SCHEME OF VALUATION					
Revision:15					
Course Title: Smart Device Programming			**Course Code**:6133		
Qn No	**Qn No**	**Scoring Indicators**	**Split Score**	**Sub Score**	**Total Score**
1	1	AVD- It is used to test the android application without the need for mobile or tablet etc. It can be created in different configurations to emulate different types of real devices.	2	2	10
	2	Direct, Bundle, Parcelable(Any two)	2*1	2	
	3	A view is a widget that has an appearance on screen. Examples of views are buttons, labels, and text boxes. A view derives from the base class android.view.View	2	2	
	4	<article>,<aside>,<figcaption>,<figure>,<footer> ,<header>,<nav>,<section>,<details>,<mark>,<meter>, <output>, <progress> (Any two))	2	2	
	5	PhoneGap is an open-source framework for mobile app development.It's recommended for those app projects which have to be developed in a cost effective manner. The single code base can be used for creating multiple versions of an app.	2*1	2	
2	1	Android Studio is the official IDE for Android Development by Google. Eclipse is an open-source software, backed by IBM Compare the following factors 1. Build System, 2. Advanced Code Completion\Restructuring, 3. User Interface Design/Layout, 4. Project Organization/Maintenance, 5. IDE Performance/Stability (Any 4)	4*1.5	6	
	2	1. Install the Java Development Kit (JDK). 2. Download Eclipse IDE from its website. 3. Run the installer wizard. 4.Download the ADT plugin for eclipse 5. Configuring the ADT plugin 6. Create an Android Virtual Device (AVD) and run application	6*1	6	

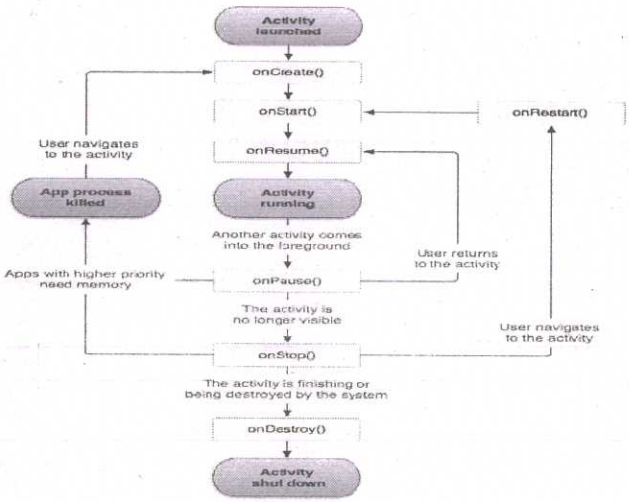
	3	Types of Android Intents- 1. Explicit 2. Implicit. **Implicit intents** specify the action which should be performed and optionally data which provides content for the action. If an implicit intent is sent to the Android system, it searches for all components which are registered for the specific action and the fitting data type **Example:** **Explicit Intent** specifies the component. In such case, intent provides the external class to be invoked. Explicit intents explicitly define the component which should be called by the Android system . **Example**:	3+3	6	
	4	**Direct**: put data into intents directly. Intent has method putExtra that allows to save inside the Intent. **Example**: Intent i = new Intent(EditActivity.this, ViewActivity.class); i.putExtra("name", edtName.getText().toString();	3+3	6	42
	5	There are two main types of menus in Android: **Options menu** — Displays information related to the current activity. In Android, activate the options menu by pressing the MENU key. **Example** **Context menu** — Displays information related to a particular view on an activity. In Android,to activate a context menu tap and hold on to it. **Example**	3+3	6	
	6	1. In the AndroidManifest.xml file, add following two permissions – SEND_SMS and RECEIVE_SMS whichever is needed.: 2. Send the message using the sendSMS() function. There are five arguments to the sendTextMessage() method 1.destinationAddress — Phone number of the recipient 2.scAddress — Service center address; use null for default SMS 3.text — Content of the SMS message 4.sentIntent — Pending intent to invoke when the message is sent 5. deliveryIntent — Pending intent to invoke when the message has been delivered	6	6	

