

### عنوان: شروع کار با Apache Cordova در ویژوال استودیو 3#

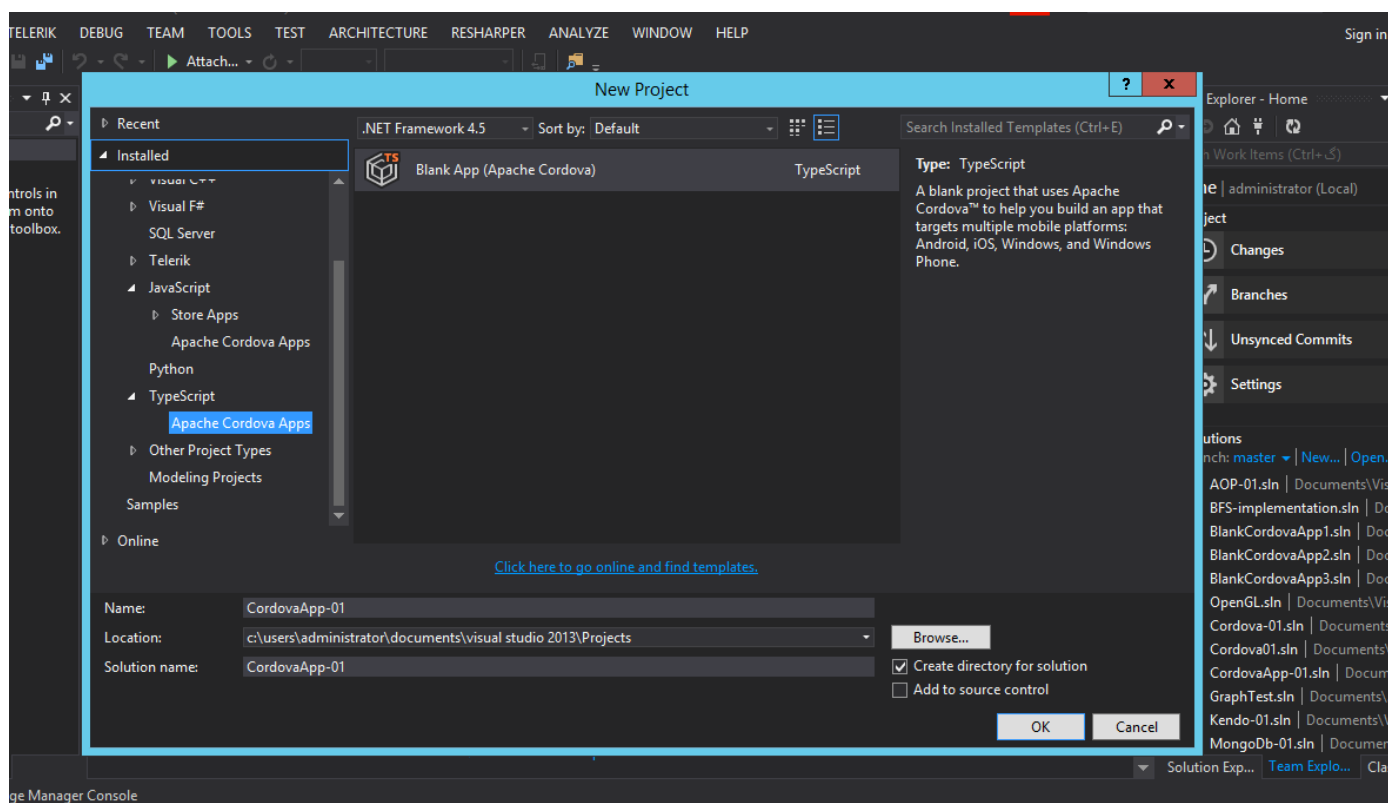
نویسنده: غلامرضا ربال

تاریخ: ۱۳۹۴/۰۱/۰۸

آدرس: [www.dotnettips.info](http://www.dotnettips.info)

