

Ciclo de vida das Activities

A Activity é um componente do Android que representa uma tela no aplicativo, contendo elementos visuais como textos, botões, e menus. Todo aplicativo precisa de pelo menos uma Activity, geralmente chamada de MainActivity. Aplicativos podem ter várias Activities, dependendo da necessidade do projeto.

Estados do ciclo de vida da Activity

As Activities possuem métodos de **callback** que controlam os estados do ciclo de vida, que vão desde a criação até a finalização. Esses estados permitem controlar o comportamento da Activity conforme o usuário navega pelo aplicativo.

Os principais estados da Activity são:

onCreate(): Criação inicial da Activity. Executa configurações estáticas como a criação de layouts. Após esse método, a Activity vai para o estado **onStart()**.

onStart(): A Activity torna-se visível, pronta para ser utilizada.

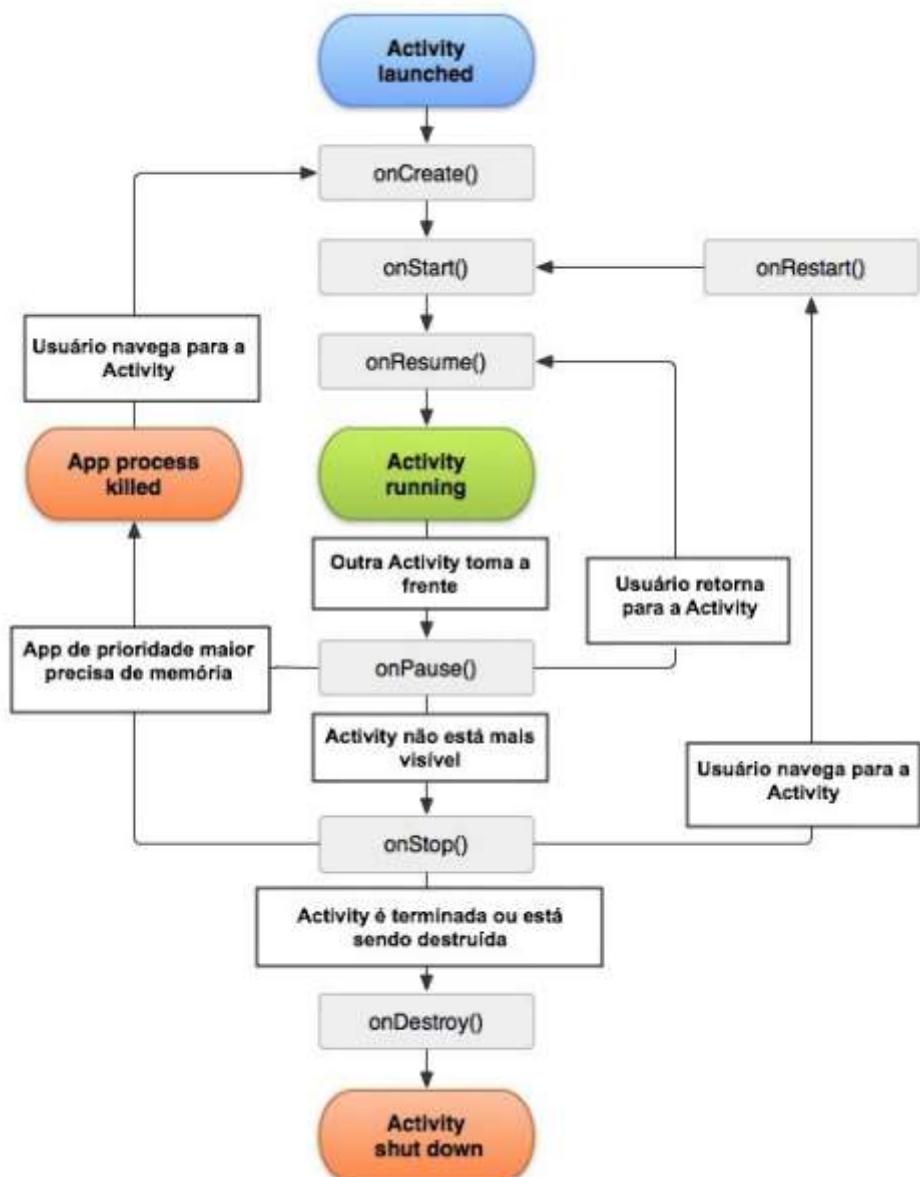
onResume(): O aplicativo entra em primeiro plano, permitindo interação com o usuário.

