# PORTADA

Instituto Tecnológico de la GAM.

Desarrollo de aplicaciones móviles.

Tutor de la materia:

Jorge Iván Rivalcoba

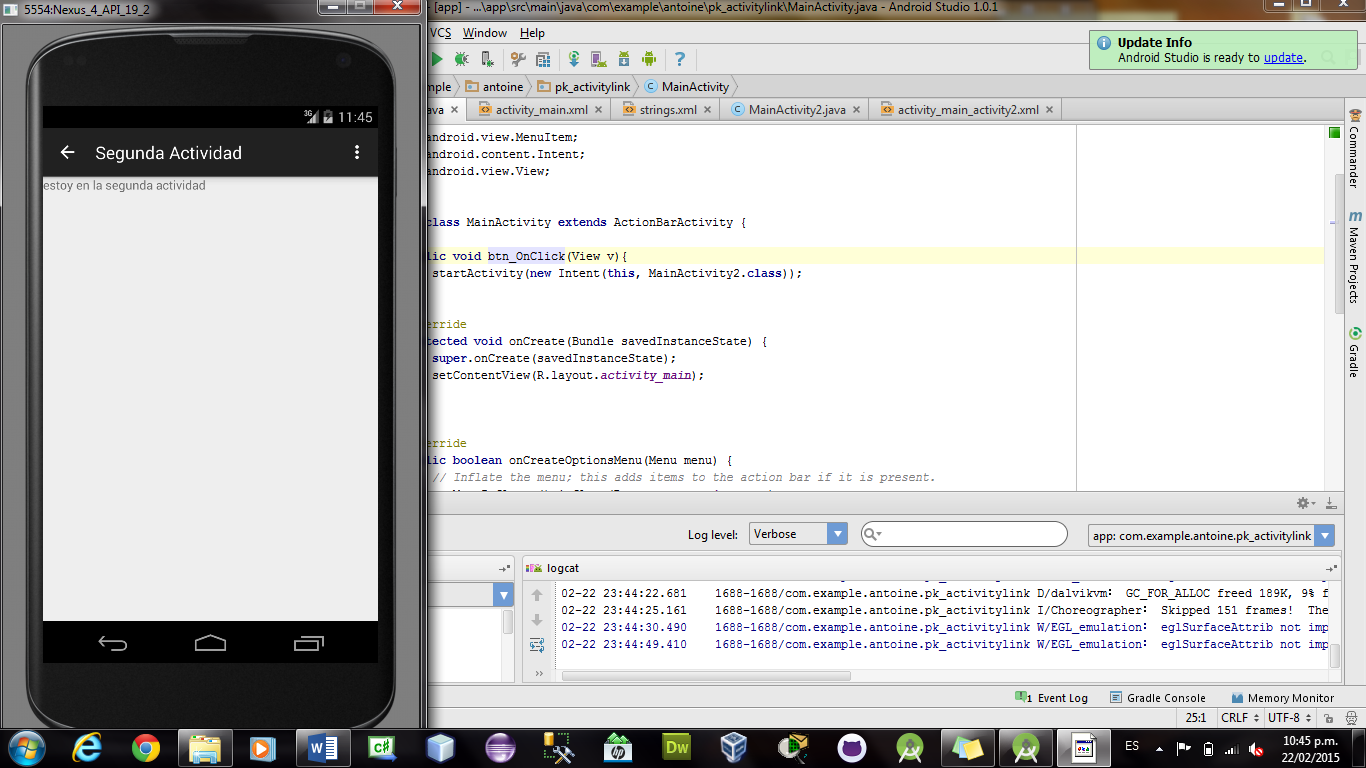
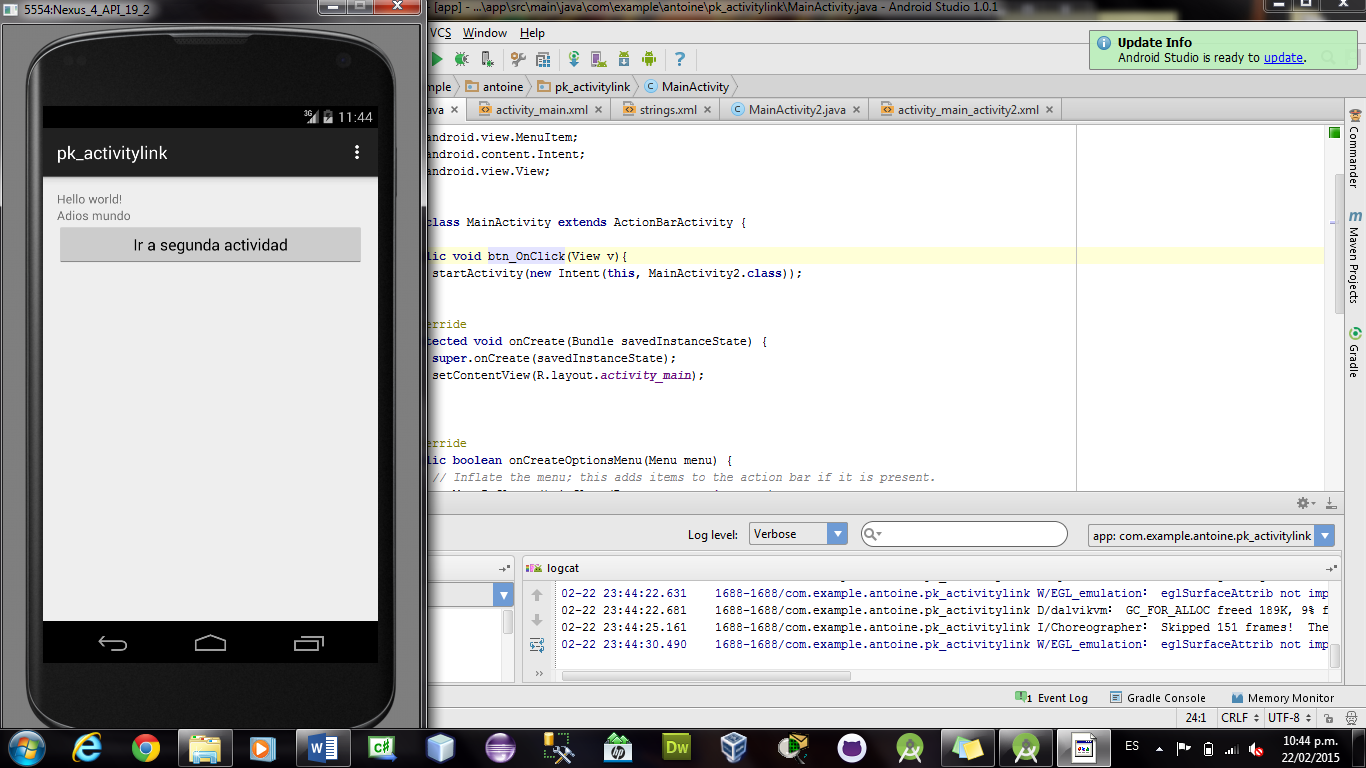
LUIS ANTONIO ROA J. FRIAS 121130033

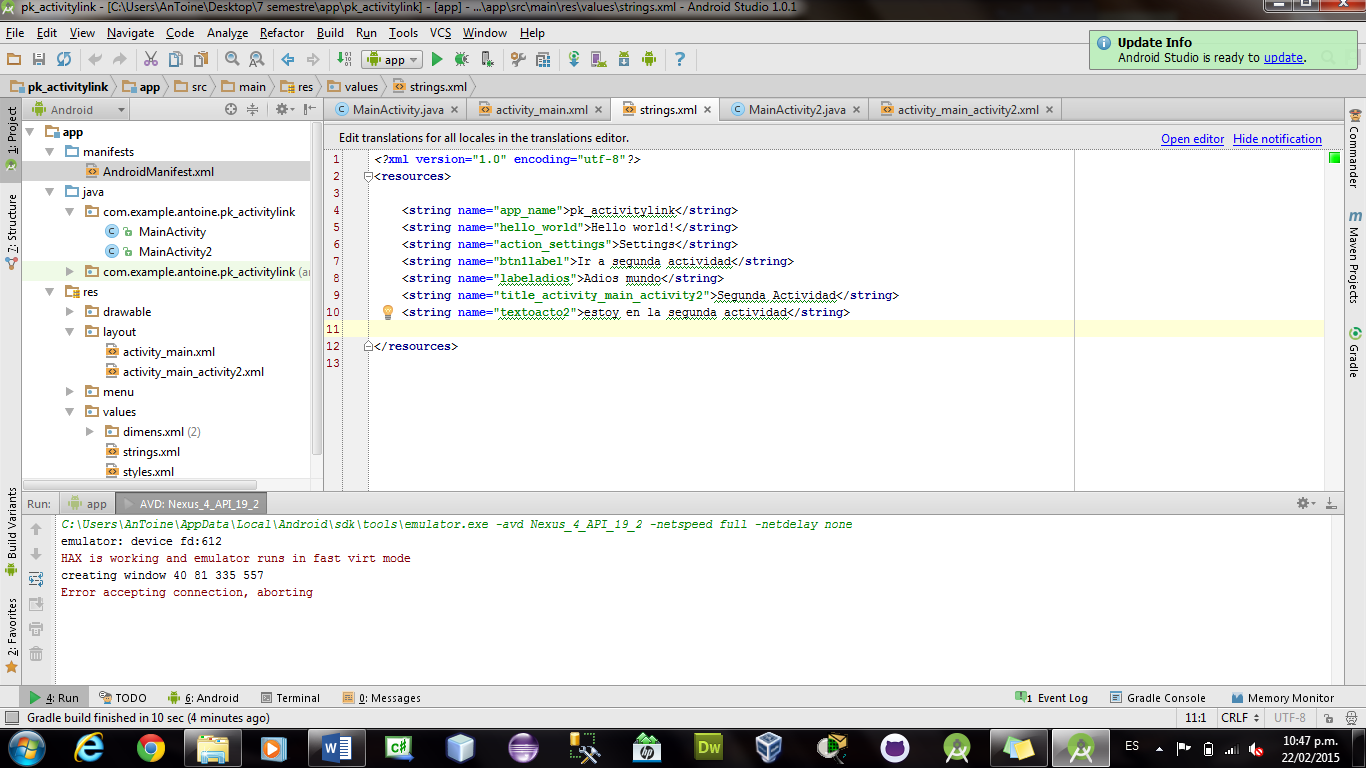
**Introducción.**

Esta una unidad vimos los INTENT tanto como implícito y explicito, generar vistas en los XM y los recursos de STRINGS para nombre los respectivos TEXTVIEW, al igual vimos el RESULTFROM que es la respuesta que recibe y al regresar a la actividad padre devuelve parámetros de un método que hemos predefinido. Al igual el INTENT-COLISSION que es mandar llamar dos actividades con un mismo filtro.

