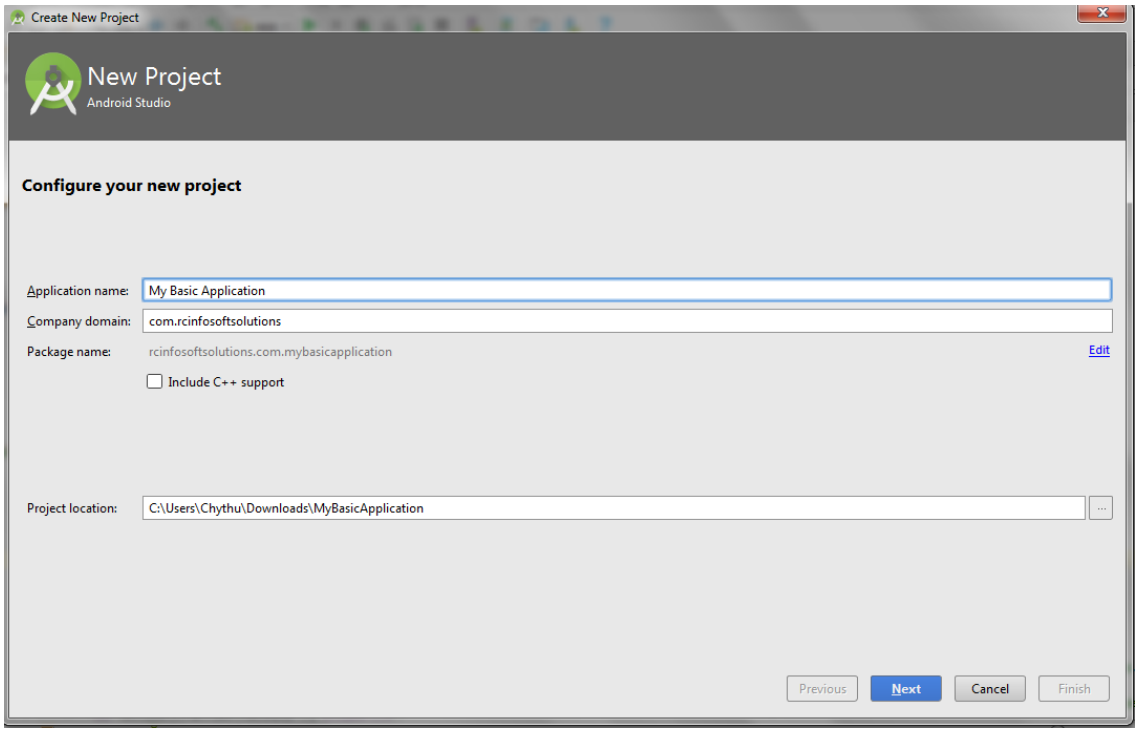


EXAMPLE PROGRAM: Java Android Program to Build a Simple Android Application.