	7	Multimedia Tags <audio>—Embedded sound files. <canvas>—Embedded dynamic graphics. <embed>—To add other technologies that don't have a specific HTML5 element. <source>—The source files for embedded sound and video. <track>—Supplementary media tracks for embedded sound and video. <video>—Embedded video files.	6*1	6	
3	a	**src** — Contains the . java source files for project. **Android library** — This item contains one file, android.jar , which contains all the class libraries needed for an Android application. **gen** — Contains the R.java file, a compiler-generated file that references all the resources found in project. **assets** — This folder contains all the assets used by your application, such as HTML, text files, databases, etc. **res** — This folder contains all the resources used in your application. It also contains a few other subfolders: drawable-<resolution> , layout , and values **AndroidManifest.xml** — This is the manifest file for Android application. Here specify the permissions needed by your application, as well as other features (such as intent-filters,receivers,etc.).	6*1.5	8	15
	b	The AVD Manager is a tool used to create update, delete, repair, and manage Android virtual devices (AVDs), which define device configurations for the Android Emulator. To launch the AVD Manager: In **Android Studio**, select Tools > Android > AVD Manager, or click the AVD Manager icon in the toolbar->Create AVD Or, use the command line to navigate to your SDK's tools/ directory and execute: $ android avd	3	7	

		In **Eclipse** ADT, Below are the steps to create an AVD from AVD manager graphical interface Go to Window ->AVD Manager and select Virtual Devices. Click on New to create a Virtual Device, give it some Name and select Target Android Platform from the drop down list Click "Create AVD"	4		
4	a	Mobile Technologies CDMA, GSM ,2G, 3G, 4G, 5G , GPRS, EDGE, UMTS, LTE,HSDPA/HSPA,EV-DO(Any 4)	4*2	8	15
	b	Compare android and apple iOS -Compare features like Customizability,Developer,OS family,Widgets,File transfer,App store,Available language(s),Video chat,Voice commands,Maps,Latest stable release and Updates,Battery life and management,Open source,File manager,Security,Supported versions,First version(Any 7)	7	7	
5	a	A service is component that runs in background to perform long running operations or to perform work for remote processes. For example, A service may play music in the background while the use is using a different application. A service may upload file to server from the server while the use is using a different application. A service does not provide user interface.	2	8	15
		Started Service (Unbounded) This type of service is created and called by Android Activities. There is no 2 way communication between Android Activity and Service. The Activity just starts the service and does not care about the status of the service. The Service will finish it's work and automatically stops when finish it's job. **Bound Service (Bounded)** This type of Android Service is for 2 way communication. Suppose an Android Activity has started a bounded service, then Activity can be notified about the status by the service.	3+3		
	b	Android has a class called Bundle where we can store our data and supports several data types like strings, chars, boolean, integer and so on. Instead of using the Intent,as data container, we store our info directly into	3		

		the bundle and then we save the bundle into the Intent.			

		Example
		Intent i = new Intent(EditActivity.this, ViewActivity.class);
		Bundle b = new Bundle();
		b.putString("name", edtName.getText().toString());
		b.putString("surname", edtSurname.getText().toString());
		b.putString("email", edtEmail.getText().toString());
		i.putExtra("personBdl", b);
		startActivity(i);
			4	7	

6	a	LIFE CYCLE OF AN ACTIVITY  onCreate- Called when activity is first created. Used to initialize the activity onStart- Called when activity is becoming visible to the	5 8	8	

		user.onResume- Called if the activity get visible again and the user starts interacting with the activity again. onPause- Called once another activity gets into the foreground. Always called before the activity is not visible anymore. onStop- Called once the activity is no longer visible. onRestart-Called after your activity is stopped, prior to start. onDestroy- Called before the activity is destroyed	3		15
	b	A broadcast receiver is a component that receive information, broadcasted by other applications or by the system, and take some actions accordingly. For example, An app can schedule an alarm to notify the user at any specified time. It's same as we set an alarm to notify at a specific time. Another example is notify the user when battery is low.	4	7	