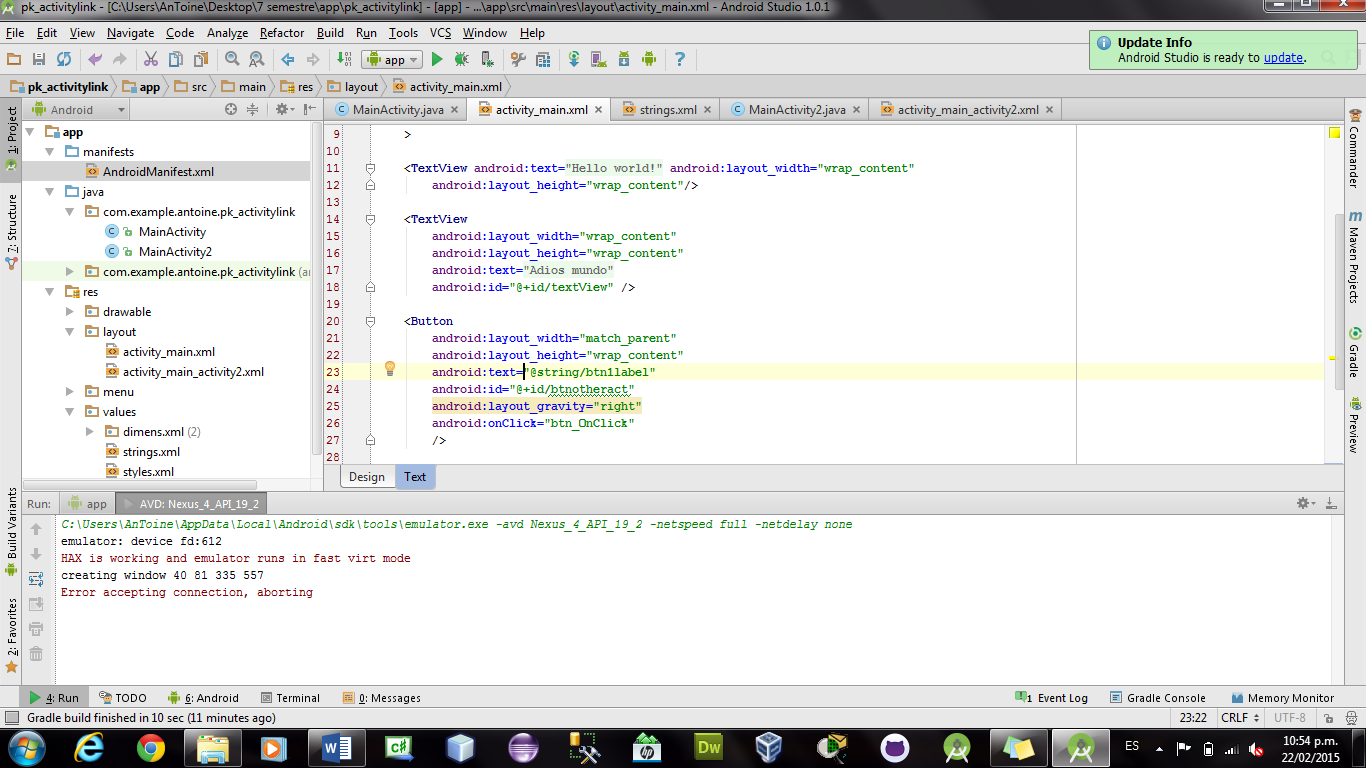
ACTIVITY LINK INTENT EXPLICITO y RECURSOS.

Se crea el interfaz en la primera actividad se agrega dos TEXTVIEW y sus respectivos recursos @STRING/helloworld que viene ya predifinido y uno que se creara en la carpeta values STRINGS donde se inserta una linea de codigo mostrado en la siguiente pagina o al igual podemos hardcodear, despues se agrega un boton al igual generando su recurso, en la segunda actividad, con el nombre de SECONDACTIVITY se crea un interfaze de un solo TEXTVIEW con su respectivo recurso.



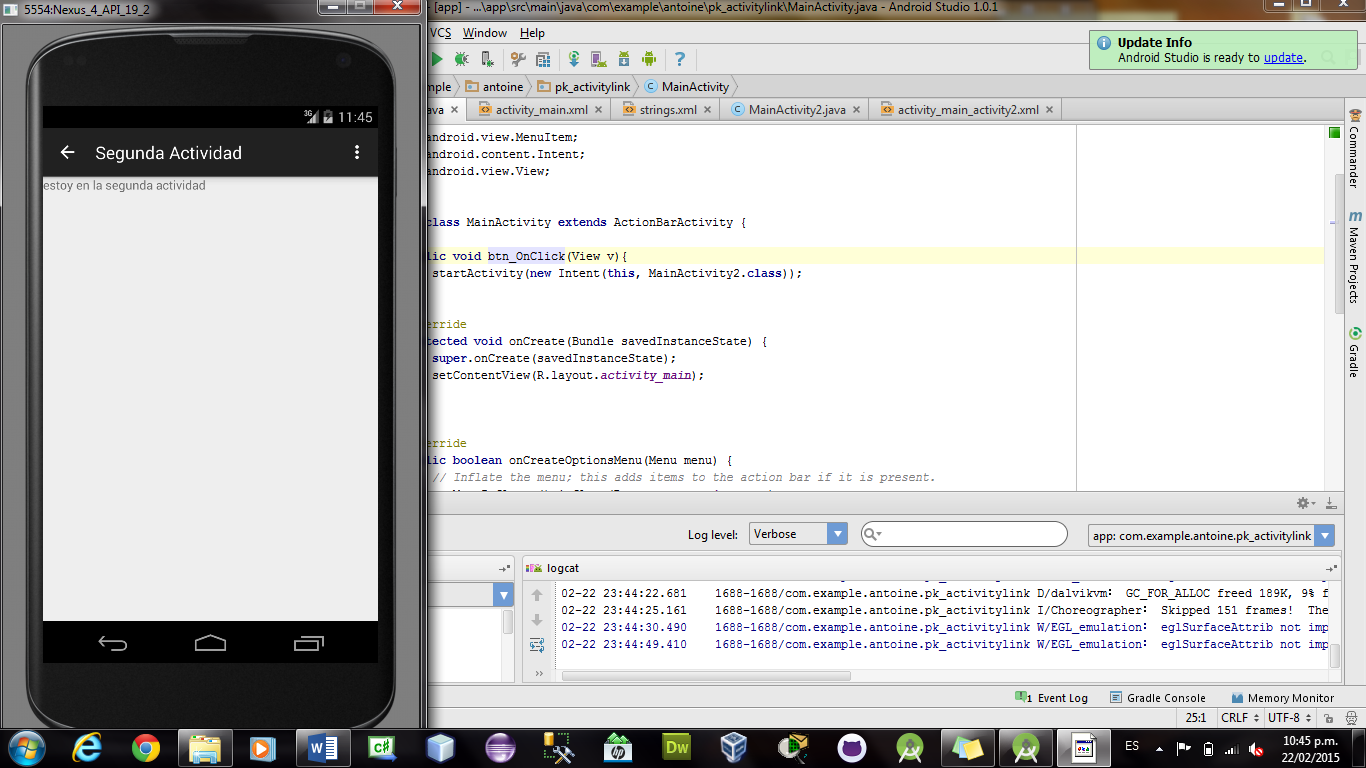
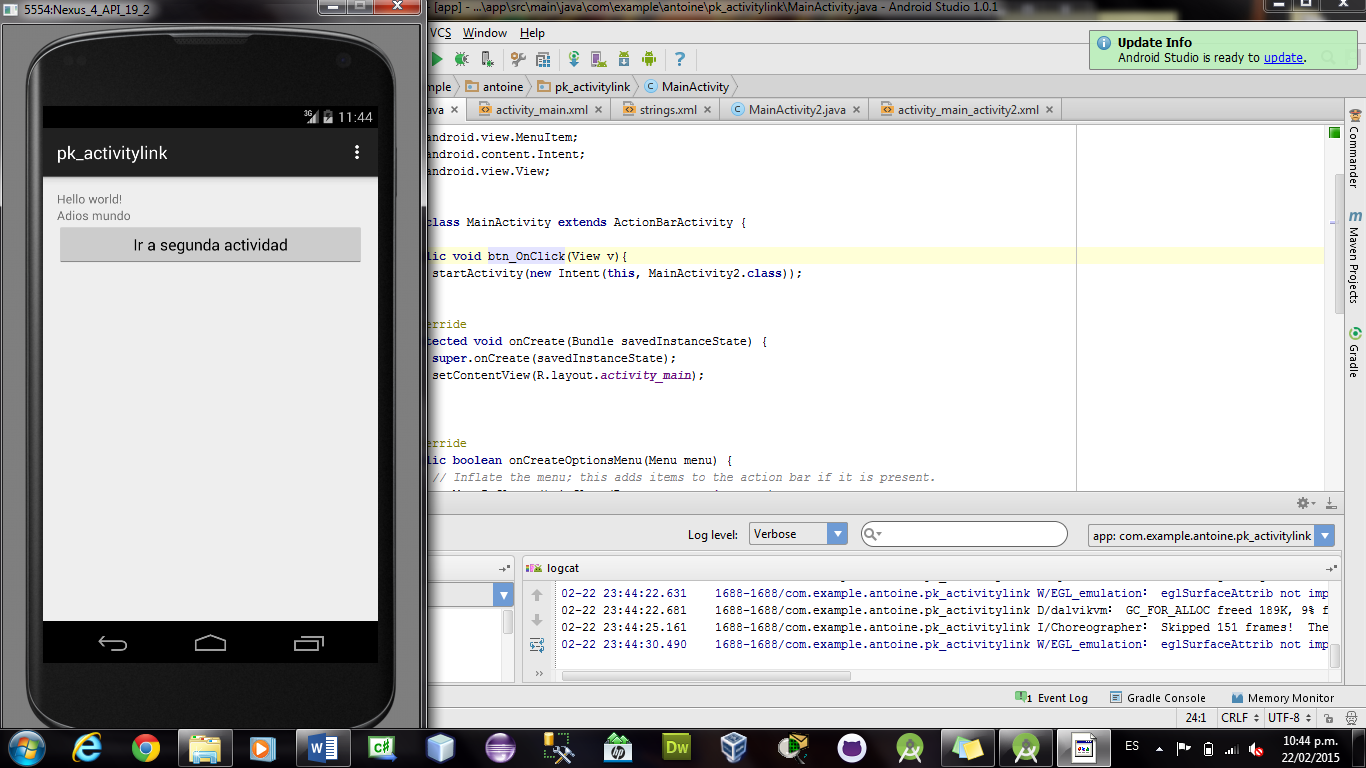
En esta ventana se visualiza mas, que tenemos dos actividades asi como sus XML y sus clases JAVA, en la parte de values STRINGS se genera los recursos de tipo cadena de caracteres que son invocados por los objetos con android:text=@String/ que manda el texto guardado. Y este texto o recurso que se genera para el texto su linea es como se muestra en la foto, y como se inserta el texto, la parte verde es el nombre de como mandaremos a llamar dicho recurso de caracteres.   


