

Dispositivos M veis

Aula 15 - Distribui o de Aplica es

Apresentação

Chegamos à última aula da disciplina! Foi um longo caminho percorrido até agora, no qual aprendemos diversos conteúdos interessantes para o desenvolvimento de aplicações Android. Agora que já somos capazes de desenvolver um aplicativo para dispositivos móveis, com funcionalidades interessantes e úteis, vamos aprender como distribuir essa aplicação e quais as formas de comercialização mais conhecidas. Na aula de hoje, veremos todos os passos para tornar uma aplicação instalável em qualquer aparelho que utiliza o sistema operacional Android. Passo a passo, estudaremos como preparar tudo e testar, a fim de gerarmos uma aplicação que possa ser distribuída por e-mail, site ou até mesmo pelo Google Play! Vamos adiante.



Vídeo 01 - Apresentação

Objetivos

Ao final desta aula, você deverá ser capaz de:

- Preparar uma aplicação para publicação.
- Gerar um arquivo assinado, pronto para ser instalado em qualquer dispositivo Android.
- Publicar o arquivo em um site, por e-mail ou no Google Play.

Preparando a Aplicação para Distribuição

Chegamos até aqui e, ao longo deste curso, conseguimos habilidades para desenvolver uma aplicação. Tendo a aplicação pronta, como poderemos distribuí-la? Bem, o primeiro passo nesse processo é preparar a aplicação para ser distribuída. Essa preparação envolve diversos passos, alguns no código, outros não.

Vamos acompanhar cada um desses passos nas próximas subseções!

Reúna os Materiais

Como você deve imaginar, para se lançar uma aplicação não é necessário apenas o código desta aplicação. Outros materiais também são necessários para garantir uma aplicação de qualidade. O primeiro material extra, necessário para a publicação de uma aplicação, é uma chave privada, para, por meio dela, assinar a aplicação. Veremos como gerar essa chave em uma subseção posterior, quando tivermos aprendendo a compilar uma aplicação para distribuição.

Outro material igualmente importante para a publicação de sua aplicação é o ícone dela. O mercado de aplicações está lotado de aplicações com diferentes propósitos, desenvolvidas de diferentes maneiras. Nesse mercado tão cheio, sua aplicação precisa se destacar em relação às outras. Isso é feito, principalmente, por meio do ícone da sua aplicação e da descrição da mesma. Tenha um ícone que se encaixe nos padrões do Android e que chame a atenção do usuário. Também é importante que o ícone seja desenvolvido segundo o guia para desenvolvimento de ícones do Android. Esse guia pode ser encontrado no [sítio http://www.google.com/design/spec/style/icons.html#](http://www.google.com/design/spec/style/icons.html#). Perceba que o ícone, nas versões mais atuais do Android, deve levar em consideração, assim como os outros recursos, as diferentes opções de densidade de tela que estão disponíveis. Tendo o ícone da aplicação, deve-se continuar a aquisição dos materiais para o lançamento dela com o desenvolvimento de uma End User License Agreement (EULA). A EULA é uma licença distribuída junto às aplicações para garantir que os usuários que a utilizam concordem com os termos necessários que aquela aplicação impõe. No

caso de aplicações pequenas, ou que serão distribuídas apenas para amigos, isso não é exatamente necessário, mas conhecer a existência dessa licença é importante para qualquer desenvolvedor. Pesquise mais sobre a EULA, caso tenha interesse!

Por fim, devem-se reunir os outros materiais que sejam necessários para o lançamento da aplicação. Esses materiais não são necessários para o funcionamento da aplicação, mas sim para a sua divulgação, seja no Google Play ou em qualquer outra forma de distribuição. Dentre esses materiais estão incluídos os textos promocionais da aplicação, vídeos de demonstração, screenshots e qualquer outro material de divulgação que o desenvolvedor julgue necessário. No caso do alvo da aplicação ser o Google Play, alguns desses materiais passam a ser obrigatórios, como uma versão maior do ícone da aplicação, um texto promocional (preferencialmente em inglês, além do idioma padrão) e, no mínimo, duas capturas de tela feitas dentro da aplicação. Esse material serve para ajudar usuários a conhecer melhor o seu aplicativo dentro da loja, antes de comprá-lo ou baixá-lo, e são de grande importância para o sucesso da aplicação. Elabore um texto explicativo e bem escrito que descreva a aplicação. Esse texto será utilizado para que os usuários saibam do que a aplicação se trata.