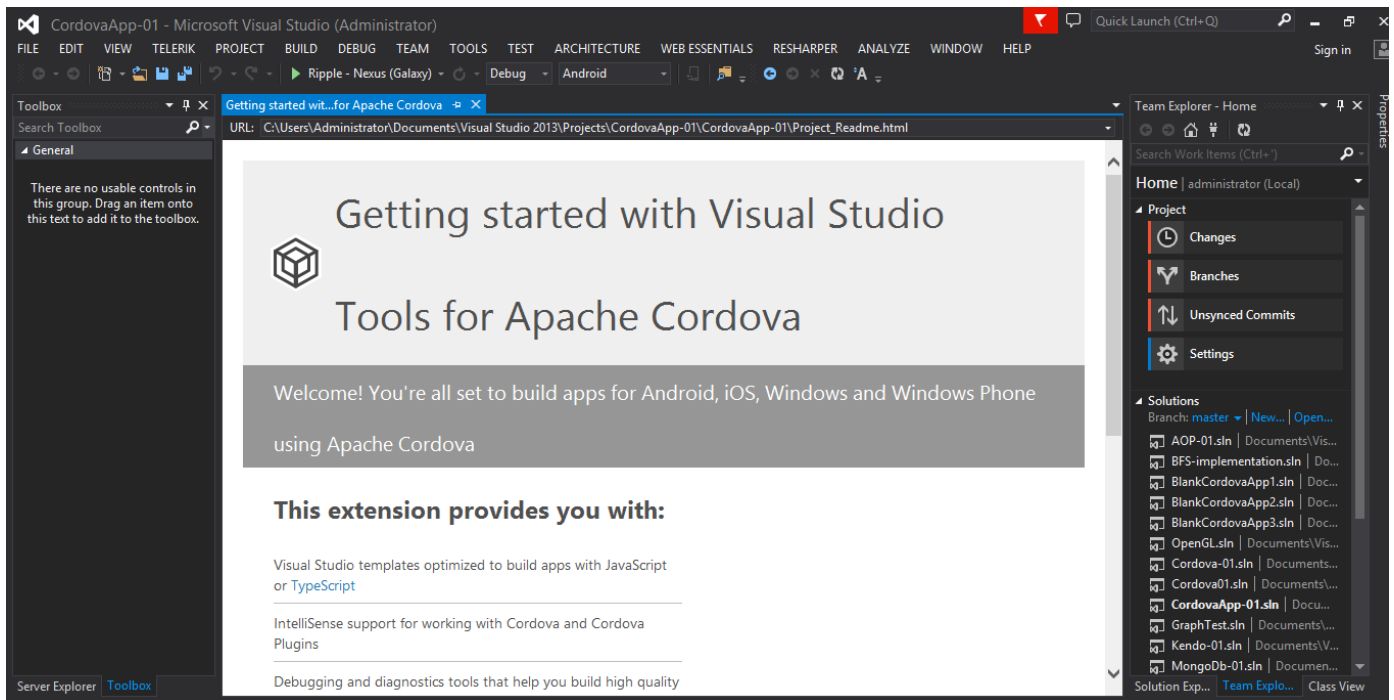
گروه‌ها: Visual Studio 2013, Apache Cordova, Mobile Programming

در قسمت قبل توانستیم ابزارهای لازم را برای Apache Cordova، نصب کنیم. در این قسمت یک پروژه‌ی ساده را ایجاد کرده و در مورد ساختار آن توضیح خواهیم داد. در ادامه‌ی مقالات از AngularJS، Bootstrap، TypeScript و jQuery Mobile هم در پروژه‌ها استفاده خواهیم کرد. برای شروع، از قسمت JavaScript یا TypeScript، یک پروژه‌ی از نوع Blank App ایجاد کنید. به شکل زیر:

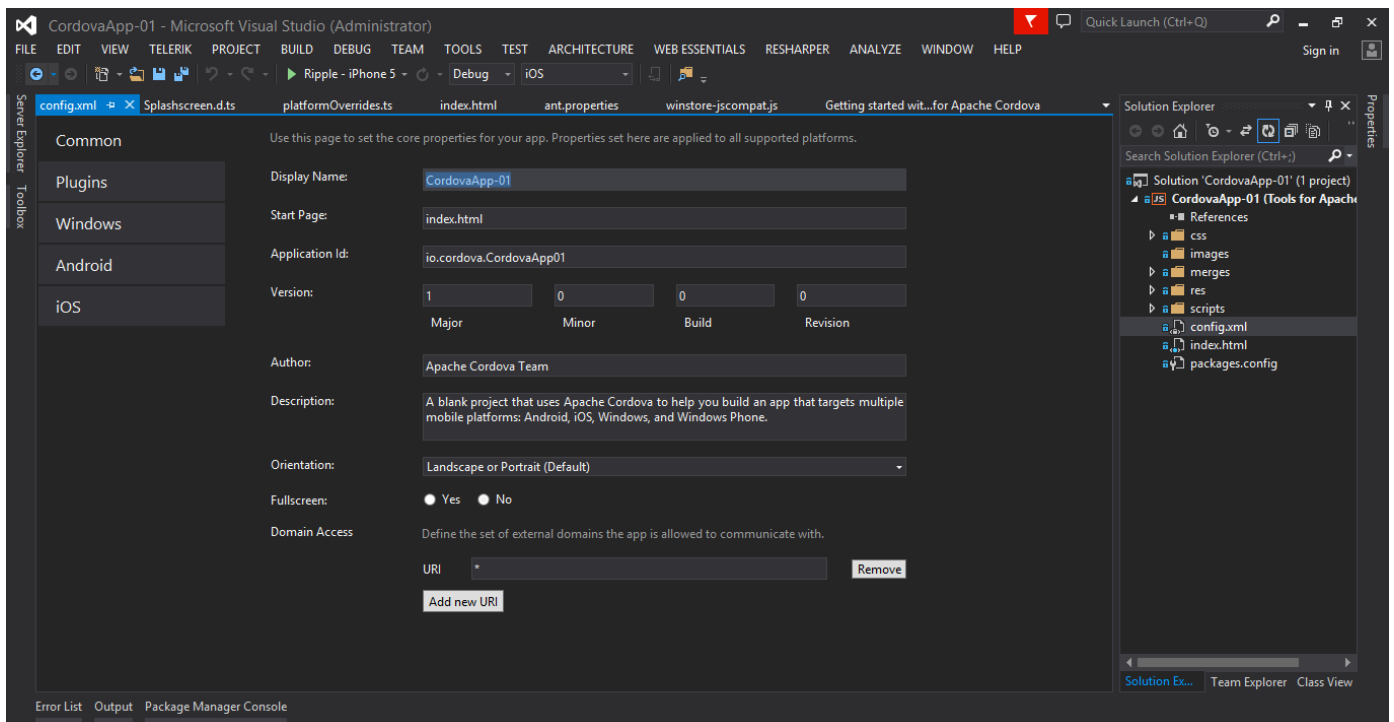


ترجیحا نوع TypeScript را انتخاب کردم. البته در داخل فایل ts، امکان نوشتن جاوا اسکریپت هم هست. بعد از ایجاد پروژه اگر با تصویری شبیه به تصویر زیر روبرو شدید، در نتیجه تنظیمات نصب و راه اندازی به درستی صورت گرفته است.

### شروع کار با Apache Cordova در ویژوال استودیو 3#



اگر به قسمت solution explorer دقت کنید، فایلی به نام config.xml را مشاهده خواهید کرد. با کلیک بر روی این فایل، یک صفحه‌ی گرافیکی باز خواهد شد که این امکان را به شما می‌دهد که پلاگین‌های مورد نیاز خود، تنظیمات مربوط به نرم افزار تولیدی (مانند تنظیم ورژن ویندوزی که می‌خواهید app شما بر روی آن اجرا شود) و تنظیمات مربوط به هر یک از پلتفرم‌ها را به صورت مجزا در اختیار داشته باشید.