En la siguiente foto, vemos como mandamos a llamar el recurso en este ejemplo el del boton y su recurso se llama btn1label, el cual dice ir segunda actividad, y tambien vemos android:onclick” “ aquí es donde al dar click al boton mandara a llamar un metodo de la clase java el cual es el de la siguiente foto.



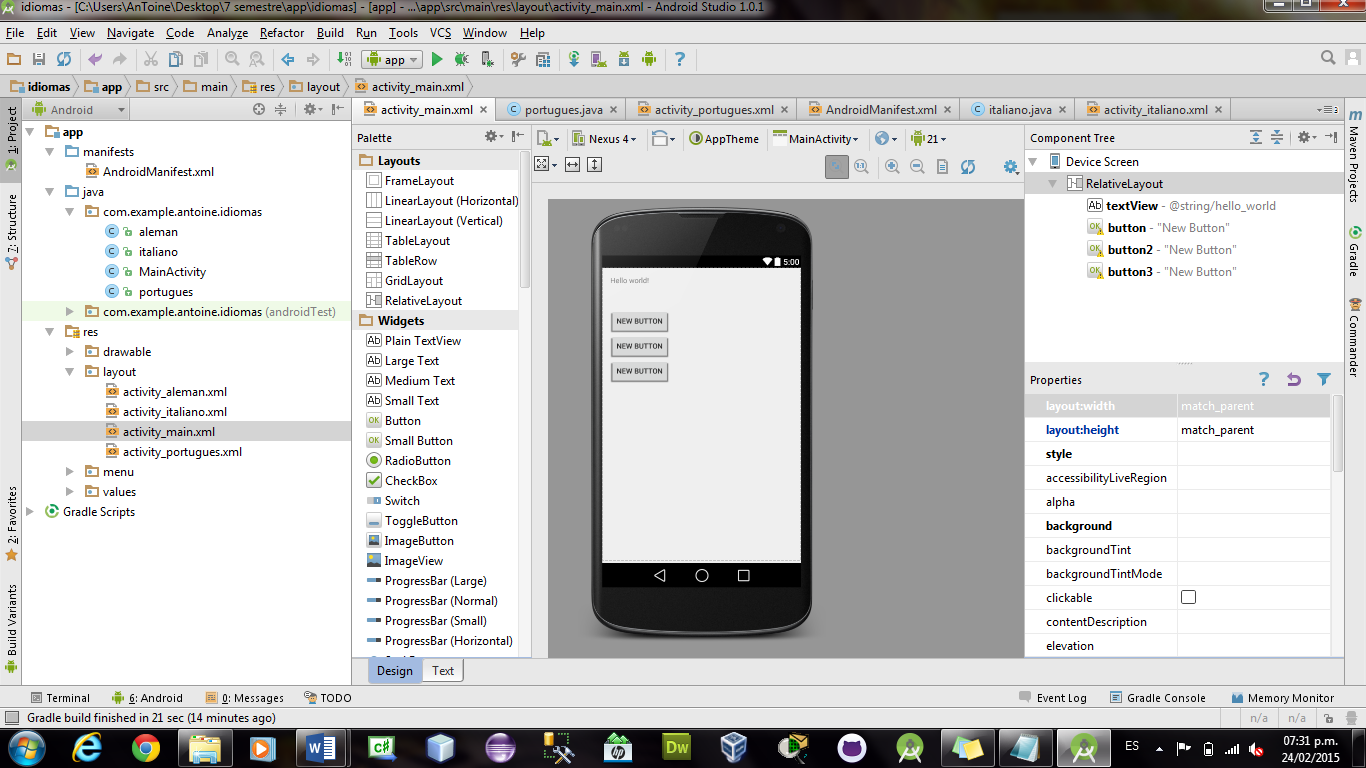
Con este metodo es de forma explicita ya que generamos un INTENT que es un intento que manda llamar de la actividad actual a la actividad nombrada, en este caso a second activity que es nuestra segunda actividad al ir. Al mandar a llamar este metodo llamado btn\_onclick hara dicha funcion de cambiar de actividad

