

Guia do Desenvolvedor

POS Digital | Getnet

Documentação

Versão 4.2.1

CLASSIFICAÇÃO DO DOCUMENTO – CONFIDENCIAL

Este documento pode conter informação confidencial e/ou privilegiada. Se você não for o destinatário ou a pessoa autorizada a receber este documento, não deverá utilizar, copiar, alterar, divulgar a informação nele contida ou tomar qualquer ação baseada nessas informações. Se você recebeu este documento por engano, por favor, avise imediatamente o responsável, devolvendo o documento em seguida. Agradecemos sua cooperação.

Sumário

HISTÓRICO DE ALTERAÇÕES	3
INTRODUÇÃO	4
ANTES DE COMEÇAR	4
REQUISITOS PARA DESENVOLVIMENTO	5
REQUISITOS MANDATÓRIOS.....	5
BOAS PRÁTICAS E SEGURANÇA.....	6
PROIBIÇÕES.....	6
<i>Permissões no AndroidManifest.xml</i>	<i>7</i>
INTEGRADORAS (TEF)	9
<i>Implementação da tela pin (TEF).....</i>	<i>9</i>
EQUIPAMENTOS SUPORTADOS	10
GETSTORE – LOJA DE APLICATIVOS	12
O QUE É A GETSTORE?.....	12
PLANOS DE ASSINATURA	12
<i>Modo de cobrança.....</i>	<i>13</i>
<i>Código do Plano.....</i>	<i>13</i>
<i>Como identificar o Código do Plano pelo seu aplicativo.....</i>	<i>16</i>
SUBMETENDO UM APLICATIVO.....	16
<i>Processo de Certificação.....</i>	<i>17</i>
<i>Relatório dos Testes.....</i>	<i>18</i>
DÚVIDAS FREQUENTES.....	19
ANDROID.....	19
PAGAMENTO	19
TERMINAL.....	19
WiFi/3G/4G.....	20
REBATEDOR	20
COMANDOS ADB.....	21
SDK	21
CERTIFICAÇÃO	22
CONTATO DOS FABRICANTES.....	22

Histórico de alterações

Versão	Data	Descrição
3.0	13/05/2020	Ajustes na organização da documentação.
3.6	07/10/2020	Nova Funcionalidade: Modo Quiosque
3.6.2	26/01/2021	Ajustes nos requisitos mandatórios.
3.7.0	01/10/2021	Nova seção GetStore com informações sobre Código do Plano.
4.0	19/10/2021	Melhorias e correções.
4.0.1	26/11/2021	Melhorias e correções.
4.1	15/07/2022	Melhorias e correções.
4.2	29/11/2022	Inclusão de informações sobre novo terminal.
4.2.1	08/12/2022	Melhorias e correções.

Introdução

Esta documentação aborda os principais assuntos que você deve saber antes de iniciar o envio de seu aplicativo.

Na parte [Requisitos para desenvolvimento](#) você vai encontrar informações básicas, boas práticas e obrigatoriedades na hora de desenvolver seu aplicativo. Vai encontrar também os requisitos para utilizar integradoras TEF e verificar os equipamentos suportados pela Getnet.

Na parte [GetStore – Loja de aplicativos](#), você vai entender como fazer a gestão de seu aplicativo desde o envio até a distribuição para os estabelecimentos. Você também encontrará informações sobre como criar diferentes tipos de planos e oferecer condições específicas para cada perfil de cliente. Por fim vai conhecer como ocorre nosso processo de certificação de aplicativos.

Antes de começar

Abaixo algumas informações importantes para você saber antes de iniciar o processo de envio do seu aplicativo.

- **Terminais:** Para adquirir os terminais suportados pela Getnet, entre em contato com os canais de venda dos fabricantes:
 - **Ingenico APOS A8:** payments@scansource.com / renato.fonseca@scansource.com
 - **Newland N910:** sergio.paulillo@tefti.com.br
 - **TecToy Sunmi P2:** rodrigo.santos@tectoy.com
- **Valor recebido pelo uso do aplicativo:** O lojista pagará pela utilização de seu aplicativo mensalmente. Você poderá oferecer condições específicas para cada perfil de cliente e definir um valor mensal a ser cobrado pelo uso do aplicativo. Deste valor mensal será retirada a **Comissão Getnet**.
- **Programa de Parcerias:** Para fazer parte do Programa de Parcerias do POS Digital e ter vantagens especiais entre em contato.
- **Contato:** Em caso de dúvidas verifique a seção [Dúvidas Frequentes](#) deste documento ou entre em contato pelo e-mail parceiros_posdigital@getnet.com.br.

Requisitos para desenvolvimento

Toda documentação e aplicativos necessários estão disponíveis no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>), no menu Documentação e SDKs.

É obrigatório que seu aplicativo funcione em todos nossos modelos de terminais do POS Digital. Para adquirir estes terminais entre em contato com os canais de venda dos fabricantes:

- **Ingenico APOS A8:** payments@scansource.com / renato.fonseca@scansource.com
- **Newland N910:** sergio.paulillo@tefti.com.br
- **TecToy Sunmi P2:** rodrigo.santos@tectoy.com

Para informações técnicas destes terminais, veja a seção “Equipamentos suportados” deste documento.

