1. **Design and implement an App that will calculate a person's BMI (**Body Mass Index**). The metric formula is** *BMI = Mass/(height * height).* **Include the following in your App:**

   a) Create your *own* App icon in 3 different resolutions (doesn't need to be beautiful!)

   b) At least one of: TextView, EditText and Button components in the main activity. Use two layout managers e.g. one LinearLayout inside another LinearLayout. Any text relevant to the components should be stored in the *strings.xml* file.

   c) Process the events on at least one of the GUI components in **b)** above.

   d) A trigger on the first Activity launches a second Activity and passes some data to it for display. A Third Activity that is triggered by the first Activity and passes data back to the first Activity.

   e) Localise your App's text, including it's name, for a second language of your choice. Change the language in your AVD to see your App correctly & automatically switch to and from the second language.

   f) Create a Second layout file for horizontal orientation – preserve at least 1 app variable using *onSaveInstanceState* & *onRestoreInstanceState* methods.

   (10 marks ea, 60 marks total)

2. **Three Advanced features or innovation/creativity; here are some examples but you can do different ones of equivalent difficulty:**

   g) Show the user a list of choices. Show the user an Alert using AlertDialog class.

   h) Save the App state when quitting. Restore on re-launch. NB method *onSaveInstanceState* does NOT do that.

   i) Draw a graphic to display some app related info.

   (10 marks ea, 30 marks total)

3. **Documentation includes the following:**

   1) Commented code & the use a coding standard. Use extra classes where appropriate.

   2) This document stating what was done including screen shots of code snippets & the running app code (on pages 3+). Include list of classes, class or block diagram, what each is for. Also include an evaluation – how has it worked out & future outlook – what could be done differently.

   (5 marks ea, 10 marks total)

Assignment Procedure guide or (includes a coding standard)

Demo in the lab (mandatory for on site courses):
- Demonstrate the running program in the 1st lab following the due date (not for online courses). You may be asked to make changes to the code after your demo. The purpose of the demo is to show what you have done and that you understand what you have done.

- **Submission**
  Submit to Blackboard: This edited document & Project code.. All goes in 1 archive file. *Please name the file* as in this example: **YourClass-YourName-MOB-Asn1.7z**

  "By submitting this electronically, using an account & password only I have access to, I hereby certify that this material which I now submit for assessment is entirely my own work and has not been taken from the work of others, save and to the extent, that such work has been cited and acknowledged within the text of my work."

.

Completed? Yes

K. Grabe

Oct 2014

1 (b) At least one of: TextView, EditText and Button components in the main activity. Use two layout managers e.g. one LinearLayout inside another LinearLayout. Any text relevant to the components should be stored in the strings.xml file.

Completed? Yes

```
 1  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
 2      xmlns:tools="http://schemas.android.com/tools"
 3      android:layout_width="match_parent"
 4      android:layout_height="match_parent"
 5      android:id="@+id/background"
 6      android:background="@drawable/galaxy_wallpaper"
 7      android:paddingBottom="@dimen/activity_vertical_margin"
 8      android:paddingLeft="@dimen/activity_horizontal_margin"
 9      android:paddingRight="@dimen/activity_horizontal_margin"
 0      android:paddingTop="@dimen/activity_vertical_margin"
 1      tools:context="mobileapp.bmi_ator.MainActivity" >
 2
 3      <Button
 4          android:id="@+id/Bt_Cancel"
 5          android:layout_width="wrap_content"
 6          android:layout_height="wrap_content"
 7          android:layout_alignBaseline="@+id/Bt_Submit"
 8          android:layout_alignBottom="@+id/Bt_Submit"
 9          android:layout_toLeftOf="@+id/textView1"
 0          android:text="@string/cancel" />
 1
 2      <LinearLayout
 3          android:id="@+id/LinearLayout01"
 4          android:layout_width="wrap_content"
 5          android:layout_height="wrap_content"
 6          android:layout_below="@+id/textView1"
 7          android:layout_centerHorizontal="true"
 8          android:layout_marginTop="30dp"
 9          android:orientation="horizontal" >
 0
 1          <TextView
 2              android:id="@+id/TextView01"
 3              android:layout_width="wrap_content"
 4              android:layout_height="wrap_content"
 5              android:text="@string/name"
 6              android:textAppearance="?android:attr/textAppearanceMedium"
 7              android:typeface="@+assets/Gravity_Light.ttf" />
 8
 9          <EditText
 0              android:id="@+id/EditTextName"
 1              android:layout_width="89dp"
 2              android:layout_height="wrap_content"
 3              android:layout_weight="1"
 4              android:ems="10"
 5              android:inputType="text" />
```
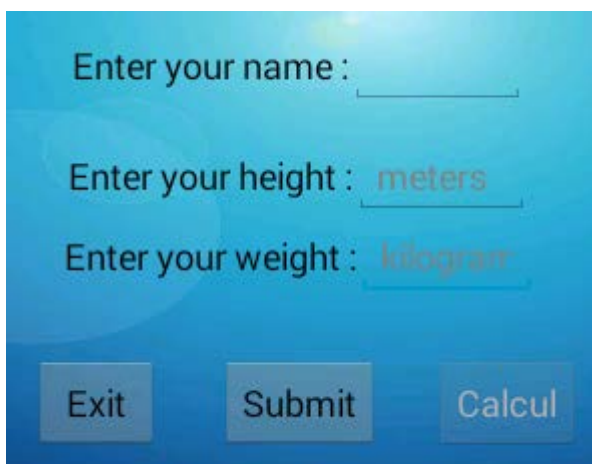
1 (c) Process the events on at least one of the GUI components in b) above.