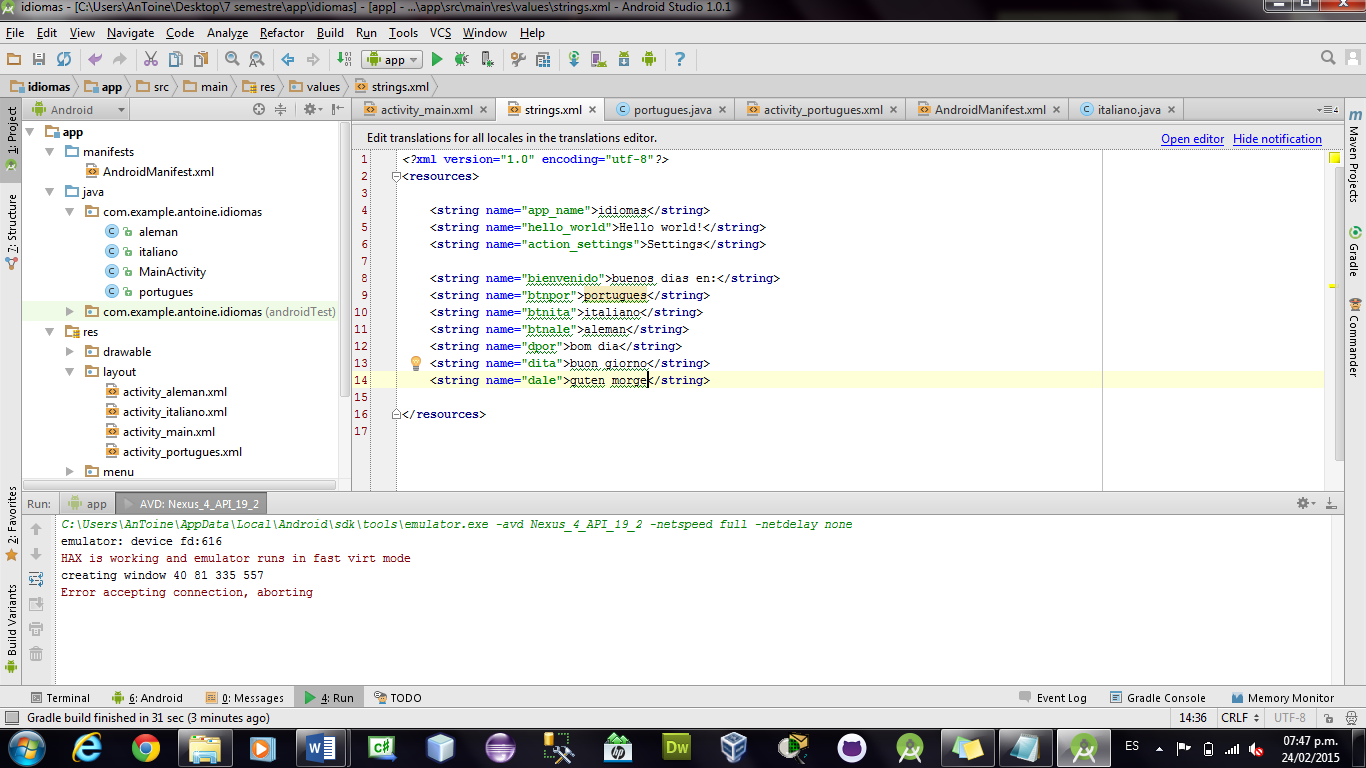


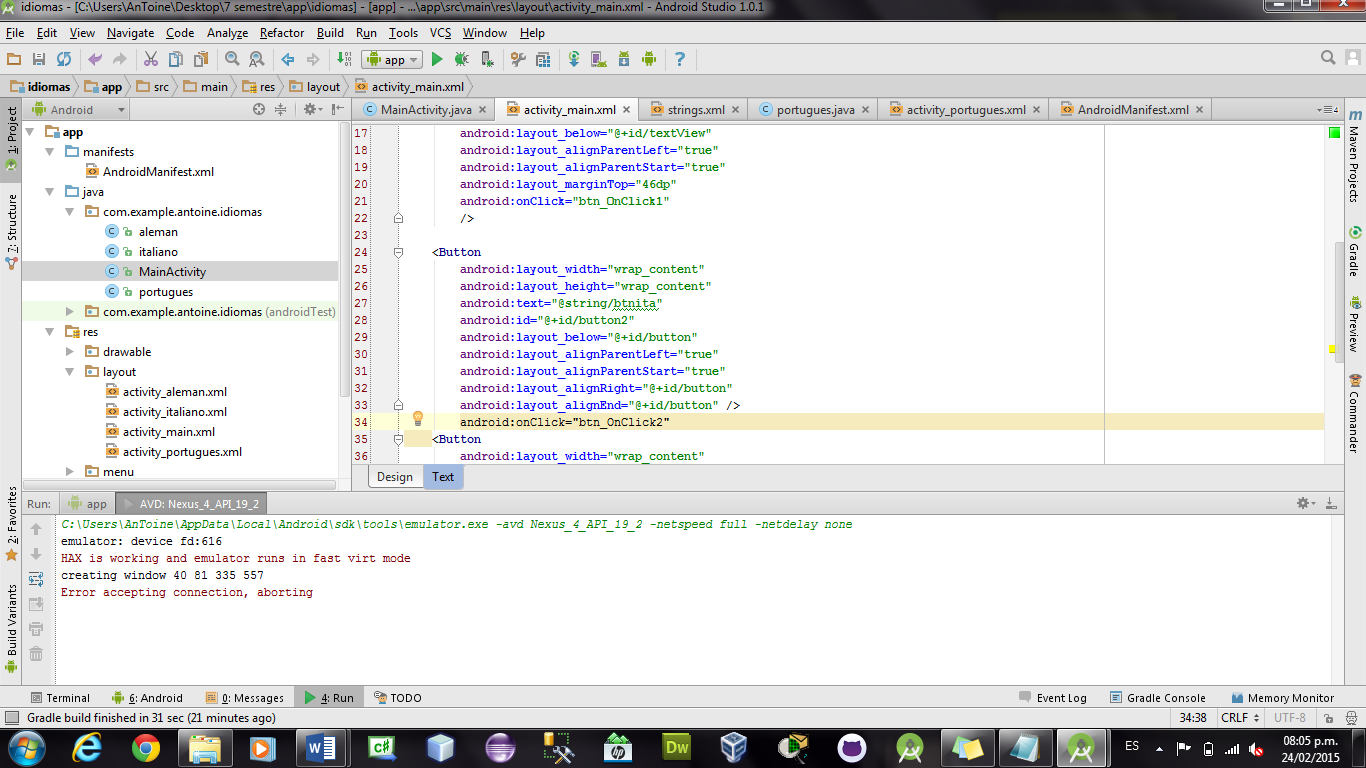
Así es el resultado como se vería.  


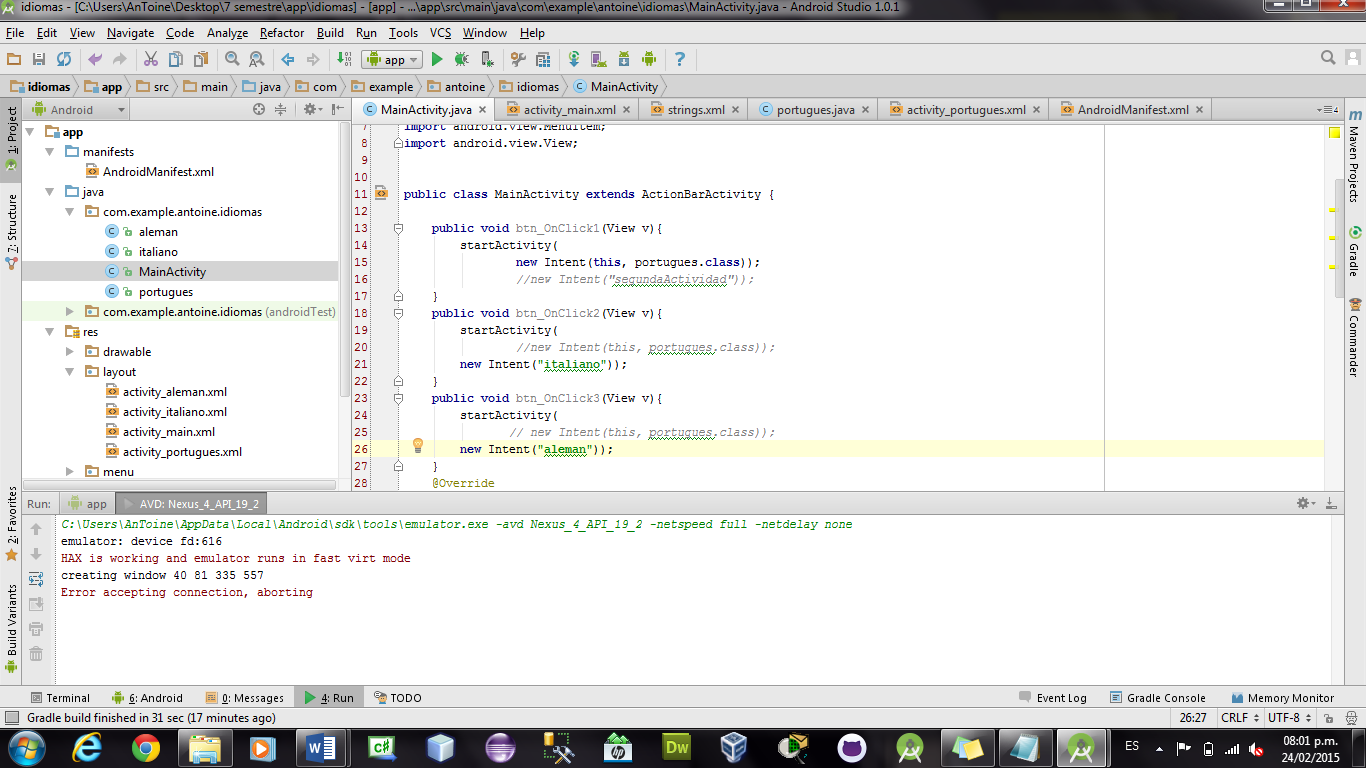
IDIOMAS (TRADUCTOR) con INTENT IMPLICITO

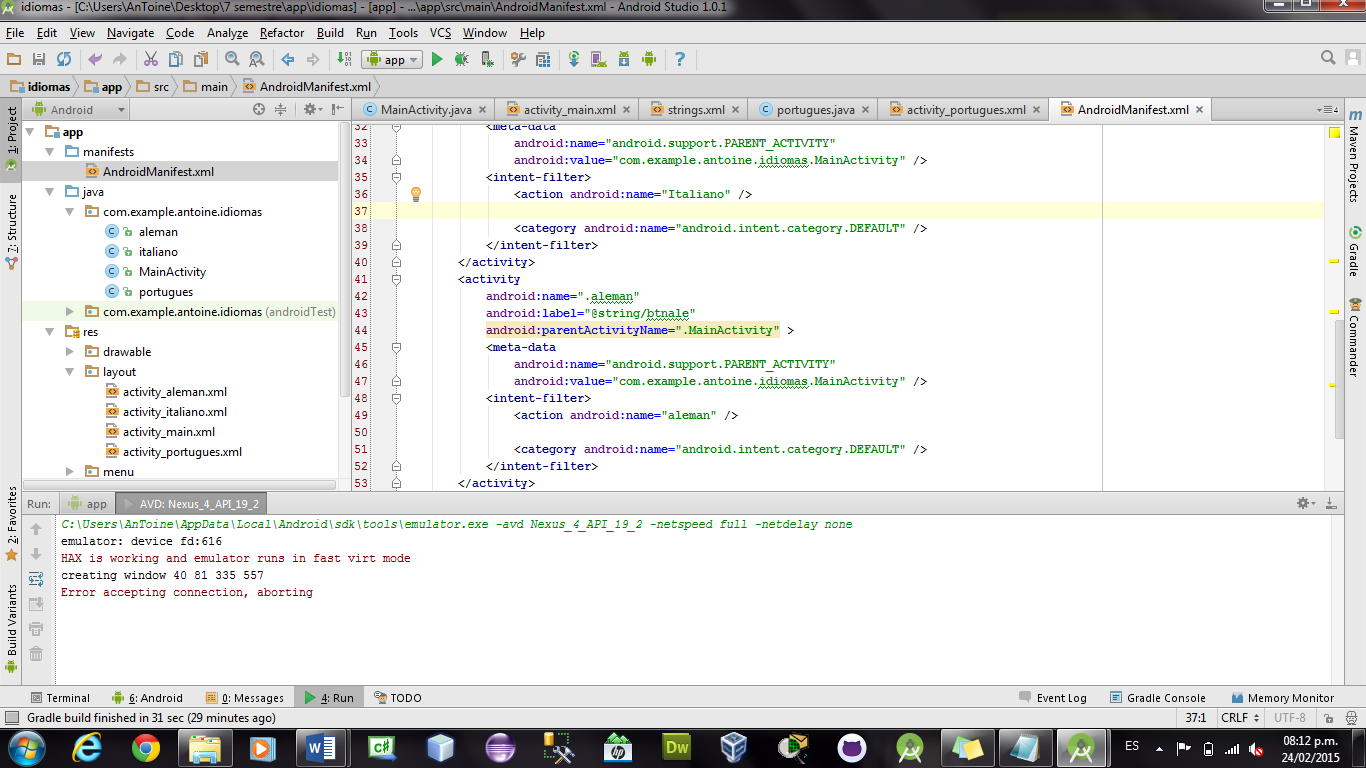
Crearemos un proyecto de nombre IDIOMAS, y en el crearemos tres actividades aparte de la actividad principal, una es alemán, otra italiano y ultima portugués, en el interfaz o XML de la actividad principal vamos agregar aparte del TEXTVIEW tres botones que nos permitirán ir a otras actividades con la traducción de BUENOS DIAS a su respectivo idioma, el primer botón portugués lo haremos de forma explícita ya aprendida anteriormente y los demás de forma implícita que posteriormente se enseñara su funcionamiento.



