

# Resumo

## 1. Manifesto (Manifest):

**Arquivo:** AndroidManifest.xml

**Finalidade:** O manifesto é um arquivo XML obrigatório que descreve as informações essenciais sobre o aplicativo, como nome do pacote, permissões, componentes (atividades, serviços, receptores de transmissão) e configurações de lançamento.

**Finalidade:** O arquivo **AndroidManifest.xml** é essencial para todo projeto Android. Ele descreve informações críticas sobre o aplicativo, como suas permissões, componentes (atividades, serviços, receptores de transmissão), versão do aplicativo, configurações de segurança, ícones e muito mais. O Android usa o manifesto para entender como o aplicativo deve ser executado e interagir com o sistema operacional.

**Particularidades:** O manifesto é fundamental para a definição da estrutura do aplicativo e suas interações com o sistema operacional Android.

## 2. Código-fonte (Source Code):

**Finalidade:** Esta seção contém os arquivos que implementam as funcionalidades do aplicativo, incluindo atividades, fragmentos, adaptadores e classes de utilidade.

**Finalidade:** Essa seção contém o código-fonte real do aplicativo Android. Os principais tipos de arquivos armazenados aqui são

**Particularidades:** O código-fonte é onde os desenvolvedores escrevem a lógica do aplicativo, respondendo a eventos e interações do usuário.

**Arquivos:** Os arquivos de código-fonte Java e/ou Kotlin que compõem a lógica do aplicativo.

**Arquivos Java (.java):** Contêm a lógica do aplicativo, como atividades, fragmentos, classes de utilidade e muito mais.

**Arquivos Kotlin (.kt):** Kotlin é outra linguagem de programação oficialmente suportada pelo Android, e os arquivos **.kt** também são usados para definir a lógica do aplicativo.

## 3. Recursos (Resources):

**Finalidade:** Os recursos são usados para definir a aparência, o conteúdo e a experiência visual do aplicativo. Eles podem ser referenciados no código-fonte para personalizar a interface do usuário.

**Finalidade:** Os recursos são arquivos que não são código-fonte, mas são usados para fornecer conteúdo estático, layouts, strings, imagens, ícones e outros ativos para o aplicativo. Alguns dos tipos de arquivos comuns nesta seção incluem

**Particularidades:** Organizar recursos de forma eficiente é importante para manter a coesão e a facilidade de manutenção do aplicativo.

**Arquivos:** Recursos como layouts XML, imagens, arquivos de string, estilos, arquivos de animação, valores de dimensão, etc.

**Layouts XML (.xml):** Descrevem a aparência das interfaces do usuário (UI) do aplicativo.

**Strings (.xml):** Armazenam todas as strings usadas no aplicativo, facilitando a localização (tradução) do aplicativo.

**Imagens e ícones (.png, .jpg, .svg, etc.):** Gráficos usados na interface do usuário.

**Recursos de animação (.xml):** Define animações que podem ser aplicadas a elementos da interface do usuário.

**Recursos de valores (.xml):** Armazenam valores, cores, dimensões e estilos usados em todo o aplicativo.

#### **4. Gradle:**

**Finalidade:** Os arquivos Gradle definem as configurações de compilação e dependências do projeto Android. O arquivo do módulo configura as informações específicas do aplicativo, enquanto o arquivo do projeto define as configurações globais do projeto.

**Finalidade:** O sistema de construção do Android é gerenciado pelo Gradle, e esta seção contém arquivos relacionados à configuração e construção do projeto. Os arquivos mais importantes aqui são:

**Particularidades:** O Gradle automatiza a compilação, a construção e a gestão de dependências, facilitando o desenvolvimento e a implantação do aplicativo.

**Arquivos:** build.gradle (módulo) e build.gradle (projeto).

**build.gradle (Módulo do app):** Define as dependências do aplicativo, configurações de compilação e plugins específicos do módulo do aplicativo.

**build.gradle (Projeto):** Configurações globais para o projeto, incluindo versões do Android SDK e outros aspectos gerais.

**settings.gradle:** Lista os módulos do projeto que serão construídos.