

Evolución Android

Del framework a la
supervivencia del más fuerte



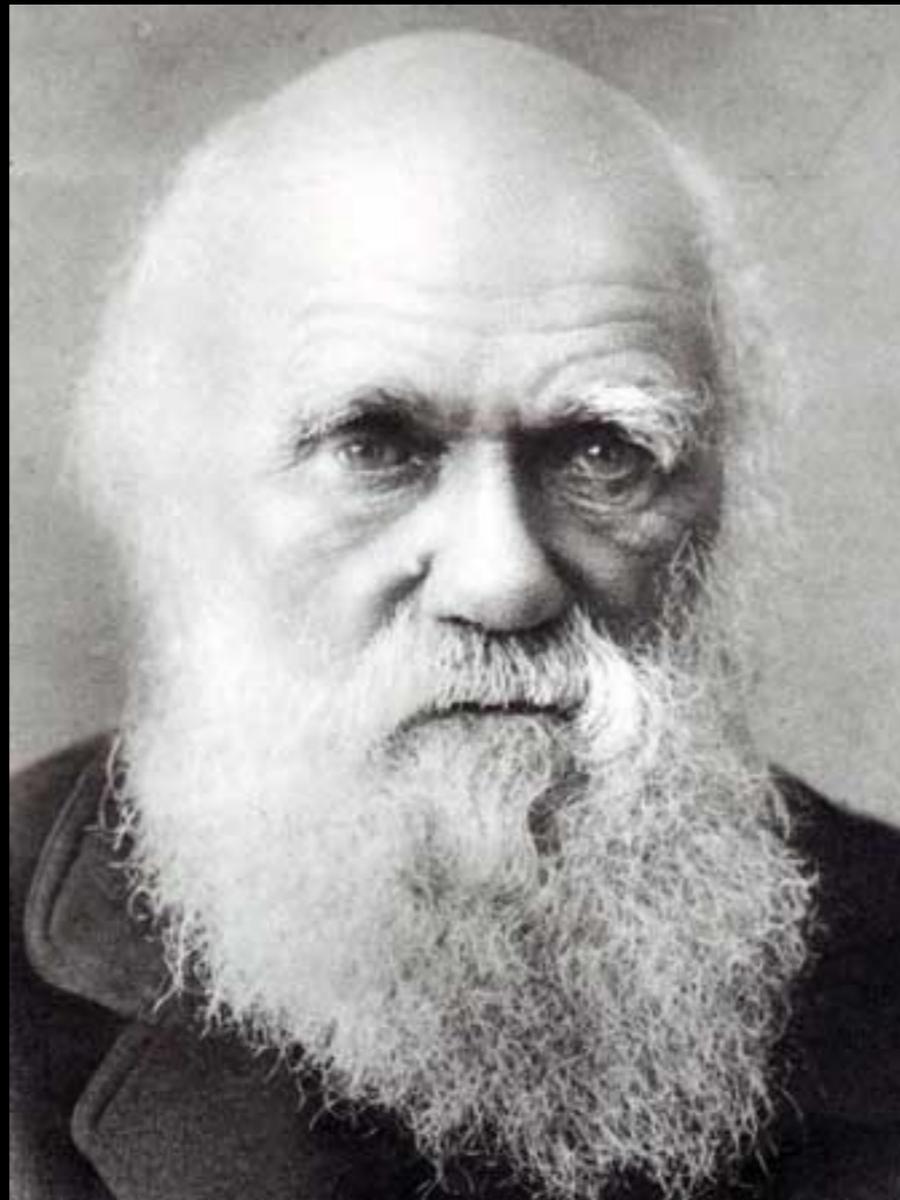
Who?



Rubén Serrano Núñez (@Akelael)

