



Universidad Nacional José María Arguedas

*Identidad y Excelencia para el Trabajo Productivo y el Desarrollo*



## MANUAL DE PROGRAMADOR

### TEMA:

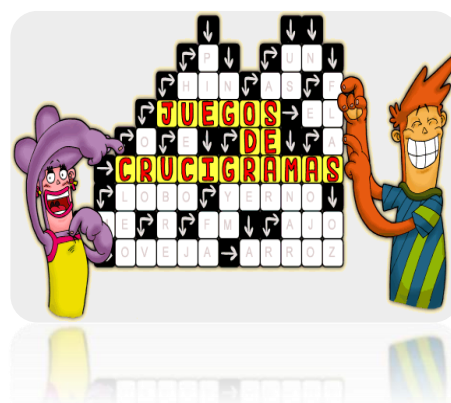
- Aplicación Móvil - CRUCIGRAMA

### DOCENTE:

- Ing. Iván Soria Solís.

### ESTUDIANTES:

- ❖ Apahuasco Saccaco Eber Jesús
- ❖ Escalante Coronel Diego Milker



# Andahuaylas – Apurímac

## 2015

## 1. Descargar los siguientes programas.

Los siguientes programas son lo esencial para poder desarrollar y/o mantener nuestra aplicación que son los siguientes:

- Como primero **Java Development Kit (jdk)**.
- Android Studio.
  - ✓ SDK según emulador o móvil lo requiera.
- Framework AndEngine

### ¿Cómo y de dónde descargar?

Gracias al avance tecnológico la descarga se realiza con mucha facilidad hoy en día, con tan simplemente contar con internet de banda ancha adecuado. Ya case todo está solucionado; Como ya primero mencionado descargar **Java Development Kit (jdk)**, nos vamos a esta página web <http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>.