onPause(): Ocorre quando algo interrompe o aplicativo, como uma ligação. O aplicativo ainda está visível, mas não em foco total.

onStop(): A Activity torna-se invisível, liberando recursos desnecessários.

onRestart(): Chama-se quando a Activity é retomada após ser pausada ou parada.

onDestroy(): Ocorre quando a Activity é destruída, seja por encerramento do aplicativo ou mudança de configuração.



Ciclos de vida na prática

As alterações no ciclo de vida ocorrem automaticamente no Android, mas podem ser visualizadas por meio de logs no Android Studio. Usar o `Log.i()` em cada estado da Activity permite monitorar seu comportamento em tempo real via Logcat.

Essas alterações das etapas do ciclo de vida acontecem por trás do funcionamento do nosso aplicativo e por isso não vemos elas acontecendo normalmente. Porém, podemos utilizar um recurso muito bacana do Android Studio que são os Logs.

Através deles, podemos exibir mensagens em tempo real a respeito do funcionamento do nosso aplicativo e visualizá-las por meio da janela Logcat.

Para utilizar essa ótima ferramenta, vamos utilizar o log de informação (Log.i()) e inseri-lo no código da Activity em cada um dos estados. O código final dessa parte da Activity ficaria assim:

```
const val TAG = "Estado da Activity"

class ListaProdutosActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(binding.root)
        configuraRecyclerView()
        configuraFab()
        Log.i(TAG, "onCreate: ")
    }

    override fun onStart() {
        super.onStart()
        Log.i(TAG, "onStart: ")
    }

    override fun onResume() {
        super.onResume()
        adapter.atualiza(dao.buscaTodos())
        Log.i(TAG, "onResume: ")
    }

    override fun onPause() {
```

```

        super.onPause()
        Log.i(TAG, "onPause: ")

    }

    override fun onStop() {
        super.onStop()
        Log.i(TAG, "onStop: ")
    }

    override fun onRestart() {
        super.onRestart()
        Log.i(TAG, "onRestart: ")
    }

    override fun onDestroy() {
        super.onDestroy()
        Log.i(TAG, "onDestroy: ")
    }

```

- Quando executamos o aplicativo pela primeira vez, são chamados os callbacks `onCreate()`, `onStart()` e `onResume()`, que, respectivamente, fazem a criação, exibição e liberação para interação da Activity;
- Quando clicamos no botão de adicionar um novo produto, o aplicativo sai da Activity principal e vai para a Activity de cadastro de produto e, nesse momento, são chamados o `onPause()` e o `onStop()`, que indicam que a Activity ficou fora do plano principal de visualização e logo em seguida ficou totalmente invisível para a pessoa usuária;
- Ao voltarmos para a Activity principal, é chamado o callback `onRestart()`, que é acionado quando a Activity está no estado

de parada e é iniciada novamente, onde são acionados os callbacks de inicialização `onStart()` e `onResume()`;

- Verificamos, também, o comportamento quando há rotação da tela do aparelho. Nesse caso, o sistema chama o callback `onDestroy()` e destrói a Activity e, em seguida, cria automaticamente uma nova instância para esta nova configuração, chamando os callbacks `onCreate()`, `onStart()` e `onResume()`;
- Por fim, temos o fechamento completo do aplicativo, em que o sistema chama o `onDestroy()` e finaliza o ciclo daquela Activity.