

## L'informatica al servizio dell'ambiente

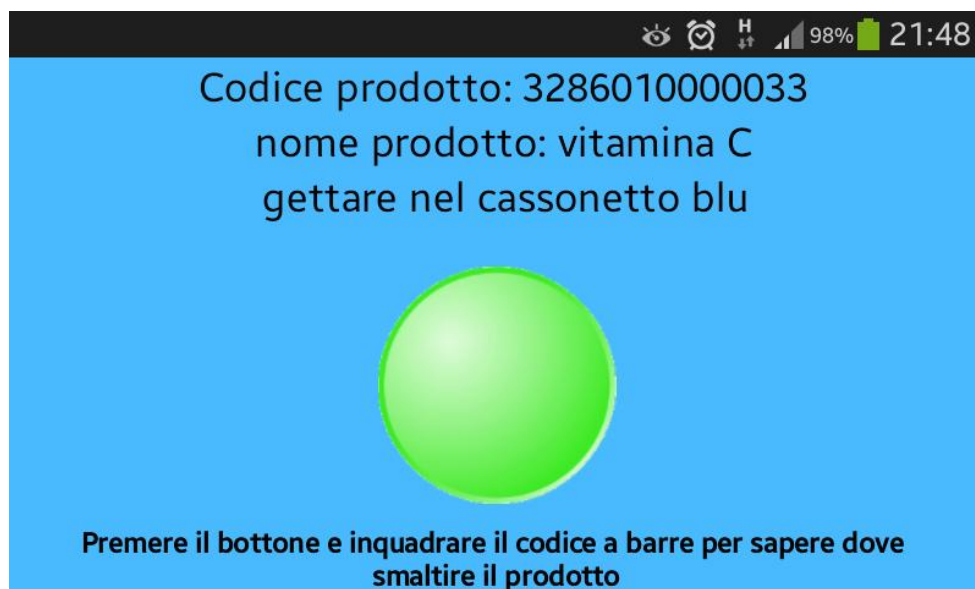


Esemplare è il contributo di Rossano Ercolini che nel libro *Non bruciamo il futuro* propone la possibilità di un riciclaggio totale dei materiali. Nell'aprile del 2013, per il suo impegno come attivista ed educatore, Rossano Ercolini riceve il *Goldman Environmental Prize* e viene invitato alla Casa Bianca dal presidente degli Stati Uniti Barack Obama. Il premio in denaro, 150.000 dollari destinati al proseguimento delle attività, rappresenta la più grande somma corrisposta per l'attivismo ambientale. La Fondazione Goldman ha voluto premiare Rossano Ercolini, poiché, "quando senti parlare dei progetti di edificazione dell'inceneritore nel suo comune, ritenne di avere la responsabilità, come educatore, di proteggere il benessere degli studenti e di informare la comunità in merito ai rischi dell'inceneritore e alle soluzioni per la gestione sostenibile dei rifiuti domestici del paese".

Ercolini possiede un network e un sito dal nome "0 rifiuti" per combattere l'inquinamento in tutte le sue forme.

Proprio come Ercolini (maestro elementare e attivista pro ambiente) ha evitato la costruzione di un impianto di termovalorizzazione nel suo comune (Capannori), tramite la sensibilizzazione al riciclaggio dei rifiuti con strumenti di tipo informatico, così io sono convinto che informatica e ambientalismo possano collaborare anche all'interno di realtà come l'istituto Aldini di Bologna.

A tal fine ho pensato allo sviluppo di un'app Android che, attraverso la lettura dei codici a barre dei prodotti, aiuti l'utente nella separazione degli imballaggi dei generi alimentari consumati nella scuola: buste di caramelle/patatine, lattine di bibite, bottiglie di plastica e contenitori di succhi di frutta.



Screenshot di esempio

Linguaggio utilizzato:	<ul style="list-style-type: none"> <li>• Java per android</li> <li>• Php per server e “json generator”</li> </ul>
Limiti del programma	<ul style="list-style-type: none"> <li>• Ho scelto di utilizzare java principalmente per la maggiore esperienza che ho nel programmare in questo linguaggio.</li> <li>• Il server è stato creato in locale con XAMPP e poi caricato su un sito di hosting gratuito: Altrivista</li> <li>• È possibile scansionare solo un prodotto alla volta</li> <li>• La ricerca è possibile solo tramite codice a barre</li> <li>• L'applicazione è formata da un'unica schermata</li> </ul>

Il sistema si compone di tre parti:

1. Applicazione utente android
2. JSON generator
3. Server

L'intera applicazione android è stata creata con l'ausilio del software Android Studio.

il funzionamento è il seguente:

1. Una volta premuto il tasto centrale verde, viene avviata l'applicazione Barcode Scanner, applicazione esterna creata da google, tramite la quale si ricava il valore numerico del codice a barre
2. Il codice viene passato alla prima applicazione
3. il codice viene aggiunto alla fine di un url preconstituito che richiama, lato server, il software php “JSON generator”:

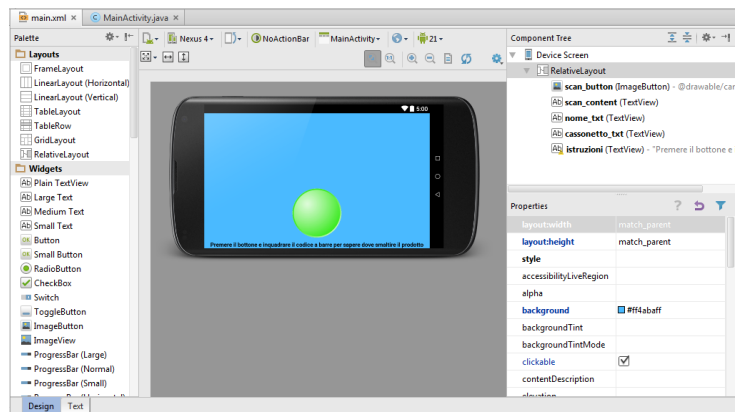
String url = "http://barcode readeraldini.altrivista.org/jsongenerator.php?id="+scanContent;

```

1 <?php
2 mysql_connect("localhost","barcode readeraldini","redpocapt58");
3 mysql_select_db("my_barcode readeraldini");
4
5 $query=mysql_query("select * from prodotto.cassonetto WHERE cassonetto.id_cassonetto=prodotto.cod_cassonetto and prodotto.barcode = " . $_GET['id']");
6 or die(mysql_error());
7 while($e=mysql_fetch_assoc($query))
8     $output[]=$e;
9
10 print(json_encode($output));
11 echo $output;
12
13 mysql_close();
14 ?>

```

4. JSON generator ha il compito di contattare il server e di recuperare i dati richiesti eseguendo una query
5. I dati vengono rispediti all'applicazione android e mostrati all'utente tramite l'interfaccia grafica



Progettazione dell'interfaccia grafica con Android Studio



Barcode Scanner serve solo per leggere il codice a barre grafico e ricavarne il valore numerico, tutte le altre azioni vengono svolte dall'applicazione principale

## Il seguente codice costituisce l'applicazione

```
1 package it.barcode;
2
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.util.Log;
7 import android.view.View;
8 import android.view.View.OnClickListener;
9 import android.widget.ImageButton;
10 import android.widget.TextView;
11 import android.widget.Toast;
12 import com.google.zxing.integration.android.IntentIntegrator;
13 import com.google.zxing.integration.android.IntentResult;
14 import org.apache.http.HttpEntity;
15 import org.apache.http.HttpResponse;
16 import org.apache.http.StatusLine;
17 import org.apache.http.client.HttpClient;
18 import org.apache.http.client.methods.HttpGet;
19 import org.apache.http.impl.client.DefaultHttpClient;
20 import org.json.JSONObject;
21 import java.io.*;
22
23 public class MainActivity extends Activity implements OnClickListener{
24
25     private ImageButton scanBtn;
26     private TextView contentTxt, nomeTxt, cassonettotxt;
27
28     public String readJSONFeed(String URL) { //funzione per connessione a database
29         StringBuilder stringBuilder = new StringBuilder();
30         HttpClient httpClient = new DefaultHttpClient();
31         HttpGet httpGet = new HttpGet(URL);
32         try {
33             HttpResponse response = httpClient.execute(httpGet);
34             StatusLine statusLine = response.getStatusLine();
35             int statusCode = statusLine.getStatusCode();
36             if (statusCode == 200) {
37                 HttpEntity entity = response.getEntity();
38                 InputStream inputStream = entity.getContent();
39                 BufferedReader reader = new BufferedReader(new InputStreamReader(inputStream));
40                 String line;
41                 while ((line = reader.readLine()) != null) {
42                     stringBuilder.append(line);
43                 }
44                 inputStream.close();
45             } else {
46                 Log.d("MainActivity", "Failed to download file");
47             }
48         } catch (Exception e) {
49             Log.d("MainActivity", e.getLocalizedMessage());
50         }
51         return stringBuilder.toString();
52     }
53     /** Called when the activity is first created. */
54     @Override
55     public void onCreate(Bundle savedInstanceState) {
56         super.onCreate(savedInstanceState);
57         setContentView(R.layout.main);
58
59         scanBtn = (ImageButton) findViewById(R.id.scan_button); //testo scan
60         contentTxt = (TextView) findViewById(R.id.scan_content); //valore codice a barre
61         nomeTxt = (TextView) findViewById(R.id.nome_txt); //valore nome prodotto
62         cassonettotxt = (TextView) findViewById(R.id.cassonetto_txt); //valore colore cassonetto
63         scanBtn.setOnClickListener(this);
64     }
65
66     public void onClick(View v) {
67         if(v.getId() == R.id.scan_button){
68             IntentIntegrator scanIntegrator = new IntentIntegrator(this);
69             scanIntegrator.initiateScan();
70         }
71     }
72
73     private String scanContent;
74     @Override
75     public void onActivityResult(int requestCode, int resultCode, Intent intent) { //dati da mostrare all'utente
76         IntentResult scanningResult = IntentIntegrator.parseActivityResult(requestCode, resultCode, intent);
77
78         if (scanningResult != null) {
79             scanContent = scanningResult.getContents();
80             //1'url al database cambia a seconda dell'oggetto scansionato: l'id viene aggiunto in fondo
81             String risultato = readJSONFeed("http://barcode readeraldini.altervista.org/jsongenerator.php?id=" + scanContent);
82             try {
83                 JSONObject jsonObject = new JSONObject(risultato);
84                 contentTxt.setText("Codice prodotto: " + jsonObject.getString("barcode"));
85                 nomeTxt.setText("nome prodotto: " + jsonObject.getString("nome"));
86                 cassonettotxt.setText("colore del cassonetto " + jsonObject.getString("cassonetto"));
87             } catch (Exception e) {
88                 Log.d("MainActivity", Log.getStackTraceString(e));
89             }
90         } else {
91             Toast toast = Toast.makeText(getApplicationContext(), "nessun dato ricevuto!", Toast.LENGTH_SHORT);
92             toast.show(); // in caso di errore, appare un popup (toast) che avverte l'utente che non sono stati ricevuti dati
93         }
94     }
95 }
96
97 }
```