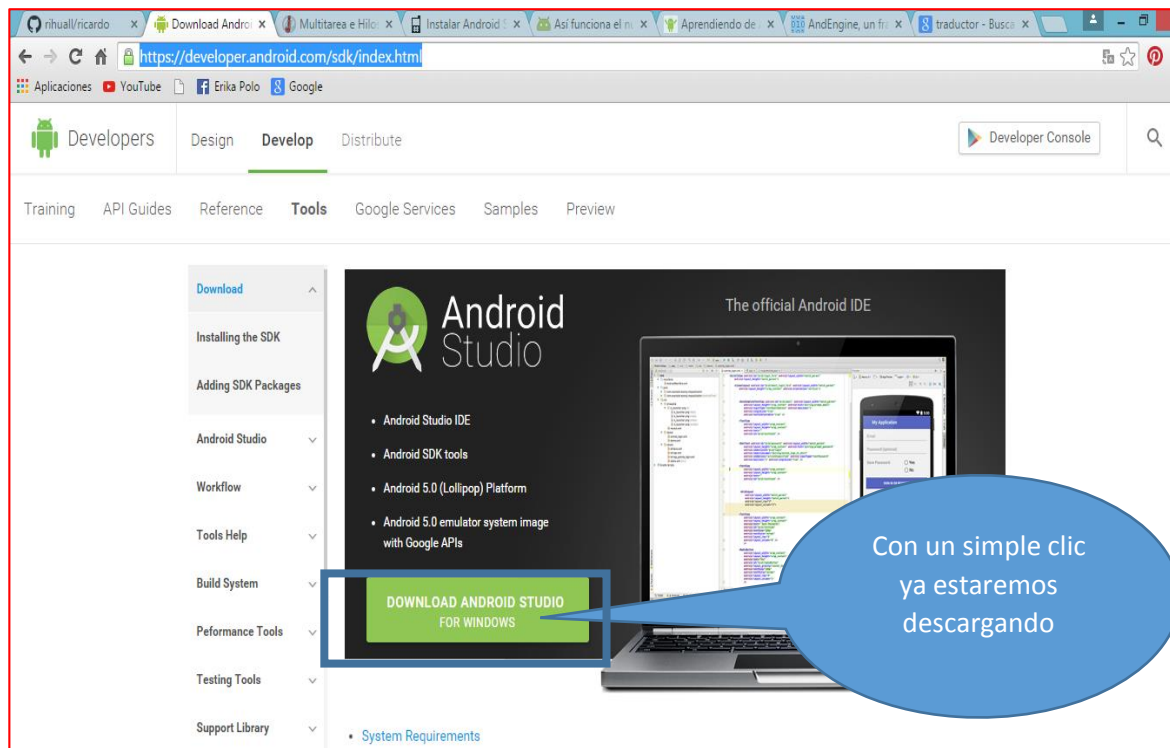


Con un simple clic  
ya estaremos  
descargando



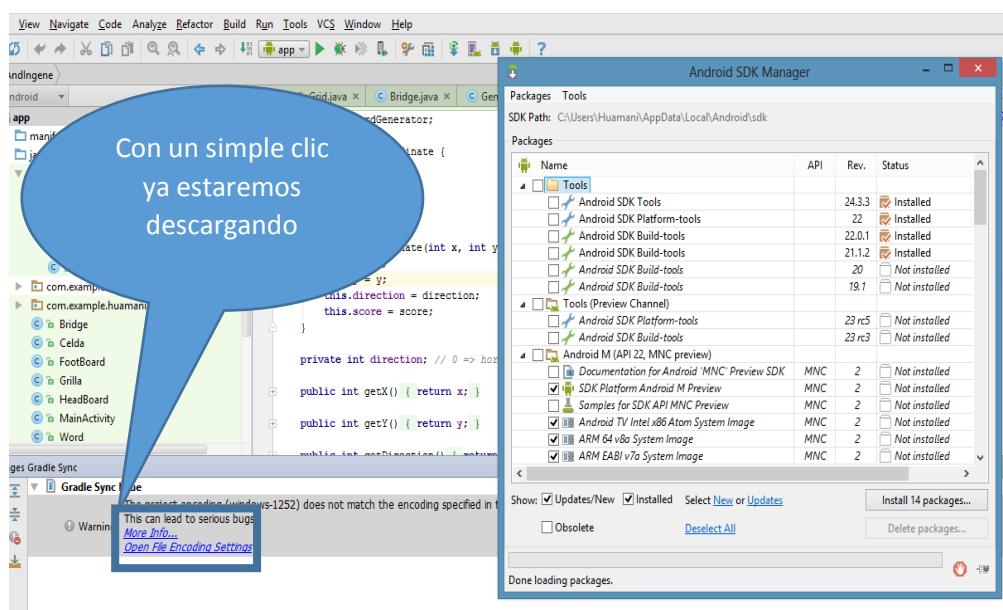
Para el siguiente:

Ya para tal no vamos a esta página web <https://developer.android.com/sdk/index.html>, en el cual podremos descargar.



