

# Curso de moviles

*Ejercicios propuestos*

*Mayo 2018*

### Descripción de la aplicación a desarrollar.

La aplicación es una lista de tareas.

Las tareas clasifican en tareas ToDo, aquellas con 0% de avance, Doing, aquellas con un avance mayor a 0% y menor a 100% y Done, aquellas con un avance de 100%.

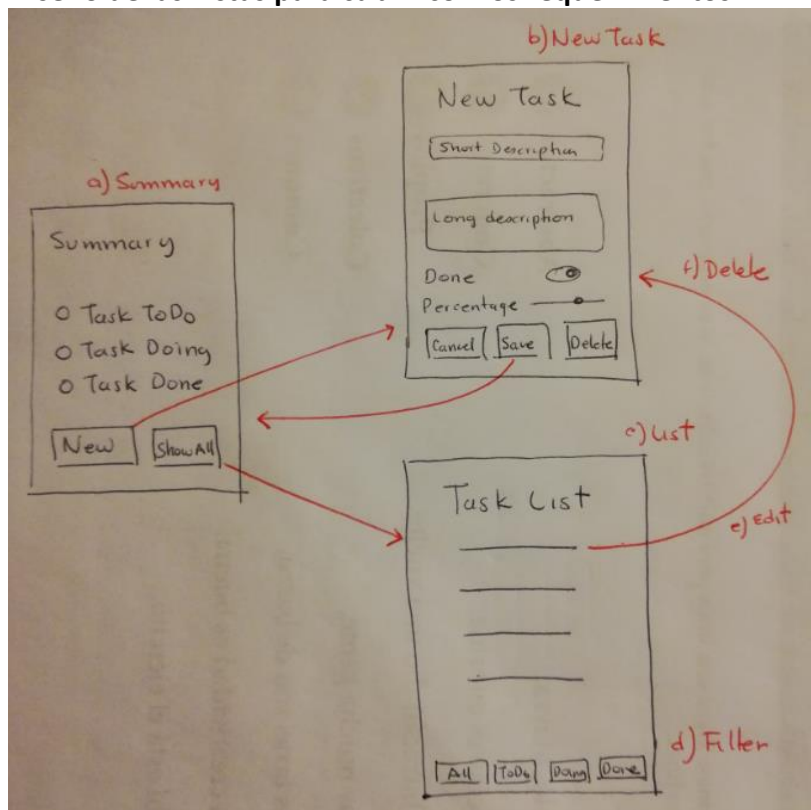
### Requerimientos.

1. Mostrar un resumen de la cantidad de tareas de cada tipo.
2. Agregar una nueva tarea.
3. Mostrar una lista de todas las tareas.
4. Filtrar las tareas para mostrar solo aquellas ToDo, Doing o Done.
5. Editar una tarea.
6. Eliminar una tarea.

### Antes de empezar, preparar el ambiente de desarrollo.

1. Instalar el software requerido<sup>1</sup>.
2. Crear y ejecuta una aplicación para probar su ambiente de desarrollo<sup>2</sup>.

### Diseño de las vistas para cubrir con los requerimientos.



<sup>1</sup> Ver el procedimiento 1 del documento de Procedimientos para la capa de presentación.

<sup>2</sup> Ver el procedimiento 2 del documento de Procedimientos para la capa de presentación.

## Contents

Ejercicio 1. ....	5
• Crear la app, llamarla task-app .....	5
<i>*Para Xamarin.Forms, llamarla AppTask, no acepta guiones.</i> .....	5
• Diseñar la pantalla inicial donde se muestra el resumen de las tareas. ....	5
• Crear un repositorio local para control de versiones y hacer un push inicial. ....	5
Ejercicio 2. ....	11
• Identificar los eventos en el ciclo de vida de una (actividad/view controller), enviar un mensaje a consola en cada uno de ellos. ....	11
Ejercicio 3. ....	12
• Crear una nueva [actividad (llámela: NewTaskFormActivity) / View Controller scene] para agregar una nueva tarea. ....	12
• Diseñar la pantalla agregando los controles requeridos para que el usuario capture los datos, de acuerdo al diseño. ....	12
• Hacer que el botón New responda al evento Click para invocar esta nueva actividad. ....	12
Ejercicio 4. ....	16
• Agregue un listener al Switch y otro al Seekbar, agregue la lógica para que en el texto abajo se reflejen los cambios. ....	16
Ejercicio 5. ....	17
• Agregue una clase para modelar una tarea. ....	17
• Cree una tarea a partir de los datos de la forma. ....	17
Ejercicio 6. ....	19
• Cree una DB con una entidad para guardar tareas. ....	19
• Al crear la tarea con los datos de la forma, sálvela en la DB. ....	19
• Muestre en el log las tareas agregadas. ....	19
Ejercicio 7. ....	20
• Crear una nueva (actividad) para mostrar la lista de tareas, llamarla TaskListActivity. ....	20
• Diseñar la vista para que contenga un elemento de lista y los botones para filtrar las tareas en la parte inferior. ....	20
• Hacer que el botón ShowAll inicie esta actividad. ....	20
• Agregar la lógica para mostrar la lista de tareas. ....	20
Ejercicio 8. ....	23

• Agregar la lógica para mostrar el resumen de las tareas en la actividad principal.....	23
Ejercicio 9. ....	23
• Agregar la lógica para filtrar las tareas en la lista de tareas. ....	23
Ejercicio 10. ....	24
• Agregar un Toolbar Back Button / Navigation Controller. ....	24
Ejercicio 11. ....	25
• Detectar el click de un elemento de la lista para mostrar el detalle. ....	25
• Eliminar o editar la tarea seleccionada. ....	25
Ejercicio 12. ....	<b>Error! Bookmark not defined.</b>

## Ejercicio 1.

- Crear la app, llamarla task-app

*\*Para Xamarin.Forms, llamarla AppTask, no acepta guiones.*

*\*Probando Xamarin Android en la MAC, la llamé xamarinAndroidTaskApp.*

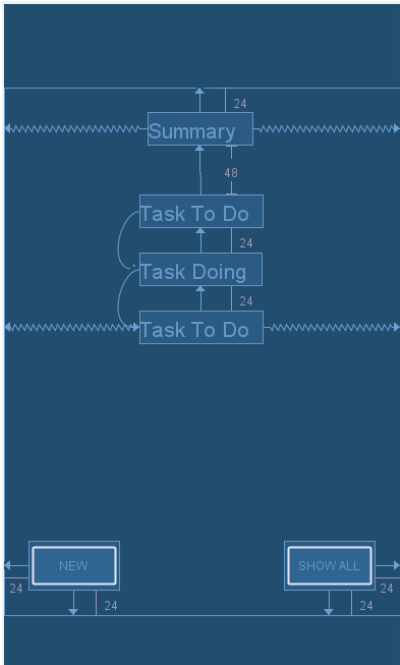
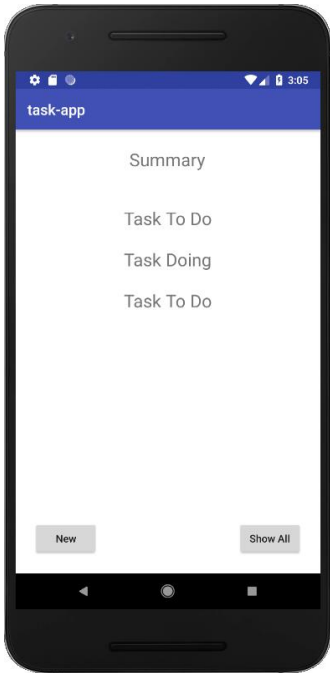
*Ver procedimiento 2.*

- Diseñar la pantalla inicial donde se muestra el resumen de las tareas.

*Ver procedimiento 3.*

- Crear un repositorio local para control de versiones y hacer un push inicial.

*Ver procedimiento 4.*

Android
<p><u>Diseño del layout:</u></p> <div></div> <div></div>
<p><u>Contenido del archivo activity_main.xml</u></p> <p>Ver archivo anexo llamado: <i>activity_main (Android Ejercicio 1).xml</i></p> <p>Ver el siguiente link: <a href="https://github.com/LilianaGF/android/blob/c0185742a5f17f2240c5f6b1f677326dea90cf90/app/src/main/res/layout/activity_main.xml">https://github.com/LilianaGF/android/blob/c0185742a5f17f2240c5f6b1f677326dea90cf90/app/src/main/res/layout/activity_main.xml</a></p> <p><u>Id de los elementos arrastrados al blueprint.</u></p>

#### Component Tree

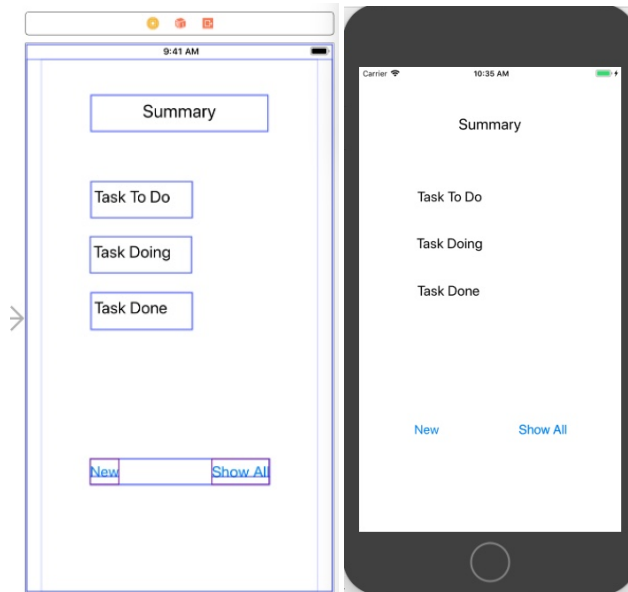
```
▼ ConstraintLayout
  Ab TextViewSummary - "@string/SummaryTitle"
  ■ ButtonNewTask - "@string/New"
  ■ ButtonShowAll - "@string/ShowAll"
  Ab TextViewTaskToDoCount - "@string/TaskToDo"
  Ab TextViewTaskDoingCount - "@string/TaskDoing"
  Ab TextViewTaskDoneCount - "@string/TaskToDo"
```

#### Contenido del archivo strings.xml

```
<resources>
  <string name="app_name">task-app</string>
  <string name="SummaryTitle">Summary</string>
  <string name="TaskToDo">Task To Do</string>
  <string name="TaskDoing">Task Doing</string>
  <string name="TaskDone">Task Done</string>
  <string name="New">New</string>
  <string name="ShowAll">Show All</string>
</resources>
```

#### iOS

#### Diseño del layout:



#### Contenido del archivo activity\_main.xml

Ver el siguiente link:

<https://github.com/LilianaGF/iOS/blob/08e7c3d3456dc55497938e5500f4f6d2c103b599/task-app/Base.lproj/Main.storyboard>

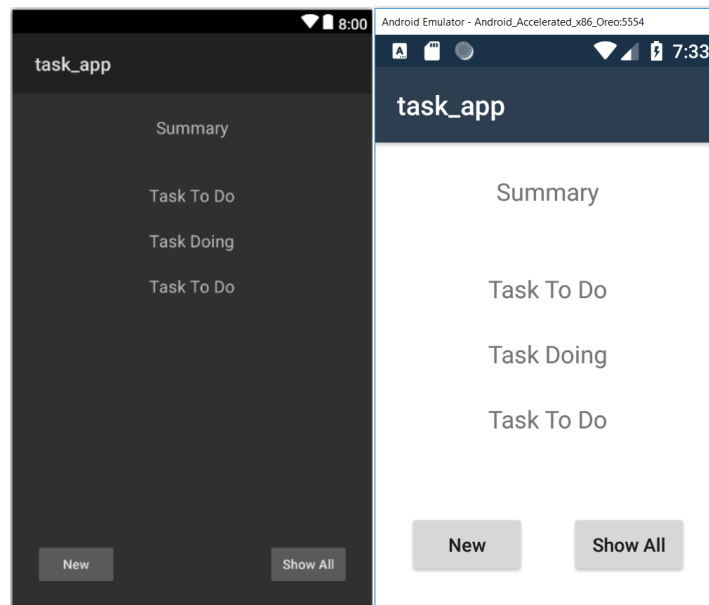
### Elementos arrastrados al Canvas.



### Xamarin.Android

### Trabajando en Visual Studio 2017 - Windows

### Diseño del layout:



### Contenido del archivo activity\_main.xml

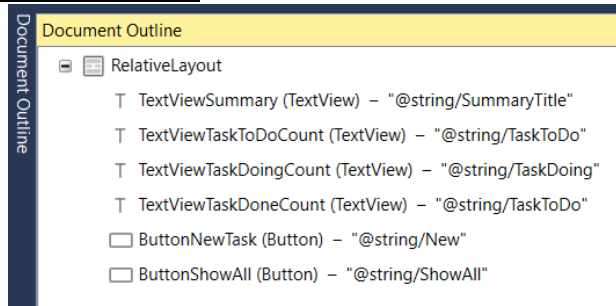
Ver archivo anexo llamado:

*activity\_main (Xamarin Android Ejercicio 1).axml*

Ver el siguiente link:

[https://github.com/LilianaGF/xamarin\\_android/blob/dd0e206d9204b39cdf8fccbfc5c2c8a7eba78546/task-app/Resources/layout/activity\\_main.axml](https://github.com/LilianaGF/xamarin_android/blob/dd0e206d9204b39cdf8fccbfc5c2c8a7eba78546/task-app/Resources/layout/activity_main.axml)

## Elementos agregados.



## Contenido del archivo strings.xml

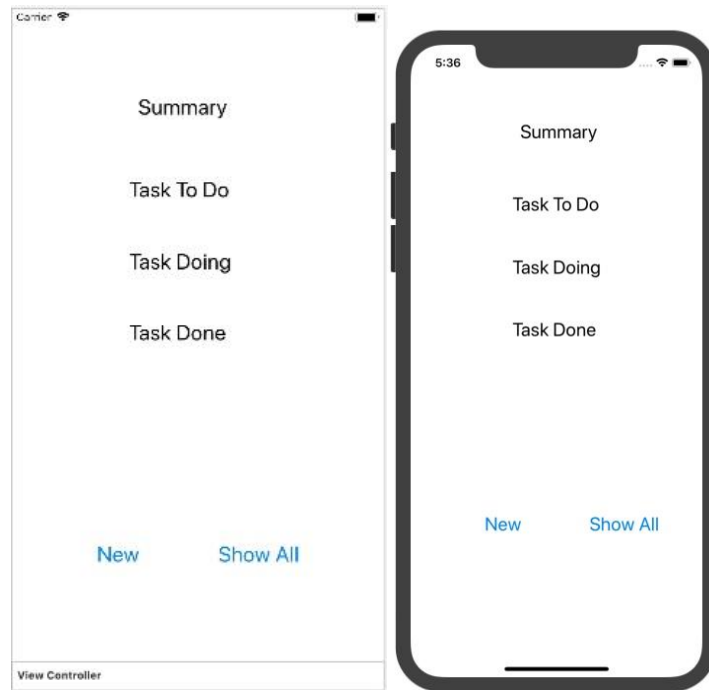
```
<resources>
  <string name="app_name">task_app</string>
  <string name="action_settings">Settings</string>
  <string name="SummaryTitle">Summary</string>
  <string name="TaskToDo">Task To Do</string>
  <string name="TaskDoing">Task Doing</string>
  <string name="TaskDone">Task Done</string>
  <string name="New">New</string>
  <string name="ShowAll">Show All</string>
</resources>
```

## Trabajando en Visual Studio 2017 - Mac

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/bee80d8e910bd0b54354ba46bab5d99e02c0598f/xamarinAndroidTaskApp/Resources/layout/Main.axml](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/bee80d8e910bd0b54354ba46bab5d99e02c0598f/xamarinAndroidTaskApp/Resources/layout/Main.axml)



Diseño del *layout*:



Contenido del archivo activity\_main.xml

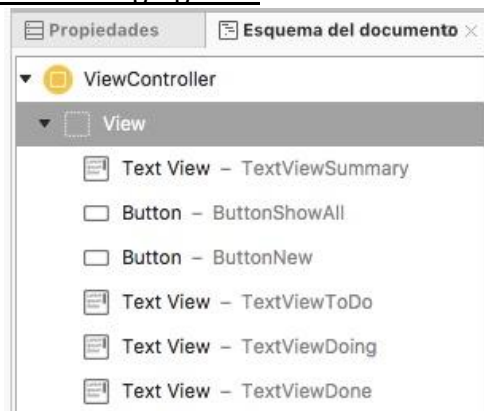
Ver archivo anexo llamado:

*MainPage (Xamarin Forms Ejercicio 1).xaml*

Ver el siguiente link:

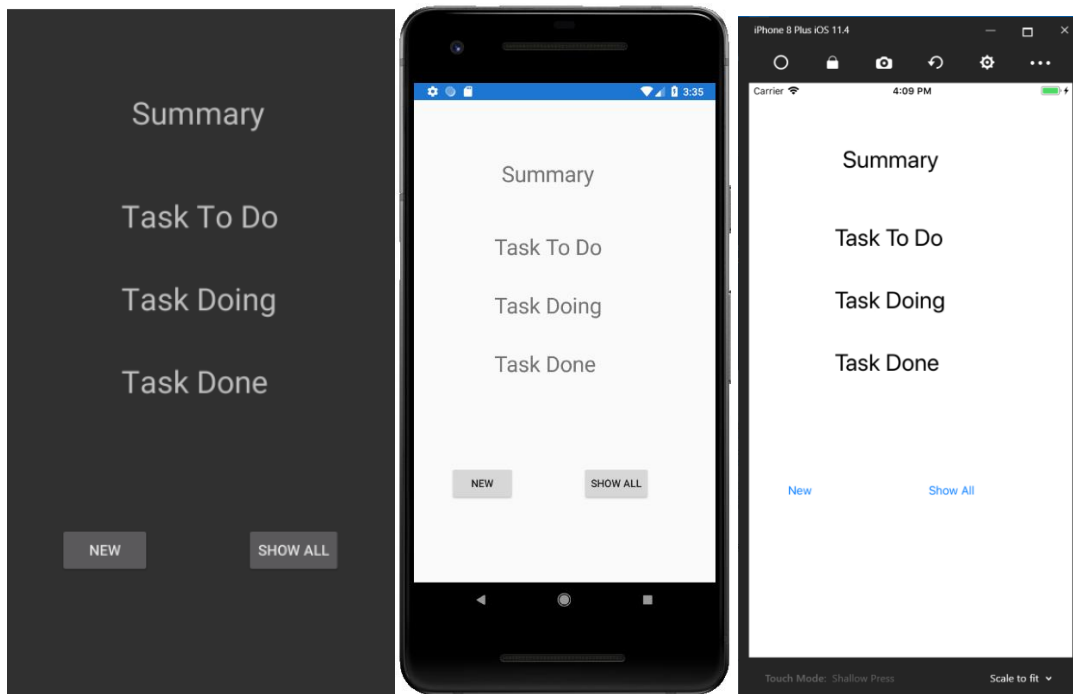
[https://github.com/LilianaGF/xamarin\\_ios/blob/36dc172b1782d82e320436b14b829157f442b15c/task-app/Main.storyboard](https://github.com/LilianaGF/xamarin_ios/blob/36dc172b1782d82e320436b14b829157f442b15c/task-app/Main.storyboard)

Elementos agregados.



## Xamarin.Forms

### Diseño del *layout*:



### Contenido del archivo *activity\_main.xml*

Ver archivo anexo llamado:

*MainPage (Xamarin Forms Ejercicio 1).xaml*

Ver el siguiente link:

[https://github.com/LilianaGF/xamarin\\_forms/blob/d9b9c90a4dd7a6b83ba8b1ee1c5be75af0dadd5e/AppTask/AppTask/MainPage.xaml](https://github.com/LilianaGF/xamarin_forms/blob/d9b9c90a4dd7a6b83ba8b1ee1c5be75af0dadd5e/AppTask/AppTask/MainPage.xaml)

## Ejercicio 2.

- Identificar los eventos en el ciclo de vida de una (actividad/view controller), enviar un mensaje a consola en cada uno de ellos.

*Ver procedimientos 5 y 6.*

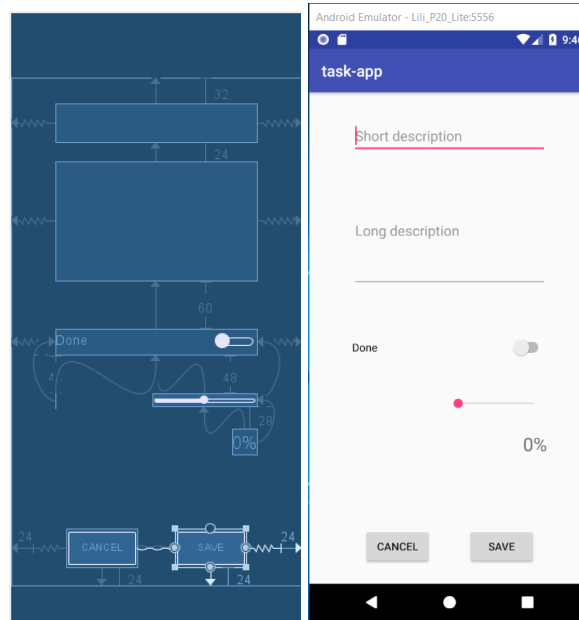
Android
<u>Contenido del archivo MainActivity.java</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/android/blob/3c06cb78e3ef6274a93d88cfd660ce95c2946cd0/app/src/main/java/towa/task_app/MainActivity.java">https://github.com/LilianaGF/android/blob/3c06cb78e3ef6274a93d88cfd660ce95c2946cd0/app/src/main/java/towa/task_app/MainActivity.java</a>
iOS
<u>Contenido del archivo ViewController.swift</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/iOS/blob/1ce28e9d75c9af4de6ef90f952ad2943cb3f8d2b/task-app/ViewController.swift">https://github.com/LilianaGF/iOS/blob/1ce28e9d75c9af4de6ef90f952ad2943cb3f8d2b/task-app/ViewController.swift</a>
Xamarin.Android
<u>Contenido del archivo MainActivity.cs</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/xamarin_android/blob/07df1d0a237c63df0e002d73ff0819a2c653725e/task-app/MainActivity.cs">https://github.com/LilianaGF/xamarin_android/blob/07df1d0a237c63df0e002d73ff0819a2c653725e/task-app/MainActivity.cs</a>
Xamarin.iOS
<u>Contenido del archivo x</u> Ver el siguiente link:
Xamarin.Forms
<u>Contenido del archivo x</u> Ver el siguiente link:

### Ejercicio 3.

- Crear una nueva [actividad (llámela: NewTaskFormActivity) / View Controller scene] para agregar una nueva tarea.  
*Ver procedimiento 8.*
- Diseñar la pantalla agregando los controles requeridos para que el usuario capture los datos, de acuerdo al diseño.
- Hacer que el botón New responda al evento Click para invocar esta nueva actividad.  
*Ver procedimientos 9 y 10.*  
*Haga que los botones Cancel y Save en la actividad que acaba de agregar, regresen a la MainActivity.*

#### Android

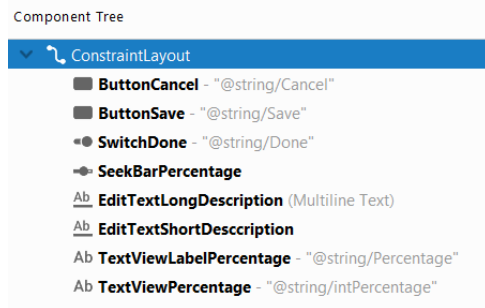
##### Diseño del layout:



##### Contenido del archivo strings.xml

```
<resources>
...
<string name="Cancel">Cancel</string>
<string name="Save">Save</string>
<string name="Done">Done</string>
<string name="LongDescription">Long description</string>
<string name="ShortDescription">Short description</string>
<string name="Percentage" />
<string name="intPercentage">0%</string>
</resources>
```

## Id de los elementos arrastrados al *blueprint*.



## Método que responde al click en al botón New y que abre la nueva actividad.

```
public void ShowNewTaskForm(View view)
//                                     //onClick-ButtonNewTask (New button)
{
    //                                     //To verify the click get the method.
    Log.d(tag, "click ButtonNewTask");

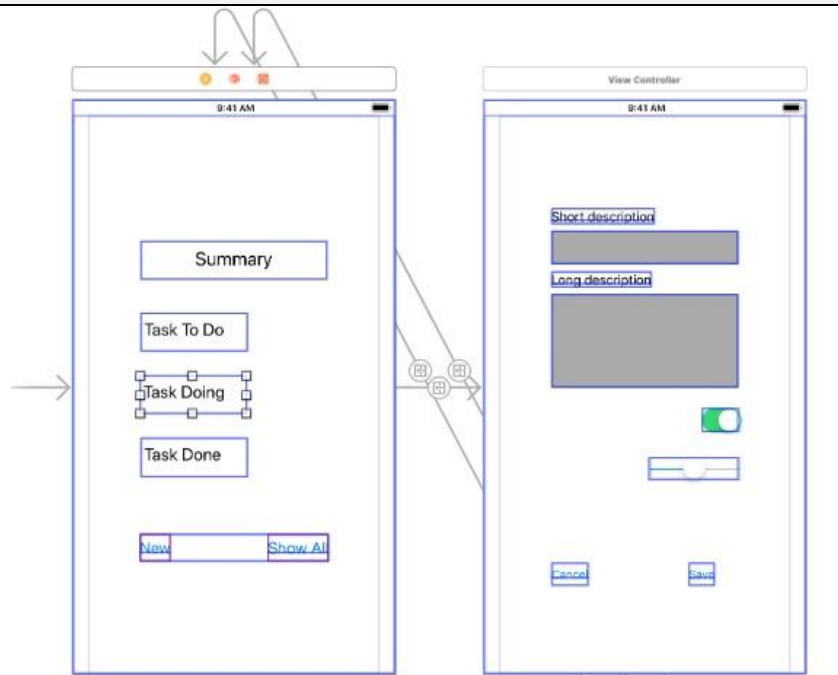
    //                                     //Explicit Intent to start NewTaskFormActivity.
    Intent intent = new Intent(
        getApplicationContext(), NewTaskFormActivity.class);
    startActivity(intent);
}
```

### Código

Ver el siguiente link:

<https://github.com/LilianaGF/android/tree/16e50d0dd7718b7d4910140e089c9fc4b8c50c53/app/src/main>

## iOS

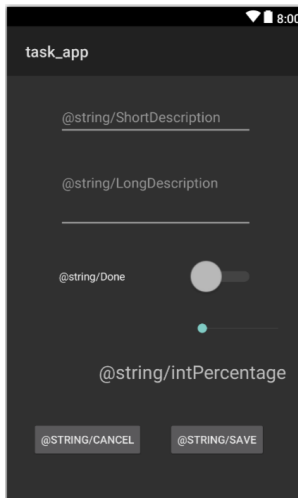


<https://github.com/LilianaGF/iOS/tree/1ce28e9d75c9af4de6ef90f952ad2943cb3f8d2b>

Ejercicios propuestos para el curso de móviles.

## Xamarin.Android

### Resultados en Windows:

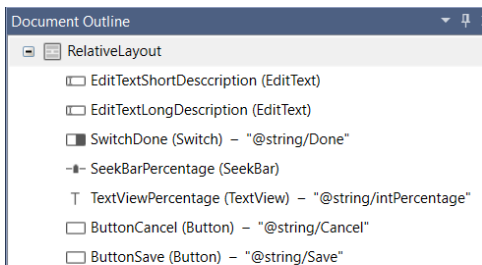


<resources>

...

```
<string name="Cancel">Cancel</string>
<string name="Save">Save</string>
<string name="Done">Done</string>
<string name="LongDescription">Long description</string>
<string name="ShortDescription">Short description</string>
<string name="Percentage">Percentage</string>
<string name="intPercentage">0%</string>
```

</resources>



### Resultados en Mac:

#### Main.axml

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/Resources/layout/Main.axml](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/Resources/layout/Main.axml)

#### activity\_new\_task\_form.axml

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/Resources/layout/activity\\_new\\_task\\_form.axml](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/Resources/layout/activity_new_task_form.axml)

#### MainActivity.cs

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/MainActivity.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/MainActivity.cs)

#### NewTaskFormActivity.cs

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/NewTaskFormActivity.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/250c8d4d41cb05b831d4ad9905f099c4ba87997b/xamarinAndroidTaskApp/NewTaskFormActivity.cs)

Xamarin.iOS
Xamarin.Forms

#### Ejercicio 4.

- Agregue un listener al Switch y otro al Seekbar, agregue la lógica para que en el texto abajo se reflejen los cambios.  
Si el Switch está checado, el valor del cuadro de texto debe ser 100%, independientemente de la posición del Seekbar  
Si el Switch no está checado, el valor del texto debe corresponder con el de la posición del Seekbar.  
*Ver procedimientos 11 y 12.*

Android
<u>Código</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/android/blob/de0b964e5ef7ae8e4176b1c7e6adc1a52434474d/app/src/main/java/towa/task_app/NewTaskFormActivity.java">https://github.com/LilianaGF/android/blob/de0b964e5ef7ae8e4176b1c7e6adc1a52434474d/app/src/main/java/towa/task_app/NewTaskFormActivity.java</a>
iOS
<u>Código</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/iOS/blob/84621925c7c5bbc85e218ad1c38e2fb6c30d8942/task-app/NewTaskFormViewController.swift">https://github.com/LilianaGF/iOS/blob/84621925c7c5bbc85e218ad1c38e2fb6c30d8942/task-app/NewTaskFormViewController.swift</a>
Xamarin.Android
<u>Código</u> Ver el siguiente link: <a href="https://github.com/LilianaGF/xamarinAndroid_Mac/blob/5efb9fb4c912e4a6aabaf21e259874e115f834e5/xamarinAndroidTaskApp/NewTaskFormActivity.cs">https://github.com/LilianaGF/xamarinAndroid_Mac/blob/5efb9fb4c912e4a6aabaf21e259874e115f834e5/xamarinAndroidTaskApp/NewTaskFormActivity.cs</a>
Xamarin.iOS
Xamarin.Forms



### Ejercicio 5.

- Agregue una clase para modelar una tarea.
- Cree una tarea a partir de los datos de la forma.

#### Android

```
package towa.task_app;  
public class Task {  
    private String shortDescription;  
    private String longDescription;  
    private int percentage;  
    //getters y setters  
}
```

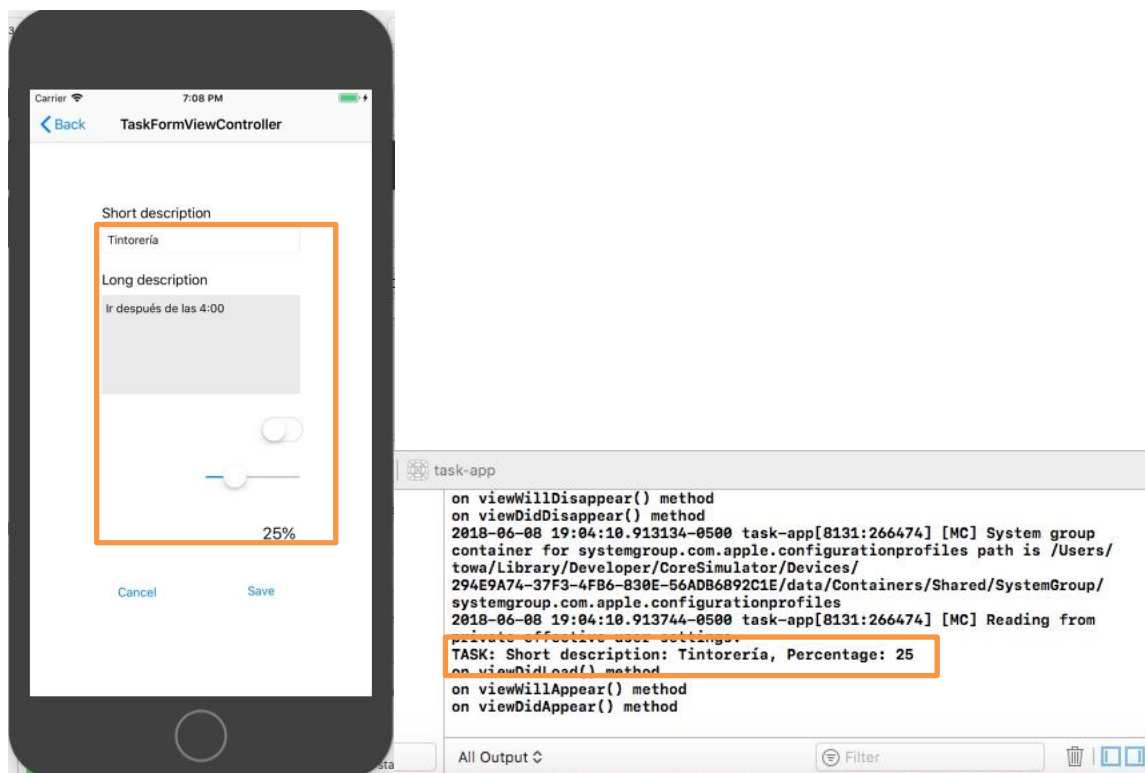
#### Código

Ver las siguientes link:

[https://github.com/LilianaGF/android/blob/1fad506f9a2fc7d9eb5a348d6489dd3075a5092d/app/src/main/java/towa/task\\_app/Task.java](https://github.com/LilianaGF/android/blob/1fad506f9a2fc7d9eb5a348d6489dd3075a5092d/app/src/main/java/towa/task_app/Task.java)

[https://github.com/LilianaGF/android/blob/1fad506f9a2fc7d9eb5a348d6489dd3075a5092d/app/src/main/java/towa/task\\_app/NewTaskFormActivity.java](https://github.com/LilianaGF/android/blob/1fad506f9a2fc7d9eb5a348d6489dd3075a5092d/app/src/main/java/towa/task_app/NewTaskFormActivity.java)

#### iOS



#### Código

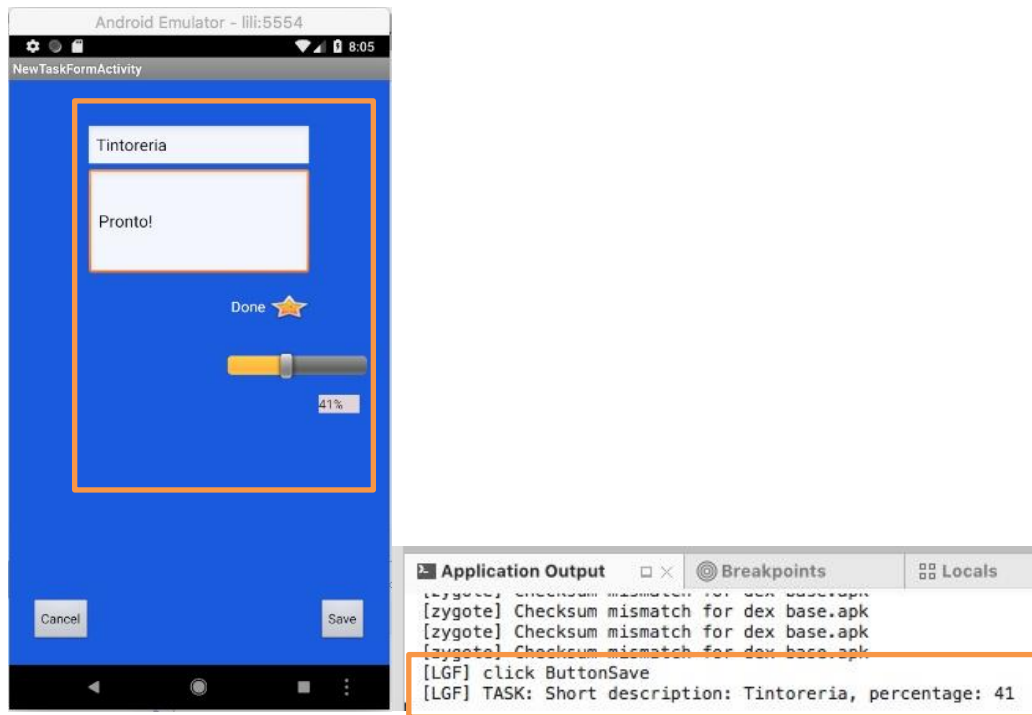
Ver las siguientes link:

<https://github.com/LilianaGF/iOS/blob/e8ac6eef58c55a745e34530fd9c647af5ad2b612/task-app/Task.swift>

<https://github.com/LilianaGF/iOS/blob/e8ac6eef58c55a745e34530fd9c647af5ad2b612/task-app/NewTaskFormViewController.swift>

Ejercicios propuestos para el curso de móviles.

## Xamarin.Android



### Código

Ver las siguientes link:

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/15c5e56e622343d5c73189ad5b623dbfd83f4ef7/xamarinAndroidTaskApp/Task.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/15c5e56e622343d5c73189ad5b623dbfd83f4ef7/xamarinAndroidTaskApp/Task.cs)

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/15c5e56e622343d5c73189ad5b623dbfd83f4ef7/xamarinAndroidTaskApp/NewTaskFormActivity.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/15c5e56e622343d5c73189ad5b623dbfd83f4ef7/xamarinAndroidTaskApp/NewTaskFormActivity.cs)

## Xamarin.iOS

## Xamarin.Forms

## Ejercicio 6.

- Cree una DB con una entidad para guardar tareas.
- Al crear la tarea con los datos de la forma, sálvela en la DB.
- Muestre en el log las tareas agregadas.

*Ver procedimiento 14.*

Android
<b>Room Persistence Library</b> <code>@Entity</code> <code>public class Task {</code> <a href="https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/Task.java">https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/Task.java</a> <code>@Dao</code> <code>public interface TaskDAO {</code> <a href="https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/TaskDAO.java">https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/TaskDAO.java</a> <code>@Database(entities = {Task.class}, version = 1)</code> <code>public abstract class TaskDB extends RoomDatabase {</code> <a href="https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/TaskDB.java">https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/TaskDB.java</a> <code>public class DBUtil {</code> <code>public static void DBSaveNewTask</code> <code>public static List&lt;Task&gt; DBGetAllTask</code> <a href="https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/DBUtil.java">https://github.com/LilianaGF/android/blob/a293572984babfb99c60d3c9343290df84c633a0/app/src/main/java/towa/task_app/DBUtil.java</a>
iOS
<b>Xamarin.Android</b>
<pre>// //path string for the database file string dbpath = Path.Combine( System.Environment.GetFolderPath(System.Environment.SpecialFolder.Personal), "dbTest.db");  // //setup the db connection var db = new SQLiteConnection(dbpath); // //setup a table db.CreateTable&lt;Task&gt;(); // //store the task in the table db.Insert(task);  // //acceder a los datos var table = db.Table&lt;Task&gt;(); foreach(var row in table){     Task PersistenceTask =         new Task(row.shortDescription, row.longDescription, row.percentage);     Log.Debug(tag, "FROM THE DB TASK: " + PersistenceTask.ToString()); }</pre> <a href="https://github.com/LilianaGF/xamarinAndroid_Mac/blob/e60f47de5d37cdf355046c0ecf74b21250a258ad/xamarinAndroidTaskApp/NewTaskFormActivity.cs">https://github.com/LilianaGF/xamarinAndroid_Mac/blob/e60f47de5d37cdf355046c0ecf74b21250a258ad/xamarinAndroidTaskApp/NewTaskFormActivity.cs</a>
<b>Xamarin.iOS</b>
<b>Xamarin.Forms</b>

*Ejercicios propuestos para el curso de moviles.*

### Ejercicio 7.

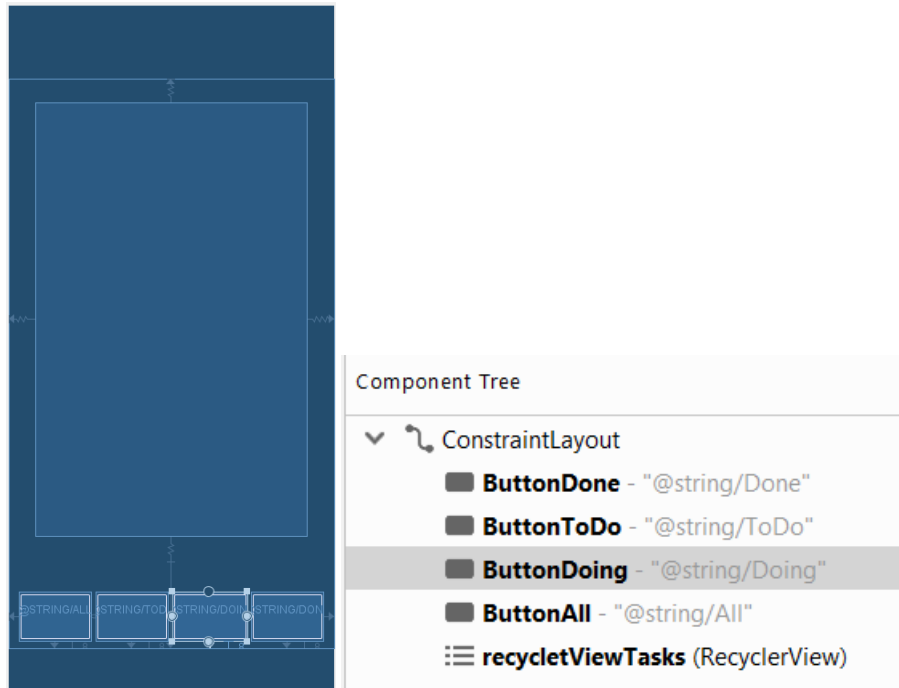
- Crear una nueva (actividad) para mostrar la lista de tareas, llamarla TaskListActivity.
- Diseñar la vista para que contenga un elemento de lista y los botones para filtrar las tareas en la parte inferior.
- Hacer que el botón ShowAll inicie esta actividad.
- Agregar la lógica para mostrar la lista de tareas.

*Ver procedimiento 15.*

Android

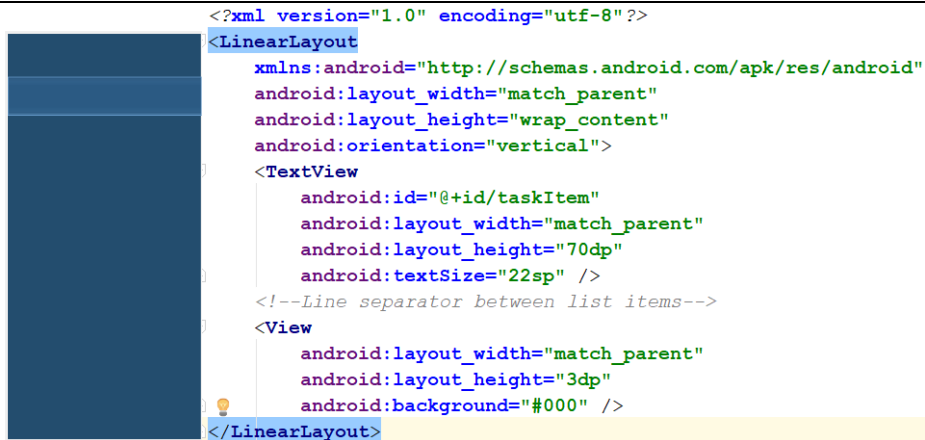
#### Solución con un RecyclerView:

Diseño del *layout*:



[https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/res/layout/activity\\_task\\_list.xml](https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/res/layout/activity_task_list.xml)

Diseño del *layout* de un elemento de la lista:



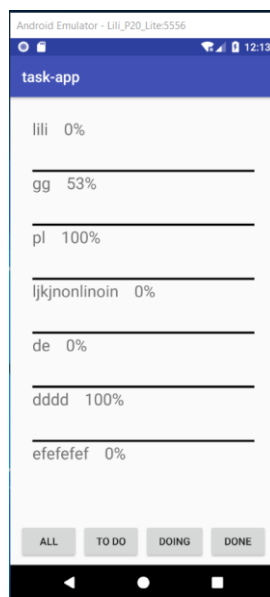
[https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/res/layout/single\\_task\\_item.xml](https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/res/layout/single_task_item.xml)

#### Adaptador para el RecyclerView:

[https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/java/towa/task\\_app/ListTaskAdapter.java](https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/java/towa/task_app/ListTaskAdapter.java)

#### Código de la Actividad:

[https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/java/towa/task\\_app/TaskListActivity.java](https://github.com/LilianaGF/android/blob/beb4390ab3b4421ca5d3a89e5ea5f549d15b531b/app/src/main/java/towa/task_app/TaskListActivity.java)



iOS

## Xamarin.Android

### Solución con un ListView:

#### Diseño del *layout*:

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/Resources/layout/activity\\_task\\_list.xml](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/Resources/layout/activity_task_list.xml)

#### Diseño del layout de un elemento de la lista:

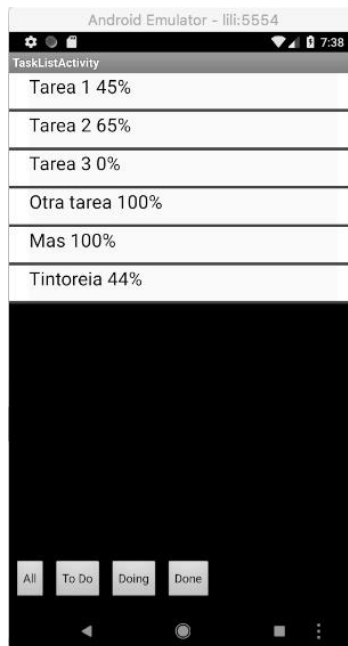
[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/Resources/layout/single\\_task\\_item.xml](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/Resources/layout/single_task_item.xml)

#### Adaptador para el ListView:

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/ListTaskAdapter.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/ListTaskAdapter.cs)

#### Código de la Actividad:

[https://github.com/LilianaGF/xamarinAndroid\\_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/TaskListActivity.cs](https://github.com/LilianaGF/xamarinAndroid_Mac/blob/073aedf74a3ca2eaedb753a56bb626191f92077b/xamarinAndroidTaskApp/TaskListActivity.cs)



### Solución con un RecyclerView:

## Xamarin.iOS

## Xamarin.Forms

### Ejercicio 8.

- Agregar la lógica para mostrar el resumen de las tareas en la actividad principal.