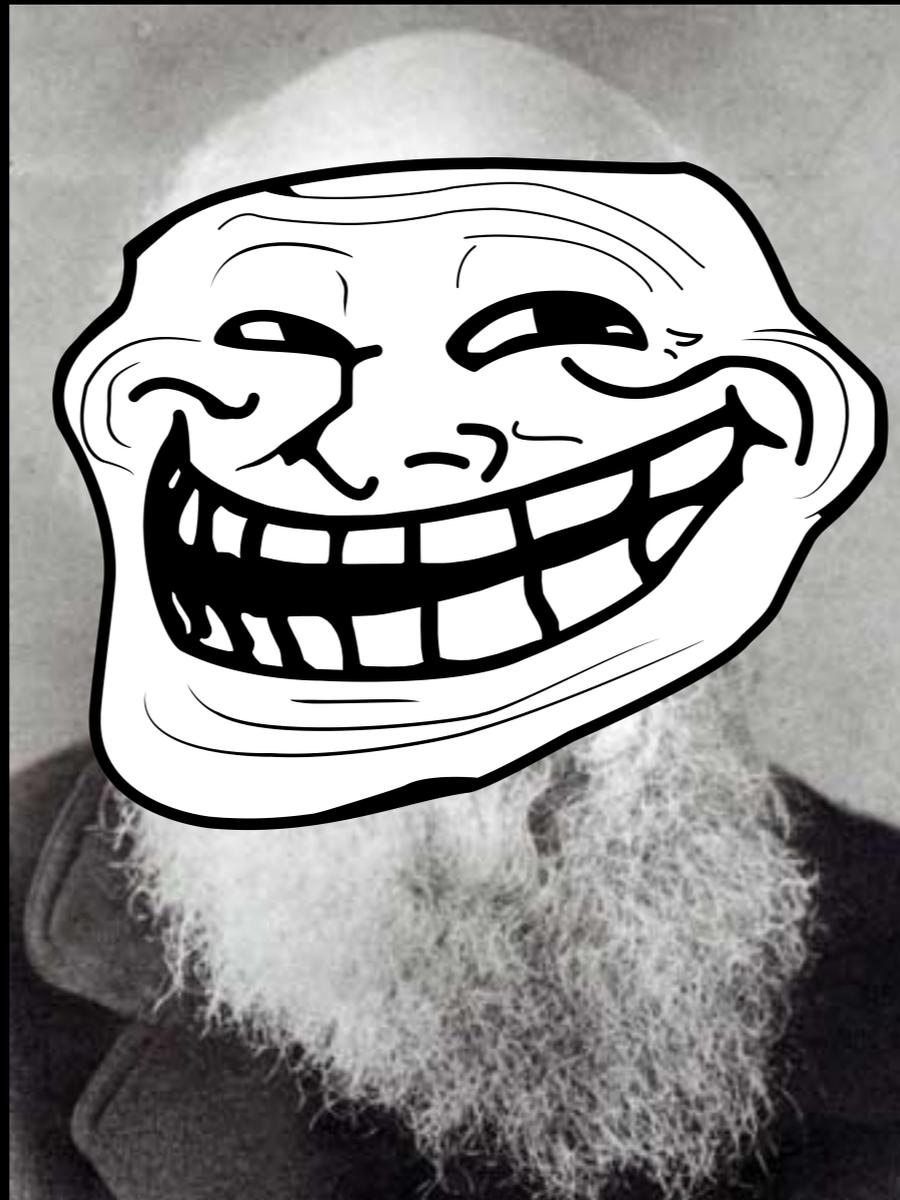
Lead Android Developer @RedboothHQ

GDG Barcelona Organizer



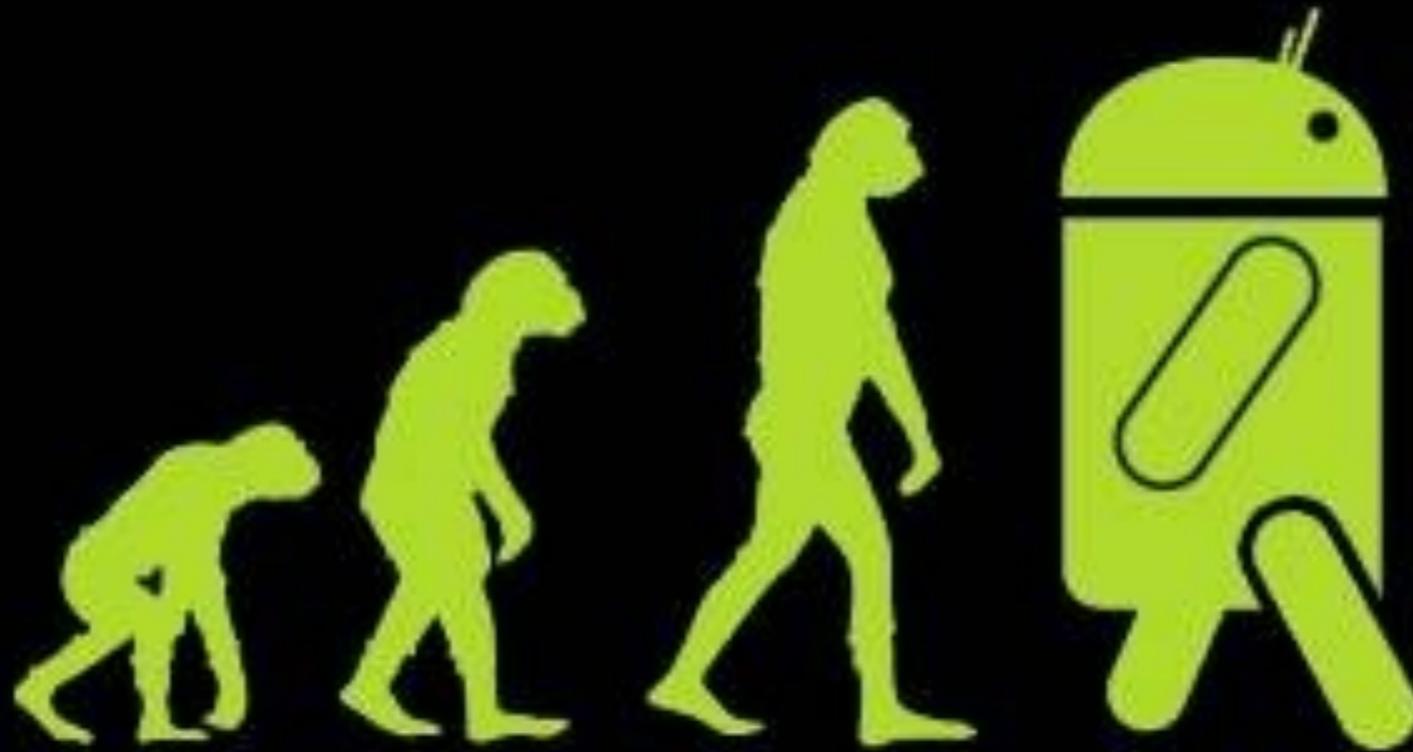
It is not the strongest of the species that survives, nor the most intelligent, but rather the one most adaptable to change.