Descargaremos **kit de desarrollo de software (sdk)** ya sea para el emulador o para su móvil con la que cuenta Ud.

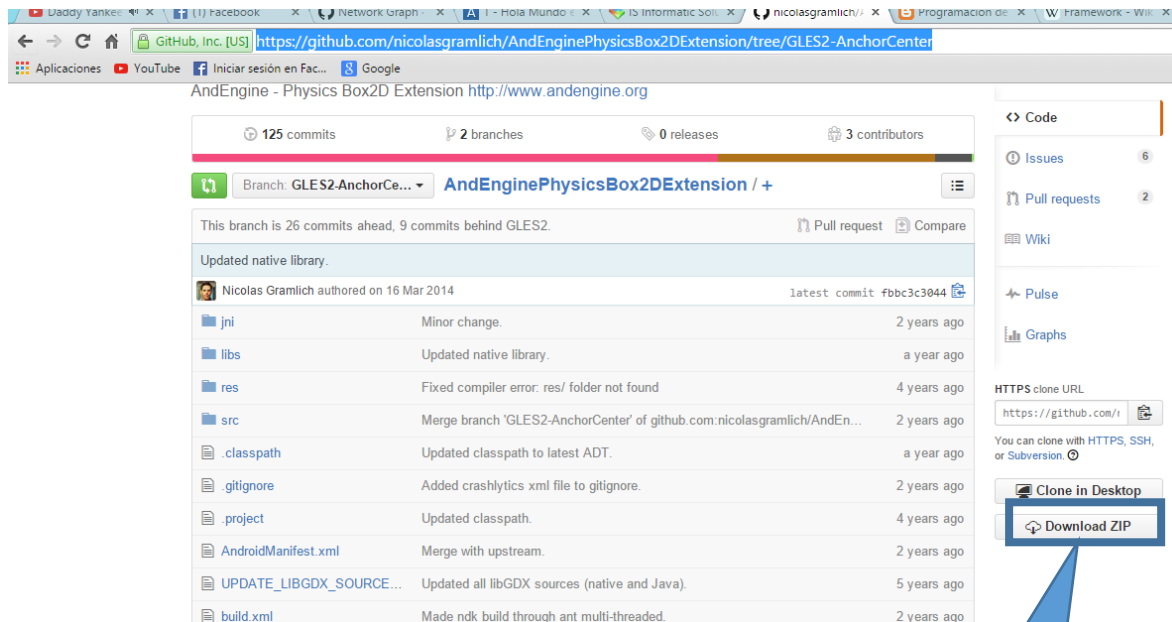
Este caso se realiza dentro de Android Studio