Requisitos mandatórios

1. Suporte ao **Android 5.1 (22)** e ao **Android 7.1 (25)**.
2. O desenvolvimento do aplicativo **pode ser nativo ou híbrido** (Exemplo: Xamarin, pois irá compilar o código para o “Android Runtime (ART)” como se fosse um aplicativo nativo).
3. O aplicativo deve estar utilizando o **SDK de Hardware da Getnet** (ver documentação do SDK de Hardware no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>)).
4. Os seguintes **atributos devem ser adicionados no build.gradle** para que seja possível assinar seu aplicativo para o modelo de terminal **TecToy Sunmi P2**.
 - **v1SigningEnabled true**
 - **v2SigningEnabled true**
 - **signingConfig signingConfigs.config**

Exemplos:

```
android {
//INICIO - Deve ser inserido esse metodo dentro do android{}
    signingConfigs {
        config {
            v1SigningEnabled true
            v2SigningEnabled true
        }
    }
//FIM - Deve ser inserido esse metodo dentro do android{}

    buildTypes {
        debug {
            debuggable true
            testCoverageEnabled true
            signingConfig signingConfigs.config //Deve ser inserida essa
linha no buildTypes
            crunchPngs false
        }
    }
}
```

```

        release {
            minifyEnabled false
            debuggable false

            signingConfig signingConfigs.config //Deve ser inserida essa
            linha no buildTypes
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
            'proguard-rules.pro'
            crunchPngs true
        }
    }
}

```

5. O aplicativo deve possuir uma seção de **"Fale Conosco"** informando ao cliente os contatos necessários para atendimento em caso de problemas ou dúvidas. Essa seção deve estar bem visível e de fácil acesso.
6. O aplicativo **deve** utilizar o **número de série do equipamento disponibilizado pela API** da Getnet, portanto, não é permitido usar o número que o Android disponibiliza ou o número gerado pelo aplicativo, pois não será o mesmo que o da etiqueta colada no verso do aparelho.
7. Utilizar **layout dinâmicos** considerando que no futuro o aplicativo pode ser instalado em outros hardwares com resoluções e densidades diferentes.
8. Deve ser possível **navegar por todas as telas** sem o aplicativo "quebrar".

Boas práticas e segurança

1. Utilize as **boas práticas** de programação Android.
2. É de **responsabilidade do desenvolvedor** garantir a **segurança da informação** dentro do aplicativo.
3. **Evite expor dados sensíveis dos clientes**, tais como, senhas e informações pessoais.
4. Integridade dos dados.
5. **Tamanho do APK** – revise o seu projeto e exclua imagens e dados que não são utilizados.
6. Evite funcionalidades com **timeout infinito** que possa deixar sua aplicação travada ou aguardando algum evento.
7. **Não manter a antena do Mifare ligada**, apenas ligar quando for ser utilizada.
8. **Uso consciente do consumo de dados.**
9. **Uso consciente do consumo de bobina de impressão** – evite imagens grandes, muitos espaços em branco, deixar grandes espaços entre linhas, espaços no início e final dos comprovantes.
10. **Uso consciente do consumo de memória e CPU.**

Proibições

1. É expressamente **proibido WebApps e WebView** devido a medidas de segurança. Desenvolvidos com tecnologias baseadas em **Webview**, são proibidas. Exemplo: Ionic, Cordova. No processo de certificação do seu aplicativo nosso time de certificação vai usar ferramentas para verificar se sua aplicação possui WebView.
2. O **GooglePlay Services não** estará disponível.
3. O aplicativo **não** pode se comunicar com nenhum concorrente direto ou indireto da Getnet ou do Grupo Santander (incluindo bancos, adquirentes, subadquirentes e wallets).
4. O aplicativo **não pode ser executado em modo de tela cheia**.
5. O aplicativo não pode realizar alterações nas configurações de sistema do Android (Ex: data, hora, localização, etc).
6. Acesso a **recursos externos impróprios** (USB, Google Play...).
7. O Ícone, logo e Nome do aplicativo **não** podem estar vinculado às identificações visuais dos itens abaixo:
 - **Getnet e seus produtos;**
 - **Banco Santander;**
 - **Concorrentes diretos ou indiretos da Getnet ou do grupo Santander** (incluindo bancos, adquirentes, subadquirentes e wallets).

Permissões no AndroidManifest.xml

As permissões abaixo **não são aceitas** por permitirem que o aplicativo acesse alguns recursos do Android ou recursos de hardware não permitidos pela Getnet, ou também por colocar em risco a operação de outros aplicativos.

A Getnet pode adicionar outras permissões proibidas durante o processo de certificação.

OBS: Caso alguma das permissões a seguir esteja em seu aplicativo, você **NÃO** vai conseguir realizar o upload do seu aplicativo no portal.