Charles Darwin



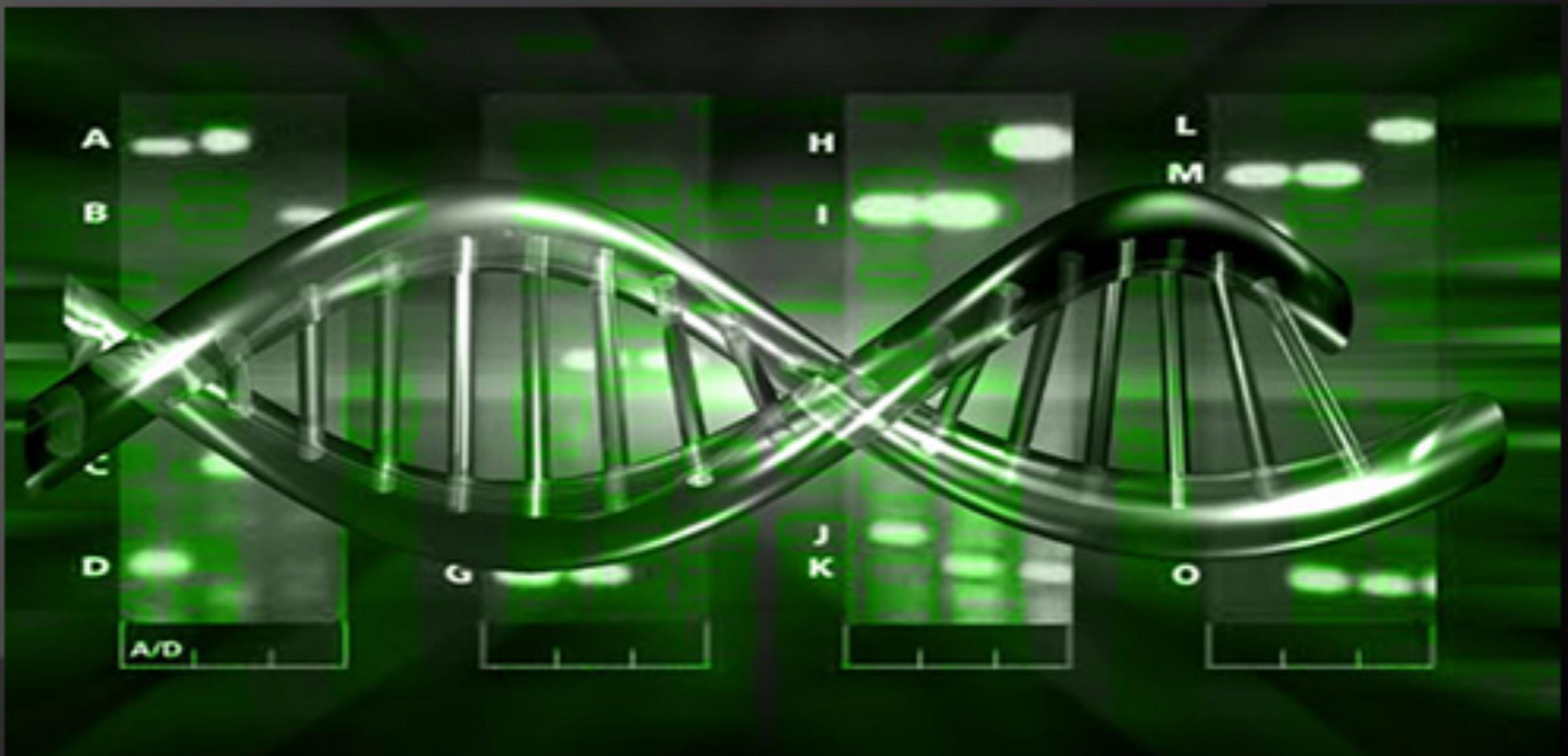
It is not the strongest of the species that survives, nor the most intelligent, but rather the one most adaptable to change.

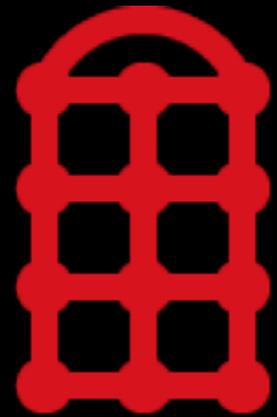
Charles Darwin



It's the same for software development

Cuestión de genes ...

