

# Guia de Build do APK - Daure Express

## Pré-requisitos

1. **Android Studio** instalado (última versão estável)
2. **JDK 17** ou superior
3. **Node.js 18+** e **Yarn**
4. Conta no **Firebase Console** (para push notifications)

## PASSO 1: Configurar Push Notifications Nativas (OBRIGATÓRIO)

Para que as notificações push funcionem no APK (como iFood), você precisa configurar o Firebase:

### 1.1 Criar Projeto no Firebase

1. Acesse [Firebase Console](https://console.firebaseio.google.com/) (<https://console.firebaseio.google.com/>)
2. Clique em “**Adicionar projeto**”
3. Nome do projeto: `daure-express` (ou outro de sua preferência)
4. Desative o Google Analytics (opcional)
5. Clique em “**Criar projeto**”

### 1.2 Adicionar App Android ao Firebase

1. No painel do Firebase, clique no ícone **Android** (
2. Preencha:
  - **Nome do pacote Android:** `com.daureexpress.app`
  - **Apelido do app:** Daure Express
  - **Certificado de assinatura SHA-1:** (veja como obter abaixo)
3. Clique em “**Registrar app**”

### 1.3 Obter SHA-1 do Certificado

No terminal do Android Studio ou CMD:

```
# Para debug (desenvolvimento)
keytool -list -v -keystore ~/.android/debug.keystore -alias androiddebugkey -storepass android -keypass android

# Para release (produção) - use sua keystore
keytool -list -v -keystore SUA_KEYSTORE.jks -alias SEU_ALIAS
```

Copie a linha **SHA1** e cole no Firebase.

### 1.4 Baixar google-services.json

1. Após registrar, clique em “**Baixar google-services.json**”
2. **IMPORTANTE:** Copie o arquivo para:  
`android/app/google-services.json`

3. Clique em “**Próximo**” até finalizar

## 1.5 Ativar Cloud Messaging

1. No Firebase Console, vá em **Configurações do projeto > Cloud Messaging**
2. Se não estiver ativo, clique em “**Ativar Cloud Messaging API (V1)**”
3. Copie a **Server Key** (você já tem configurada no backend)

## PASSO 2: Preparar o Projeto

### 2.1 Instalar Dependências

```
cd nextjs_space
yarn install
```

### 2.2 Sincronizar Capacitor

```
# Sincronizar plugins com Android
yarn cap sync android
```

## PASSO 3: Abrir no Android Studio

### 3.1 Abrir o Projeto

```
yarn cap open android
```

Ou abra manualmente o Android Studio e selecione a pasta `android/`.

### 3.2 Aguardar Sincronização do Gradle

- O Android Studio vai baixar as dependências automaticamente
- Aguarde a barra de progresso no canto inferior
- Pode levar 5-10 minutos na primeira vez

## PASSO 4: Gerar APK

### Opção A: APK de Debug (Para Testes)

1. No Android Studio: **Build > Build Bundle(s) / APK(s) > Build APK(s)**
2. Aguarde a compilação
3. Clique em “**locate**” quando aparecer a notificação
4. O APK estará em: `android/app/build/outputs/apk/debug/app-debug.apk`

### Opção B: APK Assinado (Para Distribuição)

1. No Android Studio: **Build > Generate Signed Bundle / APK**
2. Selecione **APK > Next**

3. Criar nova keystore (primeira vez):
  - Clique em **Create new...**
  - Defina localização e senha
  - Preencha os dados (Nome, Organização, Cidade, País)
  - Defina alias e senha do alias
4. Ou use keystore existente
5. Selecione **release > Finish**
6. APK estará em: `android/app/release/app-release.apk`

## Opção C: AAB para Google Play

1. No Android Studio: **Build > Generate Signed Bundle / APK**
  2. Selecione **Android App Bundle > Next**
  3. Configure a keystore
  4. Selecione **release > Finish**
  5. AAB estará em: `android/app/release/app-release.aab`
- 

## Solução de Problemas

### “Generate Signed Bundle / APK” não aparece

Isso acontece quando o Gradle ainda não sincronizou completamente:

1. **Aguarde** a sincronização do Gradle terminar (veja barra inferior)
2. **Se necessário**, vá em **File > Sync Project with Gradle Files**
3. **Fechar e reabra** o Android Studio
4. **Verifique se não há erros** no painel “Build” na parte inferior

### Erro de SDK não encontrado

Vá em **File > Project Structure > SDK Location** e configure:

- Android SDK Location
- JDK Location (deve ser JDK 17+)

### Erro “google-services.json not found”

O arquivo `google-services.json` é OBRIGATÓRIO para push notifications:

1. Certifique-se que está em `android/app/google-services.json`
2. Execute `yarn cap sync android` novamente

### Push Notifications não funcionam

1. Verifique se `google-services.json` está no lugar correto
  2. Confirme que o SHA-1 no Firebase corresponde à sua keystore
  3. Verifique se Cloud Messaging está ativado no Firebase
  4. Teste enviando uma notificação pelo Firebase Console
-

## Configurações do App

Configuração	Valor
Package Name	com.daureexpress.app
App Name	Daure Express
Min SDK	22 (Android 5.1)
Target SDK	34 (Android 14)
Server URL	<a href="https://sistemadaureredeentregas.abacusai.app">https://sistemadaureredeentregas.abacusai.app</a>

## → Instalar o APK

### Via ADB (USB)

```
adb install -r app-release.apk
```

### Via Arquivo

1. Transfira o APK para o celular
2. Abra o gerenciador de arquivos
3. Toque no APK
4. Permita instalação de “Fontes desconhecidas”
5. Instale e abra o app

## Testar Push Notifications

1. Abra o app no celular
2. Faça login como **Entregador**
3. Vá em “**Pedidos Disponíveis**”
4. Ative o status “**Online**”
5. Crie um pedido como **Cliente** pelo navegador
6. O celular deve receber uma notificação com som!

### Testar pelo Firebase Console

1. Vá em Firebase Console > **Cloud Messaging**
2. Clique em “**Criar sua primeira campanha**”
3. Selecione “**Mensagens do Firebase Notifications**”
4. Título: “Teste Push”
5. Texto: “Notificação de teste funcionando!”

6. Segmentação: Seu app Android
  7. Envie e verifique se o celular recebe
- 

## Checklist Final

---

- [ ] google-services.json está em android/app/
  - [ ] SHA-1 cadastrado no Firebase
  - [ ] Cloud Messaging ativado no Firebase
  - [ ] yarn cap sync android executado
  - [ ] Gradle sincronizado sem erros
  - [ ] APK/AAB gerado com sucesso
  - [ ] Push notifications testadas e funcionando
- 

## Suporte

---

Em caso de dúvidas, entre em contato pelo WhatsApp do suporte técnico.