

Zpravodaj moderní Programování 1/2016: Multiplatformní mobilní aplikace

Mobilní aplikace postupně nabývají na významu, zvětšuje se zájem o ně. Většinou se programují trochu jinak než klasické (desktopové) aplikace, především se ale programují úplně jinak navzájem. Co tím myslím? Roztříštěnost mobilních platforem.

V současné době existují minimálně tři hlavní mobilní platformy - Android, iOS a Windows. Plus řada menších, z nichž minimálně dřívější hit BlackBerry stojí za zmínku. No a každá platforma se programuje jinak. Android v Javě, Windows v C# nebo JavaScriptu, iOS v nějakém tom jejich jablečném céčku. Pokud chce programátor pokrýt více platforem, může se buď zbláznit, anebo najít nástroj pro multiplatformní vývoj. V tomto Zpravodaji budu řešit druhou variantu, přičemž se zaměřím na nástroje, [které jsou k dispozici ve Visual Studiu 2015](#).

Multiplatformním vývojem myslím to, že program napíšete jednou a potom ho s minimálními úpravami přeložíte pro více platforem. Že tedy nemusíte jednu věc programovat třikrát.

Ve Visual Studiu 2015 jsou k dispozici Xamarin a Apache Cordova. Obě možnosti nyní proberu.

Xamarin

Xamarin, produkt stejnojmenné firmy se sídlem v Kalifornii, je nástroj pro programování tří hlavních mobilních platforem v C#. Stránky xamarin.com.

Přesněji řečeno, je to rodina produktů, přičemž ty hlavní jsou Xamarin.Android, Xamarin.iOS a Xamarin.Forms.

Pomocí Xamarin.Android a Xamarin.iOS můžete v C# programovat **nativní** aplikace pro platformy v názvu produktu. Proč není žádný Xamarin.Windows? Protože aplikace pro Windows se v C# programují normálně, nepotřebujete nic navíc.

Všechny platformy můžete tedy pořešit pomocí C#, nicméně stále budete mít tři programy a musíte mít znalosti jednotlivých platforem - jaké jsou ovládací prvky na Androidu, jak se používají atd. Nejde tedy tak úplně o multiplatformní vývoj. Mezi projekty pro jednotlivé platformy můžete sdílet výkonný kód aplikace (též obchodní logiku), nikoli však uživatelské rozhraní.

Možnost sdílet uživatelské rozhraní přináší až Xamarin.Forms. Používáte jednotnou sadu ovládacích prvků, které se Xamarinem automaticky převedou na prvky dostupné na jednotlivých platformách. Zde jde již o plně multiplatformní vývoj.

Licence

Vypadá to krásně. Pouze ale do doby, než se na jejich stránkách podíváte na sekci „Pricing“. Podporu Xamarin.Forms a Visual Studia nabízí edice Business, která vás přijde na tisíce dolarů pro Android a dalších tisíce dolarů pro iOS! Což je tedy docela pálna a návratnost investice, pokud nesekáte mobilní aplikace jak Baťa cvičky, je problematická.

Plus háček navíc. Zmíněné platby jsou **roční**. To sice neznamená, že když se nerozhodnete pro další platbu, že vám to přestane fungovat. Znamená to ale, že přestanete mít přístup k jakýmkoli novinkám a aktualizacím.

Xamarin ve Visual Studiu

Možná jste si někteří všimli, že Xamarin si můžete nainstalovat jako součást Visual Studio Community 2015, které je zdarma. Takže jak to je? Potřebujete vůbec tu edici Business?

Součástí Visual Studio Community 2015 je **edice Starter**, která:

- omezuje velikost aplikace, ale především
- **nepodporuje Xamarin.Forms**.

Pokud si tedy třeba chcete v C# udělat nepříliš velkou aplikaci pro Android, nebo pro iOS, bezplatná edice Starter vám poslouží. Pokud potřebujete multiplatformní vývoj, pak je to se „startérem“ problematické.

Apache Cordova

Pojďme na druhou alternativu, Apache Cordova, stránky cordova.apache.org. To je nástroj pro multiplatformní mobilní vývoj v JavaScriptu, přičemž podporuje nejen tři hlavní platformy, ale i řadu menších. Podpora těchto dalších platform nicméně není integrována do Visual Studio, tam máte k dispozici Android, iOS a Windows.

Nástroj Apache Cordova vznikl ve firmě Nitobi (tuším také z Kali) pod názvem PhoneGap, který naznačoval, že se snaží o překlepnutí propasti (gap) mezi mobilními platformami. Firma Nitobi byla později koupena firmou Adobe (matka PDF), kteří ale o PhoneGap zřejmě neměli zájem, takže tento produkt byl pod názvem Cordova převeden pod Apache Software Foundation, což je organizace, která také zastřešuje známý webový server Apache. Cordova má snad být jméno ulice, kde sídlilo Nitobi, než je koupilo Adobe.

Apache Cordova je tedy nekomerční produkt vyvíjený programátorskou komunitou, jejíž součástí je také Microsoft.

Programování v Cordově

Mobilní aplikaci Apache Cordova vyvíjíte v podstatě stejně jako webovou stránku. Obsah zapisujete v HTML, formátování v CSS a s vytvořenou „webovou“ stránkou dynamicky manipulujete pomocí JavaScriptu.

Stejně jako na skutečných webových stránkách můžete i v Cordově používat JavaScriptové knihovny jako např. jQuery pro smazání rozdílů mezi prohlížeči či jQuery Mobile pro pěkně vypadající uživatelské rozhraní. Na rozdíl od skutečných webových stránek máte v Cordově navíc k dispozici hardware telefonu - kameru, akcelerometr, kontakty atd.