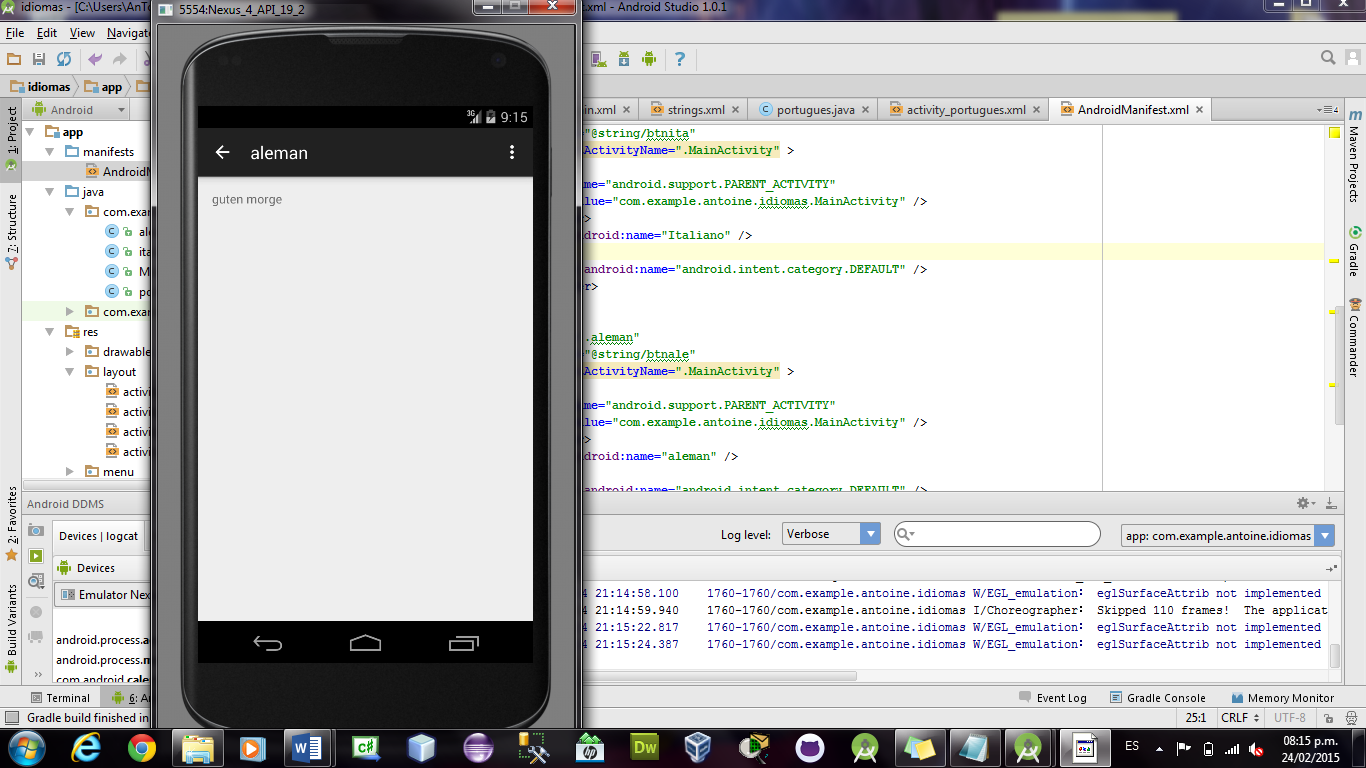
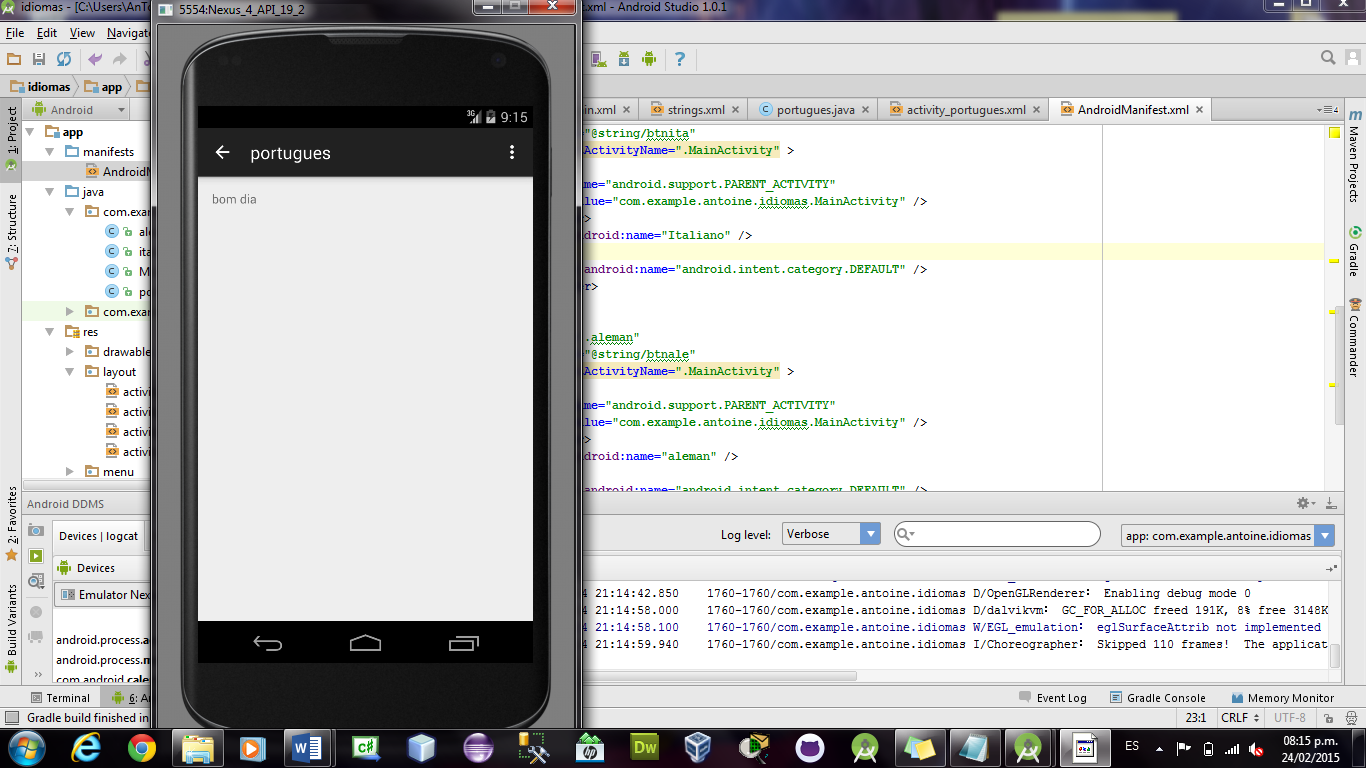
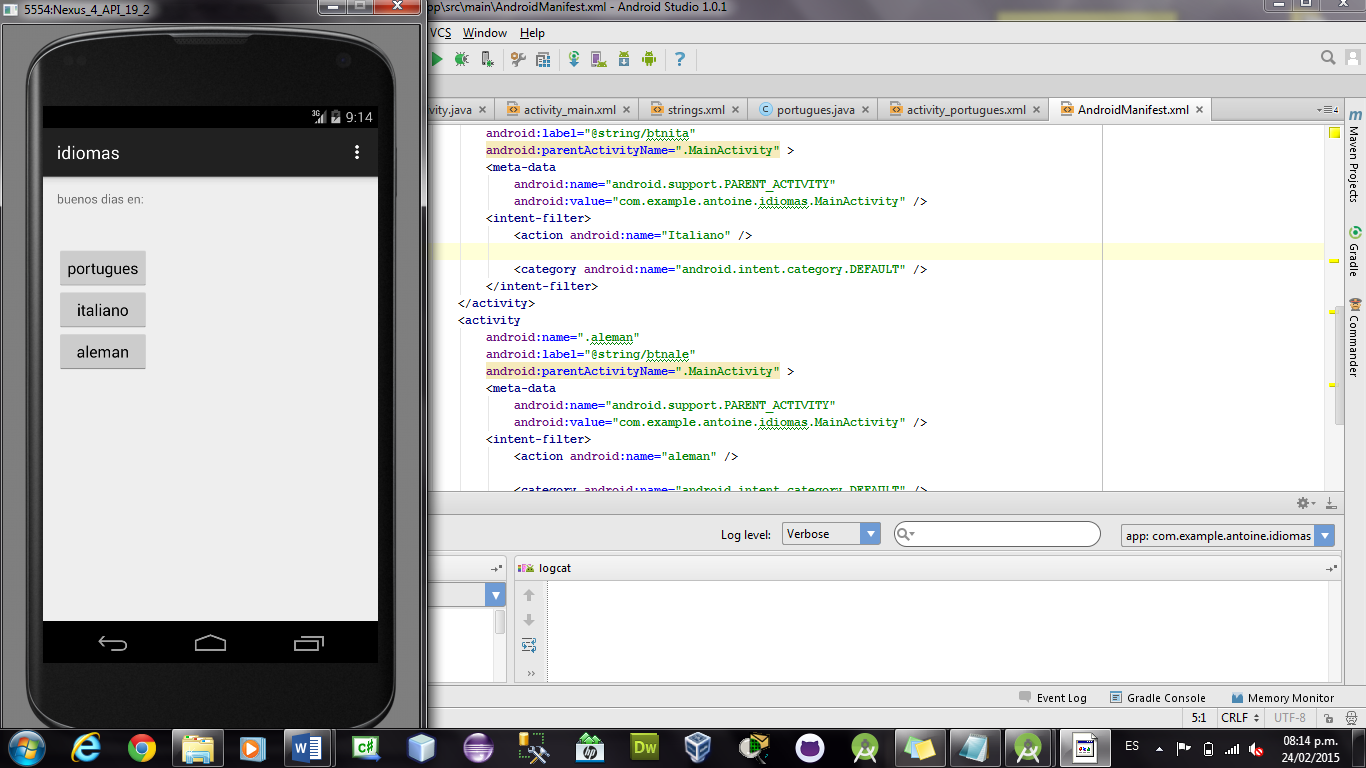
Se generaran los recursos de tipo de cadena de caracteres para darle vista y contexto a las actividades como se muestra en la siguiente imagen.  
  