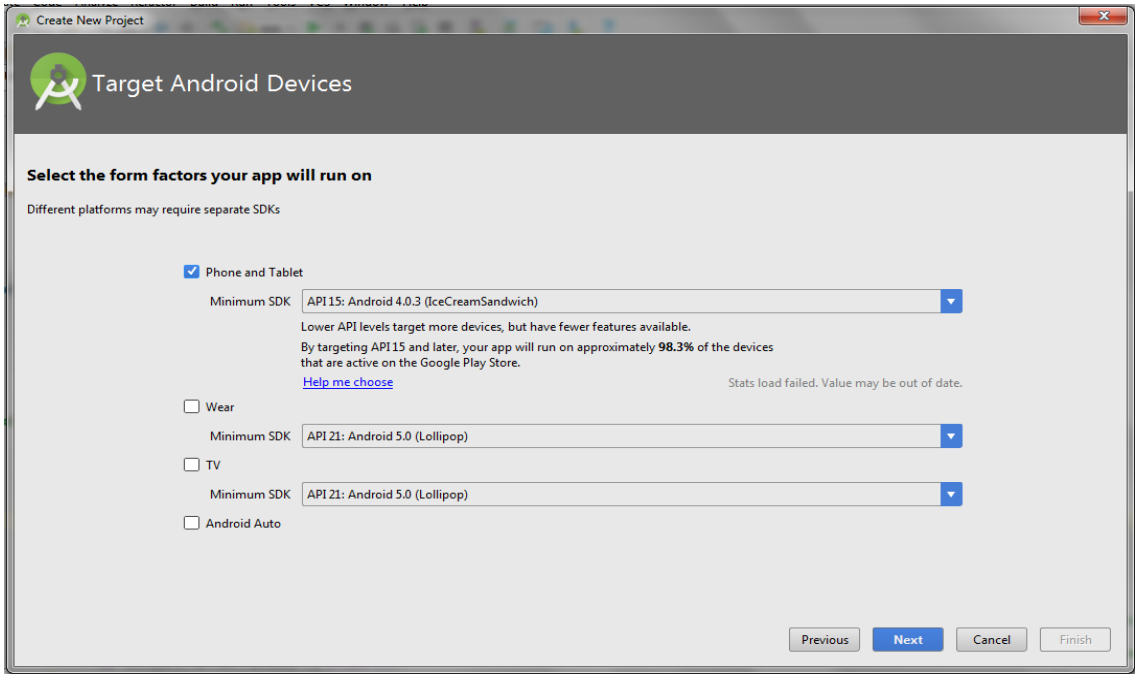
STEPS TO CREATE ANDROID APPLICATION:

Step-1: Select File menu -> New -> New Project. Then the following screen will be displayed.

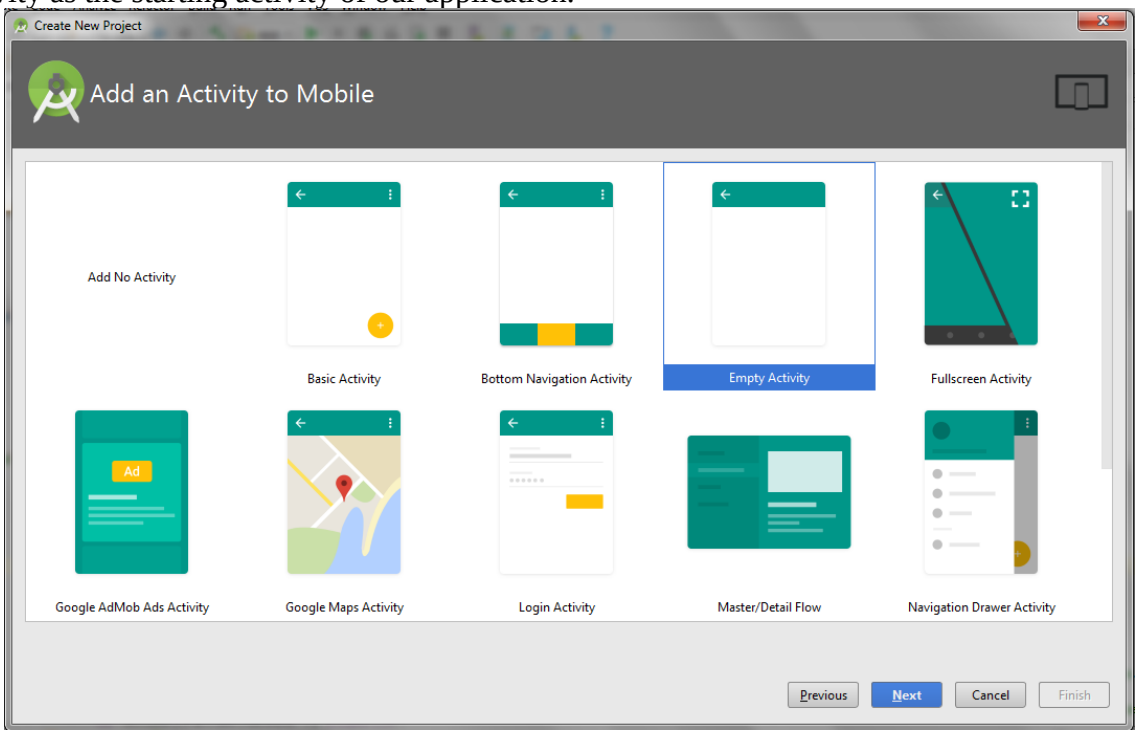
Step-2: Change the name of the application as per your requirement. Then click on the Next button.