- ACCESS_INSTANT_APPS
- BROADCAST_STICKY
- CHANGE_NETWORK_STATE
- CHANGE_WIFI_STATE
- DELETE_PACKAGES
- DOWNLOAD_WITHOUT_NOTIFICATION
- DUMP
- INSTALL_PACKAGES
- INSTALL_SHORTCUT

- INTERACT_ACROSS_USERS_FULL
- KILL_BACKGROUND_PROCESSES
- MOUNT_UNMOUNT_FILESYSTEMS
- REQUEST_INSTALL_PACKAGES
- SET_TIME
- SET_TIME_ZONE
- SET_WALLPAPER
- SET_WALLPAPER_HINTS
- SYSTEM_ALERT_WINDOW
- USB_PERMISSION
- USB_SET
- USE_PERIPHERAL_IO
- WAKE_LOCK
- WHITE_CALENDAR
- WRITE_APN_SETTINGS
- WRITE_OWNER_DATA
- WRITE_SECURE_SETTIN
- WRITE_SETTINGS

Integradoras (TEF)

Seu aplicativo utiliza SDK de TEF

Se você tem interesse em usar um **SDK de TEF de integradores** para fazer pagamentos, tais como:

- **Auttar**
- **Software Express**
- **NCR**
- **Linx**
- **VBI**
- **PayGO**

É necessário adotar os seguintes requisitos:

1. **É obrigatório o uso do SDK de Hardware Getnet** (ver a documentação do SDK de Hardware no Portal do Desenvolvedor “<https://getstore.getnet.com.br/#/home>”).
2. A Integradora deve entrar em contato com o fabricante para usar versão de **Biblioteca Compartilhada (BC)** compatível com o firmware dos fabricantes.
3. A **Getnet pode precisar atualizar o firmware e Kernel** dos terminais e a **Integradora será avisada** com antecedência para fazer testes e possíveis adaptações.
4. Durante o **processo de certificação** do aplicativo, a **integradora precisa estar apontando para um ambiente funcional** (que não é da Getnet) e que esteja configurado para usar a chave no slot 3.
5. A tela de pin deverá ser **implementada** de acordo com o fabricante conforme capítulo abaixo “Implementação de tela pin (TEF)”.

Implementação da tela pin (TEF)

Ingenico

A Ingenico tem um mecanismo de skin para montar a tela de senha. A integradora terá que implementar conforme o exemplo abaixo:

```
private void configurePinpadSkin() {  
    // Configura layout customizado do PINPAD  
    Intent configureSkin = new  
Intent("com.landicorp.pinpad.pinentry.server.SET_SKIN");  
    configureSkin.putExtra("disorder", false);  
    // Força texto de entrada (os '**') a ser apresentada  
    configureSkin.putExtra("show_input", true);  
    // Define o nome da Skin  
    configureSkin.putExtra("skin_name", "GETNET4");  
    // Totalmente transparente  
    configureSkin.putExtra("dim_amount", 0);  
}
```

```
// Tamanho em altura do PinPad
configureSkin.putExtra("skb_height", 580);

// Comunica a configuração ao serviço pinentry
mContext.sendBroadcast(configureSkin);
}
```

Este mecanismo enviará um broadcast para o service da Ingenico que quando for chamado um **goOnChip** ou **getPin**, a tela de senha será invocada pelo service e ficará em primeiro plano.

Atualmente, além da skin default do fabricante existem 3 opções de skin customizadas: **GETNET4**, **SKIN_LIGHT**, **SKIN_DARK**. Para ter acesso a essas skins baixe o aplicativo “Teclados de Senha para TEF” no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>).

Para usar a skin default do fabricante basta remover a linha:

“**configureSkin.putExtra("skin_name", "GETNET4");**”. Para as outras você deverá substituir o GETNET4 pelas demais citadas anteriormente.

Newland

Buscar suporte junto ao fabricante:

daniel@npi.newlandpayment.com

Equipamentos suportados



Fabricante	Ingenico	Newland	TecToy Sunmi
Nome	APOS A8	N910	P2
Processador	ARM Cortex-A7 Quad Core + Cortex-M4 32-bit	High performance Quad-core processor + Special secure CPU	ARM Cortex-A53 Quadcore 1.5GHz CPU
Memória	1GB RAM+ 8GB FLASH ou 2GB RAM + 16GB Flash	1GB RAM, 8GB Nand Flash	1GB RAM + 8GB ROM
SO	Android 5.1 (22)	Android 5.1 (22)	Android 7.1 (25)
SIM	1	1	1