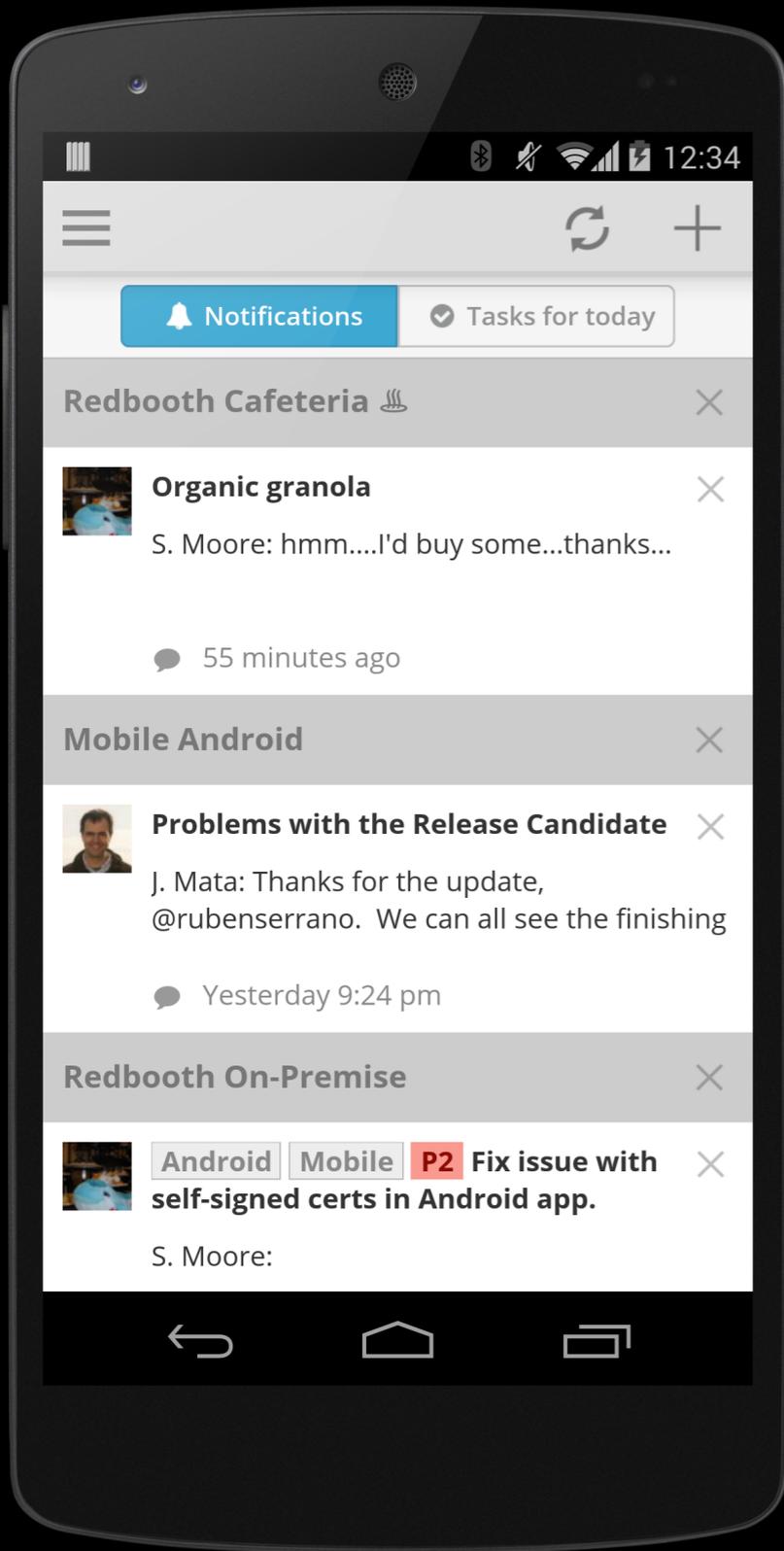




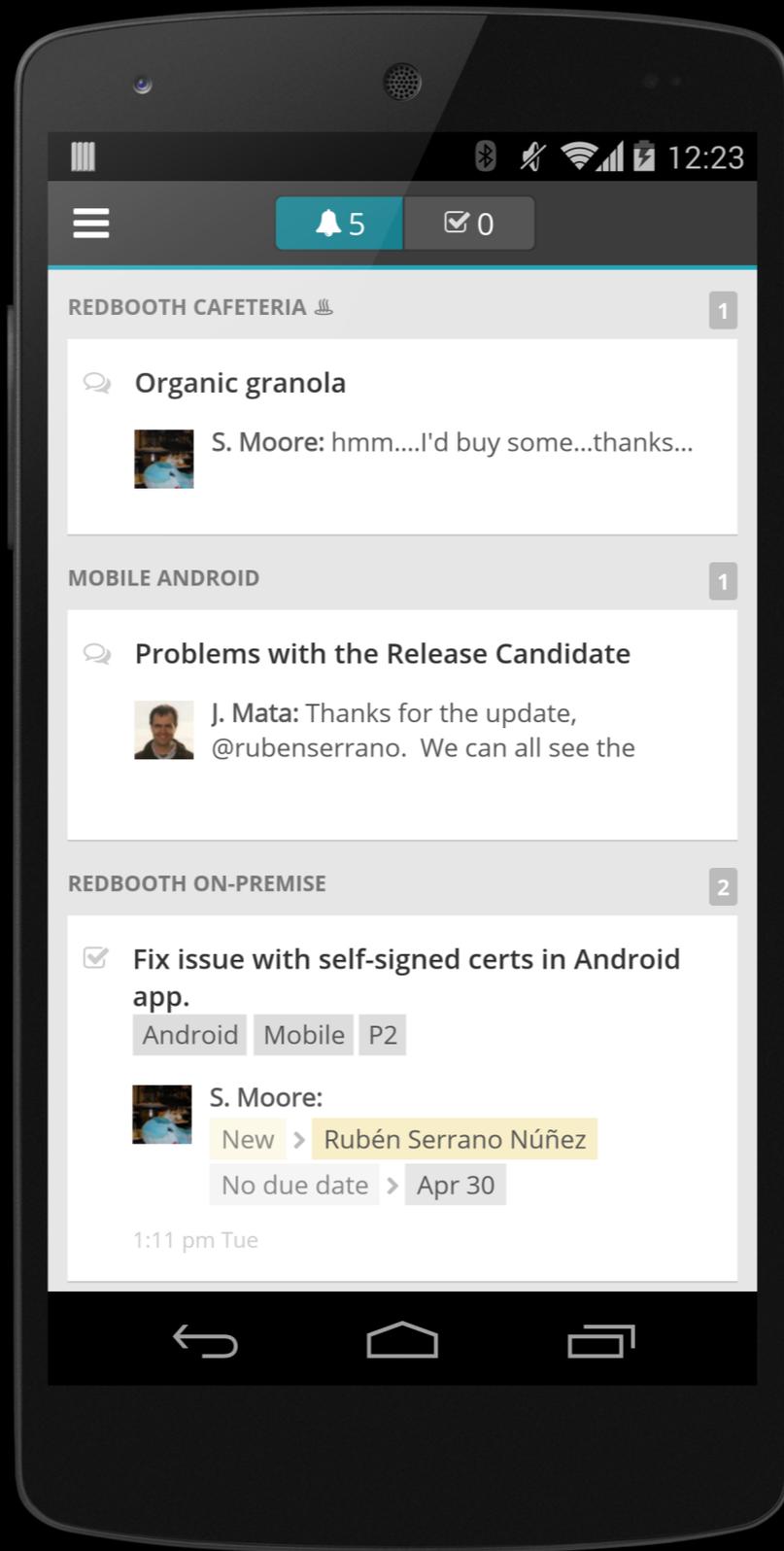
Redbooth

work smarter

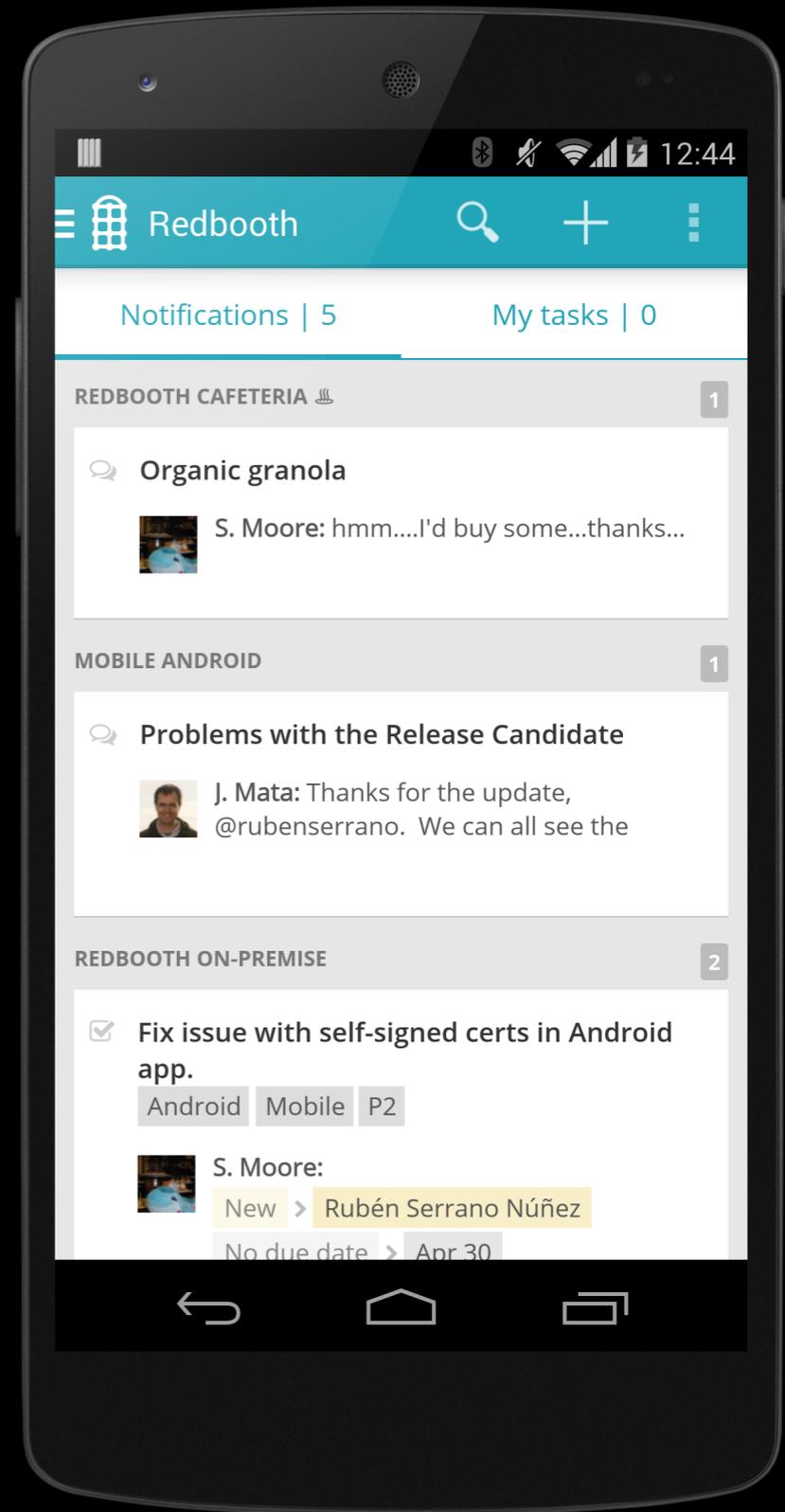
- Junio 2012 - Versión Android inicial
- Julio 2013 - Cambio del equipo de desarrollo
- Noviembre 2013 - Desarrollo Android in-house



Teambox



Redbooth Beta



Redbooth

WTF?! Un mes?!?!





“The Flying Spaghetti Monster approves Spaghetti Code”

“It’s the maintenance!”