Posteriormente estos recursos de @String se ponen de acuerdo a su identidad, y a los botones vamos a darle la sentencia ONCLICK para mandar a llamar un método respectivo a cada botón, el cual el método se describirá en la siguiente imagen



Se crean tres metodos por los tres respectivos botones, uno que es btn\_Onclick1 que es de manera explicita ya que este manda a llamar la clase con el nombre exacto de la actividad. Por lo tanto los otros dos botones es similar pero no manda a llamar directamente la actividad, si no manda a llamar por un nombre en especifico que va a buscar dentro de un filtro que tenga las actividades por ese nombre, estos filtros se editan en el manifest que posteriormente lo definiremos, por ahora es cambiar el (THIS, ACTIVIDAD.CLASS) por entre comillas el nombre de la actividad.  


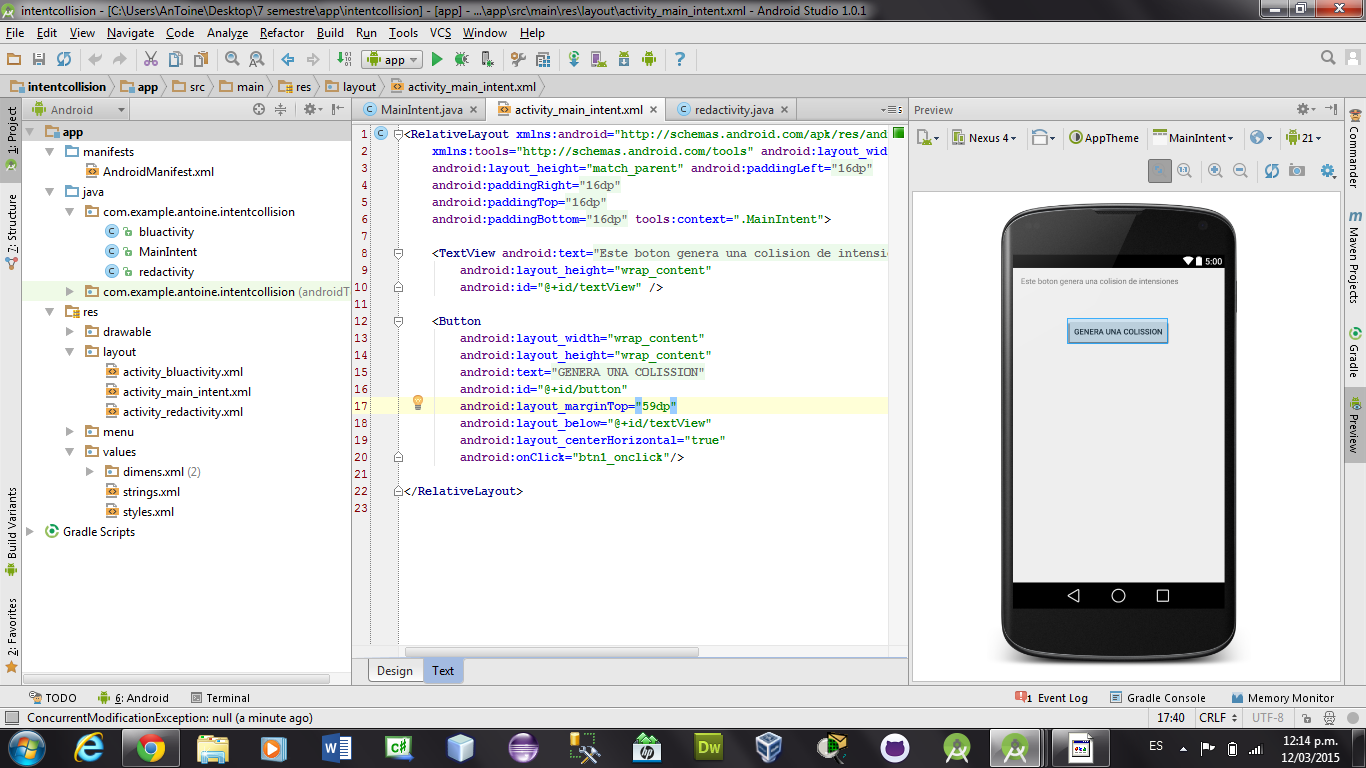
Una vez en el MANIFEST vamos a declarar los filtros, con las etiquetas <intent-filter></intent-filter> el cual dentro de esas etiquetas definiremos el nombre que va hacer filtrado la actividad para mandar a llamar con un INTENT y su categoría lo cual le daremos por default, para ser mandado a llamar.  


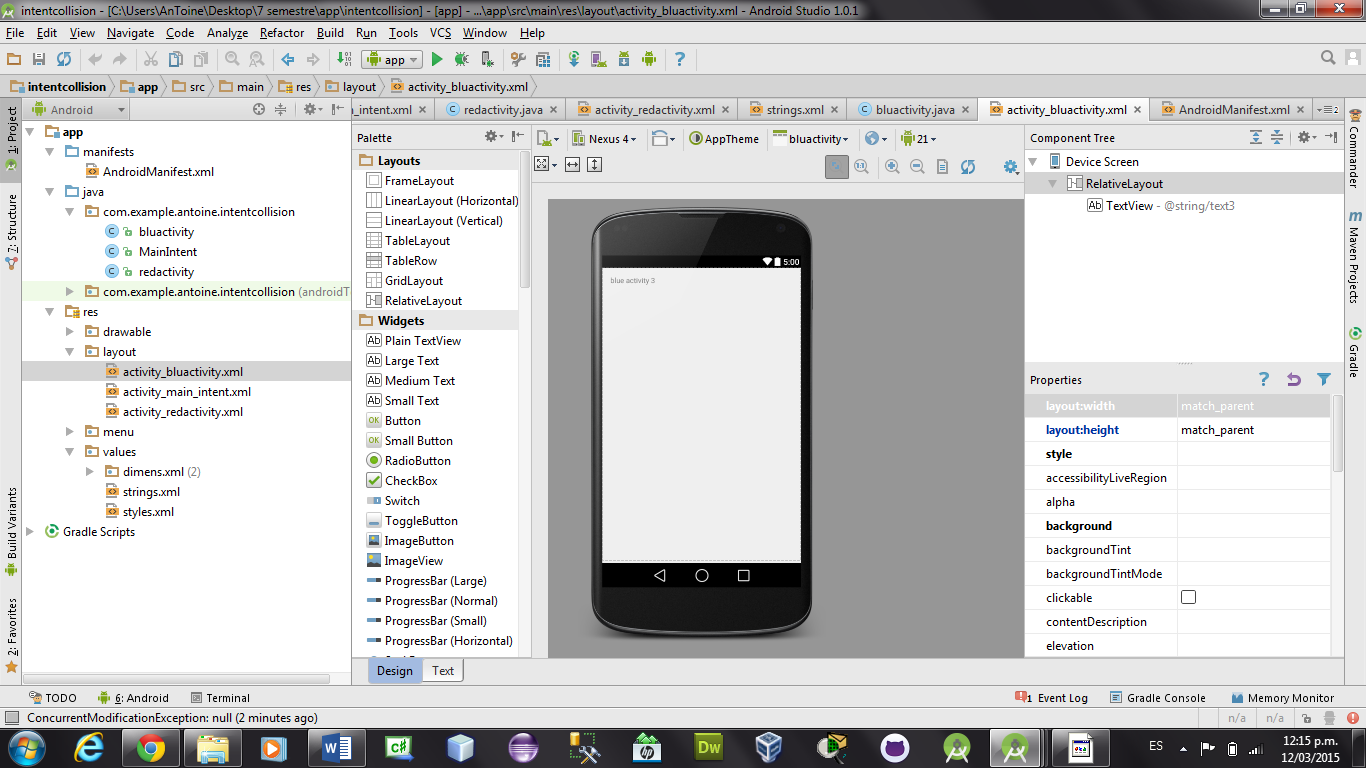
Asi es como quedaria nuestro proyecto, los tres botones con sus respectivos recursos y sus metodos a llamar, cada boton tiene su metodo onclick y al llamar a cada metodo respectivo abrira la actividad que tiene por designada el cual hara la traduccion ya escrita por un recurso.