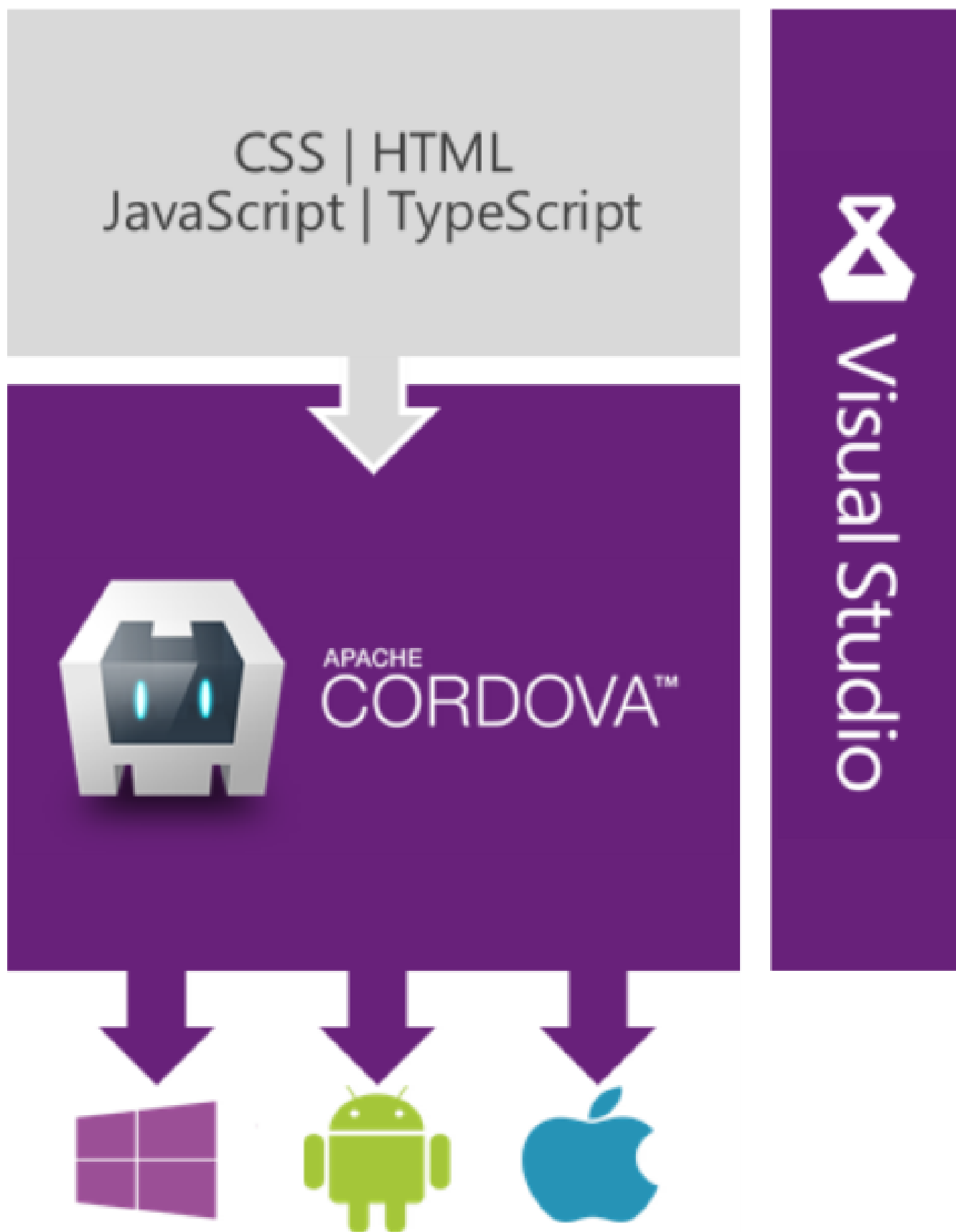


یک فایل index.html هم در قالب پیش‌فرض قرار داده شده که بعداً می‌توانید آن را تغییر دهید و یا صفحات دیگری را اضافه کنید.

همان طور که در قسمت‌های قبل گفته شد، قرار است ما یک وب اپلیکیشن طراحی کنیم و آن را درون Container بومی Cordova بسته بندی کنیم. لذا محدودیتی برای استفاده‌ی از کتابخانه‌های مرتبط با HTML ، CSS و JavaScript نداریم و در ادامه‌ی مقالات با مثال‌های متعددی از آن‌ها استفاده خواهیم کرد.

در فولدر `scripts-->typings-->cordova-->plugins` اینترفیس‌هایی که برای دسترسی به امکانات بومی دستگاه تلفن فعلا در Cordova پشتیبانی می‌شوند، قرار گرفته است.

برای استفاده از تکنولوژی‌های وب در محیط بومی دستگاه، در طی فرآیند کامپایل، Cordova یک اپلیکیشن را به وسیله دو چیز مهم که در زیر اشاره شده است، خواهد ساخت.



یک اپلیکیشن با یک کامپوننت WebView که با مرورگر یکپارچه شده است.

یه سری از منابعی که در داخل فایل‌های اپلیکیشن وب ما قرار دارند.

برای یکپارچه شدن APIهای Cordova با وب پیج موجود، اندکی کد نیاز داریم که برای انکار لینکی شبیه لینک زیر را در فایل html خود استفاده می‌کنیم که فقط بعد از کامپایل وجود خارجی دارد؛ به صورت زیر:

```
<script src="cordova.js"></script>
```

در پایان هم برای فهمیدن اینکه APIهای Cordova در دسترس هستند، می‌توانیم رخداد مربوط به deviceready را مدیریت کنیم؛ به صورت زیر:

```
document.addEventListener("deviceready", onDeviceReady, false);  
function onDeviceReady() { /* INIT */ }
```