Completed? Yes

In the MainActivity, if you don't complete the EditText and you click on Submit :


```java
104      submit_Btn.setOnClickListener(new View.OnClickListener() {
105
106          @Override
107          public void onClick(View v) {
108
109
110              final int l1 = Et_Name.getText().toString().length();
111              final int l2 = Et_Size.getText().toString().length();
112              final int l3 = Et_Weight.getText().toString().length();
113
114
115
116              if ((l1 == 0)) {
117                  Et_Name.setError(getString(R.string.error_required));
118              }
119
120
121              if ((l2 == 0)) {
122                  Et_Size.setError(getString(R.string.error_required));
123              }
124
125
126              if ((l3 == 0)) {
127                  Et_Weight.setError(getString(R.string.error_required));
128              }
129
130
131
132              if ((l1 != 0) && (l2 != 0) && (l3 != 0)) {
133
134
135                  calcul_Btn.setEnabled(true);
136                  submit_Btn.setEnabled(false);
```

1 (d) A trigger on the first Activity launches a second Activity and passes some data to it for display. A Third Activity that is triggered by the first Activity and passes data back to the first Activity.

Completed? Yes

Your BMI is :

Calcul

24.835646457268076
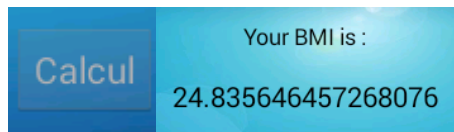
THIS BUTTON IN THE MAIN ACTIVITY LAUNCH THE SECOND ACTIVITY AND PASSES BMI VALUE.

```
169      calcul_Btn.setOnClickListener(new View.OnClickListener() {
170
171          @Override
172          public void onClick(View v) {
173
174              EditText Et_Size = (EditText) findViewById(R.id.EditTextHeight);
175              EditText Et_Weight = (EditText) findViewById(R.id.EditTextWeight);
176
177              double size = Double.parseDouble(Et_Size.getText().toString());
178              double weight = Double.parseDouble(Et_Weight.getText()
179                      .toString());
180              String BMI = Double.toString(weight / (size * size));
181
182              Intent launcher = new Intent(MainActivity.this,
183                      SecondActivity.class);
184
185              launcher.putExtra(BOX_BMI, BMI);
186
187              startActivityForResult(launcher, 1);
188
189              final ProgressDialog dialog = ProgressDialog.show(MainActivity.this, "","Loading..Wait.." , true);
190              dialog.show();
191              Handler handler = new Handler();
192              handler.postDelayed(new Runnable() {
193                  public void run() {
194                      //your code here
195                          dialog.dismiss();
196                  }
197              }, 1000);   // 3000 milliseconds
198
199          }
200      });
201
```

```
32          Intent intent = getIntent();
33          final double BMI = Double.parseDouble(intent.getStringExtra(BOX_BMI));
34
35
36
37          if (intent != null) {
38              TextView Value_BMI = (TextView) findViewById(R.id.Value_BMI);
39              Value_BMI.setText(intent.getStringExtra(BOX_BMI));
40
41          }
```

I couldn't do the the Third Activity that is triggered by the first Activity and passes data back to the first Activity but I put a trigger on the second Activity that launches a fourth Activity and passes value of BMI in order to place an ImageView.

