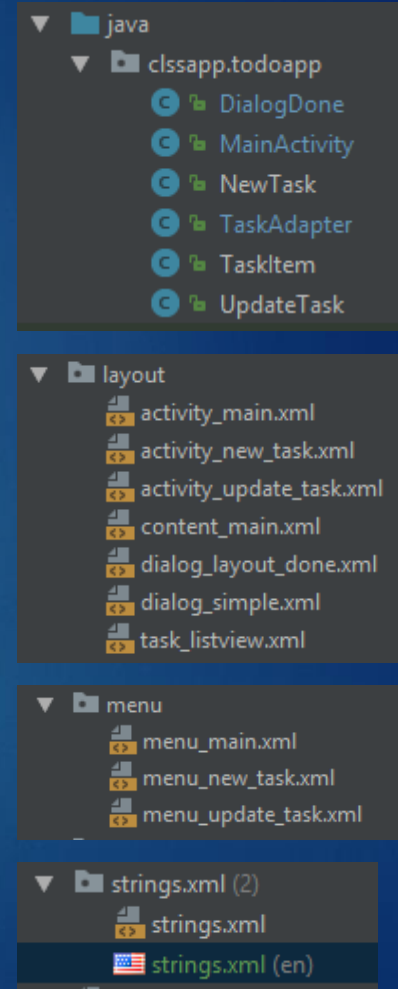
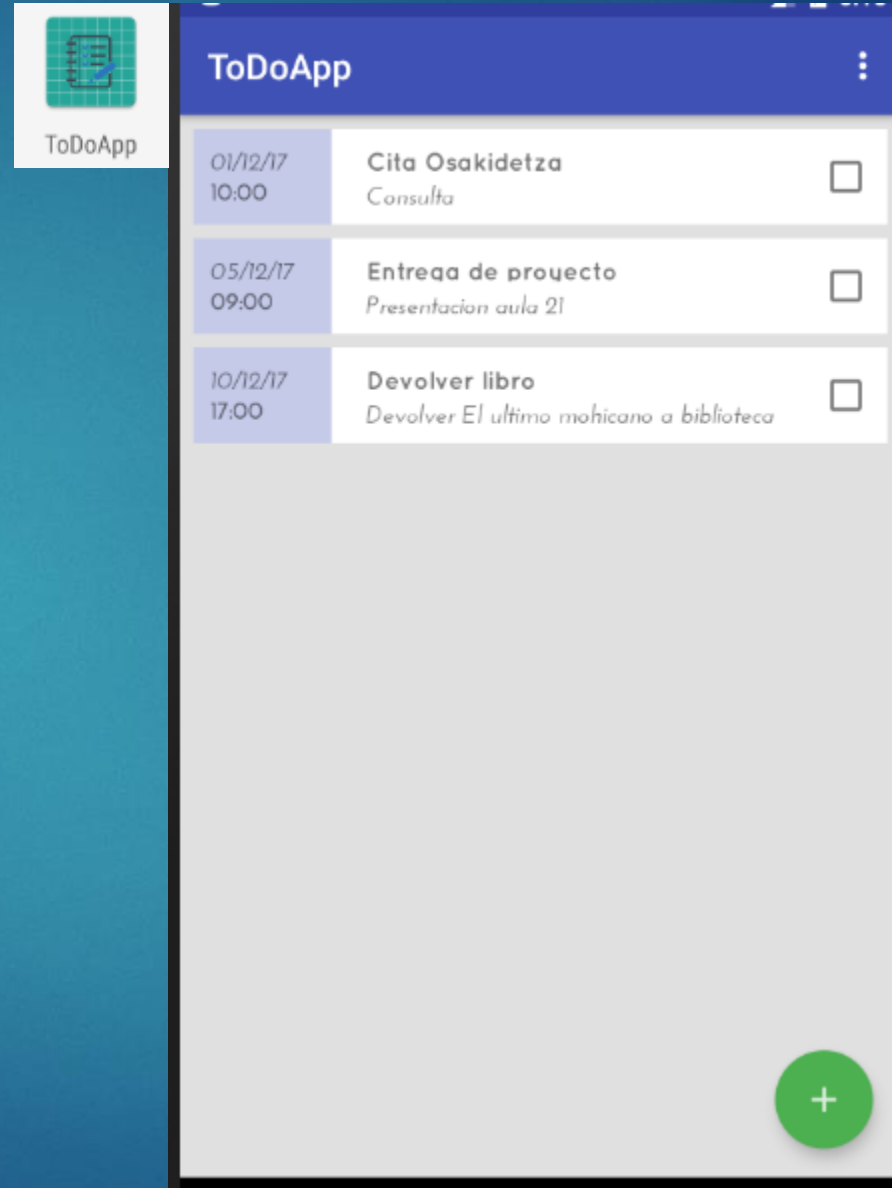


# ToDo

## LISTA DE TAREAS

Lista de tareas facil de usar que nos ayuda a organizar el tiempo y llevar un registro de las actividades en el día a día.



# 3 Actividades

layout

- activity\_main.xml
- activity\_new\_task.xml
- activity\_update\_task.xml
- content\_main.xml
- dialog\_layout\_done.xml
- dialog\_simple.xml
- task\_listview.xml

menu


- menu\_main.xml
- menu\_new\_task.xml
- menu\_update\_task.xml

← ToDoApp ✓

¿Tarea a realizar?

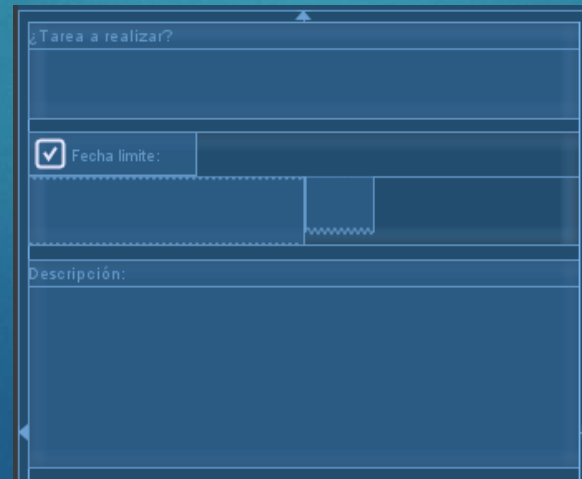
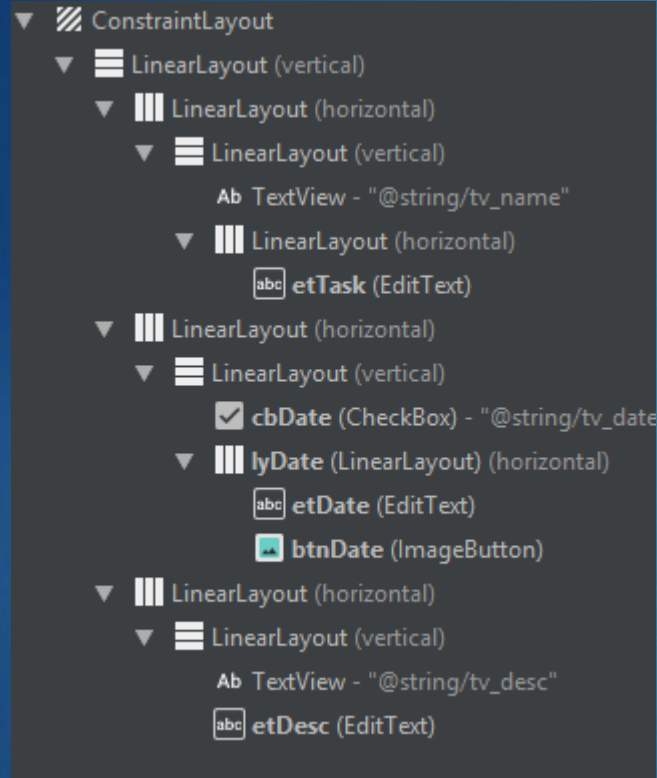
---

Fecha limite: ☒



Descripción:

# Diferentes layouts



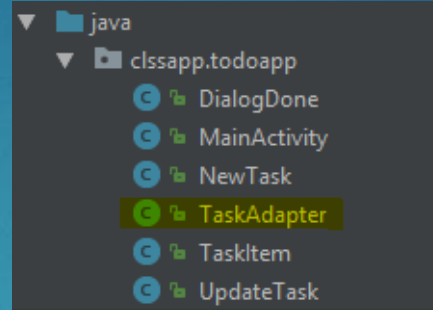
```
<LinearLayout
    android:id="@+id/lyDate"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:visibility="visible"
    android:weightSum="4">

    <EditText
        android:id="@+id/etDate"
        android:layout_width="0dp"
        android:layout_height="50dp"
        android:layout_weight="2"
        android:ems="10"
        android:inputType="none"
        android:maxLength="10"/>

    <ImageButton
        android:id="@+id/btnDate"
        android:layout_width="0dp"
        android:layout_height="42dp"
        android:layout_marginTop="-2dp"
        android:layout_weight="0.5"
        android:background="@color/icon_bg_dark"
        android:maxWidth="40dp"
        app:srcCompat="@drawable/ic_form_date"/>
</LinearLayout>
```

# ListView Adapter

```
public class TaskAdapter extends ArrayAdapter<TaskItem> {  
    public TaskAdapter(Context context, ArrayList<TaskItem> users) {  
        super(context, resource: 0, users);  
    }  
  
    @Override  
    public View getView(int position, View convertView, ViewGroup parent) {...}
```



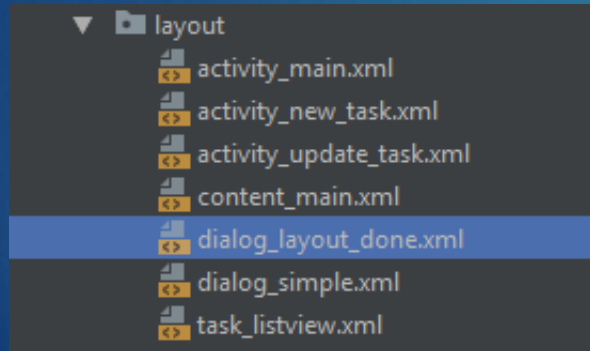
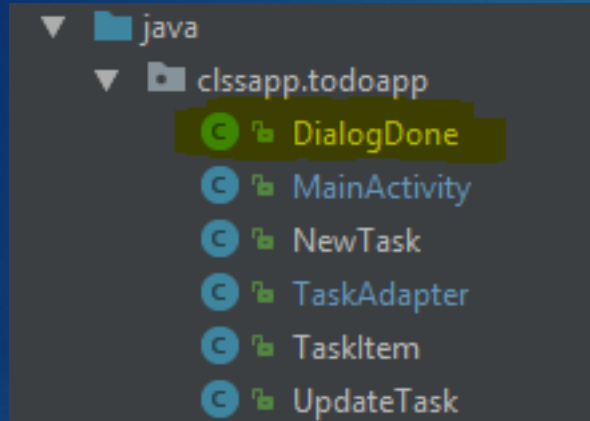
ToDoApp		
01/12/17 10:00	Cita Osakidetza Consulta	<input type="checkbox"/>
05/12/17 09:00	Entrega de proyecto Presentacion aula 21	<input type="checkbox"/>
10/12/17 17:00	Devolver libro Devolver El ultimo mohicano a biblioteca	<input type="checkbox"/>

```
@Override  
public View getView(int position, View convertView, ViewGroup parent) {  
    // Get the data item for this position  
    final TaskItem taskItem = getItem(position);  
    // Check if an existing view is being reused, otherwise inflate the view  
    if (convertView == null) {  
        convertView = LayoutInflater.from(getContext()).inflate(R.layout.task_listview,  
    }  
  
    // Lookup view for data population  
    TextView tvTask = (TextView) convertView.findViewById(R.id.tvTask);  
    TextView tvDate = (TextView) convertView.findViewById(R.id.tvDate);  
    TextView tvTime = (TextView) convertView.findViewById(R.id.tvTime);  
    TextView tvDesc = (TextView) convertView.findViewById(R.id.tvDesc);  
    CheckBox cbItemDone = (CheckBox) convertView.findViewById(R.id.cbItemDone);  
  
    // Populate the data into the template view using the data object  
    tvTask.setText(taskItem.getTask());  
    tvDate.setText(taskItem.getDate());  
    tvTime.setText(taskItem.getTime());  
    tvDesc.setText(taskItem.getDescription());
```