Uma vez que todos os materiais necessários para o lançamento da aplicação tenham sido adquiridos, podemos prosseguir com a preparação do código para o lançamento. Deixe de lado, por enquanto, a divulgação e vamos voltar a mexer no código da aplicação.

Prepare sua Aplicação para o Lançamento

Neste momento, vamos configurar os outros aspectos necessários para o lançamento da aplicação. O primeiro desses aspectos que devemos prestar atenção é em relação ao nome do pacote no qual a aplicação está contida.

Escolha o Nome do Pacote da Aplicação

Como vimos no começo do curso, o nome do pacote tem grande importância para o sistema na hora de definir o escopo de cada uma das aplicações instaladas. Para evitar conflitos ou mesmo problemas no Google Play, deve-se escolher um nome de pacote que seja único. Uma boa dica para escolher um nome de pacote que seja único é utilizar um domínio que você possua na internet, porém, ao contrário. Caso você não possua nenhum domínio, tente reduzir as possibilidades utilizando a criatividade. Por exemplo, o domínio `br.suainstituicao.suaturma.seunome` pode ser um exemplo de pacote.

É importante que você pense bastante no nome do pacote a ser utilizado pela sua aplicação, pois, uma vez que o aplicativo seja lançado, o nome do pacote a ser utilizado em versões futuras e atualizações deve ser o mesmo por todo o ciclo de vida do aplicativo. Caso o nome do pacote seja trocado, o Android enxergará aquele aplicativo como um novo aplicativo e passará a tratá-lo diferentemente do aplicativo anterior, perdendo, inclusive, todos os dados envolvidos. Por esse motivo, é de grande importância que o nome do pacote seja único e reutilizável.

Desabilite o *Debug* e as Chamadas à Classe Log

Agora que chegou a hora de publicar, a aplicação não necessita mais ser "debugada", então, devemos desabilitar essa opção definitivamente. Para fazer isso, devemos ir ao arquivo **AndroidManifest.xml** e, então, configurar o parâmetro `android:debuggable` para falso, ou mesmo apagar esse atributo do Manifest, já que o valor padrão para ele é falso. Uma vez que tenhamos removido essa linha, a aplicação não estará mais disponível para ser "debugada". Também é importante que qualquer método específico de *debug* seja removido do seu código.

Em seguida, devem-se remover as chamadas de Log que estejam em sua aplicação. É comum, durante o desenvolvimento, adicionarmos algumas chamadas ao Log, ou impressões em console para ver o valor de variáveis ou mesmo saber qual caminho a aplicação está tomando dentro do código. Uma vez que a aplicação

esteja sendo distribuída, essas chamadas se tornam código morto e podem até ser deselegantes, caso o usuário tenha acesso ao Log de seu aparelho durante a execução do aplicativo. Sendo assim, faça uma busca por todo o código de qualquer chamada desse tipo e as remova.

Atualize os Diretórios de sua Aplicação

O próximo passo na publicação é a remoção de arquivos não mais utilizados na aplicação, bem como a atualização dos arquivos que serão utilizados. Pode ser que algumas vezes, durante o desenvolvimento da aplicação, você tenha trocado um plano de fundo, ou mesmo o ícone que seria utilizado pela sua aplicação. É comum que nos esqueçamos de remover esses arquivos das pastas de recursos do Android e, assim, eles terminam sendo colocados no pacote da aplicação que será distribuída. Para evitar isso e diminuir o tamanho final da aplicação, é importante que antes da publicação façamos uma verificação em cada uma das pastas de recursos para garantir que os recursos que estão por lá sejam os mais atualizados e, também, para remover os que não mais são utilizados.

Fazendo isso, garantimos que a aplicação não será maior do que precisa ser e assim ela não vai ocupar espaço desnecessariamente no aparelho, quando instalada.

Revise o AndroidManifest e os Arquivos de Build do Gradle

Agora que a aplicação já está finalizada, podemos ter certeza de todos os recursos que ela está utilizando. Baseado nisso, é preciso definir no Manifest e no Gradle cada um desses recursos para que o Android possa dar permissão ao aplicativo de utilizá-las. Revise cada uma das permissões que o aplicativo precisa, garantindo que todas foram declaradas.