		A broadcast receiver is implemented as subclass of BroadcastReceiver.
		public class MyReceiver extends BroadcastReceiver {
		public void onReceive(context,intent){}
		}
			3		

7	a	Steps to publish an app in Google Playstore 1. Create an account 2. Familiarise yourself with Developer Console 3. Fill in the necessary account details 4. Link your merchant account 5. Upload your app 6. Alpha and beta testing of app 7. Provide details for store listing 8. Add pricing and distribution details 9. Publishing the application 10. Device Filtering option	8	8	15
	b	Content providers can help an application manage access to data stored by itself, stored by other apps, and provide a way to share data with other apps. They encapsulate the data, and provide mechanisms for defining data security. Content providers are the standard interface that connects data in one process	3		

		with code running in another process			
		Advantages of content providers Content providers offer granular control over the permissions for accessing data. Can use a content provider to abstract away the details for accessing different data sources in your application. **Disadvantages of content providers** Content providers can only use SQLite as a storage mechanism. More complicated queries, with joins and distinct, are not possible with content provider.	2+2	7	
8	a	Android supports the following Layouts 1. LinearLayout 2. AbsoluteLayout 3. TableLayout 4. RelativeLayout 5. FrameLayout(Any 4)	4*2	8	15
	b	**Steps to write:** 1. To save text into a file, use the FileOutputStream class. 2. The openFileOutput() method opens a named file for writing, 3. To convert a character stream into a byte stream, use an instance of the OutputStreamWriter class. 4. Use its write() method to write the string to the file. 5. To ensure that all the bytes are written to the file, use the flush() method. 6. Finally, use the close() method to close the file: **Steps to read**: 1. To read the content of a file from the device's internal storage in android app, use the FileInputStream class, together with the InputStreamReader class: 2. The read() method of the InputStreamReader object reads the number of characters read and returns -1, if the end of the file is reached.	4+3	7	
9	a	**Features of HTML 5** New Semantic Elements			

		Forms 2.0 Persistent Local Storage WebSocket Server-Sent Events Canvas Audio & Video Geolocation Microdata Drag and drop (Any 8)	8*1	8	
	b	**Advantages of PhoneGap** 1. Single code base can be used to create apps for iOS, Android, Windows Phone and mobile web. 2. Even if you have developed the app using PhoneGap, the app can be distributed on App store and Google Play. 3. With Phonegap, it is easy to do rapid testing and deployment of an app 4. Can create hybrid apps to save the development time and efforts. 5. If there is little time to go to market, Phonegap is the technology to be used. 6. There is no need of additional development skills 7. It is possible to tape into the hardware of device such as camera, geo-location, accelerometer and various others 8. If the app is going to be the extension of your existing enterprise app, hybrid will prove to be best choice with PhoneGap 9. Hybrid is the best technology to launch a minimal viable product to rapidly figure out that how market reacts on certain app 10. Phonegap can be the best technology to support those apps which do not require heavy usage of native features of an OS. (Any 7)	7*1	7	15
10	a	JavaScript is a programming language that you can use to affect the user interface and create dynamic websites. Use of JavaScript for mobile application development 1.Rollovers 2.Verifying form data 3.Opening new windows 4. Setting cookies	4*2	8	

	b	The HTML 5 tags related to form handling <input type=color> . <input type=date> . <input type=datetime> . <input type=datetime-local> . <input type=email> . <input type=month> . <input type=number> . <input type=range> . <input type=search> . <input type=time> . <input type=tel> . <input type=url> . <input type=week> (Any 7)	7*1	7	15