1 (e) Localise your App's text, including it's name, for a second language of your choice. Change the language in your AVD to see your App correctly & automatically switch to and from the second language.

Completed? Yes

```
⊿ 📂 values
      🗋 dimens.xml
      🗋 strings.xml
      🗋 styles.xml
⊿ 📂 values-fr
      🗋 dimens.xml
      🗋 strings.xml
      🗋 styles.xml
```

```xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3
4      <string name="app_name">BMI App</string>
5      <string name="welcome">Welcome On BMI App !</string>
6      <string name="action_settings">Settings</string>
7       <string name="next">Next</string>
8      <string name="cancel">Exit</string>
9      <string name="name">Enter your name : </string>
10     <string name="height">Enter your height : </string>
11     <string name="weight">Enter your weight : </string>
12     <string name="previous">Previous</string>
13      <string name="bmi_value">Your BMI is :</string>
14     <string name="submit">Submit</string>
15       <string name="calcul">Calcul</string>
16       <string name="title">YOUR POSITION</string>
```

```xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3
4      <string name="app_name">IMC App</string>
5      <string name="welcome">Bienvenue Sur IMC App !</string>
6      <string name="action_settings">Paramètres</string>
7       <string name="next">Suivant</string>
8      <string name="cancel">Sortir</string>
9      <string name="name">Entrer votre nom : </string>
10     <string name="height">Entrer votre taille : </string>
11     <string name="weight">Entrer votre poids : </string>
12     <string name="previous">Précédent</string>
13      <string name="bmi_value">Votre IMC est :</string>
14     <string name="submit">Soumettre</string>
15       <string name="calcul">Calcule</string>
16       <string name="title">VOTRE POSITION</string>
```

Welcome On BMI App !

Bienvenue Sur IMC App !

1 (f) Create a Second layout file for horizontal orientation – preserve at least 1 app variable using onSaveInstanceState & onRestoreInstanceState methods.

Completed? Yes

▲ 📂 layout
    📄 activity_fourth.xml
    📄 activity_main.xml
    📄 activity_second.xml
    📄 activity_third.xml
▲ 📂 layout-land
    📄 activity_main.xml
    📄 activity_second.xml
    📄 activity_third.xml

```
275
276    @Override
277    public void onSaveInstanceState(Bundle savedInstanceState) {
278
279
280        // Save the user's current index state
281         EditText Et_Name = (EditText) findViewById(R.id.EditTextName);
282        savedInstanceState.putString("index", Et_Name.getText().toString());
283
284        // Always call the superclass so it can save the view hierarchy state
285            super.onSaveInstanceState(savedInstanceState);
286    }
287
288    public void onRestoreInstanceState(Bundle savedInstanceState) {
289        // Always call the superclass so it can restore the view hierarchy
290        super.onRestoreInstanceState(savedInstanceState);
291
292        // Restore state members from saved instance
293        EditText Et_Name = (EditText) findViewById(R.id.EditTextName);
294        Et_Name.setText(savedInstanceState.getString("index"));
295
296    }
```

If you change of orientation, the value you wrote into the EditText Name doesn't disappear.

2 (g) Show the user a list of choices. Show the user an Alert using AlertDialog class.