## Implementación desde MainActivity

```
ArrayList<TaskItem> taskList = new ArrayList<>();  
Cursor c = db.rawQuery( sql: "SELECT * FROM item", selectionArgs: null);  
  
if (c.getCount() == 0) {  
    //lista.add("No hay registros");  
} else {  
    while (c.moveToNext()) {  
        taskList.add(new TaskItem(Integer.parseInt(c.getString(0)),  
            c.getString(2), c.getString(3), c.getString(4),  
            Boolean.parseBoolean(c.getString(6)), c.getString(5));  
        System.out.println(c.getString(5));  
    }  
}  
  
// Create the adapter to convert the array to views  
TaskAdapter taskAdapter = new TaskAdapter( context: this, taskList);  
// Attach the adapter to a ListView  
lvToDo.setAdapter(taskAdapter);
```

# Dialog Fragment



```
public class DialogDone extends DialogFragment implements Button.OnClickListener {
    DialogDoneInterface ddi;
    TaskItem taskItem;
    private EditText etMsg;
    private RadioGroup rbgDone;
    private RadioButton rbPos;
    private RadioButton rbNeg;
    private Button btnAccept;
    private Button btnCancel;

    @Override
    public void onResume() { ... }

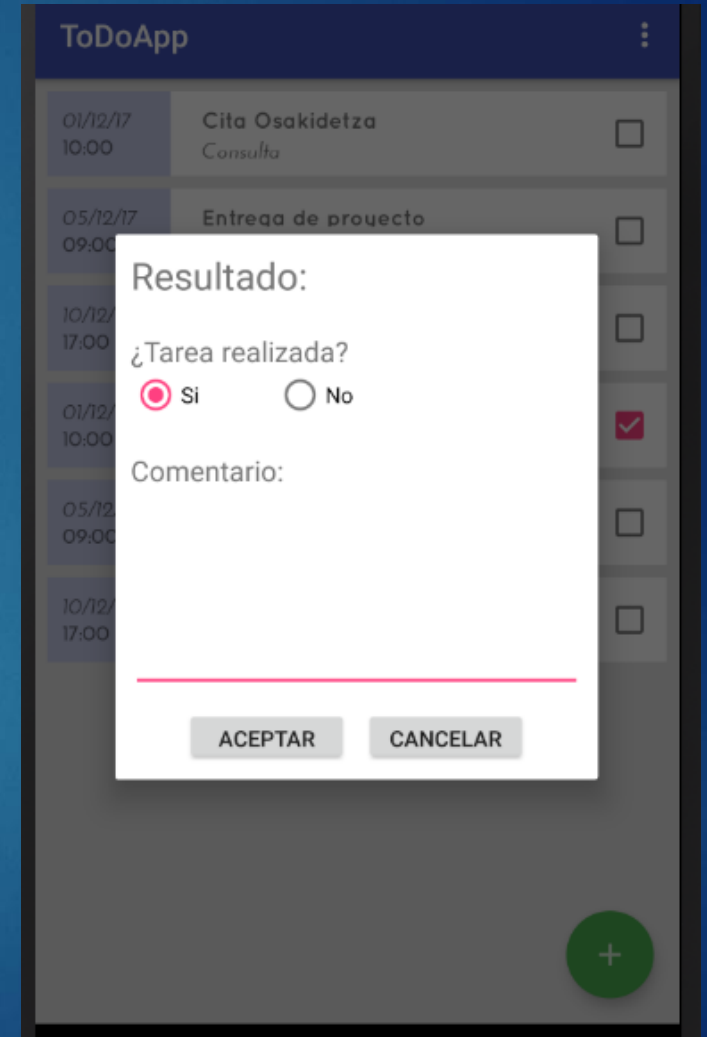
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) { ... }

    private void findViewsById(View rootView) { ... }

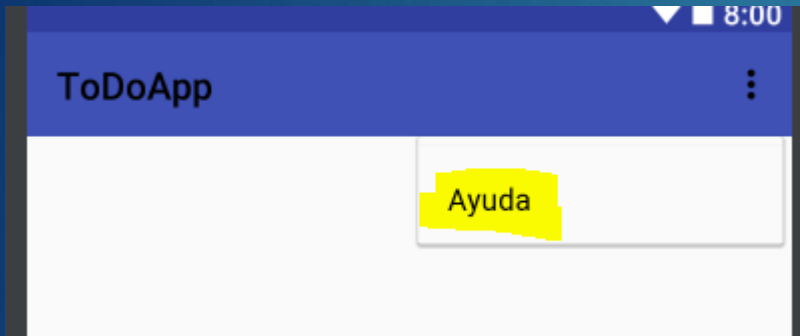
    public void setTaskItem(TaskItem taskItem) { this.taskItem = taskItem; }

    @Override
    public void onClick(View view) { ... }

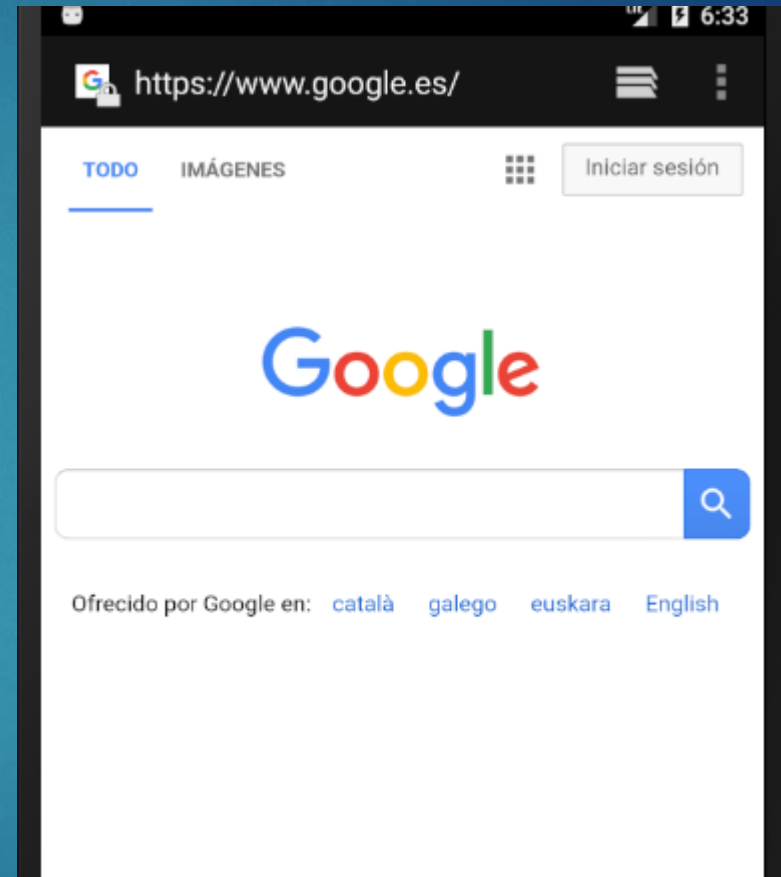
    public interface DialogDoneInterface { ... }
}
```