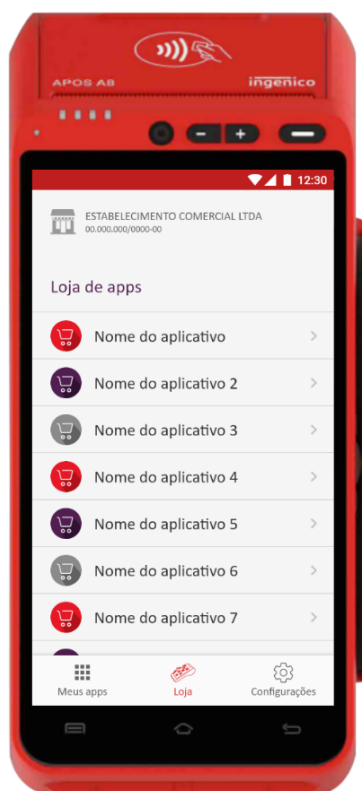
SAM	2	1	1
Interface	Tarja Magnética ISO 1/2/3	Tarja Magnética ISO 1/2/3	Tarja Magnética ISO 1/2/3
Smart	Card EMV Level 1	Card EMV Level 1	ISO7816/EMV/PBOC3.0
Contactless	EMV LEVEL 1&2 compliant	ISO/IEC 14443 Type A/B, Mifare®, NFC	PBOC 3.0, ISO/ICE 14443 TypeA&B, Mifare Card, qPBOC, PayPass, PayWave
Display	5,5"	5"	5,5"
Resolução	1280 x 720 pixels	1280 x 720 pixels	1440 x 720 pixels
Teclado	Teclado virtual	Teclado virtual	Teclado virtual
Conectividade	4G cat4 / 3G / 2G + WiFi bgn / WiFi bgn / BT 3.x	4G/WCDMA/GPRS/802.11b/g/n / Bluetooth 2.1~4.0	2G/3G/4G LTE/UMTS/GSM + WiFi 2.4G/5G Suporte IEEE 802.11 a/b/g/n + Bluetooth Suporte a 2.1/3.0/4.0 BLE, ibeacon
Bateria	Carregamento 2600 mAh x 2	Carregamento 2600 mAh x 2	Bateria polímero de lítio 7.6VCC/2540mAh
GPS	Sim	Sim	Sim
Câmera	Câmera Traseira - 5M pixel com zoom automático Código de Barras QRCode - Leitor via câmera (1D/2D)	Câmera Traseira - 5M pixel com zoom automático Código de Barras QRCode - Leitor via câmera (1D/2D)	Câmera Traseira: 5,0MP autofocus + LED flash
Áudio	1 alto-falante - 1 entrada de fone de ouvido - 1 Microfone	1 alto-falante - 1 Microfone	1 alto-falante
Impressora	Tipo - Térmica	Tipo - Térmica	Tipo - Térmica
Velocidade	High speed	High speed	High speed
Diâmetro da bobina	Ø 40 mm	Ø 40 mm	Ø 40 mm
Dimensões	183 x 84 x 64 mm	190 x 81 x 54.9 mm	222 x 81.6 x 17.6 mm
Peso	450g	450g	400g

GetStore – Loja de aplicativos

O que é a GetStore?

GetStore é a loja de aplicativos do POS Digital, ela possibilita que você faça toda a gestão de seu aplicativo. Desde o envio do aplicativo, até o download pelos lojistas, além de permitir a distribuição exclusiva para seus estabelecimentos parceiros ou testar novas versões.

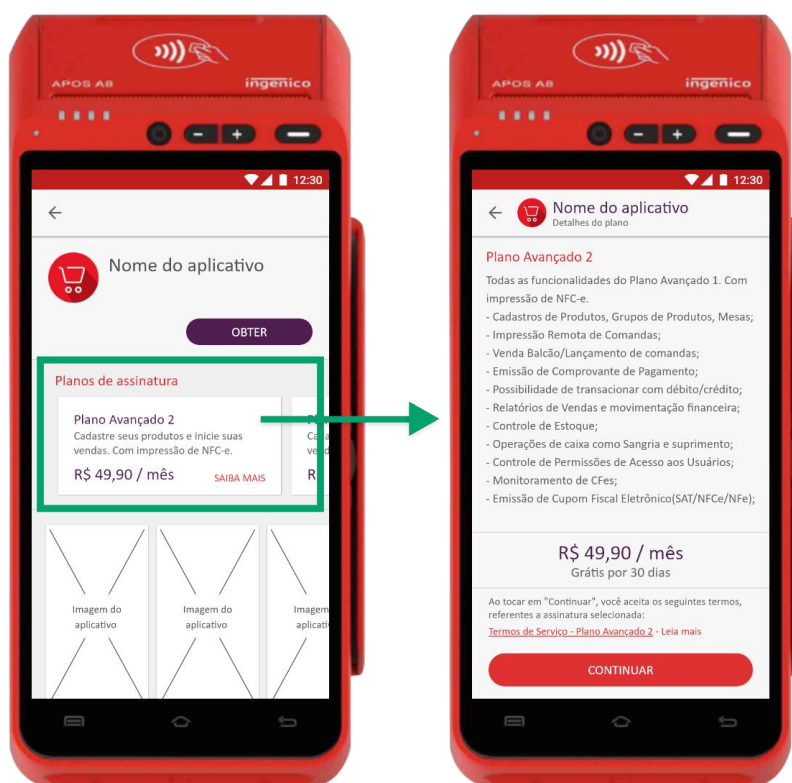
Ao publicar um aplicativo a partir do Portal do Desenvolvedor este será disponibilizado no POS Digital no aplicativo GetStore, conforme imagem de exemplo abaixo:



Planos de Assinatura

O Plano de Assinatura é a forma que a GetStore possibilita vincular aplicativos à terminais de lojistas através de licenças, estas licenças são geradas no momento da assinatura destes planos. Os Planos de Assinatura são responsáveis por conter informação sobre o tipo de cobrança a ser gerada para o lojista.

Abaixo exemplo dos detalhes de um aplicativo e a posição e detalhes de um Plano de Assinatura:



Modo de cobrança

Ao contratar um aplicativo através de um plano do tipo Pago, o lojista pagará pela utilização deste mensalmente.

Crie diferentes planos e ofereça condições específicas para cada perfil de cliente. Você **deve** definir um valor mensal a ser cobrado pelo uso do aplicativo. Deste **valor mensal** será retirada a **Comissão Getnet**.

OBS: Os Aplicativos que fazem parte do programa de parcerias devem entrar em contato com parceiros_posdigital@getnet.com.br para negociações especiais.

Código do Plano

O que é o Código do Plano?