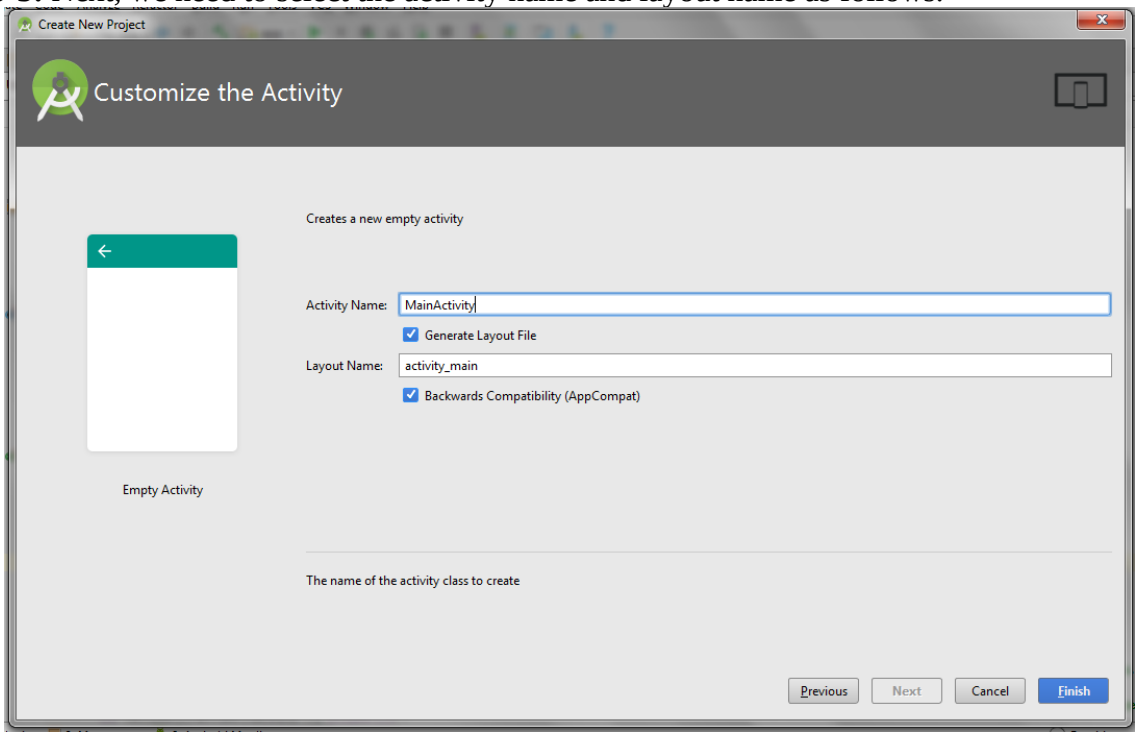
Step-3: Next, we need to select the Android API level on which our application will run. We need to select the minimum level of API level to run on the most of the devices.



Step-4: Next, we need to add an activity to our application; here we need to select the empty activity as the starting activity of our application.



Step-5: Next, we need to select the activity name and layout name as follows.



Step-6: Now click on Finish, now the application will be created and the MainActivity.java file and layout_main.xml file will be opened.

CODE:

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context="rcinfosoftsolutions.com.mybasicapplication.MainActivity">
   <TextView
      android:id="@+id/textView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="SREC"
      android:textSize="52sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintLeft_toLeftOf="parent"
      app:layout_constraintRight_toRightOf="parent"
      app:layout_constraintTop_toTopOf="parent"/>
</android.support.constraint.ConstraintLayout>
```

MainActivity.java:

```
package rcinfosoftsolutions.com.mybasicapplication;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }
}
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
        package="rcinfosoftsolutions.com.mybasicapplication">
   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">
      <activity android:name=".MainActivity">
```

```
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
    </application>
</manifest>
```

OUTPUT: