

Qué es AndroidAnnotations

<http://androidannotations.org/>

<https://github.com/androidannotations/androidannotations/wiki>

Goal

AndroidAnnotations **facilita** la **escritura** y el **mantenimiento** de las aplicaciones de Android. Permite la realización de **código simple y claro**.

<https://github.com/androidannotations/androidannotations/wiki#goals>

Características

Esto lo iremos viendo a lo largo de los cursos de AndroidAnnotations:

- Nos permite inyección de dependencias de manera simple.
- Binding de eventos.
- REST client de manera simple y sencilla.
- Genera un código de menos de 150k

<https://github.com/androidannotations/androidannotations/wiki#features>

Dependency injection: inject views, extras, system services, resources, ...

Simplified threading model: annotate your methods so that they execute on the UI thread or on a background thread.

Event binding: annotate methods to handle events on views, no more ugly anonymous listener classes! REST client: create a client interface, AndroidAnnotations generates the implementation.

No magic: As AndroidAnnotations generate subclasses at compile time, you can check the code to see how it works.

AndroidAnnotations provide those good things and even more for less than 150kb, without any runtime perf impact!

¿Cómo funciona?

Vamos a ver como añadir una anotación, [recuerda lo visto en este enlace](#).

Además deberemos modificar el AndroidManifest

<https://github.com/androidannotations/androidannotations/wiki/HowItWorks>

AndroidAnnotations works in a very simple way. It automatically adds an extra compilation step that generates source code, using the standard Java Annotation Processing Tool.

What source code? For each enhanced class, for example each `@EActivity` annotated activity, a subclass of this activity is generated, with the same name plus an underscore appended at the end.

```
package com.some.company;
@EActivity
public class MyActivity extends Activity {
    // ...
}
```

vs.

```
package com.some.company;
public final class MyActivity_ extends MyActivity {
    // ...
}
```

That is the reason why you must add `_` to your activity names in `AndroidManifest.xml`:

```
<activity android:name=".MyListActivity_" />
```

Más info

En este enlace podremos ver diferentes usos de las anotaciones (las iremos viendo a lo largo de los tres cursos de `AndroidAnnotations` con más profundidad):

<https://github.com/androidannotations/androidannotations/wiki/Cookbook>

Apps

Algunos ejemplos de apps que usan `AndroidAnnotations`:

- Lidl
- Timely
- Starbucks Card

<https://github.com/androidannotations/androidannotations/wiki/Apps-using-AndroidAnnotations>