Creationists

Evolutionists

vs





Creacionista



Creacionista

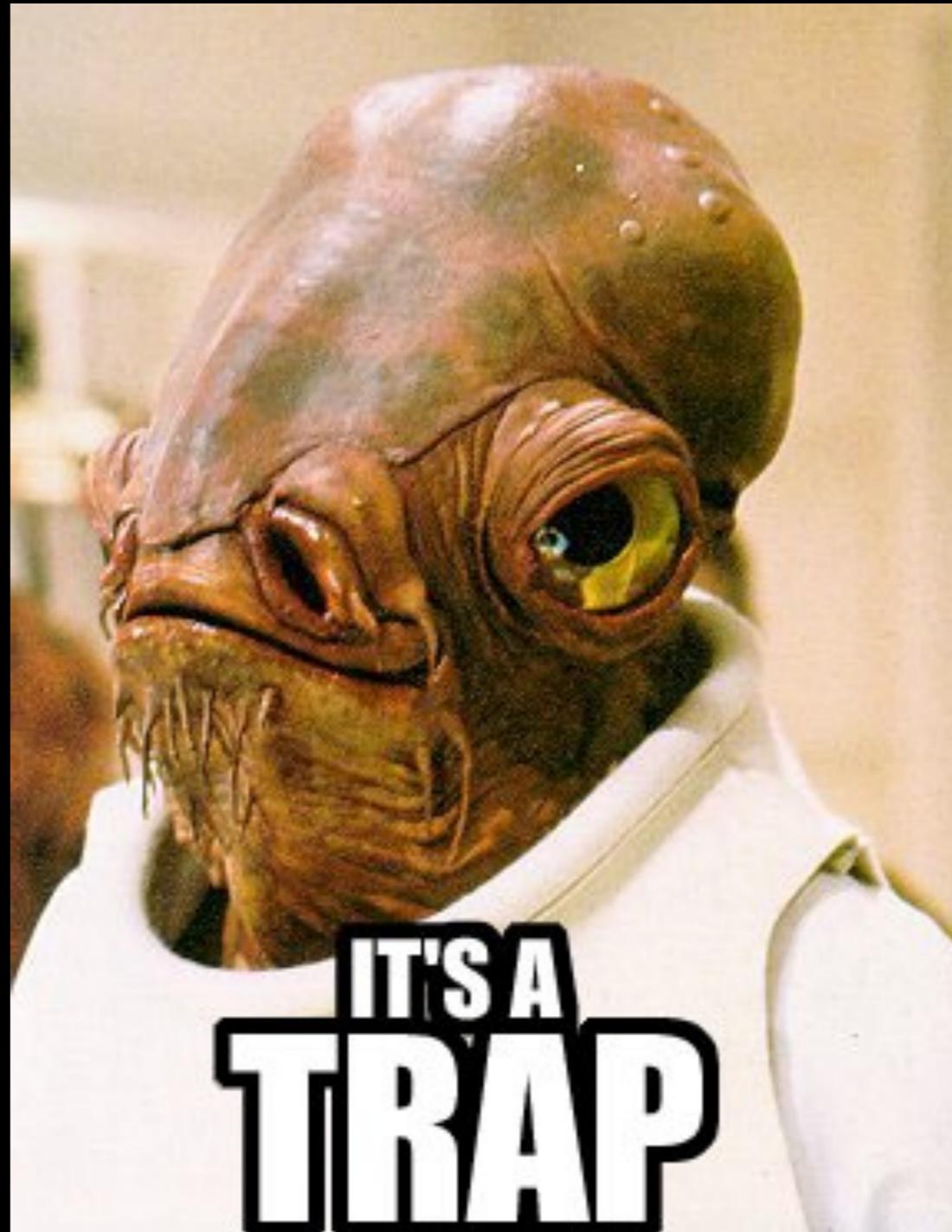


Evolucionista

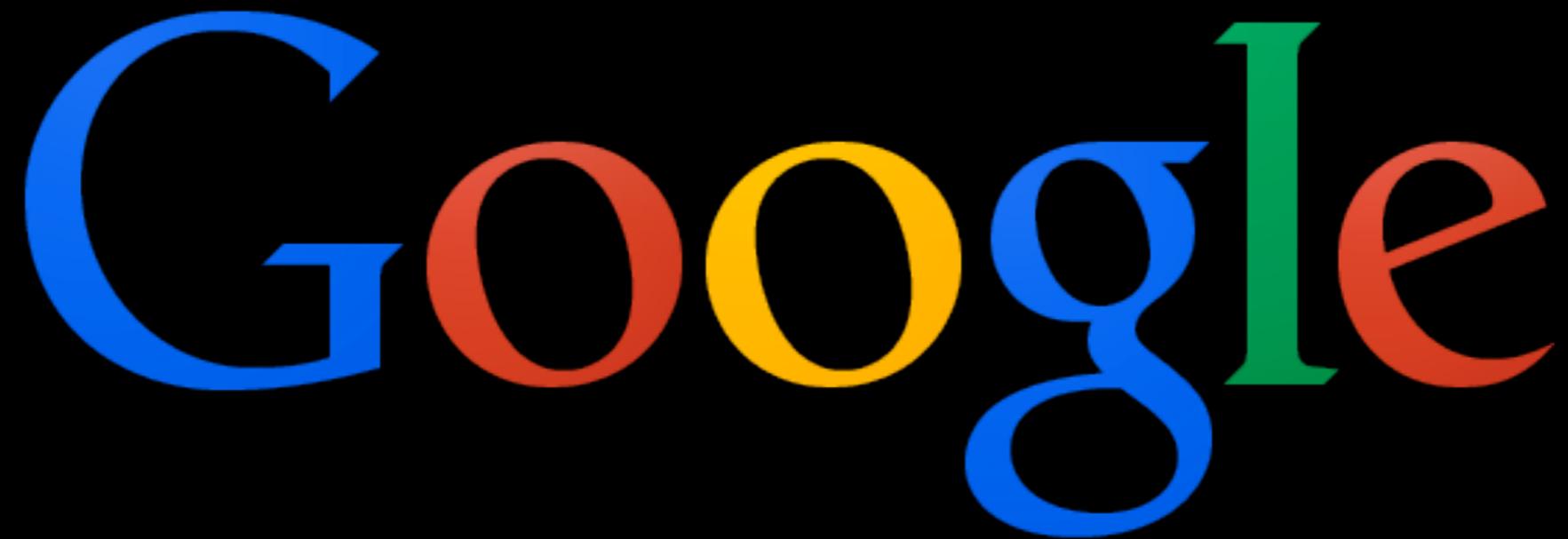


Evolucionista

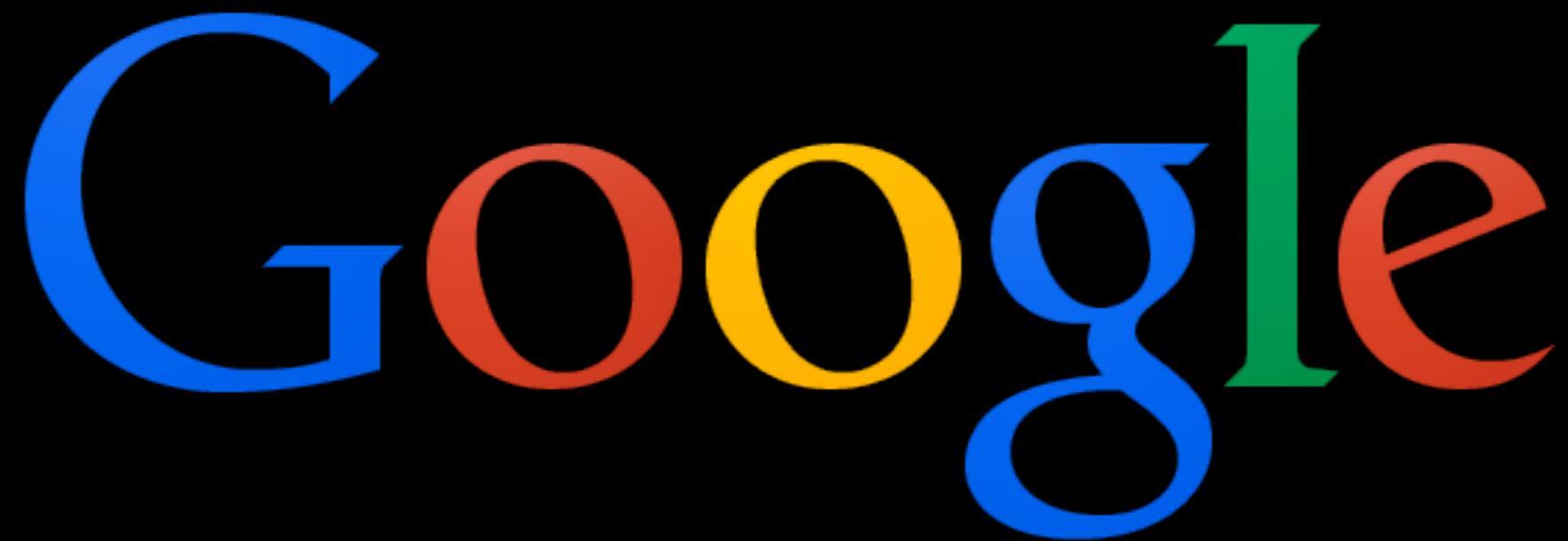
Los peligros de Android



El más peligroso de todos

The Google logo is displayed in its characteristic multi-colored font. The letters are: 'G' (blue), 'O' (red), 'O' (yellow), 'g' (blue), 'l' (green), and 'e' (red).