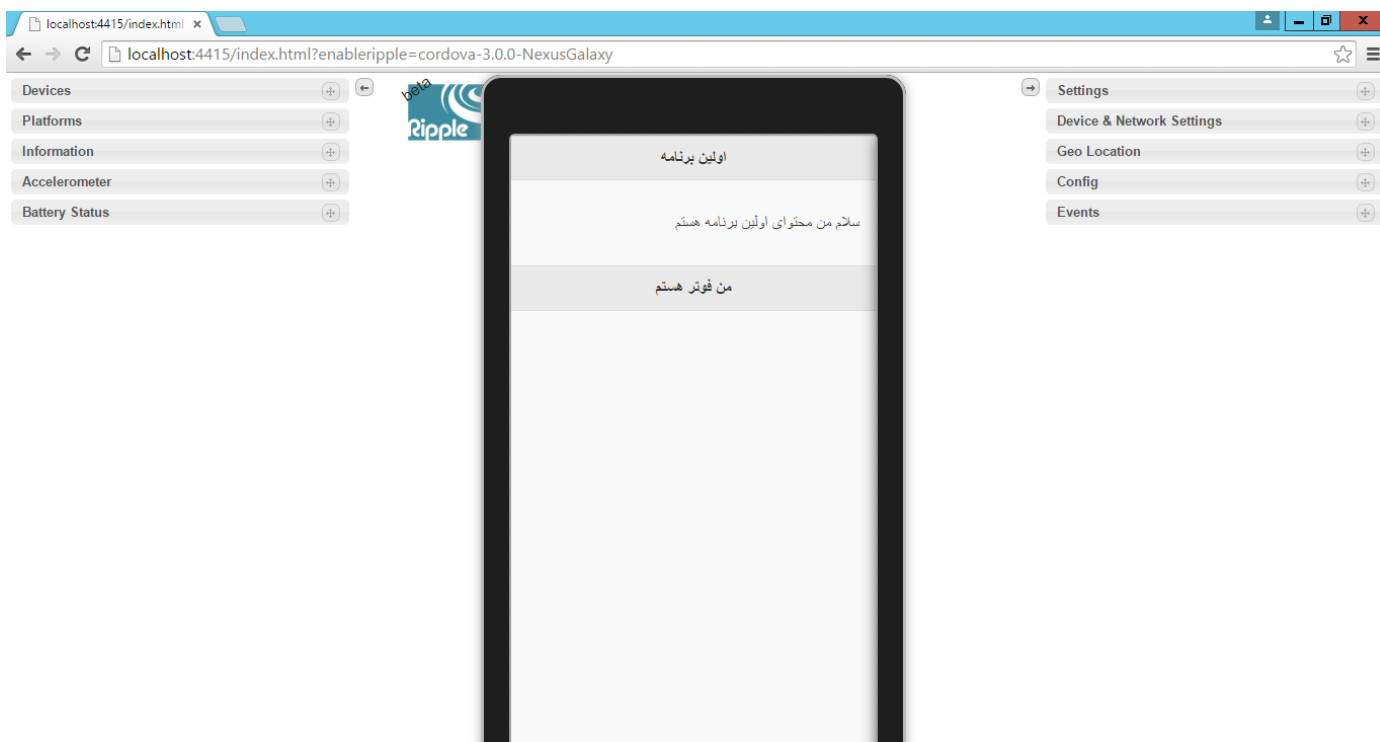
برای مدیریت رخدادهای مربوط به pause و resume هم که نشان دهنده‌ی ادامه برنامه (خارج شدن از حالت pause) و حالت تعلیق هستند، می‌توان به شکل زیر عمل کرد:

```
function onDeviceReady() {  
    // Handle the Cordova pause and resume events  
    document.addEventListener('pause', onPause, false);  
    document.addEventListener('resume', onResume, false);  
  
    // TODO: Cordova has been loaded. Perform any initialization that requires Cordova here.  
}  
  
function onPause() {  
    // TODO: This application has been suspended. Save application state here.  
}  
  
function onResume() {  
    // TODO: This application has been reactivated. Restore application state here.  
}
```

حال قصد داریم پروژه‌ی خود را که قرار است یک متن ساده را نشان دهد، با استفاده از شبیه ساز اجرا کنیم. برای این منظور از قسمت toolbar ویژوال استودیو، Solution Platform خود را انتخاب کنید و سپس می‌توانید شبیه ساز مورد نظر خود را انتخاب کرده و برنامه را اجرا کنید. در اینجا محیط مورد نظر من اندروید است و برای این منظور هم می‌توانم از شبیه ساز Android Emulator یا Ripple استفاده کنم. به دلیل سرعت کم شبیه ساز اندروید، می‌توانید شبیه ساز [YouWave](#) را دانلود و اجرا کرده و در قسمتی که شبیه ساز را از toolbar ویژوال انتخاب می‌کردید، این بار گزینه‌ی Device را انتخاب کنید. بعد از کامپایل برنامه‌ی شما، فایل apk تولید شده بر روی شبیه ساز نصب خواهد شد و شما قادر خواهید بود آنرا اجرا کنید.