Em seguida, atente para o nome e o ícone da aplicação. Verifique se a propriedade de label do Manifest está definida corretamente, bem como a propriedade icon. Essas são as propriedades responsáveis por definir o nome e o

ícone que será mostrado pela aplicação no menu dos usuários que a instalarem. Garanta que ambos estão corretamente definidos.

Outras duas propriedades importantes a serem definidas são as propriedades de versão do Android a qual a aplicação se destina. Ao configurar os atributos `android:minSdkVersion` e `android:targetSdkVersion`, o desenvolvedor é capaz de comunicar aos usuários a qual versão do Android aquela aplicação se destina, assim como quais são as versões em que a aplicação deve funcionar corretamente. Esses dois atributos ajudam o usuário a escolher se a aplicação está de acordo com seu aparelho, bem como ajuda o Google Play a filtrar os usuários que verão a aplicação.

Por fim, outro par de atributos muito importante de se configurar no `AndroidManifest` e `Gradle` são `android:versionCode` e `android:versionName`. O atributo `android:versionCode` é o atributo responsável pela indicação da versão atual do aplicativo. É por meio desse atributo que indicamos aos usuários quando um aplicativo instalado no aparelho necessita ser atualizado (através de um processo feito automaticamente pelo Google Play). É sempre interessante que esse atributo seja utilizado de forma sequencial, para que o aplicativo tenha suas versões bem estabelecidas. Já o atributo `android:versionName` indica ao usuário e ao Play qual o nome da versão atual. Diversos sistemas de nomeação de versões são utilizados em aplicações. Escolha um e procure segui-lo durante a vida de seu aplicativo. Um exemplo seria utilizar a versão como `x.y.z`, onde `x` indica a versão do aplicativo, `y` indica atualizações de aprimoramento dessa versão e `z` indica atualizações corrigindo erros da versão atual. Lembre-se que esse é apenas um exemplo e que qualquer sistema de nomeação de versões pode ser utilizado para esse atributo. Com esses atributos, finalizamos a configuração do `Manifest` e do `Gradle` e temos mais uma parte do nosso código na versão final para o lançamento. Vamos ao próximo passo.

Trate os Problemas de Compatibilidade

Existem diversas ferramentas no Android que podem testar a compatibilidade do seu aplicativo com diferentes versões e tamanhos e densidade de telas de dispositivos variados. É importante que, antes de publicar sua aplicação, você

verifique se todas as compatibilidades estão de acordo, incluindo otimizações para tablets, suportes a diferentes formatos de tela e o uso adequado da biblioteca Support.

Ajuste os Endereços de Servidores e Serviços

Esse é um passo importante da preparação da aplicação para o lançamento e pode, muitas vezes, passar despercebido, causando graves problemas à aplicação. Antes de lançar a aplicação, caso ela possua acesso a qualquer endereço remoto, é importante conferir se esses endereços estão inseridos corretamente e não apontando para endereços de teste. Imagine o problema que seria lançar uma aplicação com o serviço de *login* apontando para 10.0.2.2, como vimos na aula passada

Problema!

Também é importante preparar e testar o lado servidor de sua aplicação, caso ela utilize. Após trocar os endereços da aplicação para os definitivos, teste a aplicação novamente utilizando o endereço definitivo e garanta que os usuários finais verão o que devem ver.

Configure o Sistema de Licença

Essa última configuração a ser feita está inclusa nesta aula apenas como um lembrete. O estudo do sistema de licenças não está no escopo dessa disciplina. Caso tenha interesse de saber um pouco mais sobre licenças, a leitura complementar desta aula indica o link respectivo do Android Developers que aborda esse tema.

O sistema de licenças é uma biblioteca que o próprio Google Play disponibiliza para a inserção em sua aplicação, que permite ao desenvolvedor tratar diversos aspectos diretamente com o Google Play. Por meio dessa API, o desenvolvedor pode inserir, em diversos pontos da aplicação, chamadas a métodos que verificam o status da versão local da aplicação em relação à versão disponível no Google Play. Nessas chamadas, é possível saber do Google Play se a aplicação naquele aparelho

foi instalada a partir do Google Play ou não, se a aplicação está em versão mais atual, se aquele usuário tem permissão para utilizar a aplicação, entre outras informações. Isso é especialmente útil em aplicações que serão distribuídas no Play com algum custo. Por meio desse método é possível ao desenvolvedor consultar o Google Play para saber se a aplicação instalada naquele dispositivo foi comprada ou não e, então, bloquear o acesso à aplicação, caso não tenha sido.

