Building Frigate Mobile App

Prerequisites

- Node.js 18+ installed
- Android Studio (for Android builds) or Xcode (for iOS builds)

Quick Start - Local Development

```
# Install dependencies
npm install

# Start development server
npx expo start
```

Then:
- Scan QR code with Expo Go app
- Press ‘a’ for Android emulator
- Press ‘i’ for iOS Simulator
- Press ‘w’ for web browser

Building APK for Android (Easiest)

Method 1: EAS Build (Recommended)

```
# Install EAS CLI globally
npm install -g eas-cli

# Login to Expo (create free account at expo.dev)
eas login

# Configure EAS
eas build:configure

# Build APK
eas build --platform android --profile preview
```

This builds in the cloud and gives you a download link for the APK.
Free tier: 30 builds/month

Method 2: Local Build

```
# Generate Android project
npx expo prebuild --platform android

# Build APK locally (requires Android SDK)
cd android
./gradlew assembleRelease

# APK will be at:
# android/app/build/outputs/apk/release/app-release.apk
```

Building for iOS

Using EAS (Cloud Build)

```
# Build for iOS
eas build --platform ios --profile preview

# Or build for TestFlight/App Store
eas build --platform ios --profile production
```

Note: iOS builds require an Apple Developer account ($99/year)

Local Build (on Mac with Xcode)

```
# Generate iOS project
npx expo prebuild --platform ios

# Open in Xcode
open ios/frigatemobileapp.xcworkspace

# Build and run from Xcode
# or from command line:
npx expo run:ios
```

Building Both Platforms

```
# Build for both Android and iOS simultaneously
eas build --platform all
```

Distribution Options

Android:

- **Sideload APK**: Install directly on device via USB or file transfer
- **Google Play**: Upload AAB file to Play Console
- **Internal Testing**: Share APK link via EAS

iOS:

- **TestFlight**: Beta testing (up to 10,000 testers)

- **Ad-hoc**: Limited to 100 devices
- **App Store**: Public distribution

Signing Keys

Android

EAS handles signing automatically, or create your own keystore:

```
keytool -genkeypair -v -storetype PKCS12 -keystore my-upload-key.keystore \
  -alias my-key-alias -keyalg RSA -keysize 2048 -validity 10000
```

iOS

EAS handles certificates and provisioning profiles automatically with your Apple Developer account.

Troubleshooting

"EXPO_PUBLIC_API_BASE_URL not found"

This is expected - the app doesn't need a backend, it connects directly to your Frigate instance.

Build fails with "SDK location not found"

Set ANDROID_HOME environment variable:

```
export ANDROID_HOME=$HOME/Library/Android/sdk
```

iOS build fails with code signing error

Ensure you're logged into Xcode with your Apple ID (Xcode > Preferences > Accounts)

App Configuration

Edit `app.json` to customize:
- App name: `"name": "Your App Name"`
- Bundle ID: `"bundleIdentifier": "com.yourname.frigateapp"`
- Package name: `"package": "com.yourname.frigateapp"`
- Version: `"version": "1.0.0"`
- Icons and splash screen

Testing Your Frigate Connection

1. Launch the app
2. Enter your Frigate URL:
   - Local: `http://192.168.101.4:5000`
   - Remote: `https://frigate.beckettnet.org`
3. Login with your Frigate username and password
4. Test camera streaming:
   - **HD**: WebRTC (requires port 1984 accessible)
   - **High**: MSE streaming
   - **Low**: MJPEG fallback

Support

For issues:
- Expo documentation: https://docs.expo.dev
- Frigate API: https://docs.frigate.video
- React Native docs: https://reactnative.dev