O Código do Plano é um conjunto de caracteres alfanuméricos que você poderá criar para diferenciar cada plano dentro de um aplicativo.

Para que serve?

O Código do Plano permite identificar qual plano foi assinado durante a instalação do seu aplicativo. Por exemplo, se um plano do seu aplicativo possuir funcionalidades básicas e outro tiver funcionalidades mais avançadas, com o Código do Plano você consegue verificar e controlar quais lojistas têm cada um de seus planos assinados.

Exemplo de divisão de planos:



Plano Avançado 1
Cadastre seus produtos e inicie suas vendas. Sem impressão de NFC-e.
R\$ 29,90 / mês

Código do Plano: avancado001
Quero liberar todas as funcionalidades, mas sem impressão de nota fiscal.

Plano Avançado 2
Cadastre seus produtos e inicie suas vendas. Com impressão de NFC-e.
R\$ 49,90 / mês

Código do Plano: avancado002
Quero liberar todas as funcionalidades, inclusive com impressão de nota fiscal.

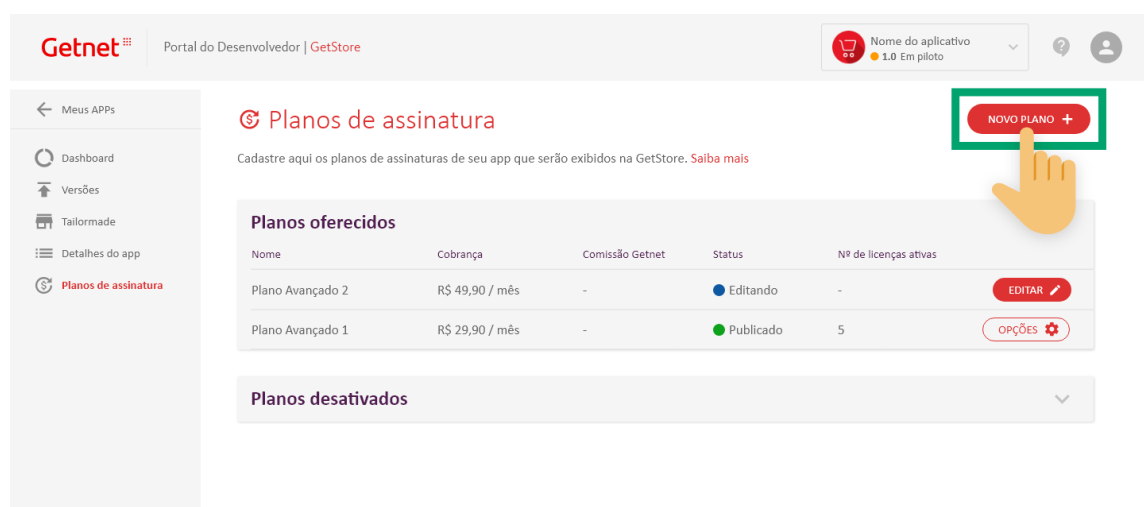
O que você precisa saber antes de criar o Código do Plano?

Você pode criar ou editar o Código do Plano quando quiser, exceto se ele for um plano expirado, nesse caso ele fica disponível apenas para visualização. Ao criá-lo ou editá-lo você deverá utilizar somente letras sem acentuação e números, e o máximo de 30 caracteres.

Como criar o Código do Plano?

Em um novo plano - Você deve preencher o campo “Código do Plano” localizado na página de criação:

Exemplo de criação de novo plano:



Getnet Portal do Desenvolvedor | GetStore

Plano de assinatura

Cadastre aqui os planos de assinaturas de seu app que serão exibidos na GetStore. Saiba mais

Nome	Cobrança	Comissão Getnet	Status	Nº de licenças ativas
Plano Avançado 2	R\$ 49,90 / mês	-	● Editando	-
Plano Avançado 1	R\$ 29,90 / mês	-	● Publicado	5

Planos desativados

Exemplo de preenchimento do campo Código do Plano:

Código do plano

Com este **código** você poderá identificar qual plano foi assinado durante a instalação de seu aplicativo. Para saber como **utilizar esse código** no seu aplicativo, acesse a documentação [Guia do Desenvolvedor](#) na página [Documentação e SDKs](#). Este campo **não é obrigatório**.

Atenção: nosso sistema faz distinção de letras maiúsculas ou minúsculas.

Código do plano

avancado001

Utilize somente letras sem acentuação e números

20/30

Informações financeiras

Em um plano já publicado – Acesse os detalhes do plano no botão “Opções” e então no campo “Código do Plano” você terá a opção de editá-lo:

Exemplo – Botão Opções de um plano publicado:

Getnet

Portal do Desenvolvedor | GetStore

Nome do aplicativo

1.0 Em piloto

Meus APPs

- Dashboard
- Versões
- Tailormade
- Detalhes do app
- Planos de assinatura**

Planos de assinatura

Cadastre aqui os planos de assinaturas de seu app que serão exibidos na GetStore. [Saiba mais](#)

NOVO PLANO +

Planos oferecidos	Nome	Cobrança	Comissão Getnet	Status	Nº de licenças ativas	
Plano Avançado 2	R\$ 49,90 / mês	-	Editando	-	EDITAR	
Plano Avançado 1	R\$ 29,90 / mês	-	Publicado	5	OPÇÕES	

Planos desativados

Exemplo – Botão Editar do campo “Código do Plano”:

Código do plano

Com este **código** você poderá identificar qual plano foi assinado durante a instalação de seu aplicativo. Para saber como **utilizar esse código** no seu aplicativo, acesse a documentação [Guia do Desenvolvedor](#) na página [Documentação e SDKs](#). Este campo **não é obrigatório**.