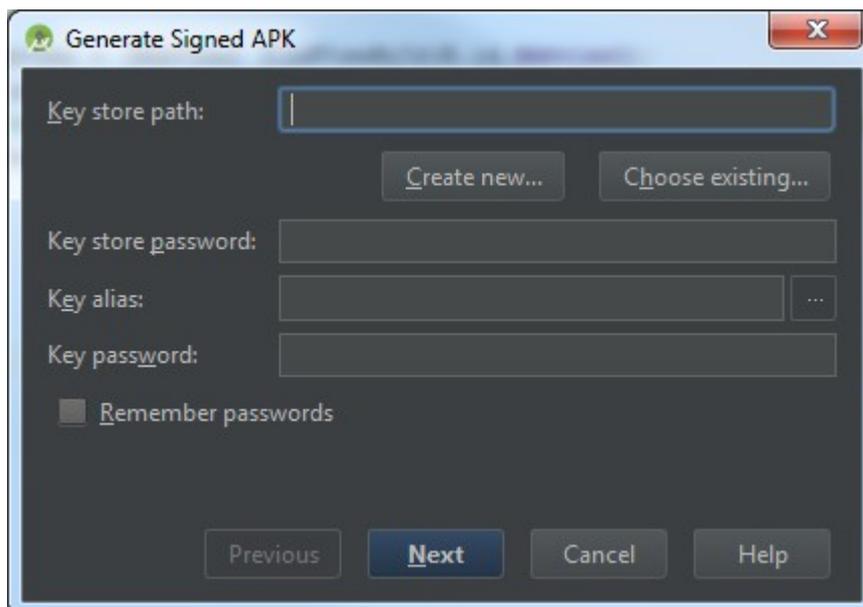
Com isso, concluímos a preparação da nossa aplicação para o lançamento. Agora, vamos aprender a compilar a aplicação em modo *release* e também quais os pontos importantes a se prestar atenção ao fazer isso.

Compilando a Aplicação

Agora que a aplicação está preparada para o lançamento, devemos compilá-la e assiná-la para que possa ser distribuída. Diferentemente de quando estamos testando uma aplicação, precisamos passar por um pequeno processo antes de gerar o .apk que será instalado nos aparelhos Android.

Para iniciarmos a compilação da aplicação de maneira assinada no Android Studio devemos, primeiramente, selecionar a opção Build -> Generate Signed APK. Feito isso, teremos acesso a uma janela que nos dá a opção de escolher uma chave já existente ou criar uma nova, como pode ser visto na **Figura 1**. Selecionaremos a opção Create New.

Figura 01 - Build → Generate Signed APK



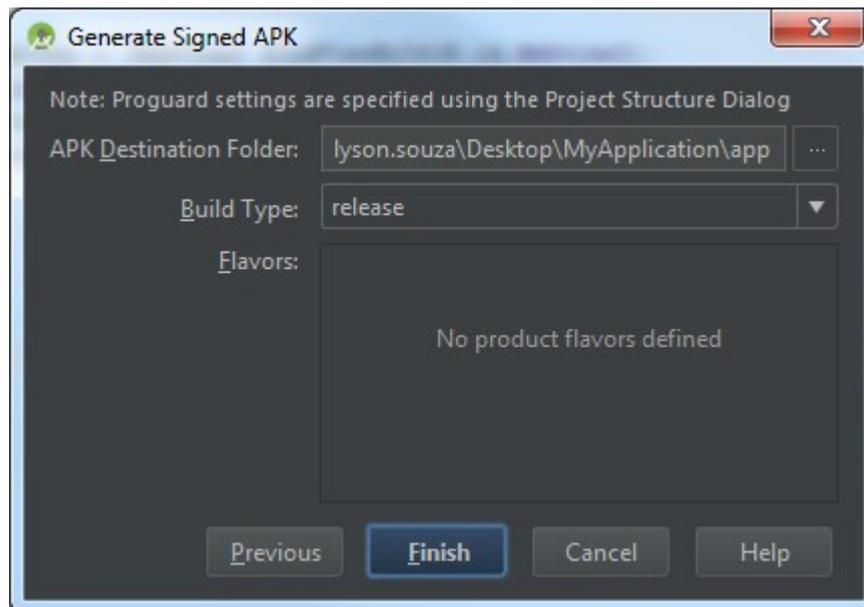
Após selecionar a opção indicada, uma nova janela será exibida solicitando o preenchimento de alguns dados. Preencha-os de acordo com o que for solicitado. Perceba que você está criando uma nova chave que será parte de um *Keystore*. Um único *Keystore* é capaz de armazenar mais de uma chave. O campo *Alias* indica exatamente qual será o nome dessa chave específica que você está criando. O ideal é associar esse campo de alguma maneira a aplicação que será assinada por essa chave para que posteriormente a chave seja lembrada.