El más peligroso de todos

The Google logo is displayed in its characteristic multi-colored font. The letters are: 'G' in blue, the first 'O' in red, the second 'O' in yellow, the first 'g' in blue, the 'l' in green, and the 'e' in red.

bueno, vale, no lo es

Algunas trastadas

- **El framework mutante (fragmentación del SO)**

Algunas trastadas

- El framework mutante (fragmentación del SO)
- **“No uses singleton, usa Application”**

**Lo puse todo
en Application
y ...**

**¡Quita eso de
ahí, guarro!**



Algunas trastadas

- El framework mutante (fragmentación del SO)
- “No uses singleton, usa Application”
- **Librería de compatibilidad**

Algunas trastadas

- El framework mutante (fragmentación del SO)
- “No uses singleton, usa Application”
- Librería de compatibilidad
- **“Usa RelativeLayout... mejor no lo uses”**

God Activity



God Activity

- Descargar datos de un servidor
- Parser JSON
- Guardar datos en la BBDD
- Recuperar datos de la BBDD
- Mostrar datos en la pantalla

God Activity

- ~~Descargar datos de un servidor~~
- ~~Parser JSON~~
- ~~Guardar datos en la BBDD~~
- ~~Recuperar datos de la BBDD~~
- **Mostrar datos en la pantalla**

God Activity: el Origen del Mal

startManagingCursor(Cursor c)

This method allows the activity to take care of managing the given Cursor's lifecycle for you based on the activity's lifecycle.



SOLID?

- **S**ingle Responsibility Principle
- **O**pen/closed Principle
- **L**iskov Substitution Principle
- **I**nterface Segregation Principle
- **D**ependency Inversion Principle

God Activity: el Final

Desacoplar lógica y modelo:

- SOLID
- MVP, MVC, MVVM, etc
- Ports & Adapter, Clean Architecture, etc

AsyncTask

- Mal usada, interfiere con el código de la Activity
- Solía utilizarse como excusa para hacer la carga del modelo en la Activity
- Presenta problemas de performance

AsyncTask + startManagingCursor



Loaders!!



Loaders Callbacks

- Modelo acoplado a la activity (~~SRP~~)
- CursorLoader
- Mismos problemas de rendimiento que AsyncTask
- Da rabia, porque realmente funciona bien

Fixing Loaders?

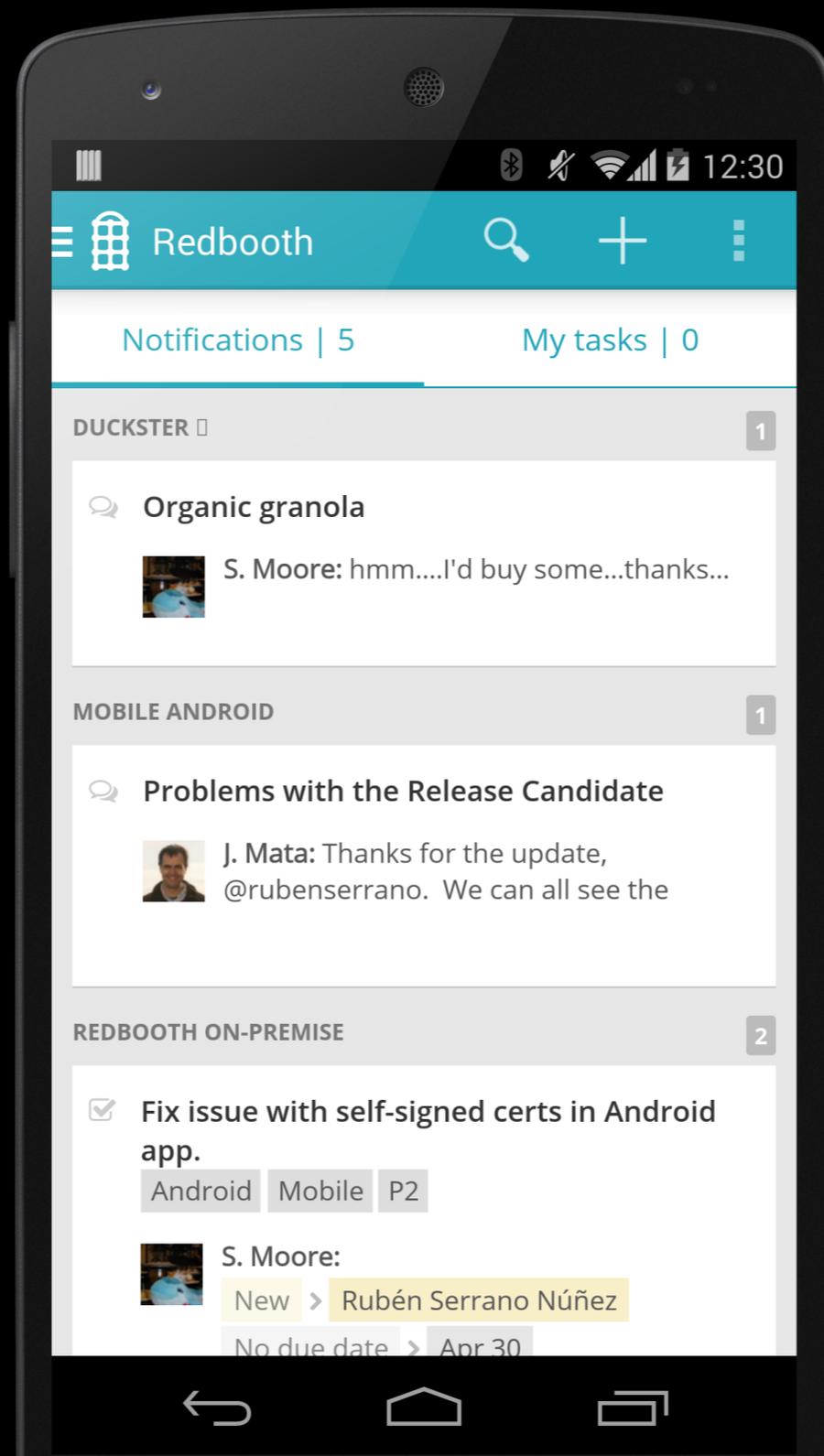
1. No uses loaders
2. Desacopla la lógica de negocio y el modelo
3. Aplica la carga asíncrona en el modelo
4. Utiliza un bus de evento o `LocalBroadcastReceiver` para anunciar eventos de carga de datos

Fixing Loaders?



