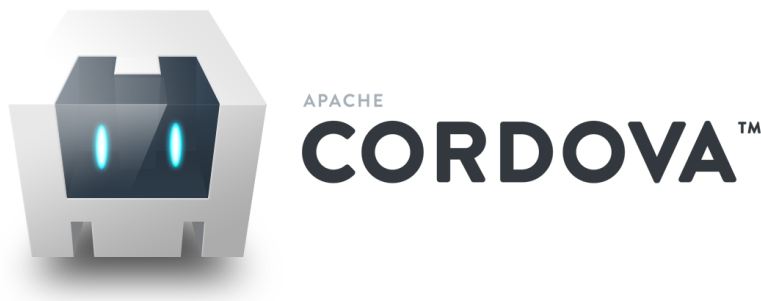


Convert Website to **Android** App Using **Apache Cordova** & Publishing app to **Google Play Store**



Contents of Tutorial

1. Installation of required software.
2. Converting Website to App.
3. Publishing app to Google Play Store.



1. Installation of required software.

Before proceed we need following software to be installed and setup on your computer.

- I. Java Development Kit (JDK)
- II. Android SDK
- III. NodeJS
- IV. Apache Cordova

Installing JDK:

Download and install jdk from below link

<https://www.oracle.com/java/technologies/javase-jdk16-downloads.html>

Installing Android SDK

Downloading link <https://developer.android.com/studio#downloads>

You will need to download the Android SDK without Android Studio bundled. Go to [Android SDK](#) and navigate to the **SDK Tools Only** section. Copy the URL for the download that's appropriate for your build machine OS.

Get just the command line tools

If you do not need Android Studio, you can download the basic Android command line tools below. You can use the included [sdkmanager](#) to download other SDK packages.

These tools are included in Android Studio.

Platform	SDK tools package	Size	SHA-256 checksum
Windows	tools_r25.2.3-windows.zip	292 MB (306,745,639 bytes)	23d5686ffe489e5a1af95253b153ce9d6f933e5dbabe14c494631234697a0e08
Mac	tools_r25.2.3-macosx.zip	191 MB (200,496,727 bytes)	593544d4ca7ab162705d0032fb0c0c88e75bd0f42412d09a1e8daa3394681dc6
Linux	tools_r25.2.3-linux.zip	264 MB (277,861,433 bytes)	1b35bcb94e9a686dff6460c8bca903aa0281c6696001067f34ec00093145b560

See the [SDK tools release notes](#).

Installing NodeJS/NPM

Download and install NodeJS from below links

<https://nodejs.org/en/download/>



Please subscribe our channel from <https://youtube.com/sulochanatutorials>

Installing Apache Cordova

Cordova is installed using the Node Package Manager (npm). Type the following in the command window to install:

```
npm install -g cordova
```

You can follow below tutorial to setup apache cordova

<https://cordova.apache.org/docs/en/10.x/guide/platforms/android/index.html>



Converting Website to Android App

1. Creating a cordova project.

```
cordova create directory_name package_name app_title
```

Example:

```
cordova create hello com.example.hello HelloWorld
```

Once you create project change directory to project directory and type following command to check all requirements installed properly

```
cordova requirements
```

```
Administrator: C:\Windows\System32\cmd.exe

C:\Users\SulochanaTutorials\Desktop\agridoctornew>cordova requirements

Requirements check results for android:
Java JDK: installed 1.8.0
Android SDK: installed true
Android target: installed android-29,android-28,android-22
Gradle: installed C:\Gradle\gradle-6.3\bin\gradle.BAT

C:\Users\SulochanaTutorials\Desktop\agridoctornew>
```

2. Add platform android to project.

Once you create project add platform android to project using following command

```
Cordova platform add android
```

3. Install necessary plug-ins.

Install any plugin using below command

```
Cordova plugin add plug-in-name
```



We need to install following plugins

- cordova-plugin-dialogs
- cordova-plugin-inappbrowser
- cordova-plugin-network-information