Outro detalhe importante dessa tela é em relação a validade da chave, em anos. Para aplicações que serão publicadas no Google Play, essas chaves deverão possuir uma validade de, pelo menos, 25 anos. Isso deve garantir que você será capaz de utilizar a mesma chave para todas as atualizações da vida do seu app.

Por fim, é MUITO importante que seu *Keystore* seja mantido em um local seguro e que você tenha sempre acesso às senhas utilizados tanto para ele quanto para a chave criada. Isso garante que você poderá utilizar a mesma chave sempre, mantendo assim a sua aplicação dentro do mesmo pacote.

Após criar o *Keystore* e a chave para assinatura da sua aplicação, você retornará a tela indicada na **Figura 1**. Caso ela já não esteja preenchida, preencha os campos, selecionando o *Keystore* criado, utilizando o password dele, selecionando o *Alias* correto e utilizando também o password deste. Feito isso, clique em **Next** para ser levado à tela da **Figura 2**.

Figura 02 - Gerando a versão final e assinada do App



Na tela indicada na **Figura 2** você deve selecionar o local onde o APK assinado será salvo e qual o tipo de Build será feito para esse APK. Por padrão, *release* já está selecionado. Caso não esteja, selecione essa opção e então clique em Finish para salvar, na localização indicada, o APK assinado. Com isso, estamos prontos para publicar nossa aplicação pelo caminho que desejemos.

Publicando sua Aplicação

Agora que já temos uma aplicação pronta, funcional, assinada e pronta para a publicação, precisamos apenas definir como publicaremos a aplicação. A maneira mais adequada de se publicar uma aplicação é por meio do Google Play, mas também é possível publicar as aplicações através da web ou mesmo por e-mail.

Publicando no Google Play

A melhor opção para a publicação de uma aplicação é o Google Play. Ele se encarrega das vendas da aplicação, oferece proteções contra a pirataria, faz a divulgação da aplicação, cria uma página personalizada para sua aplicação, com informações pertinentes, bem como faz a gerência completa do número de instalações, dos tipos de dispositivos que estão utilizando sua aplicação e até

mesmo dos erros que são gerados durante a aplicação. Ou seja, o Google Play faz todo o trabalho para você, que só tem de se preocupar com fazer a divulgação da aplicação em outros meios e o que fazer com os lucros.

Entretanto, para oferecer tantos serviços e vantagens, o Google Play cobra uma taxa aos desenvolvedores. Essa taxa é cobrada uma vez e custa 25 dólares. Além dessa taxa, 30% do valor final da venda dos aplicativos são destinados à manutenção dos servidores e da operação da loja como um todo. Ou seja, para utilizar o Google Play temos que pagar. Caso as aplicações que você vá distribuir sejam gratuitas, os 30% sobre a venda não serão cobrados. A taxa de 25 dólares, no entanto, sempre é cobrada, uma vez que está atrelada à criação da conta e não à distribuição do aplicativo em si.

Criar uma conta no Google Play é simples. Primeiramente, é necessário criar uma conta Google. Contas já existentes podem ser utilizadas (sua conta do Gmail, por exemplo). Uma vez criada a conta Google, basta acessar o endereço <https://play.google.com/apps/publish/> e preencher o cadastro para criação de sua conta. É nesta hora que devemos pagar a taxa de inscrição de 25 dólares, utilizando um cartão de crédito internacional. É importante notar que o titular do cartão de crédito utilizado na criação da conta será considerado o titular da conta.

