introdução - android

NDK - conjunto de ferramentas para desenvolvimento em C/C++ para android



Arquitetura do sistema

**Kernel Linux**

•Nível mais baixo da arquitetura, é a base para todas as demais camadas.

•Oferece um nível de abstração entre os dispositivos periféricos e suas especificidades, evitando assim a problema de ter que fazer a manipulação direta.

•O núcleo também é responsável pelo gerenciamento de rede e pelos drivers que realizam a interface com o hardware periférico.

**Bibliotecas**

Acima do kernel, consiste num conjunto de bibliotecas para:

•Armazenamento e compartilhamento de dados (SQLite)

•Navegação web (Webkit)

•Bibliotecas da ling. C (libC)

•Reprodução e gravação de conteúdo multimídia (áudio e vídeo)

•Segurança de rede (SSL)

**Android Runtime**

•Consiste na terceira camada da arquitetura e atua em conjunto com a segunda.

•Nela encontramos a DalvikVirtual Machine, uma espécie de JVM especialmente projetada e otimizada para o Android e seus apps.

Cada app roda seu próprio processo, que é uma instância da DalvikVM.

•Contém também as bibliotecas básicas que permitem aos programadores utilizarem a linguagem Java no processo de desenvolvimento.

**Framework**

•Bibliotecas baseadas em Java que são específicas para o desenvolvimento em Android.

•Oferecem recursos que facilitam a criação de interfaces gráficas, manipulação de componentes e acesso a dados.

•Servem de classes base para construção do app

**Activities: Ciclo de Vida**

•A classe *Activity* define vários *callbacks*para manipular seus eventos, que são:

•**onCreate():**quando a activity é criada

•**onStart():**sempre que se torna visível na tela

•**onResume():**quando o usuário inicia a interação

•**onPause():**quando a activity não recebe mais entrada do usuário e não pode executar qualquer trecho de código.

•**onStop():**quando a activity não está mais visível

•**onRestart():**chamada no retorno da activity após ser parada

•**onDestroy():**invocada antes de ser destruída pelo sistema

Layouts

LinearLayout – alinhamento de componentes filhos na horizontal ou vertical android:orientation

RelativeLayout – organiza os objetos filhos a direita e esquerda em relação ao principal

WebView – exibição de conteúdo HTML através do método loadUrl(String)

ListView – componentes organizados de forma se tornarem uma lista

GridView – organiza os itens a formar uma matriz bidimensional