Atenção: nosso sistema faz distinção de letras maiúsculas ou minúsculas.

Código do plano

planoAvançado001

EDITAR

Informações financeiras

Como identificar o Código do Plano pelo seu aplicativo

Com o Código do Plano você poderá identificar qual plano foi assinado durante a instalação de seu aplicativo.

Para verificar o Código do Plano pelo seu aplicativo você deverá realizar uma consulta no **Content Provider** do aplicativo **GetStore** utilizando o código abaixo:

```
String authority = "br.com.getnet.getstore.provider";
String getInfosTable = "TerminalPlanCode";
Uri getstoreGetInfosUri = Uri.parse("content://" + authority + "/" +
getInfosTable);
Cursor cursor = getContentResolver().query(getstoreGetInfosUri, null, null, null,
null);
if (cursor != null) {
    while (cursor.moveToNext()) {
        String planCode = cursor.getString(cursor.getColumnIndex("planCode"));
    }
}
```

Sua consulta irá retornar uma **String** de 1 a 30 caracteres que representa o Código do Plano que foi assinado durante a instalação de seu aplicativo.

Caso o Código do Plano não tenha sido cadastrado, o retorno desta consulta será **null**.

Atenção! Para que seu aplicativo possa consultar o **Content Provider** do aplicativo GetStore é necessário adicionar a seguinte **permissão** no arquivo **AndroidManifest.xml** da sua aplicação:

```
<uses-permission android:name="br.com.getnet.getstore.provider.READ_DICTIONARY"/>
```

Submetendo um Aplicativo

Crie um usuário no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>) e faça o upload do seu aplicativo.

Ao submeter o aplicativo na GetStore, é de **responsabilidade do desenvolvedor** conceder as informações de testes ao submeter o aplicativo para certificação, preencher os campos com as informações abaixo:

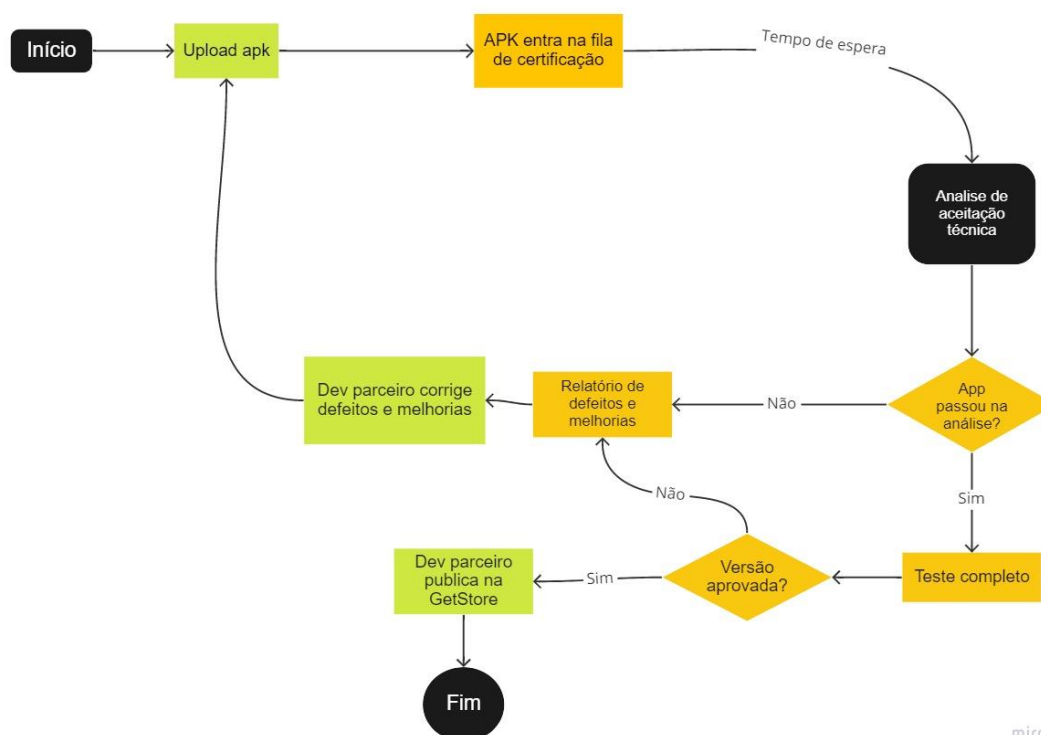
- Documentação com **todos** os fluxos de telas e informações das funcionalidades do aplicativo.
- Dados necessários para os testes (Ex.: credenciais de acesso, IPs).

- Release Notes: Nesta opção você deve informar quais foram as alterações feitas em sua aplicação, este campo será exibido para todos os usuários da GetStore.
- Quais recursos externos o aplicativo utiliza e os seus respectivos IPs/URLs (Exemplos: Crashlytics, Analytics, Firebase, servidores, etc.).
- Contatos para suporte ao aplicativo.
- **IMPORTANTE:** Não serão aceitos aplicativos com conteúdo "em construção" e "em desenvolvimento". O apk submetido sempre deve ter o *versioncode* maior que o anterior, mesmo se for uma versão de correção.