Assim que a conta for criada, acesse a página de publicação e selecione a opção de disponibilizar um novo aplicativo no Google Play. Ao selecionar essa opção, primeiro preencha todas as informações sobre a aplicação, e, em seguida, o Play irá requisitar que você faça o upload do APK de sua aplicação. Selecione o APK assinado que foi gerado anteriormente e clique em enviar. Em seguida, você verá uma página como a exibida na **Figura 3**.

Figura 03 - Publicando uma aplicação.

Google play | ANDROID DEVELOPER CONSOLE

Editar aplicativo

Publicar Salvar

Detalhes do produto Arquivos APK

Enviar recursos

Capturas de tela pelo menos duas	Adicionar uma captura de tela: <input type="button" value="Choose File"/> No file chosen	<input type="button" value="Enviar"/>	Capturas de tela: 320 x 480, 480 x 800, 480x854, 1.280x720, 1.280x800 24 bits no formato PNG ou JPEG (sem alfa) Sem margens nem bordas É possível fazer upload das capturas de tela em orientação paisagem. As miniaturas parecerão invertidas, porém, as imagens reais e suas orientações serão preservadas.
Ícone do aplicativo em alta resolução [Saiba mais]	Adicionar um ícone de aplicativo de alta resolução: <input type="button" value="Choose File"/> No file chosen	<input type="button" value="Enviar"/>	Ícone do aplicativo em alta resolução: 512 x 512 PNG ou JPEG de 32 bits Máximo: 1024 Kb
Gráfico promocional opcional	Adicionar gráfico promocional: <input type="button" value="Choose File"/> No file chosen	<input type="button" value="Enviar"/>	Gráfico promocional: 180 L x 120 A PNG ou JPEG de 24 bits (sem alfa) Sem bordas na arte
Gráfico de recursos opcional [Saiba mais]	Adicionar gráfico de recursos: <input type="button" value="Choose File"/> No file chosen	<input type="button" value="Enviar"/>	Gráfico de recursos: 1024 x 500 PNG ou JPEG de 24 bits (sem alpha) Será diminuído para tamanho mini ou micro
Video promocional opcional	Adicionar link de vídeo promocional: <input type="text" value="http://"/>		Video promocional: Insira o URL do YouTube

É a partir dessa tela que a página de sua aplicação será gerada. Para que ela possa ser mais completa possível, é importante que o maior número de campos dessa página seja preenchido com materiais promocionais. No mínimo, um ícone da aplicação em tamanho grande e duas capturas de tela da aplicação devem ser enviados, juntamente ao texto promocional da aplicação na língua do país em que ela será distribuída e/ou em inglês.

Dentre os campos a serem preenchidos, é importante notar o campo preço. Como está destacado na própria página do Play, um aplicativo que, a qualquer momento, seja distribuído como gratuito não poderá ser distribuído como pago novamente, então, preste atenção ao selecionar essa opção. Caso a opção selecionada seja **Pago**, preencha o campo com o preço em dólar e utilize a conversão automática para os outros países nos quais a sua aplicação será distribuída. Essa conversão automática facilita o gerenciamento de preços, mesmo com a variação cambial. É importante também selecionar a categoria apropriada para seu aplicativo, pois, assim, a visibilidade dele aumenta e o número de instalações também. Quando todos os campos estiverem preenchidos adequadamente, clique em **Salvar**, no menu superior, e confira novamente. Caso tudo esteja certo, clique em publicar.

A partir do momento em que o desenvolvedor clica em publicar, a aplicação é enviada ao Play. É necessário algum tempo antes da aplicação ser publicada. Esse tempo fica em torno de 24 horas e é utilizado para que o Google possa certificar sua aplicação e criar a página dela dentro do Play. Assim que a aplicação for publicada, basta procurá-la no Google Play ou divulgar o link do Play, e todos os usuários Android que tiverem dispositivos compatíveis com sua aplicação poderão baixá-la.

Depois de publicado, o aplicativo será mostrado em sua página inicial de desenvolvedor, como visto na **Figura 4**. A partir dessa tela é possível acessar as diversas estatísticas que o Google Play disponibiliza sobre seu aplicativo, como os modelos dos dispositivos que instalaram a aplicação e o país ao qual eles pertencem. Essas informações são muito precisas e interessantes para atualizações que venham a ser geradas.

Figura 03 - Publicando uma aplicação.