### نتیجه‌ی نهایی

با شبیه ساز Ripple



مطالعه بیشتر

[https://msdn.microsoft.com/en-us/library/dn879821\(v=vs.140\).aspx](https://msdn.microsoft.com/en-us/library/dn879821(v=vs.140).aspx)

<http://blog.falafel.com/getting-started-with-cordova-and-multi-device-hybrid-app-in-visual-studio>

<http://www.codeproject.com/Articles/860150/Visual-Studio-and-Apache-Cordova>

**نکته :** وقتی پروژه را برای اولین بار اجرا می‌کنید شاید کمی طول بکشد تا نتیجه‌ی نهایی را ببینید و آن هم به دلیل این است که ویژوال استودیو باید مجموعه‌ای از package های مورد نیاز Cordova را دانلود کند.

در مقاله بعد با jQuery Mobile آشنا خواهیم شد و یک مثال برای کار کردن با آن در نظر خواهیم گرفت.

ادامه دارد ...

## نظرات خوانندگان

نویسنده:

خوزستان

تاریخ:

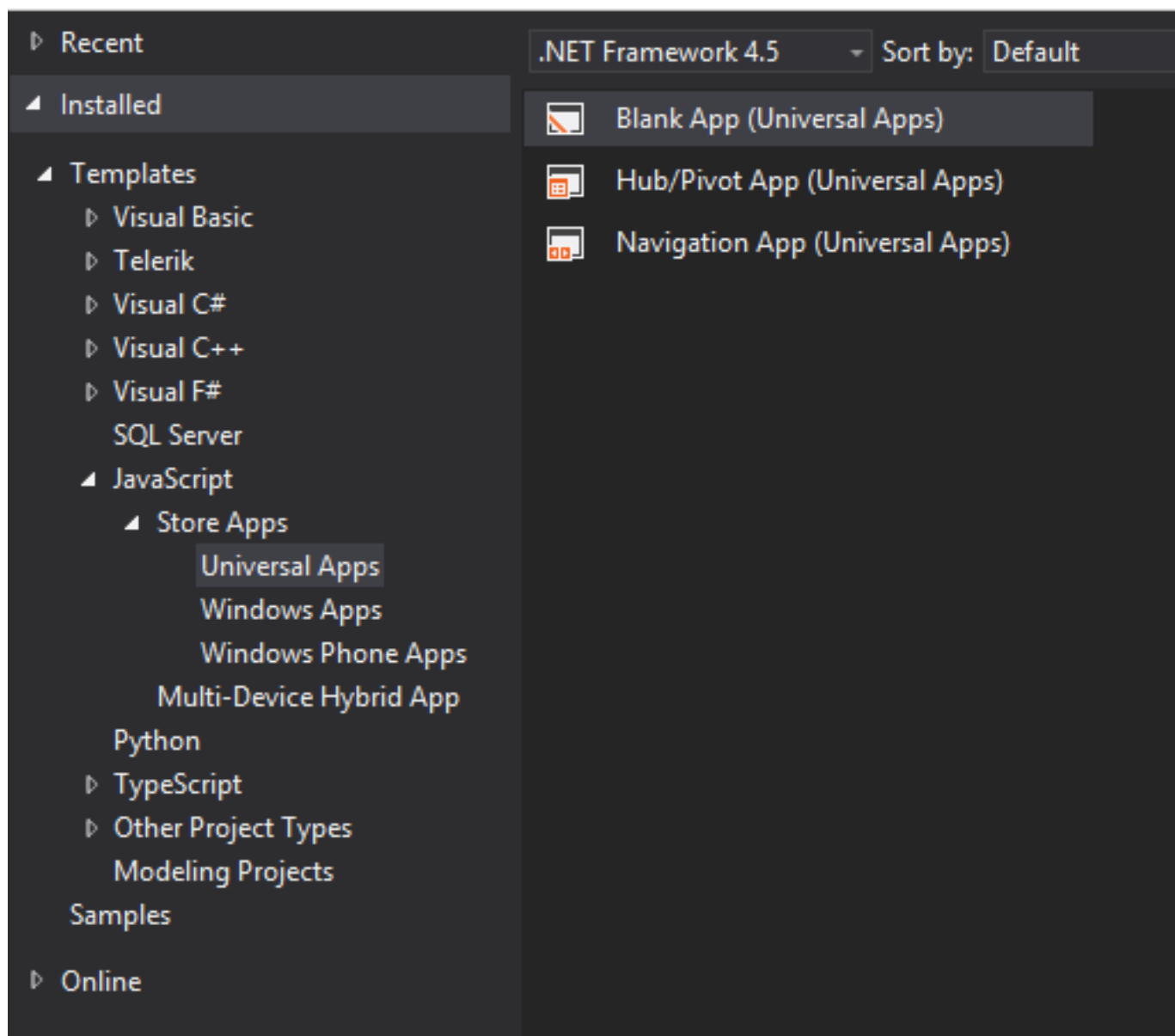
۱۰:۱۴ ۱۳۹۴/۰۱/۰۸

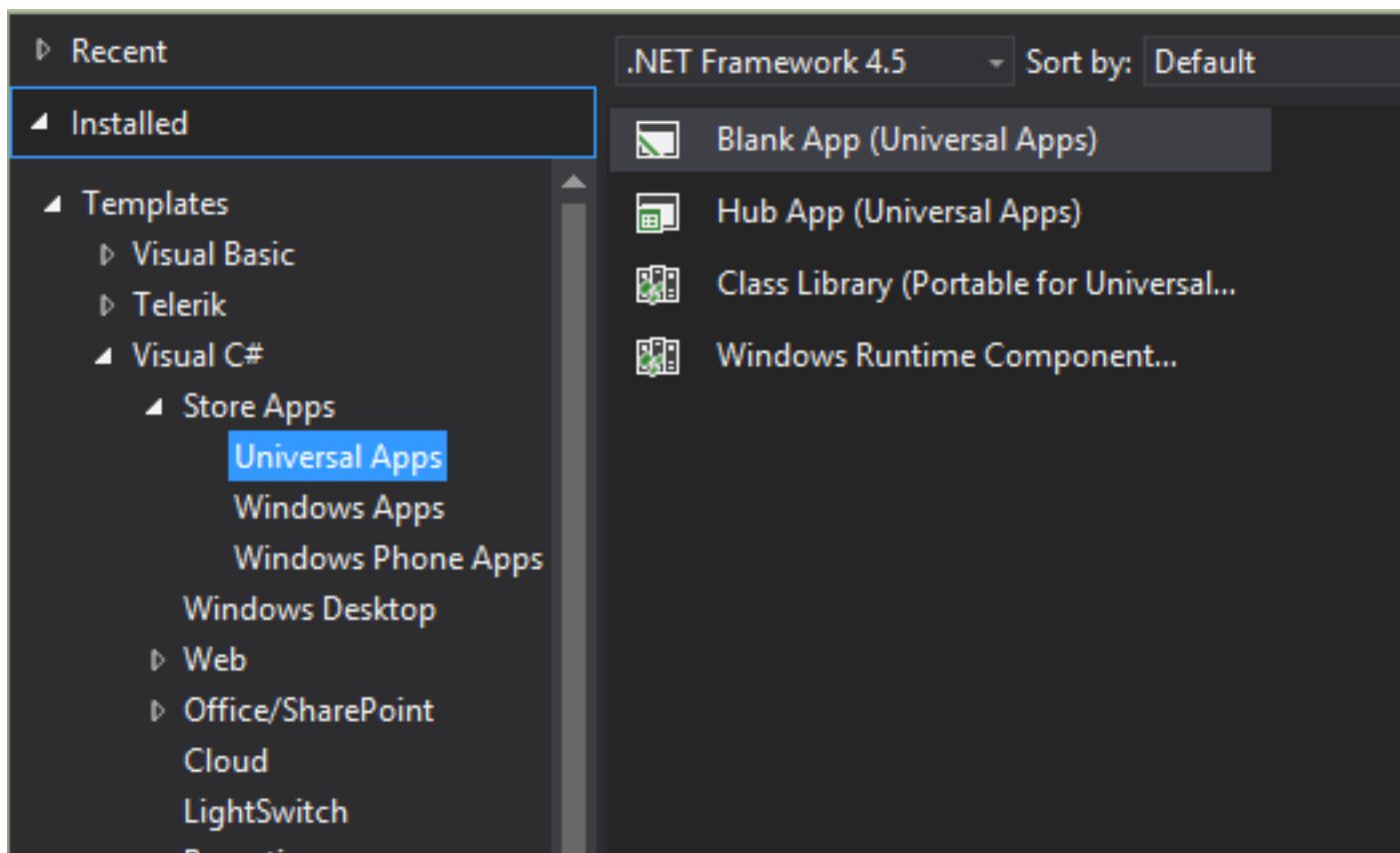
سلام ضمن تشکر بابت مقاله شما.