Completed? Yes
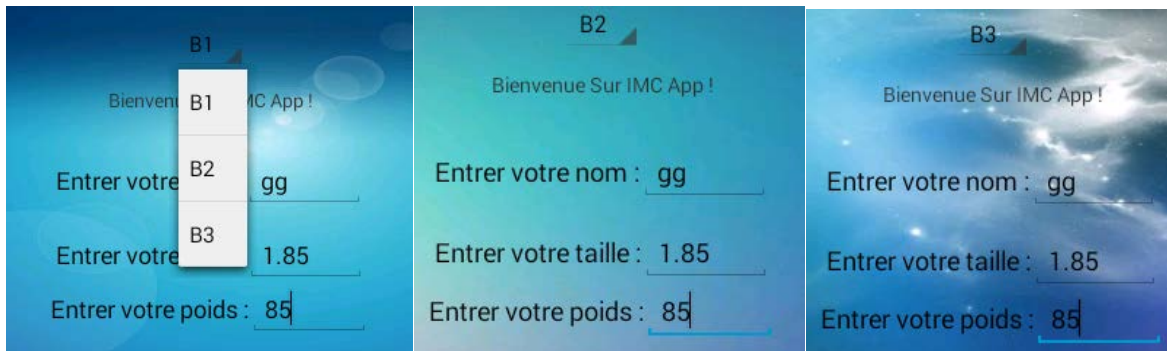

```
67      spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
68
69          @Override
70          public void onItemSelected(AdapterView adapter, View v, int i,
71                  long lng) {
72              if (i == 0) {
73                  View background = findViewById(R.id.background);
74                  background
75                          .setBackgroundResource(R.drawable.galaxy_wallpaper);
76
77              }
78
79              if (i == 1) {
80                  View background = findViewById(R.id.background);
81                  background.setBackgroundResource(R.drawable.background2);
82
83              }
84
85              if (i == 2) {
86                  View background = findViewById(R.id.background);
87                  background.setBackgroundResource(R.drawable.background3);
88
89              }
90
91          }
92
93          @Override
94          public void onNothingSelected(AdapterView arg0) {
95              Toast.makeText(getApplicationContext(), "Nothing selected",
96                      Toast.LENGTH_SHORT).show();
97
98          }
99      });
```
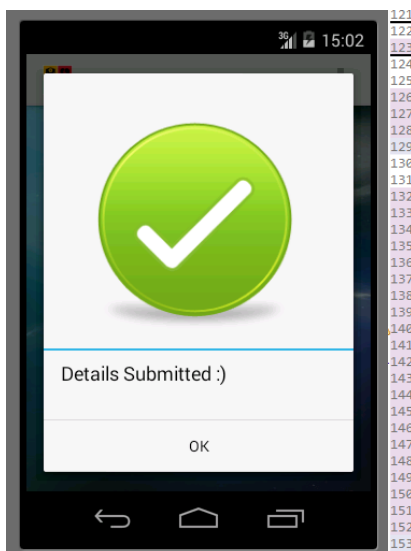


```
121
122      if ((l1 != 0) && (l2 != 0) && (l3 != 0)) {
123
124          calcul_Btn.setEnabled(true);
125          submit_Btn.setEnabled(false);
126
127          AlertDialog alertDialog = new AlertDialog.Builder(
128                  MainActivity.this).create();
129
130          // Setting Dialog Title
131          alertDialog.setTitle("Alert Dialog");
132
133          // Setting Dialog Message
134          alertDialog.setMessage("Details Submitted :)");
135
136          // Setting Icon to Dialog
137          alertDialog.setIcon(R.drawable.tick);
138
139          // Setting OK Button
140          alertDialog.setButton("OK",
141                  new DialogInterface.OnClickListener() {
142                  public void onClick(DialogInterface dialog,
143                          int which) {
144                      // Write your code here to execute after
145                      // dialog closed
146                      Toast.makeText(getApplicationContext(),
147                              "You clicked on OK",
148                              Toast.LENGTH_SHORT).show();
149                  }
150          });
151
152          // Showing Alert Message
153          alertDialog.show();
```
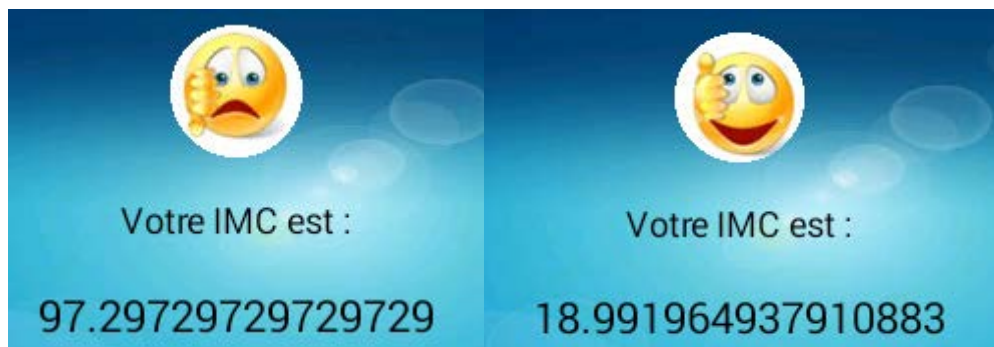
Completed? No

BUT I put 3 different events that depend of the BMI Value:

- An image in the SecondActivity (top or bad)

```
24
25              final ImageView top = (ImageView) findViewById(R.id.Img_top);
26              final ImageView bad = (ImageView) findViewById(R.id.Img_bad);
27
28            top.setVisibility(View.INVISIBLE);
29             bad.setVisibility(View.INVISIBLE);
30
```

```
43
44            if (BMI>=18.5 && BMI<24.9) {
45                 top.setVisibility(View.VISIBLE);
46                 }
47            else{
48                 bad.setVisibility(View.VISIBLE);
49            }
50
```


Votre IMC est :

97.29729729729729

Votre IMC est :

18.991964937910883

- 2 information about the BMI you have

```
94   public String Classification(double BMI){
95       String value = "";
96
97       if(BMI<18.5){
98           value = "Poor Weight";
99       }
100
101      if(BMI>=18.5 && BMI<24.9){
102          value = "Normal Weight";
103      }
104
105      if(BMI>=24.9 && BMI<29.9) {
106          value = "Overweight";
107      }
108
109      if(BMI>=29.9 && BMI<34.9) {
110          value = "Moderate Obesity" ;
111      }
112
113      if(BMI>=34.9 && BMI<39.9) {
114          value = "Severe Obesity" ;
115      }
116
117      if(BMI>=39.9){
118          value = "Massive Obesity" ;
119      }
120
121      return value;
122      }
123
```

```
124
125
126   public String Problems(double BMI){
127       String value = "";
128
129       if(BMI<18.5) {
130           value = "Health Problems Increased";
131       }
132
133       if(BMI>=18.5 && BMI<24.9){
134           value= "Health Problems Lesser";
135       }
136
137       if(BMI>=24.9 && BMI<29.9) {
138           value= "Health Problems Increased";
139       }
140
141       if(BMI>=29.9 && BMI<34.9) {
142           value= "Health Problems Moderate";
143       }
144
145       if(BMI>=34.9 && BMI<39.9) {
146           value= "Health Problems High";
147       }
148
149       if(BMI>=39.9) {
150           value = "Health Problems Very High";
151       }
152
153       return value;
154
155   }
156
```

18.991964937910883 — Normal Weight — Health Problems Lesser

25.712198685171657 — Overweight — Health Problems Increased

- An image that move on the graphic depending on BMI Value
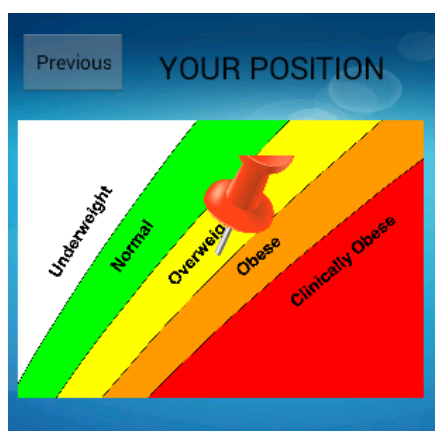
Case of Overweight: Case of Underweight :
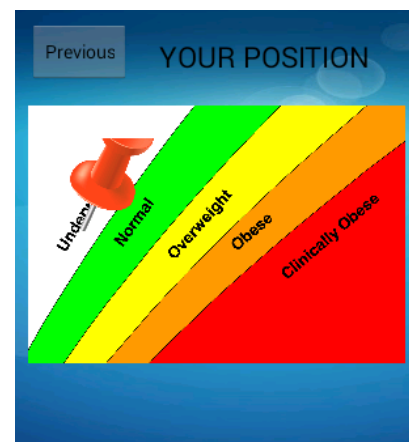

```
26
27        ImageView attach = (ImageView) findViewById(R.id.Img_attach);
28
29        if (BMI2 < 18.5) {
30            attach.scrollBy(40, 20);
31
32        }
33
34        if (BMI2 >= 18.5 && BMI2 < 24.9) {
35            attach.scrollBy(20, 20);
36
37        }
38
39        if (BMI2 >= 24.9 && BMI2 < 29.9) {
40            attach.scrollBy(-20, 20);
41        }
42
43        if (BMI2 >= 29.9 && BMI2 < 34.9) {
44            attach.scrollBy(-40, 20);
45        }
46
47        if (BMI2 >= 34.9) {
48            attach.scrollBy(-90, 20);
49        }
50
```