Android
<a href="https://github.com/LilianaGF/android/blob/5411862bfef794c028fc11b0d53b6e389a07e4b4/app/src/main/java/towa/task_app/DBUtil.java">https://github.com/LilianaGF/android/blob/5411862bfef794c028fc11b0d53b6e389a07e4b4/app/src/main/java/towa/task_app/DBUtil.java</a>
<a href="https://github.com/LilianaGF/android/blob/5411862bfef794c028fc11b0d53b6e389a07e4b4/app/src/main/java/towa/task_app/MainActivity.java">https://github.com/LilianaGF/android/blob/5411862bfef794c028fc11b0d53b6e389a07e4b4/app/src/main/java/towa/task_app/MainActivity.java</a>
iOS
Xamarin.Android
Xamarin.iOS
Xamarin.Forms

### Ejercicio 9.

- Agregar la lógica para filtrar las tareas en la lista de tareas.

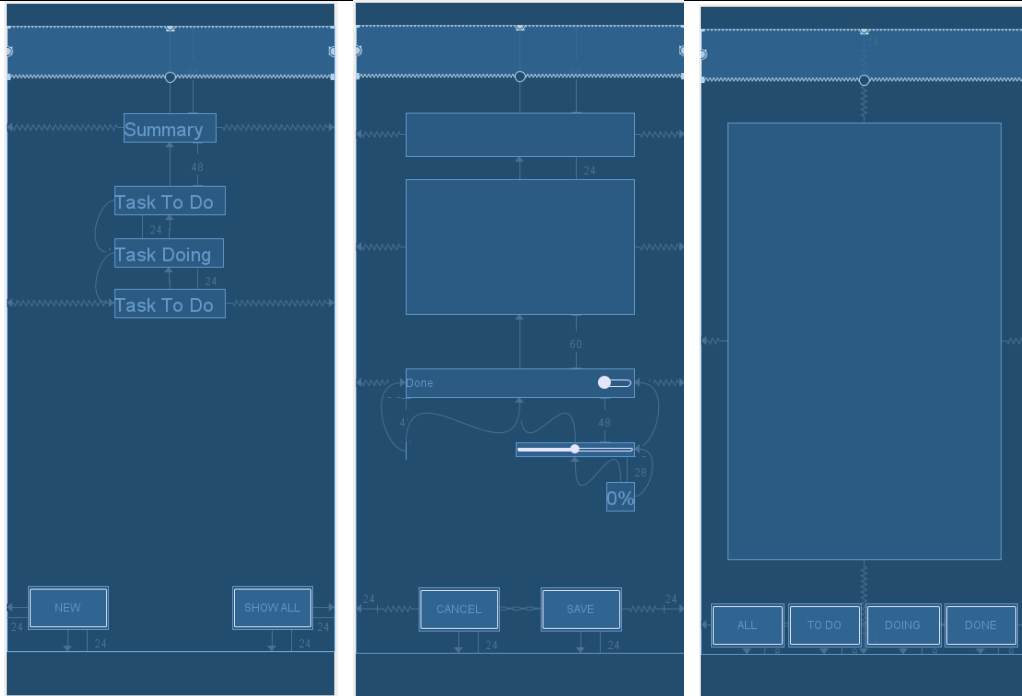
Android
<a href="https://github.com/LilianaGF/android/blob/5d3bed996c023f2fe3509322339769d432912124/app/src/main/java/towa/task_app/DBUtil.java">https://github.com/LilianaGF/android/blob/5d3bed996c023f2fe3509322339769d432912124/app/src/main/java/towa/task_app/DBUtil.java</a>
<a href="https://github.com/LilianaGF/android/blob/5d3bed996c023f2fe3509322339769d432912124/app/src/main/java/towa/task_app/TaskListActivity.java">https://github.com/LilianaGF/android/blob/5d3bed996c023f2fe3509322339769d432912124/app/src/main/java/towa/task_app/TaskListActivity.java</a>
iOS
Xamarin.Android
Xamarin.iOS
Xamarin.Forms

## Ejercicio 10.

- Agregar un Toolbar Back Button / Navigation Controller.

Ver procedimiento 16.

### Android



```
<android.support.v7.widget.Toolbar
    android:id="@+id/ToolBarMain"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:background="@attr/colorPrimary"
    android:clickable="true"
    android:focusable="true"
    android:minHeight="?attr/actionBarSize"
    android:onClick="FinishApp"
    android:theme="@attr/actionBarTheme"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:navigationIcon="@drawable/ic_arrow_white_24dp"
    app:title="Task app"
    app:titleTextColor="@android:color/white" />
```

```
<android.support.v7.widget.Toolbar
    android:id="@+id/ToolBarNewTaskForm"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:background="@attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:onClick="BackToMainActivity"
    android:theme="@attr/actionBarTheme"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:navigationIcon="@drawable/ic_arrow_white_24dp"
    app:title="Task Detail"
    app:titleTextColor="@android:color/white" />
```

```
<android.support.v7.widget.Toolbar
    android:id="@+id/ToolBarTaskList"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:background="@attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:onClick="BackToMainActivity"
    android:theme="@attr/actionBarTheme"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:navigationIcon="@drawable/ic_arrow_white_24dp"
    app:title="Task List"
    app:titleTextColor="@android:color/white" />
```

Ejercicios propuestos para el curso de moviles.



<pre>//----- public void FinishApp(View view) //                                     //onClick-toolbar (Toolbar back button) {     Log.d(tag, "Bye Li...");     finish(); }  //----- public void BackToMainActivity(View view) {     //                                     //Back to main activity.     finish(); } <a href="https://github.com/LilianaGF/android/tree/b2d9d683540ca096fc8a514b48586b2a30d78d47/app/src/main">https://github.com/LilianaGF/android/tree/b2d9d683540ca096fc8a514b48586b2a30d78d47/app/src/main</a></pre>
iOS
Pegar resultado cuando se haya agregado la lista.
Xamarin.Android
Xamarin.iOS
Xamarin.Forms

### Ejercicio 11.

- Detectar el click de un elemento de la lista para mostrar el detalle.
- Eliminar o editar la tarea seleccionada.

Android
<a href="https://gist.github.com/grantland/cd70814fe4ac369e3e92">https://gist.github.com/grantland/cd70814fe4ac369e3e92</a>
iOS
Xamarin.Android
Xamarin.iOS
Xamarin.Forms