I fixed it!

ViewHolder



ViewHolder

- Es un patrón que corrompe el código el adapter
- Soluciona un problema que ya no existe
- **Alternativa:** Renderers®

God Fragment

Remember?



SauronActivity

“Una Activity para gobernarlos a todos; una Activity para encontrarlos; una Activity para atraerlos a todos y atarlos a las tinieblas”

- El Señor de los Fragments

SauronActivity

- Porque no tener una sola activity y muchos Fragments?

SauronActivity

Múltiples fragments ==

- Horda de loaders! (degradación de la performance)
- Memory leaks
- God Activity

ContentProviders

- Ideado para compartir datos con terceras aplicaciones
- Obligatorio para CursorLoaders
- No simplifica el código de acceso a la BBDD
- God ContentProvider
- Problemas con ORMs

SyncAdapter

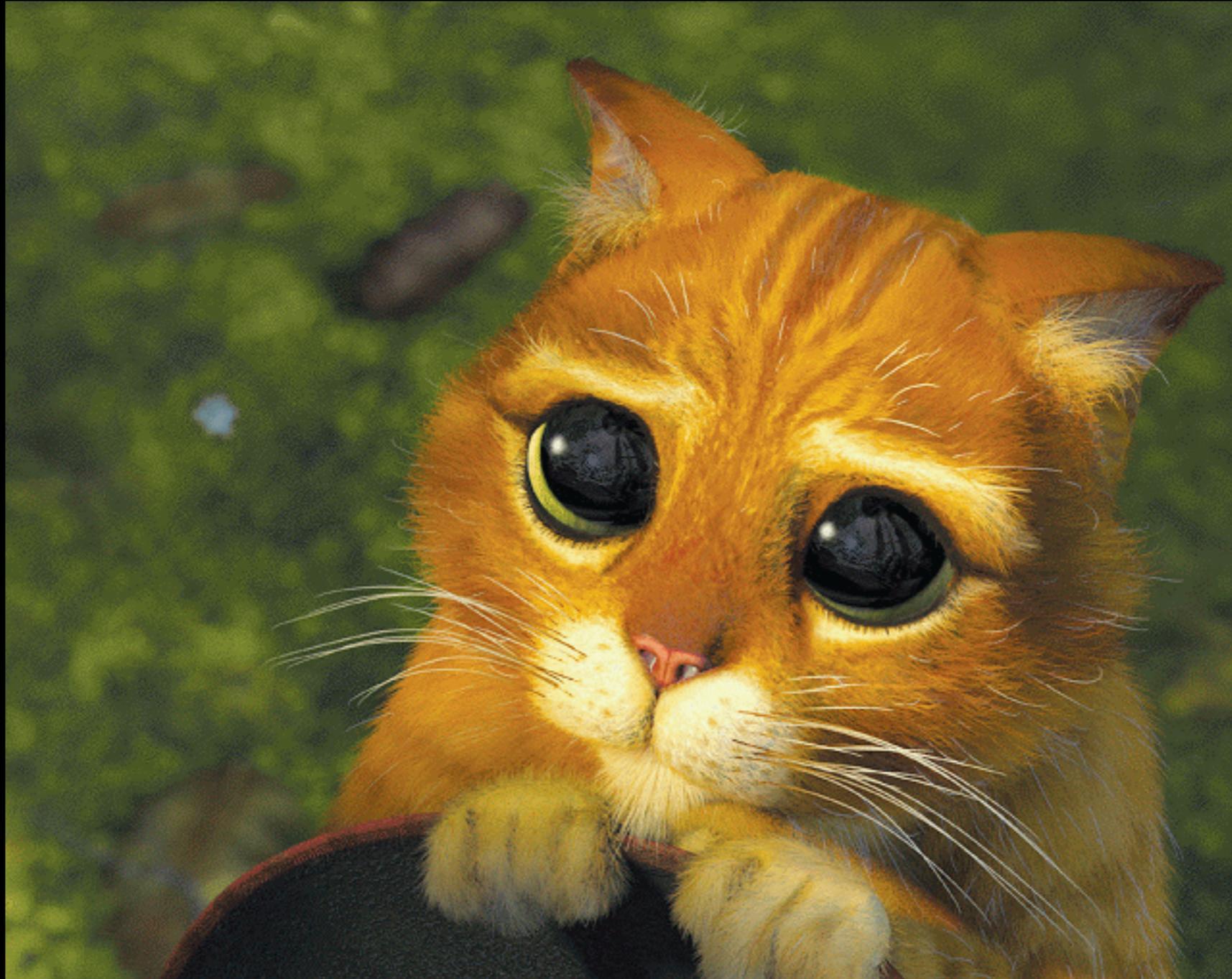
- Objetivo muy loable: sincronizar BBDD con servidor de forma que ahorre batería
- Problema: cuándo sucede?

hope



Mejorando la genética del código

- SOLID
- Testing, mejor si es con JUnit
- Patrones/arquitecturas que favorezcan lo anterior: Presenter First, Ports & Adapters, Clean Architecture, Onion Layer, etc.



Remember!

¡Muchas gracias!

Q & A

Rubén Serrano Núñez

@Akelael

plus.google.com/+RubénSerrano