**Topic**: An android application for displaying the RSSI Signal Strength of Wifi networks in range

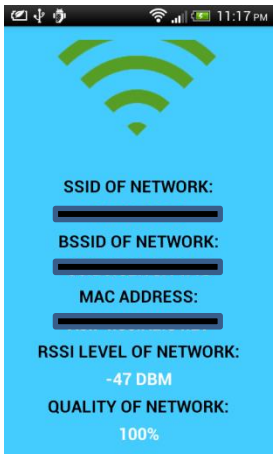
**Implementation Details:**

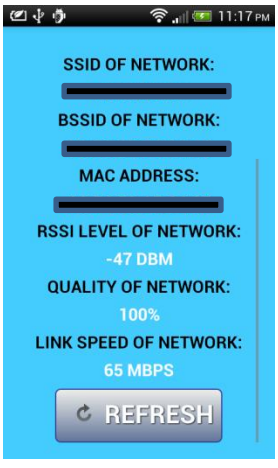

The home screen of the application contains two options:

a. Connected Network – Display the details of only the network to which the device is connected.
b. Network List – Display details regarding all the wifi networks which are in range

Each of the two options shall led to creation of a new intent and transfer of control to respective activities.

1) Connected Network:
    a. If the device is not connected to any wifi network, an error message shall be displayed to indicate the same. Refer the following leftmost picture:

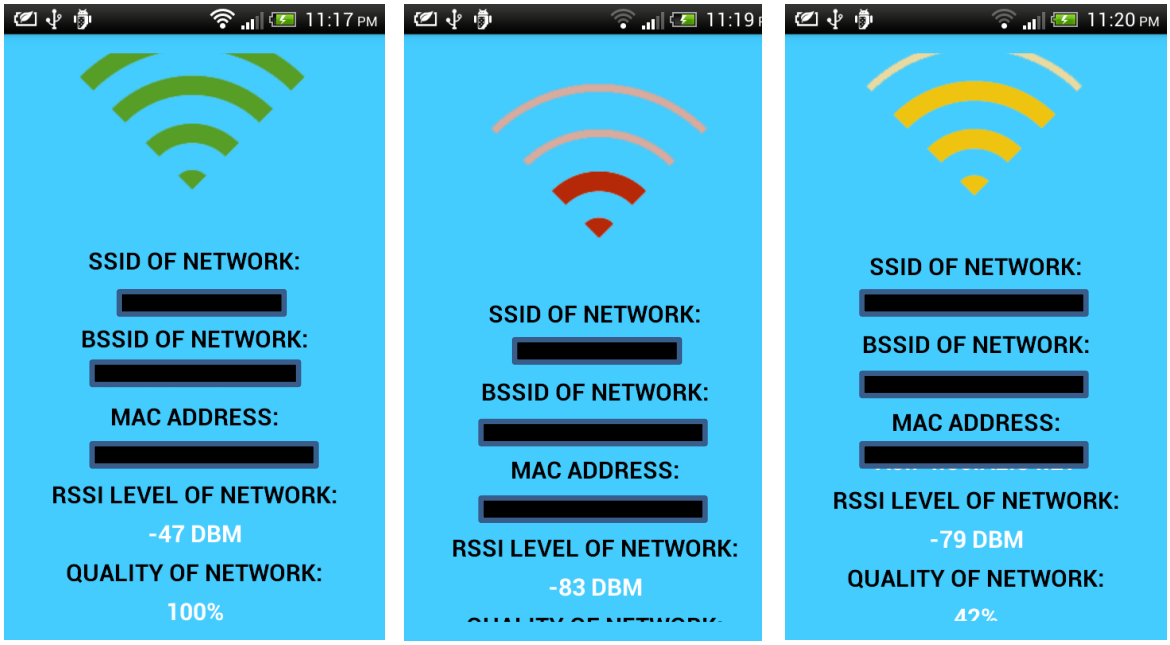




    b. In case of proper connection with a wifi network, the following details shall be displayed:
        i. SSID
        ii. BSSID