Ve fázi sestavení aplikace pro příslušnou platformu vytvoří Apache Cordova nativní kostru aplikace, která obsahuje prakticky jen prvek typu WebView (de facto webový prohlížeč, který lze integrovat jako součást aplikace, existuje na všech platformách), který vaše „webové“ stránky zobrazuje. Výjimkou je platforma

Windows, kde není WebView potřeba, neboť nativní Windows aplikace lze kromě C# vytvářet i v JavaScriptu.

O Cordova aplikacích se mluví jako o tzv. **hybridních**. Co to znamená? Výsledkem je normální mobilní aplikace,

- kterou **můžete publikovat v příslušném Storu**, ať už u Applu, Googlu či Microsoftu,
- která nepotřebuje být připojená k Internetu (samozřejmě, pokud z Internetu netaháte data) - **běží čistě na mobilním zařízení**.

Současně jde o aplikaci, kde kód není **zkompilován** do instrukcí procesoru zařízení, ale za běhu **interpretován** tím vestavěným prohlížečem. Jde tedy o mix nativní aplikace a webové stránky, proto název „hybridní“.

Výhody a nevýhody Cordovy

Výhody:

- Je zdarma, např. jako volitelná součást Visual Studio Community 2015.
- Vývoj v komunitě omezuje případné vrtochy jedné firmy. Navíc se v Cordově výrazně angažují i komerční firmy, takže **snad** to není záležitost poblouznění na jednu noc.
- Vývoj Cordova aplikace je téměř stejný jako vývoj dynamických webových stránek. Co se naučíte v Cordově, použijete na webu a naopak.

Nevýhody:

- Jako hlavní nevýhoda Cordovy se udává interpretovaná povaha JavaScriptu, což znamená nižší výkon. Domnívám se, že to se víceméně týká jen **starších** zařízení s Androidem, případně **počítačových her**. Osobně jsem při vývoji **podnikových aplikací** a jejich testování na běžných, nikoli hypermoderních zařízeních s Androidem a Windows žádný problém s výkonem nezaznamenal.

Plus dále JavaScript.

JavaScript

Samostatnou kategorií je JavaScript, který lze považovat za nevýhodu i výhodu.

Nevýhodou je, že to není C# :) Což znamená za prvé, že člověk musí zvládnout **úplně nový jazyk**, nové knihovny, nový způsob členění kódu atd. A za druhé je JavaScript **slabě typový jazyk**.

V C# vždy víte, jakého typu daná hodnota je, ať už jde o vlastnost, proměnnou, parametr či výsledek metody. To má dva příznivé důsledky:

- Výrazně se tím redukuje ošklivé, obtížně odhalitelné chyby vznikající tím, že např. do parametru přiřadíte něco úplně jiného.
- Funguje našeptávání, tudíž se vyvarujete překlepů.

Toto JavaScript z podstaty své koncepce nemá. Jistě, Microsoft se snaží ve Visual Studiu dělat zázraky a **jakési** našeptávání k dispozici má. Ale jen někdy, tak půl na půl?, jinak našeptávač mlčí, případně odhaduje a s vykřičníkem nabízí možnosti, které nejsou v dané situaci použitelné.

Na druhé straně zásadní výhodou JavaScriptu je, že je to v dnešní době **jediný jazyk, ve kterém můžete psát kód, který běží v prohlížečích**. Alternativy jako Flash či Silverlight se dále nevyvíjí, již jen dobíhají a mnohé nové prohlížeče je už ani nepodporují. To znamená, že investice do zvládnutí JavaScriptu není jednoúčelová pro Cordovu.

V JavaScriptu navíc existuje technologie **Node.js**, pomocí níž můžete dělat i případnou serverovou část webové či mobilní aplikace, pokud zatím nemáte oblíbenou jinou technologii jako např. ASP.NET či PHP. Investice do JavaScriptu tedy rozhodně není jednoúčelová.

Co potřebujete?

Prošli jsme si Xamarin i Cordovu. Ať už budete vyvíjet v jednom nebo ve druhém, potřebujete běžný počítač s Windows a nainstalované Visual Studio Community 2015. Tak můžete vyvíjet jak pro Windows, tak pro Android.

Váš program poběží i na iOS, ale na jeho **sestavení** potřebujete **počítač od Applu** (platí pro Xamarin i Cordovu)! Tento požadavek je vynucen velkou uzavřeností jejich platformy. Pro Cordovu snad existují nějaké weby, které vám sestavení pro iOS provedou bez toho, že sami musíte mít Apple, ale sám s tím zatím přímou zkušenost nemám.

Závěr

Visual Studio Community 2015 obsahuje nástroje pro multiplatformní vývoj mobilních aplikací. Na podzim 2014, kdy bylo na vývojářské konferenci Microsoftu oznámena úzká spolupráce se Xamarinem, jsem nejen já měl dojem, že Xamarin by zanedlouho snad mohl být k dispozici jako součást bezplatné či jiné edice Visual Studia.

Nemám kontakt na Billa, abych se ho zeptal, jak to vidí, takže jsem se musel rozhodnout sám. Původní dojem jsem ztratil a po pečlivém zvažování, s jehož výsledky jsem vás seznámil, jsem se rozhodl pro Cordovu. Vyvinul jsem v ní dvě aplikace podnikového typu a tato cesta vypadá schůdně.

Radek Vystavěl, 6. ledna 2016

Pokud Vám Zpravodaje moderníProgramování připadají užitečné, doporučte jejich odběr svým známým. Mohou se přihlásit na webu www.moderniProgramovani.cz.