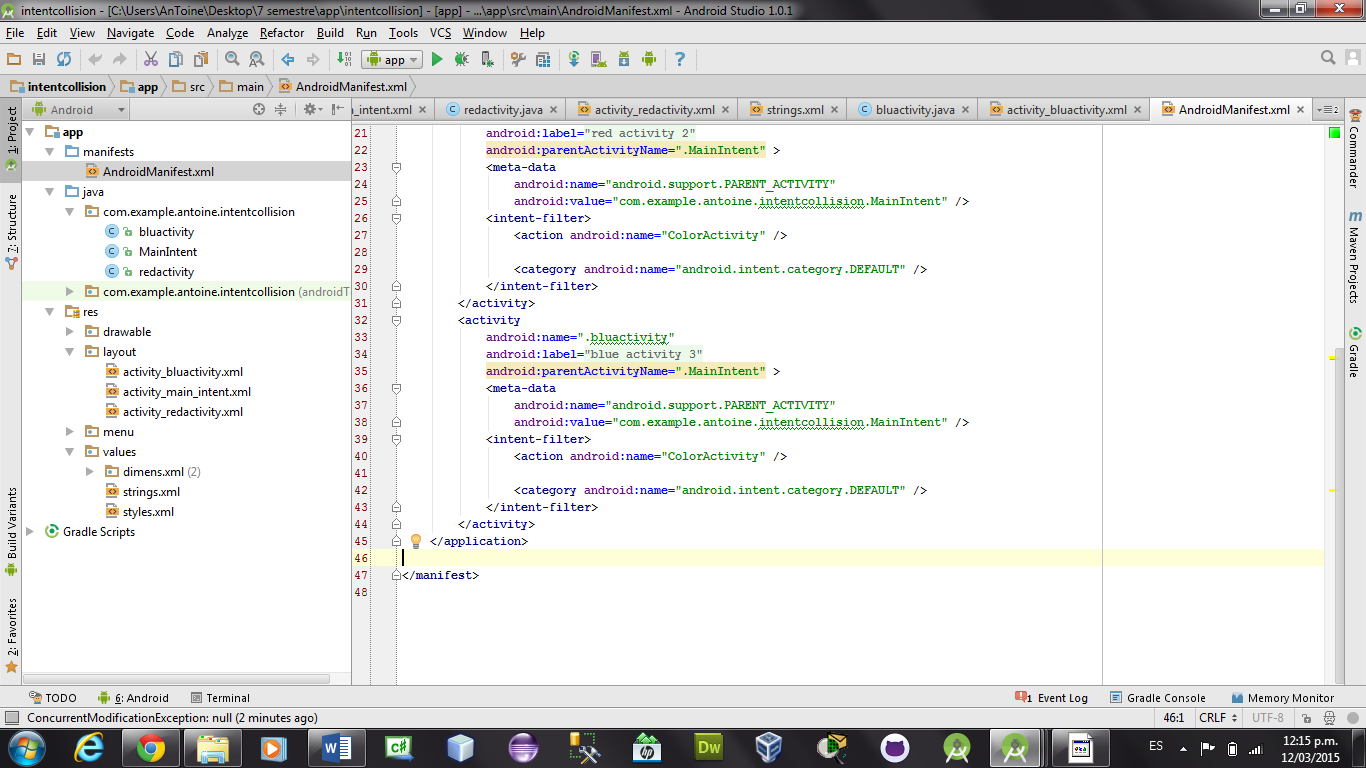
INTENT-COLISSION.

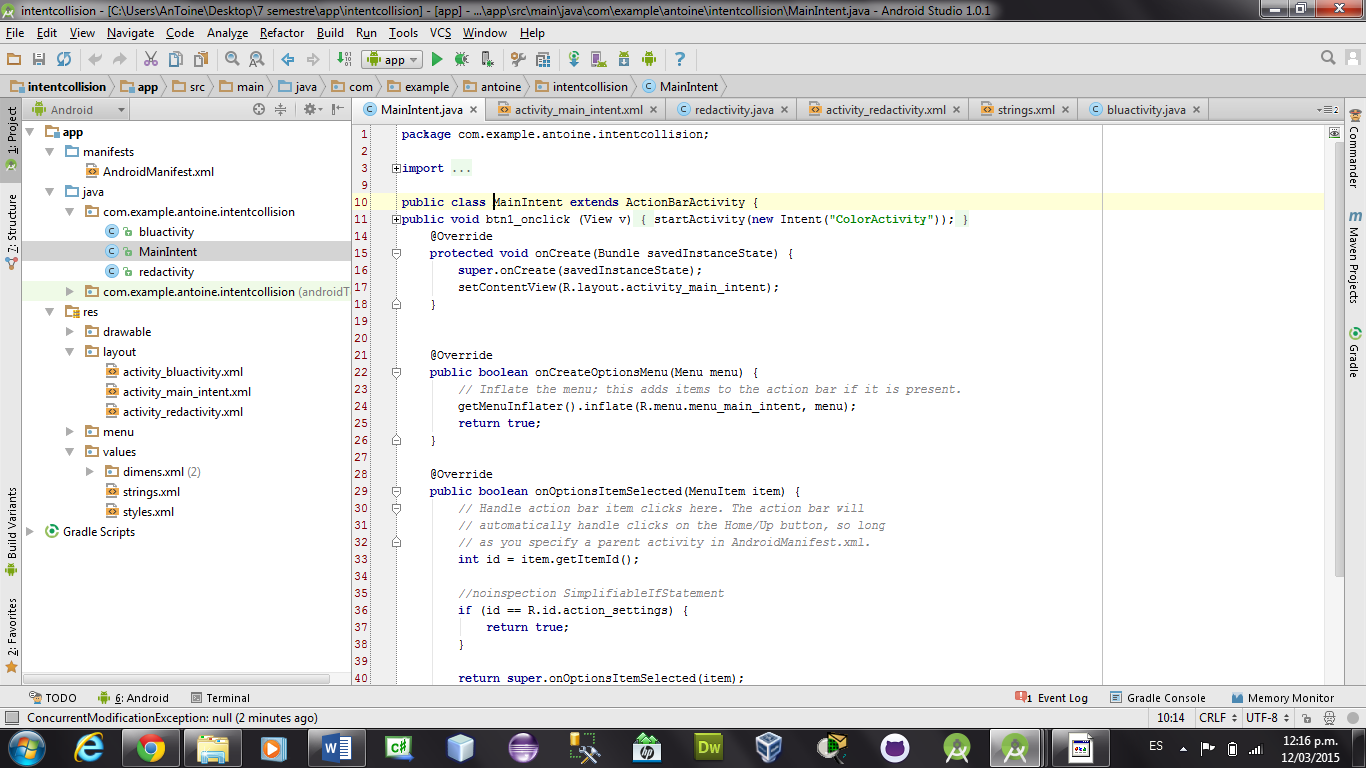
Este función como la practica anterior por INTENT IMPLICITO pero se mostrara por imágenes como se realiza y como se colisiona dos actividades al mismo tiempo.

Generamos una actividad principal al iniciar el proyecto INTENT-COLISSION y generamos una vista como se muestra en la siguiente imagen; en el botón generamos un ONCLICK que mandara llamar un método de nombre “btn1\_onclick” este método lo explicaremos posteriormente en una imagen.   


Se crea dos nuevas actividades una llamada red, y otra blue, las interfaces son similares, una contiene en el textview mensaje rojo y por otro azul.

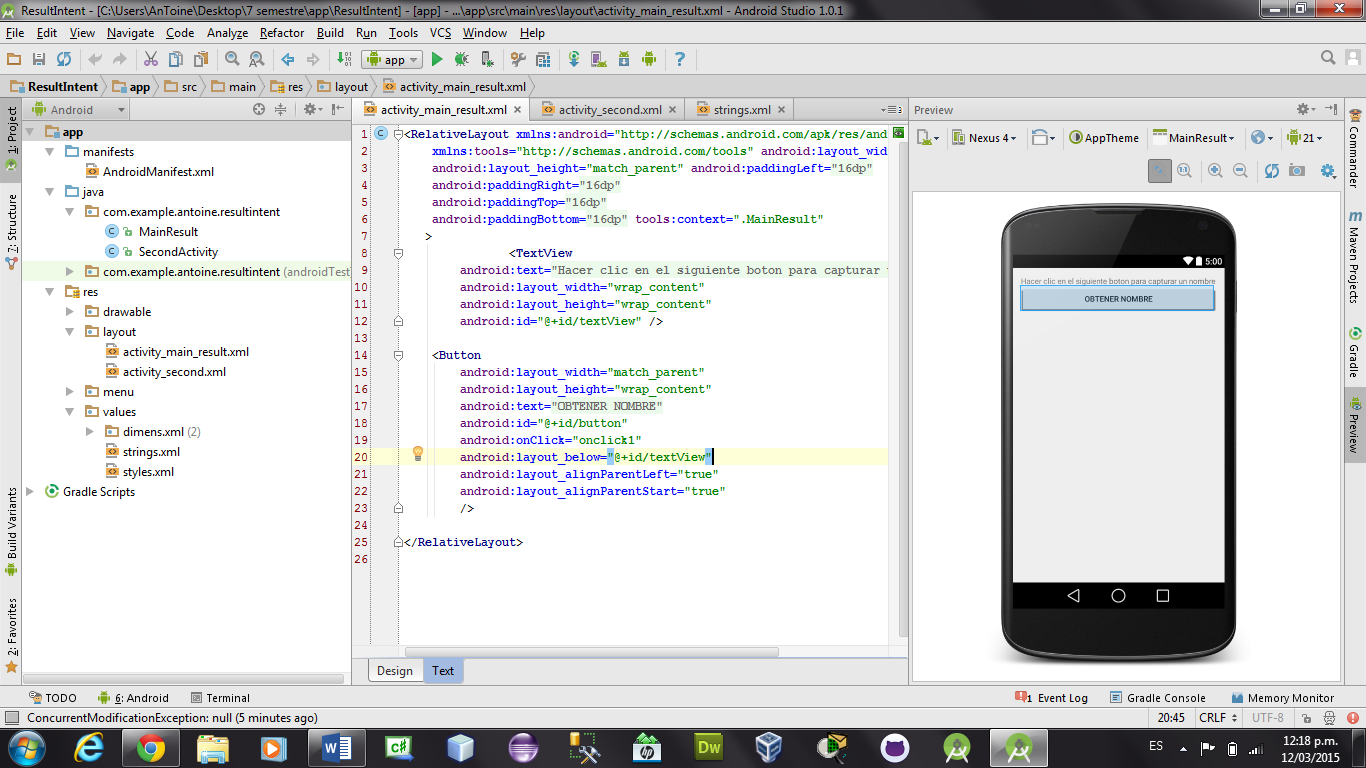
Se configura en el manifest los filtros de cada actividad de la actividad azul y rojo se pondra en el name ColorActivity a ambos para que cuando en el metodo mande a llamar de manera implicita la actividad haga colission y detecte las dos actividades y pida cual abrir de las dos.

