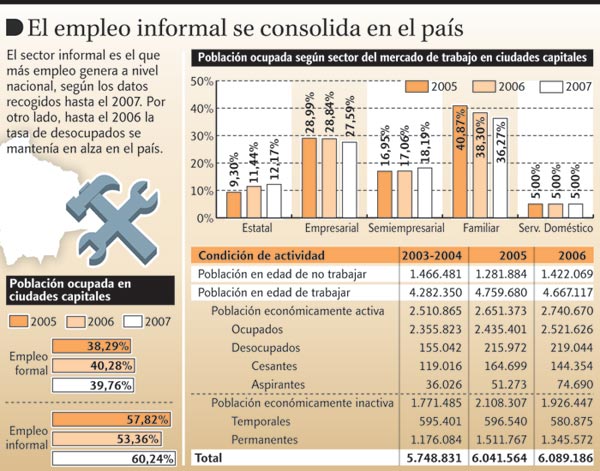
**ANEXO A**

**Tabla estadística del empleo informal en Bolivia hasta el año 2007**



Fuente: http://industriabolivia.blogspot.com/2009/03/el-estado-fomenta-el-empleo-eventual-y.html

**ANEXO B**

**Sistema de puntaje de las tablas de riegos**



Fuente: Elaboración propia

**ANEXO C**

**Servicios web**

















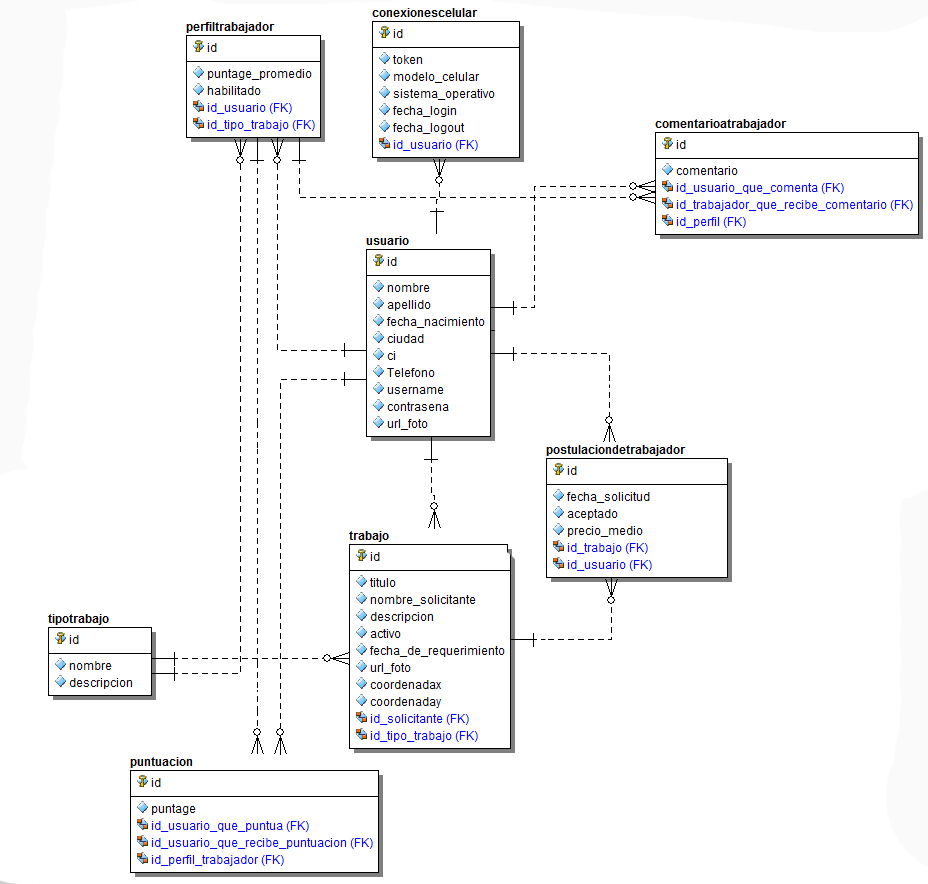




Fuente: Elaboración propia

**ANEXO D**

**Diseño de la base de datos**



Fuente: Elaboración propia

**ANEXO E**

**Diccionario de datos**





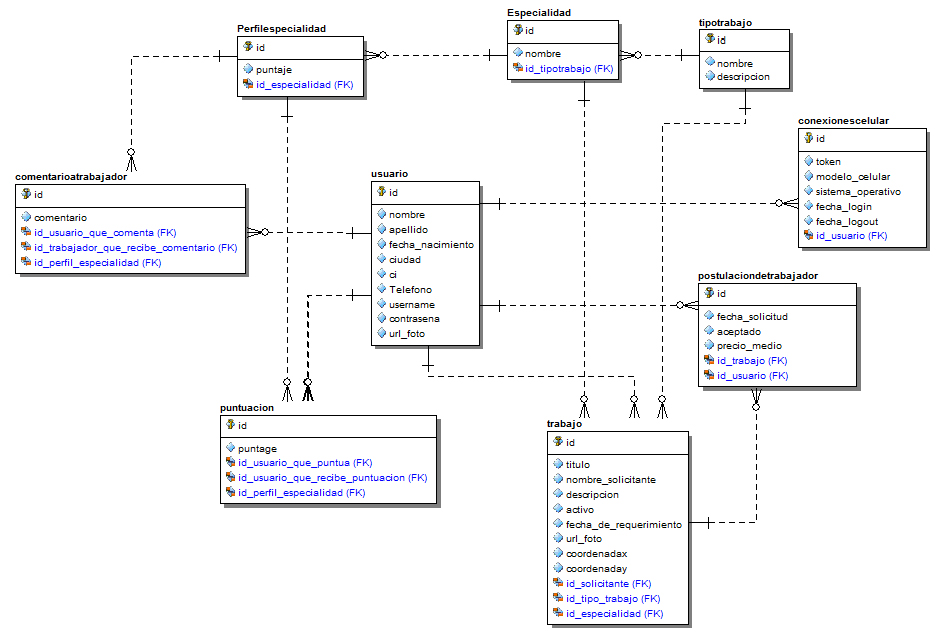




Fuente: Elaboración propia

**ANEXO F**

**Diseño de la base de datos para especialidades**



Fuente: Elaboración propia

**ANEXO G**

**Diccionario de datos para especialidades**









Fuente: Elaboración propia

**ANEXO H**

**Código para envió de información con *JSON* a servidor web**

private class NetworkOperationAsyncTask extends AsyncTask<Void, Void, String> {

@Override

protected String doInBackground(Void... params) {

// This is my URL: http://dipandroid-ucb.herokuapp.com/work\_posts.json

HttpURLConnection urlConnection = null; // HttpURLConnection Object

BufferedReader reader = null; // A BufferedReader in order to read the data as a file

Uri buildUri = Uri.parse("http://192.168.56.1:3000").buildUpon() // Build the URL using the Uri class

.appendPath("obtenerelemento.json").build();

try {

URL url = new URL(buildUri.toString()); // Create a new URL

Log.d(LOG\_TAG, buildUri.toString());

urlConnection = (HttpURLConnection)url.openConnection();

urlConnection.setRequestMethod("POST");

urlConnection.addRequestProperty("Content-Type", "application/json");

urlConnection.setDoOutput(true);

urlConnection.connect();