Processo de Certificação

Seu aplicativo será **certificado** pelo time de certificação da Getnet. O processo será **criterioso e considera as questões de segurança, como também os itens mandatórios.**

Considere o fluxo abaixo para entender como funciona o processo de certificação dos aplicativos submetidos:



Assim que for iniciado o processo de certificação de seu aplicativo, ele passará por uma Análise de Aceitação Técnica a fim de verificar se o aplicativo atende os [Requisitos para desenvolvimento](#).

Após a Análise de Aceitação Técnica, os testadores da Getnet irão focar nos testes do aplicativo de acordo com as funcionalidades do aplicativo. Durante este período, para agilizar, os testadores poderão **requisitar informações por e-mail, então mantenha seu contato atualizado no** Portal do Desenvolvedor. Após a finalização, será gerado um relatório indicando se a versão foi aprovada ou reprovada. Caso aprovada, pedimos que o desenvolvedor **pilote** a versão através da GetStore, antes da publicação definitiva.

Caso as meta-datas **priority_pay**, **kiosk_mode** ou **allow_print_permission** sejam utilizadas, serão realizados testes específicos para cada funcionalidade. Para informações de implementação, consulte a documentação **API de Pagamento**, disponível no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>).

Relatório dos Testes

O relatório de testes é dividido em 2 partes, conforme abaixo:

Defeitos

Aqui são listados todos os erros encontrados referentes a integração do aplicativo com o POS Digital da Getnet, ou seja, erros de API, SDK e recursos de hardware. Precisam ser corrigidos para aprovação da versão. Também serão listados erros do próprio aplicativo, que impeçam sua correta utilização.

Melhorias

São todas as situações que os testadores reportam sobre a usabilidade do aplicativo. A Getnet não considera que a melhoria precisa ser desenvolvida para aprovar o aplicativo.

Dúvidas Frequentes

Android

1. Pode alterar o packagename de um aplicativo?

Não, o packageName é a forma como o Android reconhece um aplicativo. O packageName é o identificador único de cada aplicativo.

2. É possível utilizar API do Android em POS de produção?

Sim, as únicas API disponíveis para uso são as do próprio Android.

Pagamento

1. Todo o controle da transação de adquirencia será realizado pela aplicação de pagamento da Getnet? Inclusive a situação das transações pendentes e confirmadas?

Sim, a aplicação de pagamento é responsável por toda parte da aquisição.

2. Quem deve imprimir o comprovante da transação de pagamento?

A aplicação de pagamento da Getnet. Quando o seu aplicativo chamar o método "StartActivityResult", a aplicação da Getnet tomará conta da tela do terminal e executará toda transação de pagamento até a impressão do CV e remoção do cartão (em caso de chip). Inclusive mostrará a mensagem na tela de sucesso ou erro. Essa mensagem será retornada para seu aplicativo com resposta da API.

3. A Getnet pode me enviar o apk de pagamento para eu fazer testes?

Não. Por questões de segurança, a aplicação de pagamento da Getnet não pode ser enviada. Por isso criamos o aplicativo "Rebatedor", com ele você pode simular o protocolo e as chamadas da API de Pagamento. Disponível para download no Portal do Desenvolvedor (<https://getstore.getnet.com.br/#/home>).

4. Quais são os cartões aceitos pela Getnet nos terminais Android?

No site <https://site.getnet.com.br/bandeiras/> você pode ver a lista de bandeiras aceitas pela Getnet.

Terminal

1. Gostaria de apresentar para meus clientes em uma tela maior, como eu posso fazer?

O equipamento não suporta o cast, mas pode ser instalado no terminal algum aplicativo de visualização, como o vysor.

2. Estou utilizando um terminal GPOS 700 para realizar os testes em minha aplicação, porém o rebatedor não funciona.

Não estamos mais trabalhando com os terminais GPOS 700, desta forma não conseguimos garantir que nossas aplicações vão funcionar para este terminal.

3. Como faço para adquirir um terminal homologado pela Getnet?

Você terá que entrar em contato com os canais de venda das fabricantes.

Ingenico venda: payments@scansource.com / renato.fonseca@scansource.com

Newland venda: sergio.paulillo@tefti.com.br

TecToy Sunmi venda: rodrigo.santos@tectoy.com

4. Não conseguimos debugar a solução no dispositivo Newland N910/Ingenico APOS A8. Há alguma dica ou solução para essa questão?

Primeiro verifique se seu terminal realmente é Mockup(dev). Para essa verificação, deve ter um ícone piscando no canto inferior esquerdo da tela ou uma imagem estática indicando ser “debug”, caso ele não seja Mockup você tem em mãos um terminal produtivo desta forma não será possível debugar sua aplicação pois a porta USB's não funcionam para debug nos terminais produtivos.

WiFi/3G/4G

1. Existe alguma limitação ou requisito para o uso do Wifi?

Sim, o POS Digital, por questões de segurança e PCI, não pode se conectar à uma rede wifi sem segurança (sem senha), assim como também não pode entrar em redes que exijam autenticação via browser.

2. Eu terei que colocar o meu simcard 3G/4G no POS?

Não, o simcard será da Getnet.