Ahora para descargar AndEngine nos vamos a este página web:

<https://github.com/nicolasgramlich/AndEnginePhysicsBox2DExtension/tree/GLES2-AnchorCenter>



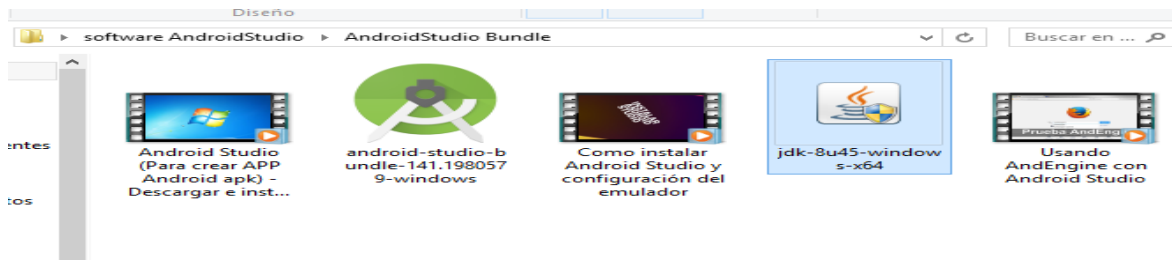
Con un simple clic ya estaremos descargando

## 2. La instalación.

Ejecutar las descargas que hicimos en el principio los formatos (.exe).

- Ejecutamos primero el jdk de java.
- Luego ejecutamos Android Studio.
  - Ya abriendo el Android studio copiamos el framework.

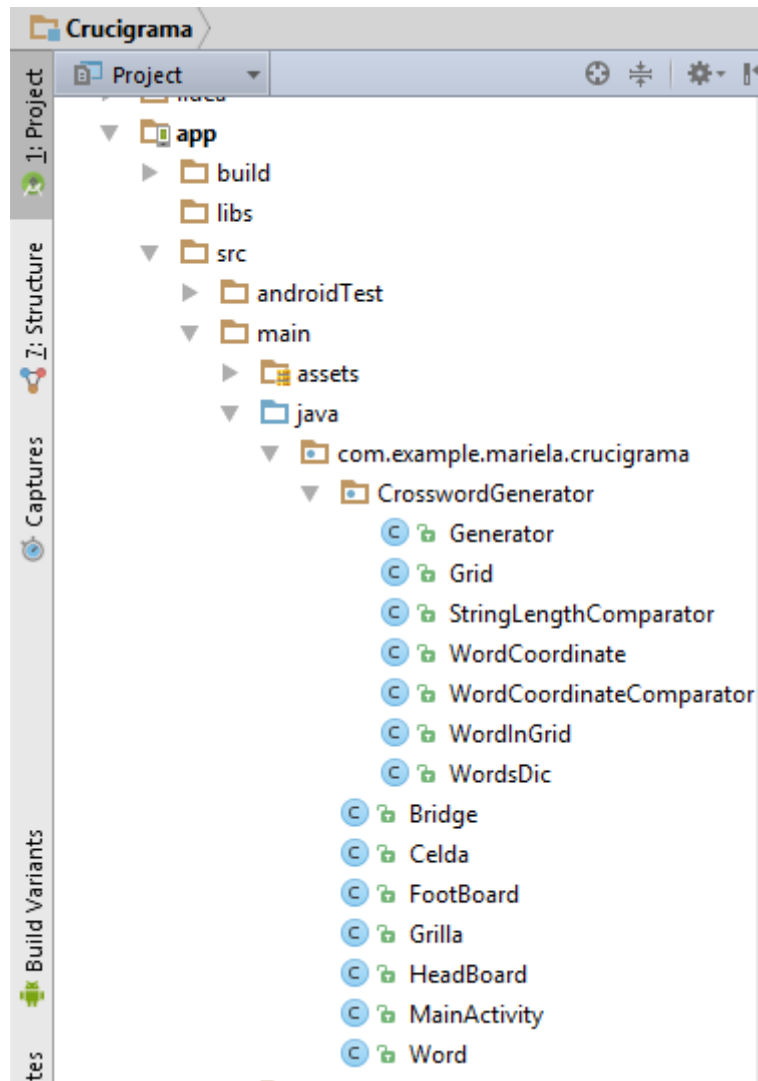
Nota: guiarse con los respectivos videos tutoriales el url se encuentra el artículo en **referencias**.