JSONObject json = new JSONObject();

json.put("id", "1");

DataOutputStream output = new DataOutputStream(urlConnection.getOutputStream());

Log.d(LOG\_TAG, json.toString());

output.write(json.toString().getBytes());

output.flush();

StringBuffer buffer = new StringBuffer();

if (urlConnection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

// OK

Log.d(LOG\_TAG, "RECIBIO CORRECTAMENTE!!!!!!!!!!!!!!");

InputStream inputStream = urlConnection.getInputStream();

reader = new BufferedReader(new InputStreamReader(inputStream));

// Save the data in a String

String line;

while ((line = reader.readLine()) != null) {

buffer.append(line).append("\n");

}

} else {

Log.d(LOG\_TAG, "NO RECIBIO NADA, NI MIERDA!!!!!");

Log.d(LOG\_TAG, Integer.toString(urlConnection.getResponseCode()));

Log.d(LOG\_TAG, urlConnection.getResponseMessage());

// Server returned HTTP error code.

}

output.close();

return buffer.toString();

} catch (IOException ex) {

Log.e(LOG\_TAG, ex.getMessage());

} catch (JSONException e) {

e.printStackTrace();

} finally {

try {

if (reader != null) {

reader.close();

}

if (urlConnection != null) {

urlConnection.disconnect();

}

} catch (IOException e) {

Log.e(LOG\_TAG, e.getMessage());

}

}

return "";

}

@Override

protected void onPostExecute(String json) {

try {

JSONArray array = new JSONArray(json);

System.out.println(array);

JSONObject jobPostJSON = array.getJSONObject(0);

System.out.println(jobPostJSON);

System.out.println(jobPostJSON.getString("nombre"));

//textonombre.setText(jobPostJSON.getString("nombre"));

//textonombre.setText(jobPostJSON.getString("apellido"));

} catch (JSONException ex) {

Log.e(LOG\_TAG, ex.getMessage());

}

}

}

Fuente: developer.android.com