Rebatedor

1. Como vou saber se a integração do meu aplicativo com a aplicação da Getnet está funcionando?

A utilização do Rebatedor garante que a API de Pagamento está de acordo, porém somente durante a certificação interna da Getnet que teremos a validação completa. Durante esta certificação, a Getnet utiliza a aplicação de pagamento oficial para os testes de integração.

2. O Rebatedor não está respondendo corretamente quando sucesso no meu smartphone, podem me ajudar?

O Rebatedor foi desenvolvido para funcionar no terminal de desenvolvimento, mas ele funciona em smartphones com Android até 8.0, recomendamos utilizar os terminais que oferecemos suporte (Ingenico APOS A8 e Newland N910)

3. O Rebatedor não pede o cartão, está correto?

Sim, o Rebatedor apenas simula as possíveis respostas do cartão.

4. Quero utilizar o Rebatedor no meu celular, é possível?

Sim é possível, porém como o Rebatedor foi desenvolvido para ser usado em terminais Ingenico APOS A8 e Newland N910 você pode encontrar algumas limitações em relação ao seu uso em um celular.

Comandos ADB

1. Meu computador não reconhece o comando ADB devices, o que eu faço?

Provavelmente ele não foi instalado corretamente, no gerenciador de dispositivos do computador deve ter um "Android > Android Composite ADB Interface (landi platforms)". Caso contrário irá aparecer em "Outros dispositivos > Android".

2. Conseguimos testar em POS Digitais produtivos?

Não, porque as portas USB's não funcionam para debug, apenas são instalados aplicativos através da GetStore. Isso para aplicativos já certificados por nós.

3. Como eu instalo meu aplicativo no terminal?

Primeiro verifique se ele realmente é Mockup(dev). Para essa verificação, deve ter um ícone piscando no canto inferior esquerdo da tela ou uma imagem estática indicando ser "debug". Sendo mockup, pode ser utilizado os comandos ADB para realizar a instalação do aplicativo. Acesse para mais informações:

<https://developer.android.com/studio/command-line/adb>.

SDK

1. A versão do android é 5.1 (Lollipop), basta minha aplicação suportar o Android 5.1 (22) ou necessariamente tenho que compilar para SDK 22?

Isso. Basta ter suporte para versão 22.

2. Posso compilar em um SDK superior e informar a minSDK 22?

O target pode ser superior, mas o minSDK deve ser 22.

3. A máquina que estamos testando veio com o sistema operacional Android na versão 5.1, existe a possibilidade de atualizarmos essa versão para uma mais moderna?

Os modelos de terminais que estamos trabalhando hoje são android 5.1 e não temos expectativa de atualizar. Pode acontecer, no futuro, de passarmos a trabalhar com outros modelos de POS com Android mais atual.

4. Tenho dúvidas quanto ao funcionamento de algumas partes do SDK, podem me ajudar?

Você pode pedir suporte pelo e-mail parceiros_posdigital@getnet.com.br

Recomendação do time de suporte: Prezados parceiros, quando entrar em contato com nosso time detalhe bem sua dúvida, envie anexos de evidência assim conseguimos diagnosticar melhor seu problema e achar uma solução com maior agilidade.

Certificação

1. **Toda vez que eu mandar uma versão nova a Getnet irá certificar?**

Sim, mas o tempo de testes pode variar.

2. **Meu aplicativo ainda não está pronto, mas eu posso iniciar a certificação da Getnet para avaliar a integração com pagamento?**

Não, pois a Getnet entende que um aplicativo submetido já está validado pelo desenvolvedor e apto para ser publicado para produção. Inclusive, se o apk for aprovado pela certificação, o mesmo será assinado pela Getnet e já estará pronto para produção.

3. **Vou enviar minha aplicação para a certificação, quais arquivos devem ser enviados?**

Você deve enviar para nosso time de certificação o **arquivo .apk** da sua aplicação e **documentação da aplicação** contendo os fluxos de telas e dados necessários para acessar a aplicação, exemplo: usuário, senha, servidor, IPs etc. A **documentação completa** ajuda nosso time a certificar seu aplicativo com mais agilidade.

4. **Como serei avisado quando meu aplicativo for certificado?**

Quando o time de certificação começar a certificar seu aplicativo será enviado um e-mail informando que seu aplicativo está em análise. É muito importante que você fique atento ao seu e-mail, pois nosso time de certificação pode precisar de um suporte referente ao seu aplicativo, este contato será feito através do e-mail cadastrado no **Portal do Desenvolvedor**.

5. **O processo de certificação da Getnet funciona para testar todo meu aplicativo?**

Não, nosso processo de certificação é avaliar se seu aplicativo segue os requisitos mandatórios de implementação e avaliar a qualidade das chamadas do seu aplicativo com nossa API de Pagamento. A qualidade da sua aplicação vai depender de como você desenvolveu seu aplicativo por isso é necessário realizar os testes de qualidade antes de submeter a versão.

Contato dos fabricantes

Ingenico venda: payments@scansource.com / renato.fonseca@scansource.com

Ingenico suporte: BRGDAPOS@ingenico.com

Newland venda: sergio.paulillo@tefti.com.br

Newland suporte: daniel@npi.newlandpayment.com

TecToy Sunmi: rodrigo.santos@tectoy.com