# Intent Implícito



```
public void launchBrowser() {  
    Intent browserIntent = new Intent(Intent.ACTION_VIEW,  
        Uri.parse("http://www.google.es"));  
    startActivity(browserIntent);  
}
```





# Sqlite

```
// Init DB
db = openOrCreateDatabase( name: "ToDoListDb", Context.MODE_PRIVATE, factory: null);

db.execSQL("DROP TABLE item");
db.execSQL("CREATE TABLE IF NOT EXISTS item(" +
    "item_id INTEGER PRIMARY KEY,task VARCHAR,description VARCHAR, date VARCHAR");
//db.execSQL("DELETE FROM item");
```

## Insert

```
public void insertMockData() {
    try {
        String query;
        for (TaskItem mockTask:mockData) {
            query = "INSERT INTO item (task, description, date, time) " +
                "VALUES ('"+mockTask.getTask()+"','"+mockTask.getDescr";
            try {
                db.execSQL(query);
            } catch (Exception e) {
                e.printStackTrace();
                logger.info( msg: "Error al insertar mockdata.");
            }
        }
    } catch (Exception e) {
        //e.printStackTrace();
        logger.info( msg: "Error al añadir datos de prueba");
    }
}
```

## Update

```
try {
    String query = "UPDATE item SET done='"+value+"',result='"+
        "WHERE item_id='"+id+"'";
    try {
        db.execSQL(query);
    } catch (Exception e) {
        e.printStackTrace();
        logger.info( msg: "Update error.");
    }
} catch (Exception e) {
    e.printStackTrace();
    logger.info( msg: "Update error.");
}
```