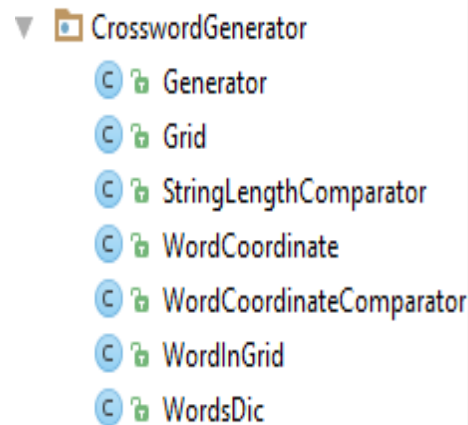
### 3. Proyecto crucigrama

Pasaremos a explicar la estructura general de nuestro juego “crucigrama” aquí una vista panorámica global



## 3.1 El algoritmo principal.

La parte más fundamental de este proyecto, mediante ella se genera toda la crucigrama son clases que interactúan mutuamente entre ellas.



La clase generator es el corazón de nuestro aplicativo.



```
package com.example.mariela.crucigrama.CrosswordGenerator;

import java.util.*;

public class Generator {

    private Grid grid;
    private List<String> words;

    //private Hashtable<Integer, List<String>> horizontalAnnex;
    //private Hashtable<Integer, List<String>> verticalAnnex;
    //private Hashtable<Integer, String> horizontalAnnex;
    //private Hashtable<Integer, String> verticalAnnex;
    //private Hashtable<Integer, WordInGrid> horizontalAnnex;
    private HashMap<Integer, WordInGrid> horizontalAnnex;
    private HashMap<Integer, WordInGrid> verticalAnnex;

    private Random rnd;

    public Generator(Grid grid, List<String> words) {...}

    private void orderWords() {...}

    public int generate() {...}















    private Object[] pickupWord(List<String> remainingWords) {...}

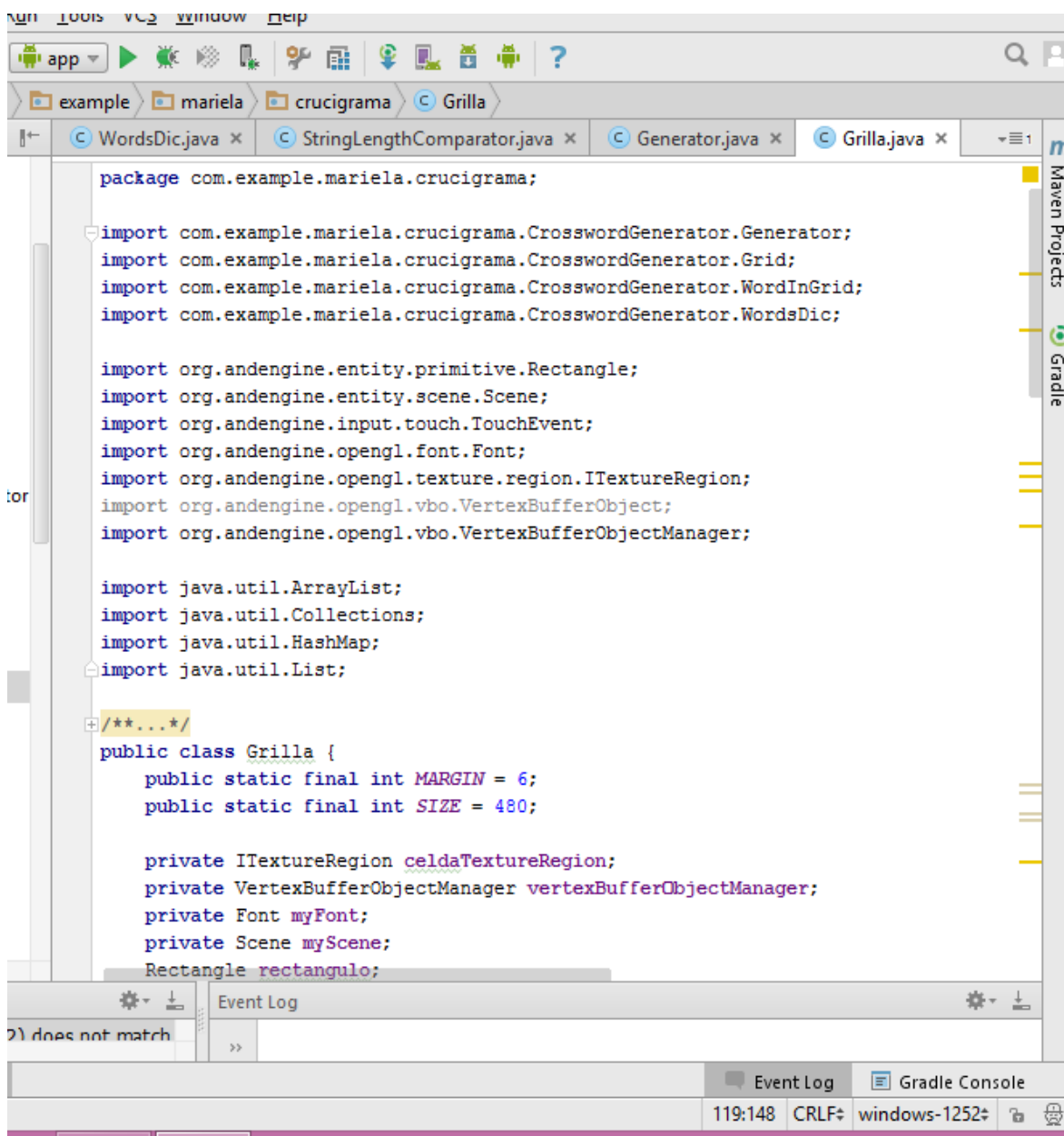
    private List<WordCoordinate> getPossibleCoordinates(String word) {...}

    private int checkScore(String word, WordCoordinate coordinate) {...}
```

## 3.2 interfaz de del juego de crucigrama

Es el código del interfaz con la cual el usuario podrá interactuar.

-   Bridge
-   Celda
-   FootBoard
-   Grilla
-   HeadBoard
-   MainActivity
-   Word



```
package com.example.mariela.crucigrama;

import com.example.mariela.crucigrama.CrosswordGenerator.Generator;
import com.example.mariela.crucigrama.CrosswordGenerator.Grid;
import com.example.mariela.crucigrama.CrosswordGenerator.WordInGrid;
import com.example.mariela.crucigrama.CrosswordGenerator.WordsDic;

import org.andengine.entity.primitive.Rectangle;
import org.andengine.entity.scene.Scene;
import org.andengine.input.touch.TouchEvent;
import org.andengine.opengl.font.Font;
import org.andengine.opengl.texture.region.ITextureRegion;
import org.andengine.opengl.vbo.VertexBufferObject;
import org.andengine.opengl.vbo.VertexBufferObjectManager;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;

/**...*/
public class Grilla {
    public static final int MARGIN = 6;
    public static final int SIZE = 480;

    private ITextureRegion celdaTextureRegion;
    private VertexBufferObjectManager vertexBufferObjectManager;
    private Font myFont;
    private Scene myScene;
    Rectangle rectangulo;
```





## 3.3 framework AndEngine