2 (i) Draw a graphic to display some app related info.
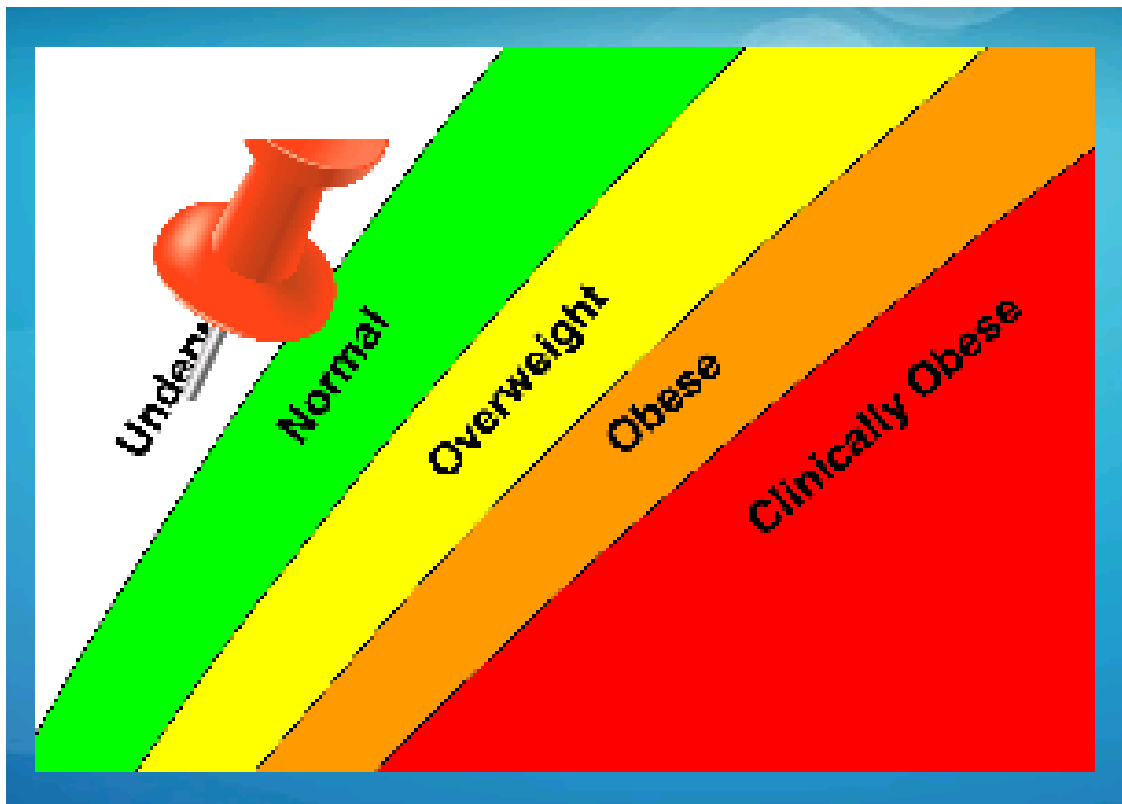
```
12
13    <LinearLayout
14        android:layout_width="wrap_content"
15        android:layout_height="wrap_content"
16        android:layout_centerHorizontal="true"
17        android:orientation="vertical" >
18
19        <ImageView
20            android:id="@+id/Img_chart"
21            android:layout_width="wrap_content"
22            android:layout_height="wrap_content"
23            android:src="@drawable/chart"
24            android:text="TextView" />
25    </LinearLayout>
26
```

1) List of classes

- MainActivity.java : Home Application where the user has to put his information (name, height, weight) and this information are supposed to be register in a database in clicking on "Submit"; he can change the background of the activity.

- SecondActivity.java : Display BMI of the user, and informations about it.

- ThirdActivity.java : Doesn't work actually. This activity was supposed to stock user details in a Database in order to avoid the user to put his details each time he start the application.

- FourthActivity.java : Display a Graphic and a pin that move in function of the bmi value.

2) How has it worked

This application was very cool to create. I mean that the language we use is very easy to understand and to manipulate. Just, I had some problems to put an image over an over image or to put a graphic component over an image.. but finally, thanks to internet, I did it.

3) Future outlook / What could be done differently

For this kind of application, it can be cool if we use PHP/SQL to create an account for the user in order to avoid the user to enter his details each time. Moreover, the user will be able to see the evolution of his BMI each time he uses the application (cf goal of ThirdActivity).
The design of my application could be better and I think I could use more activities to display functionalities (like the background personalize).

Lucas Uzan