چند سوال :

1- آیا فقط با TypeScript و یا جاوا اسکریپت یا کتابخانه JQuery Mobile میتوان تمام نیازهای یک برنامه را تامین کرد ؟

2- دو روش دیگر برای ایجاد برنامه‌های موبایل وجود دارد :





تفاوت این دو روش ، با روش شما چیست ؟

3-در دو روش بالا زبانی مثل سی شارپ مورد استفاده قرار میگیرد ، در روش شما چطور ؟

4- آیا با توجه به محبوبیت زبان جاوا برای ساخت برنامه‌های اندرویدی ، روش مورد استفاده شما (cordova) میتواند با آن برابری کند ؟

تشکر

نویسنده: محسن خان  
تاریخ: ۱۳۹۴/۰۱/۰۸ ۱۰:۲۹

universal apps برای پلتفرم‌های مختلف مایکروسافت هست فقط. این مطلب یک قسمت اول هم داره: [شروع کار با Apache Cordova در ویژوال استودیو #1](#) . اونجا توضیح داده که این روش چند سکویی هست (یعنی فقط مختص به اندروید نیست). دسترسی به امکانات native دستگاه‌ها رو هم داره.

البته فقط این روش نیست که الان استفاده از جاوا اسکریپت رو شروع کرده برای توسعه‌ی برنامه‌های موبایل چندسکویی. شرکت تلریک هم اخیرا native script رو ارائه داده: <http://www.telerik.com/nativescript>

نویسنده: غلامرضا ربال  
تاریخ: ۱۳۹۴/۰۱/۰۸ ۱۱:۲۴

سلام.

1-بله البته در کنار html5 و css ، هدف اصلی Cordova هم همین است. در مقالات قبلی گفته شد که قرار است یک وب اپلیکیشن را در Cordova Container بسته بندی کنیم .

2- [این قسمت](#) و [اینجا](#) را مطالعه کنید . بنده کار نکردم . ولی فکر میکنم برای ساخت اپلیکیشن‌های همگانی ویندوز استفاده میشوند



- 3- خیر این از زیان سی شارپ استفاده نخواهد شد. هدف Cordova این نیست.
- 4- مطمئناً نمیتواند برابری کند حداقل در زمان حال ( [مزایا و معایب خود را دارد](#) ). اگر علاقه مند به توسعه پلاگین‌های Cordova باشید باز هم نیاز است از زبان بومی آن پلتفرم استفاده کنید.

نویسنده: خوزستان  
تاریخ: ۱۳۹۴/۰۱/۰۸ ۱۲:۴۰

آیا از لحاظ کاربردی و قابلیت و توسعه ، بین [زمرین](#) و cordova تفاوتی هست ؟ آیا میکروسافتی که قصد خرید [زمرین](#) رو داشت الان بعنوان یک رقیب cordova رو معرفی کرده یا اینکه کارکرد متفاوتی دارند ؟

نویسنده: محسن خان  
تاریخ: ۱۳۹۴/۰۱/۰۸ ۱۲:۴۸

cordova در اصل محصول میکروسافت نیست. در قسمت دوم این بحث [شروع کار با Apache Cordova در ویژوال استودیو #2](#) بیشتر در این مورد بحث شده.

نویسنده: غلامرضا ربال  
تاریخ: ۱۳۹۴/۰۱/۰۸ ۱۲:۵۴

[به نیاز شما بستگی دارد](#) .  
[این مقاله](#) هم مفید خواهد بود و مقایسه خوبی انجام داده است.

نویسنده: میثم ثوامری  
تاریخ: ۱۳۹۴/۰۱/۰۹ ۱۶:۵۸

با تشکر از مطلب آموزنده شما Xamarin و Cordova رو همیشه با هم مقایسه کرد  
Xamarin از زبان C# برای موبایل استفاده میکنه که البته زحمت کامپایل Mono بر عهده داره (Mono یک کامپایلر C# برای لینوکس و MonoToch برای IOS). و البته پولی  
Cordova با استفاده از Javascript و Html5 اینکارو انجا میده