  
Ya por ultimo se crea el metodo que manda a llamar la actividad de manera implicita por ColorActivity

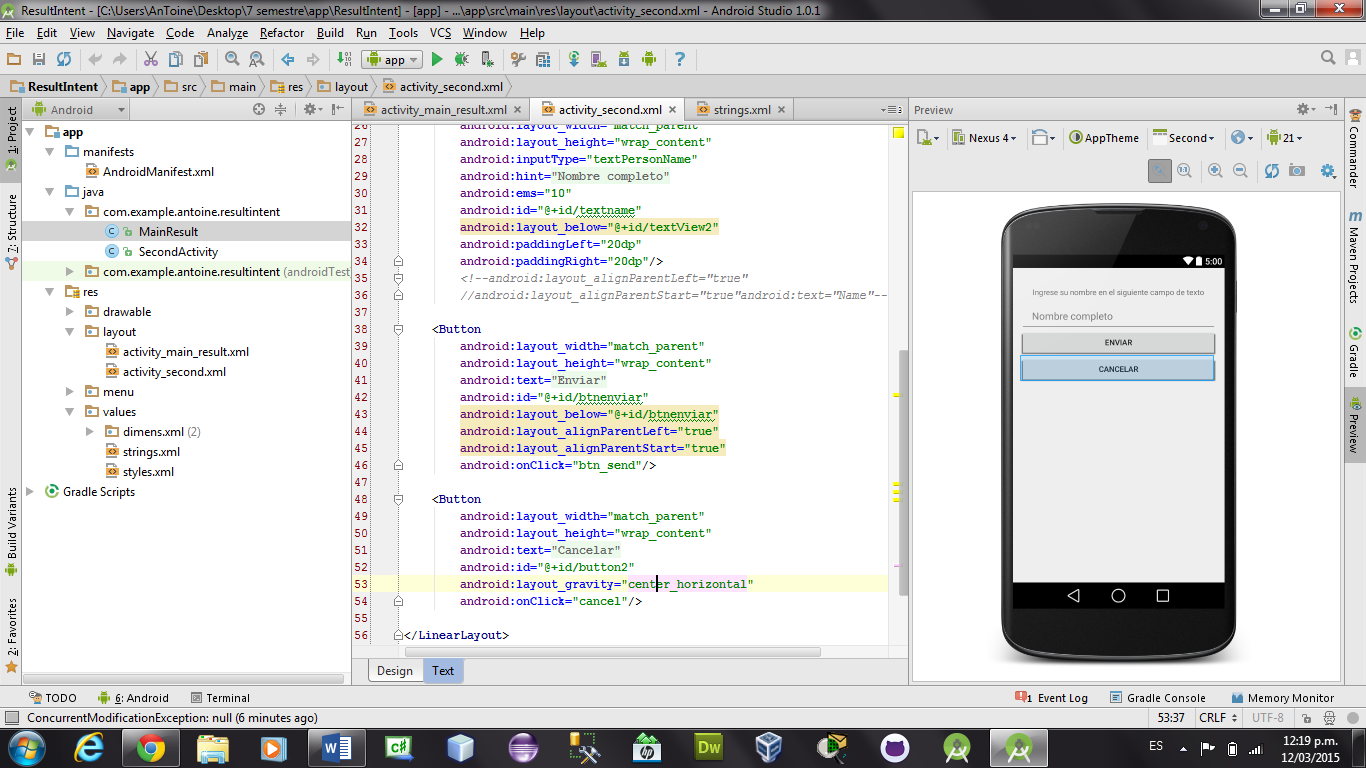


RESULT-FROM-INTENT

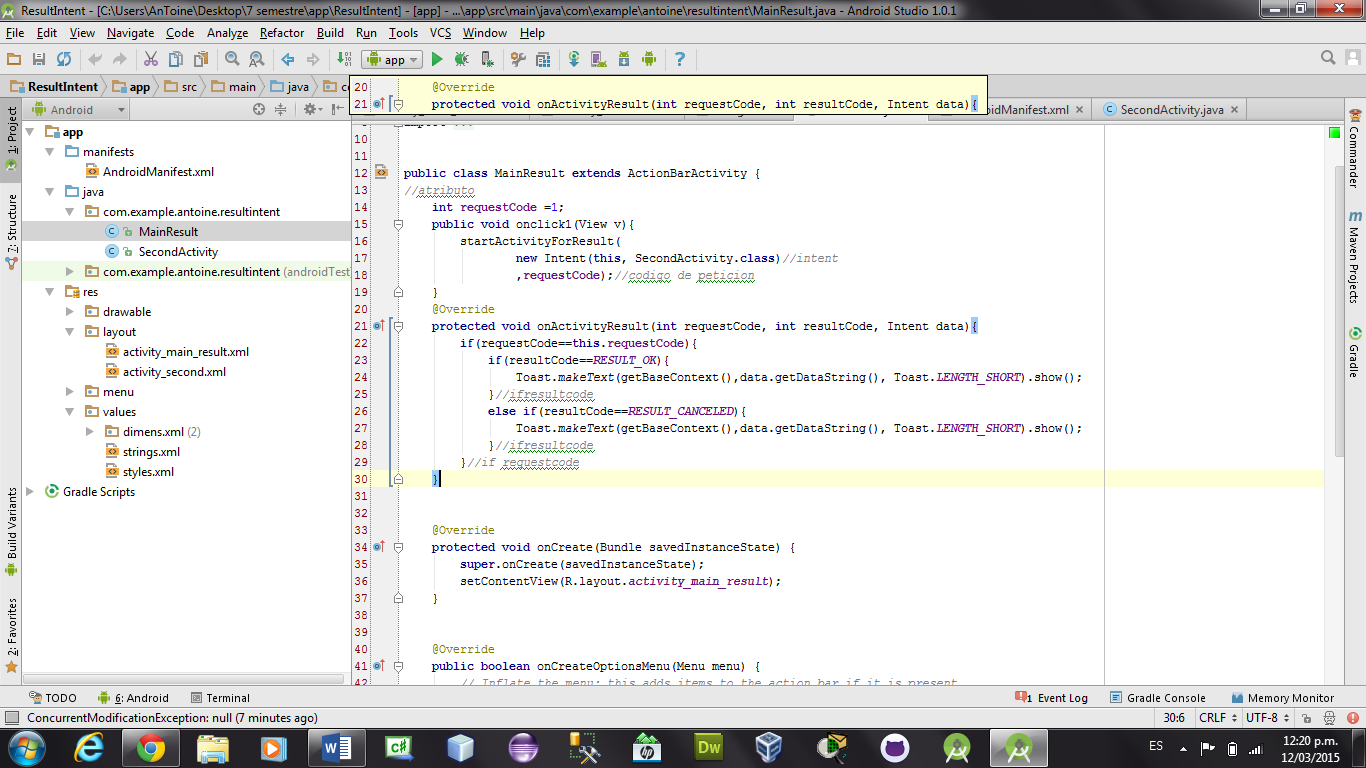
En esta práctica vamos a regresar un resultado a la actividad anterior, por lo tanto la primera actividad se creara lo básico de las otras actividades, un botón para acceder a una segunda actividad y un TEXTVIEW. El botón mandara a llamar el método onclick1.



La segunda actividad o vista XML se visualizara así como en la siguiente imagen y se le dará las siguientes propiedades mostradas en la imagen.



Después procederemos a escribir el código de los métodos que mandamos a llamar en los botones. El siguiente código muestra el método onclick1 que permite ir de manera explícita a la segunda actividad y con ONACTIVITYRESULT tendremos una respuesta de la otra clase, el cual viene una condición si es ok manda un mensaje con el nombre, puesto en la otra clase; o por lo contrario CANCELED mandara un mensaje que no se ingresó y se cancelo.



El siguiente metodo es btn\_send que estara en la segunda actividad en el boton enviar, lo que hara es validar por medio de un if si el campo esta vacio o no, si esta vacio mandara un mensaje que inserte un nombre, el cual este es de tipo EDIT TEXT y lo pasaremos a STRING para poder condicionarlo, posteriormente lo pasaremos a URI si el campo llegara tener un nombre y este es mandado o establecido a data para mandarlo en la primera actividad, se establece el resultado OK y finaliza.

Otro metodo es el de CANCEL que establece el resultCode en canceled finaliza la aplicación y lo manda para el if de la otra actividad y este retorna un mensaje, este mensaje lo almacene en una variable llamada etxt de tipo STRING que de mensaje es no se ingreso ningun nombre. Y es asi como queda el proyecto.