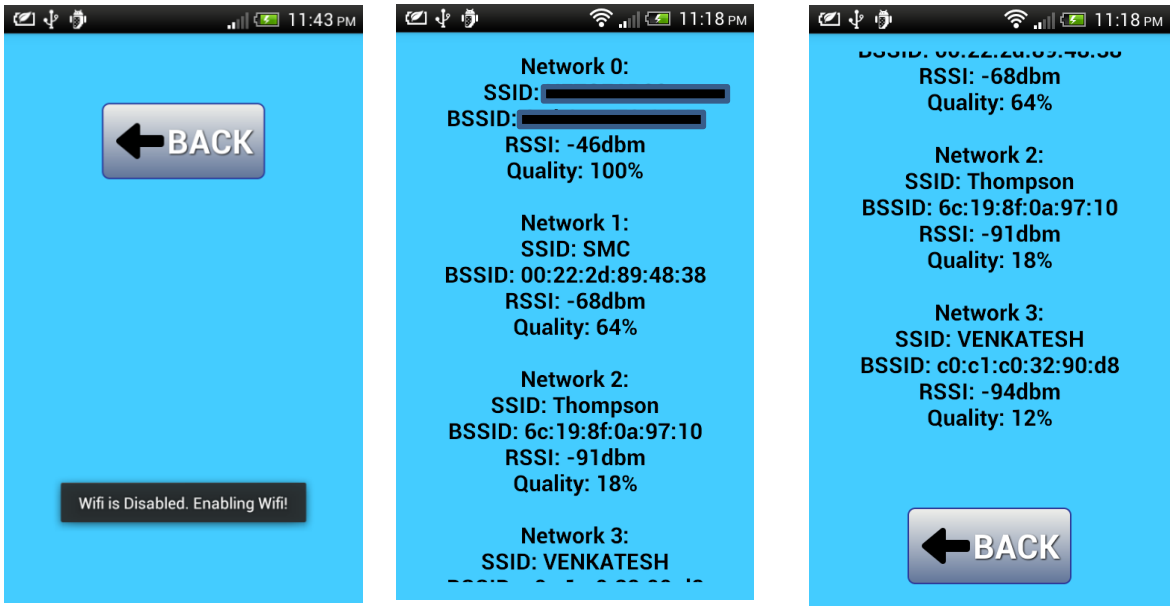
iii. MAC ADDRESS
iv. RSSI Level
v. Quality (Calculated from RSSI )
vi. Link Speed

c. We also have a refresh button in order to acquire the details again in real time and the layout is scrollable and thus even landscape orientation is perfectly supported.
d. For this, we use the geConnectionInfo() method of class WifiManager for extracting SSID,BSSID, Mac Address, RSSI and Link Speed.
e. A RSSI value of -50 dbm or more is considered to be 100%, while a value of -100 dbm or less is considered as 0% network quality. In order to calculate intermediate network quality, the formula used is quality = 2 * (RSSI + 100)
f. According to the quality of signal, we change the wifi strength picture on top of this layout. If quality is less than or equal to 40%, picture showing one signal wave of red color is displayed. If quality is greater than 40% but less than 75%, two signal waves of yellow color are shown and if quality is greater than 75% we have all the three signal waves of green color.
g. Following screenshots explain the same more effectively




2) Network List

a. In order to view all the networks in range, the wifi must be enabled and hence on clicking the Network List button on the homepage, wifi setting is enable from the application itself.




b. For this intially we use getSystemService(Context.WIFI_SERVICE) and convert it to object of class WifiMAnager. Then we check whether the wifi setting is enabled or not using isWifiEnabled() method of WifiManager object thus created. If wifi is not enabled then we use the method setWifiEnabled(True) of WifiManager object. Once it is ensured that wifi is enabled, we proceed to Network_List activity.
c. In order to scan a list of wireless networks, an object of BroadcastReceiver class needs to be registered using the onRegister() method of the class which is done in OnResume() function in the activity. For this we create a WifiScanner class which extends BroadcastReciever and create its object to get wifi networks in range using onReceive() method.
d. We obtain the WifiManager object using getSystemService(Context.WIFI_SERVICE) and then use its method startScan()
e. Using getScanResults() of WifiManager object we obtain the list of all wifi networks in range which are stored in a list of ScanResult objects.
f. ScanResult objects have attributes SSID, BSSID, RSSI and using RSSI, we calculate the quality of each network.
g. Thus for each network we display the results on the scrollable layout.
h. The network list is automatically updated after a specific duration and thus it is dynamic.
i. A back button is provided to return to the home screen for convenience and it is ensured that landscape mode works just as fine as the portrait mode.

**Permissions required for app:**

1. For accessing wifi network
2. For changing state of wifi network
3. For accessing network state

**Activities:**

1. MainActivity – activity starts with the application
2. Wifi_details – activity starts with Connected Network button is pressed
3. No_network – activity starts when device is not connected to any wifi network
4. Network_List – activity starts when Network List button is pressed

**Layout:**

1. activity_main.xml – home page layout
2. activity_network_list.xml – layout for displaying details of wifi networks in range
3. activity_wifi_details.xml – layout for displaying details of connected wifi network
4. activity_no_network.xml – layout for displaying error message when device isn't connected to any wifi network

**Addition resources:**

1. Few images and icons have been added to res folder for visual purposes and graphical user interface.
2. custombuttom.xml is used to create layout for buttons used in the application.