Type below command to list a installed plug-ins

```
cordova plugin
```

4. Update config.xml file.

Add Access Origin to your website link

```
<access origin="http://mywebsite.com*" />
<access origin="http://www.mywebsite.com*" />
<access origin="https://www.mywebsite.com*" />
<access origin="https://mywebsite.com*" />
```

Add App Icon

```
<platform name="android">
  <allow-intent href="market:*" />
  <icon src="res/images/ldpi.png" density="ldpi" />
  <icon src="res/images/mdpi.png" density="mdpi" />
  <icon src="res/images/hdpi.png" density="hdpi" />
  <icon src="res/images/xhdpi.png" density="xhdpi" />
  <icon src="res/images/xxhdpi.png" density="xxhdpi" />
  <icon src="res/images/xxxhdpi.png" density="xxxhdpi" />
</platform>
```

```
<style type="text/css">
    #body{ position: fixed; width: 100%; height: 100%; background: white; }
    #content-holder { height: 300px; position: absolute; width: 100%; top: 50%; transform:
    translate(0, -50%); text-align: center; text-align: center; }
    a{ color: #b92b46; text-decoration: none; font-weight: 900;font-size: 17px; }
</style>
<div id="body">
    <div id="content-holder">
        <br>
        <h4 style="text-align: center;text-transform: capitalize;">Please wait..</h4>
        
    </div>
</div>
```

Copy pleasewait.gif and logo.png image inside www/img/ folder

JavaScript Code: Put below JavaScript code inside index.js file

```
function redirectTo(location) {
    window.open = cordova.InAppBrowser.open;
    //showPleasewait('Redirecting','Please wait');
    var d = setInterval(function () {
        window.open(location, '_self',
        'location=no,zoom=no,toolbar=no');
        clearInterval(d);
    }, 1500);
}
var app = {
    // Application Constructor
    initialize: function () {
        document.addEventListener('deviceready',
        this.onDeviceReady.bind(this), false);
    },
    // deviceready Event Handler
    //
    // Bind any cordova events here. Common events are:
    // 'pause', 'resume', etc.
    onDeviceReady: function () {
        this.receiveEvent('deviceready');
    },
    // Update DOM on a Received Event
    receiveEvent: function (id) {
```



Converting website to android app using apache cordova and publishing app to playstore

```
        if (navigator.connection.type == Connection.NONE)
navigator.notification.alert('An internet connection is required to continue');
        } else {
            redirectTo("https://agridoctor.net");
        }
    }
};

app.initialize();
```

Replace application tag in AndroidManifest.xml file

You will find this file inside `..\platforms\android\app\src\main`

Add `android:usesCleartextTraffic="true"`

```
<application android:hardwareAccelerated="true"
android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
android:supportsRtl="true" android:usesCleartextTraffic="true">
```

5. Create debug APK file.

Create APK file using following command

```
cordova build android
```

6. Creating Signed APK.

- a. Go to your cordova project for generate our release build:

```
cordova build --release android
```

You will find unsigned APK file inside ... `android\app\build\outputs\apk\release`

- b. Key Generation:



Please subscribe our channel from <https://youtube.com/sulochanatutorials>

Command:

```
keytool -genkey -v -keystore <keystoreName>.keystore -alias <Keystore AliasName> -keyalg <Key algorithm> -keysize <Key size> -validity <Key Validity in Days>
```

Example:

```
keytool -genkey -v -keystore agridoctor.keystore -alias agridoctor -keyalg RSA -keysize 2048 -validity 10000
```

3. Sign APK file

Place the generated keystore in ...\\platforms\\android\\ant-build

To sign the unsigned APK, run the jarsigner tool which is also included in the JDK:

Syntax:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore <keystorename> <Unsigned APK file> <Keystore Alias name>
```

Example:

```
jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore agridoctor.keystore app-release-unsigned.apk agridoctor
```

Step:4

Finally, we need to run the zip align tool to optimize the APK:

```
zipalign -v 4 app-release-unsigned.apk app-release-signedapk
```