Com isso, finalizamos todo o processo de publicação. Vejamos, agora, quais são as alternativas para a disponibilização de aplicações para dispositivos Android.



Vídeo 02 - Estatísticas da sua Aplicação

Publicando Fora do Google Play

Como o Android é uma plataforma aberta, é possível publicar os aplicativos gerados fora do Google Play. Para que isso seja possível, é apenas necessário que o dispositivo que está recebendo a aplicação seja configurado para aceitar aplicações que não venham do Play. Essa opção pode ser encontrada dentro do menu de Configurações → Aplicações → Fontes Desconhecidas.

Ao habilitar essa opção, em geral, é possível instalar no aparelho aplicações assinadas que venham de fontes diferentes do Google Play. É importante ter muito cuidado ao distribuir uma aplicação assim, pois uma vez que o APK de sua aplicação seja distribuído, é muito fácil que ele seja replicado e passado adiante, tornando, assim, sua aplicação pública.

Tendo um aparelho que aceite aplicações que não sejam do Play, basta receber o APK por e-mail, instalar o APK após recebê-lo via Bluetooth ou, até mesmo, baixá-lo da web. Como distribuir a aplicação fica ao seu critério, é importante apenas habilitar a opção no aparelho que irá recebê-la.

Monetizando sua Aplicação

Além de vender nossas aplicações, também há outras formas de monetizá-las, gerando algum lucro ao desenvolvedor. Abordaremos brevemente cinco formas de monetização de aplicações: venda da aplicação, distribuição gratuita com venda de funcionalidades adicionais (Freemium Apps), assinaturas, comércio eletrônico dentro da aplicação, utilização de propaganda.

Venda de Aplicações Completas

Cobrar para que os usuários transfiram seus apps é um modelo simples e de conveniente monetização. Após criar sua conta do Google Play, você pode definir os preços para seus aplicativos no Console do Desenvolvedor. Opcionalmente, você pode incluir publicidade ou vender recursos adicionais dentro da aplicação.

Este modelo pode funcionar bem para qualquer aplicativo ou jogo, e pode ser particularmente relevante para aplicativos com características extensas ou para um nicho específico do mercado. É indicado para certas categorias de aplicativos, como jogos para crianças, que devem ser adquiridos uma única vez, ao invés de vender produtos dentro da aplicação.

No entanto, esse modelo pode limitar seu potencial de uso, particularmente em mercados ainda em desenvolvimento, em que as pessoas não costumam pagar por aplicações. Estatísticas indicam que você pode ser capaz de atingir uma maior

receita usando outras formas de monetização, como Freemium Apps, Assinaturas ou Propaganda.

Freemium Apps – Aplicações Gratuitas com Venda de Recursos Adicionais

Os usuários são mais propensos a baixar aplicativos e jogos gratuitos em comparação aos pagos. No entanto, você pode usar outras maneiras de rentabilizar os Apps gratuitos, como a venda de recursos adicionais dentro do próprio App. Você pode vender produtos digitais de tempo ilimitado (duráveis) ou consumíveis (que irão acabar). Os itens duráveis sempre estarão disponíveis para o usuário quando forem comprados, tais como características adicionais do App. Os itens consumíveis podem ser usados progressivamente ou expirarem após um período de tempo, como add-on de jogos, ou compra de notícias e informações. Uma abordagem básica é oferecer um download gratuito com recursos limitados ou recursos completos por um tempo limitado. Em seguida, use a venda dos recursos para desbloquear o aplicativo completo, ilimitado.

Assinaturas de Serviços

Assinaturas fornecem uma excelente oportunidade para criar fluxos de receita contínua. As assinaturas são semelhantes aos recursos digitais oferecidos pela venda dentro de Apps Freemium, só que elas são disponibilizadas de forma recorrente mensal ou anual.

Quando os usuários compram assinaturas em seus aplicativos, o Google Play lida com todos os detalhes da venda para que os Apps não precisem processar diretamente as transações financeiras. O Google Play processa todos os pagamentos para assinaturas através do Google Wallet, exatamente como faz com os aplicativos pagos ou com os recursos vendidos dentro de um app Freemium.

Como estratégias, você pode oferecer seus Apps com uma assinatura básica, ou usando uma assinatura gratuita para avaliação, permitindo que os usuários explorem o aplicativo e seu conteúdo antes de assiná-lo. Uma abordagem mais avançada é oferecer, dentro de seus aplicativos, recursos específicos gratuitos e itens que precisem de assinatura.

