


Open · Opened 3 weeks ago by  **Rubén Montero**

Ciclo de vida: onStop

Resumen

- Veremos las posibilidades en el ciclo de vida de una actividad Android
- Implementaremos `onStop` en nuestro `MainActivity`

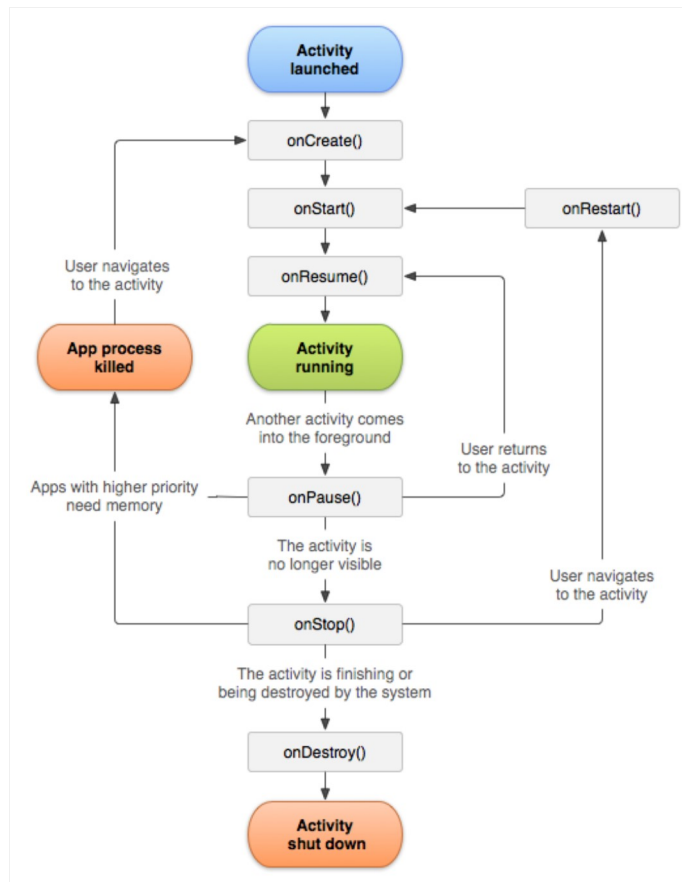
Descripción

El [ciclo de vida](#) de una actividad Android consiste en los posibles estados que puede atravesar mientras el usuario disfruta de ella.

Inicialmente se creará y entonces `onCreate` será invocado.

¿Qué más puede pasar?

Ciclo de vida



Nosotros vamos a utilizar `onStop`. Sirve para controlar que la *app* ha dejado de estar en primer plano, igual que `onDestroy`. La diferencia es que si hay escasez de memoria RAM, el sistema Android puede matar nuestra aplicación *sin* invocar `onDestroy`.

La tarea

En `MainActivity.java`, añade un nuevo método *dentro* de la clase. **Teclea**:

```
onSto
```

Verás cómo Android Studio autocompleta por nosotros. **Dale** a ENTER para que eso pase:

```
public class MainActivity extends AppCompatActivity {
```

```
private Context context = this;

@Override
protected void onCreate(Bundle savedInstanceState) { /* ... */ }

@Override
protected void onStop() {
    super.onStop();
    // ...
}
}
```

A continuación, *dentro* de `onStop` (después de `super.onStop()`), **lanza** un `Toast` con el texto `"MainActivity fue a segundo plano"`.

Una vez lo hayas escrito, **prueba** tu `app` en el emulador.

¿Qué ves si haces click en el botón inferior para empezar la otra actividad?

¿Qué ves si pulsas el botón de *Atrás* un par de veces para salir de la aplicación

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



Rubén Montero @ruben.montero changed milestone to [%Sprint 1](#) 3 weeks ago



Ania Blanco @ania.blanco mentioned in commit [6ccfb733](#) 1 week ago