E-Commerce

Você pode usar essa opção quando seu aplicativo é uma loja ou oferece produtos de terceiros para venda. Desta forma, você pode cobrar uma taxa pela venda de cada produto dentro do aplicativo. Você também pode combinar essa estratégia com seus aplicativos pagos ou Freemium, oferecendo produtos relacionados à aplicação.

Propaganda

Anúncios podem ser uma maneira rápida e fácil de lucrar mais, inclusive em comparação com aplicativos Freemium, pagos ou com assinatura. Esta é uma abordagem que vem sendo bastante utilizada e que pode gerar um bom lucro.

O Google disponibiliza alguns SDKs (AdMob e Google Mobile Ads SDK) para utilização de anúncios que serão convertidos em dinheiro, e pagos pelo próprio Google. Eles permitem que você possa adicionar publicidade aos seus aplicativos com apenas algumas linhas de código. A pergunta é: qual é o modelo que obtém os melhores resultados para seu aplicativo?

As ferramentas de anúncio do Google ajudam a descobrir qual combinação funciona melhor para seu público. Vinculando suas contas AdMob e Google Analytics você consegue obter mais informações e melhores resultados sobre as propagandas no seu aplicativo. Faça uma análise da forma ideal de distribuição da sua aplicação e siga em frente.

Com isso, finalizamos nossa aula e, conseqüentemente, nossa disciplina. Espero que tenha aprendido tanto quanto eu e seja, agora, capaz de passar por todas as etapas do desenvolvimento de uma aplicação Android! Sabemos que alguns temas não puderam ser abordados, mas a base adquirida durante esse curso é bastante para que qualquer outro conteúdo seja absorvido com um pouco de pesquisa e um pouco de dedicação.

Foi um prazer imenso ter a sua atenção e até uma próxima oportunidade!



Vídeo 03 - Faturando com o Android

Leitura Complementar

- DEVELOPERS. **Application Licensing**. Disponível em: <<http://developer.android.com/guide/market/licensing/index.html>>. Acesso em: 08 jun. 2015.
- _____. **Publishing checklist for Google Play**. Disponível em: <<http://developer.android.com/distribute/tools/launch-checklist.html>>. Acesso em: 08 jun. 2015.

Resumo

Nesta aula, vimos como dar os retoques finais em uma aplicação e torná-la pronta para distribuição. Aprendemos a configurar o Manifest e o Gradle corretamente, remover as chamadas a classe Log e todos os outros pontos necessários para a criação de um APK assinado (por uma chave que também aprendemos a criar durante essa aula) e pronto para publicação. Estudamos também como publicar essa aplicação no Google Play, bem como os custos envolvidos. Por fim, vimos como é possível publicar de outras maneiras e como monetizar a nossa aplicação de maneira adequada. Concluimos, então, esta aula e o curso de Dispositivos Móveis. Bons estudos! Até a próxima oportunidade!

Autoavaliação

1. Quais atributos/chamadas devem ser *removidos* de sua aplicação para torná-la pronta para distribuição?
2. Quais os dois atributos responsáveis pelo versionamento e como funcionam?
3. É possível a qualquer um publicar no Google Play? Quanto custa? Quais vantagens o Google Play traz ao desenvolvedor?

4. Como publicar uma aplicação via e-mail? Comente o que o desenvolvedor e o usuário devem fazer.

Referências

ANDROID DEVELOPERS. 2015 Disponível em: <<http://developer.android.com/>>. Acesso em: 08 jun. 2015.

DIMARZIO, J. **Android: A Programmer's Guide**. New York: McGraw-Hill, 2008. Disponível em: <<http://books.google.com.br/books?id=hoFl5pxjGesC>>. Acesso em: 08 jun. 2015.

LECHETA, Ricardo R. **Google android: aprenda a criar aplicações**. 2. ed. São Paulo: NOVATEC, 2010.

_____. **Google android para tablets**. São Paulo: NOVATEC, 2012.

MEIER, R. **Professional Android 2 Application Development**. Hoboken, NJ: John Wiley e Sons, 2010. Disponível em: <<http://books.google.com.br/books?id=ZthJlG4o-2wC>>. Acesso em: 08 jun. 2015.