

MORAVSKÁ VYSOKÁ ŠKOLA OLMOUC

Ústav informatiky a aplikované matematiky

Jan Sršeň

**Tvorba aplikací pro chytré mobilní telefony se systémem
Windows**

Development of Applications for Windows Based Smartphones

Bakalářská práce

Vedoucí práce: Mgr. Jiří Martinů

Olomouc 2017

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a uvedl v ní veškerou literaturu a ostatní zdroje, které jsem použil. Prohlašuji, že odevzdaná tištěná verze bakalářské práce se shoduje s elektronickou verzí vloženou do IS/STAG.

v Olomouci dne 29. 3. 2017

.....

Poděkování

Tímto bych chtěl poděkovat vedoucímu mé práce Mgr. Jirí Martinů za poskytnutí odborných a hodnotných připomínek při řešení této bakalářské práce.

Moravská vysoká škola Olomouc
Akademický rok: 2016/2017

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jan Sršeň**
Osobní číslo: **M14058**
Studijní program: **B6208 Ekonomika a management**
Studijní obor: **Podnikové informační systémy**
Název tématu: **Tvorba aplikací pro chytré mobilní telefony se systémem Windows**
Téma anglicky: **Development of Applications for Windows Based Smartphones**
Zadávající katedra: **Ústav informatiky a aplikované matematiky**

Z á s a d y p r o v y p r a c o v á n í :

Diskutujte způsoby vytváření a publikace aplikací pro chytré mobilní telefony s operačním systémem Windows. Práce může porovnat způsob programování pro běžné Windows a mobilní telefon, využívání oficiálního online distribučního kanálu Microsoftu, může také porovnat různé edice a verze mobilních operačních systémů od Microsoftu.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

LACKO, L'uboslav. Vývoj aplikací pro Windows 8.1 a Windows Phone. 1. vyd. Brno: Computer Press, 2014. ISBN 978-80-251-3822-9.

JANGLA, Kinnary. Windows 10 revealed: the Universal Windows Operating System for PC, Tablets and Windows Phone. New York NY: Apress 2015. ISBN 978-1-4842-0686-7.

SOFTWARE Falafel. Pro Windows Phone App Development. Berkeley CA:Apress, 2013. ISBN 978-1-4302-4783-8.

LEE, Henry and CHUVYROV, Eugene. Beginning Windows Phone App Development. Dordrecht:Springer 2012. ISBN 978-1-4302-4135-5.

SZOSTAK, Tomasz. Windows Phone 8 application development essentials: a practical guide to creating a Windows Phone 8 application using C#, XAML and MVVM. Birmingham: Packt Publishing 2013. ISBN 978-1-84969-676-0.

LALONDE, Lori and TOTZKE, David R. Windows Phone 8 recipes a problem-solution approach. 2013. ISBN 978-1-4302-5903-9.

MCKENNA, Sean and WHITECHAPEL, Andrew. Windows Phone 8 Development Internals. Microsoft.

TABOR Bo. Windows Phone 8.1 Development for Absolute Beginners. <http://www.LearnVisualStudio.net/> 2014.

LACKO, L'uboslav. Vývoj aplikací pre Windows Phone 7. Microsoft.

Vedoucí bakalářské práce:

Mgr. Jiří MARTINŮ


Ústav informatiky a aplikované matematiky

Datum zadání bakalářské práce: **10. prosince 2016**


Termín odevzdání bakalářské práce: **31. března 2017**

Podpis studenta: Datum: **12. 12. 2016**

Podpis vedoucího práce: Datum: **12. 12. 2016**


Mgr. Irena KOVAČIČINOVÁ
prorektorka

L.S.


PhDr. Mgr. Zdeňka KRIŠOVÁ, Ph.D.
manažer ústavu

V Olomouci dne 10. prosince 2016



Obsah

ÚVOD	6
1. MOBILNÍ OPERAČNÍ SYSTÉMY	7
1.1 POROVNÁNÍ MOBILNÍCH OPERAČNÍCH SYSTÉMŮ	7
1.2 HISTORIE MOBILNÍCH OPERAČNÍCH SYSTÉMŮ NA PLATFORMĚ WINDOWS	8
1.2.1 <i>Pocket PC 2000 (Rapier) a 2002 (Merlin)</i>	8
1.2.2 <i>Windows Mobile 2003 (Ozone) a 2003 SE</i>	8
1.2.3 <i>Windows Mobile 5 (Magnet)</i>	9
1.2.4 <i>Windows Mobile 6 (Cross Bow)</i>	9
1.2.5 <i>Windows Phone 7 a 7.5 (Mango, Refresh, Tango) a 7.8</i>	9
1.2.6 <i>Windows Phone 8 (Apollo)</i>	10
1.2.7 <i>Windows Phone 8.1 (Blue nebo Cyan)</i>	10
1.2.8 <i>Windows 10 Mobile</i>	11
1.3 HARDWAROVÁ VÝBAVA ZAŘÍZENÍ S MOBILNÍMI WINDOWS OS.....	11
1.4 HISTORICKÝ PŘEHLED MOŽNOSTÍ VÝVOJE APLIKACÍ PRO WINDOWS MOBILNÍ ZAŘÍZENÍ	13
1.4.1 <i>Windows Embedded Visual Tools</i>	13
1.4.2 <i>Microsoft Windows Visual Studio</i>	13
1.5 PŘECHOD NA WINRT	16
2. PRINCIPY VÝVOJE APLIKACÍ	17
2.1 ARCHITEKTURA	17
2.2 ŽIVOTNÍ CYKLUS APLIKACE.....	18
2.2.1 <i>Windows Phone 7.5 až 8</i>	18
2.2.2 <i>Windows Phone 8.1 a novější</i>	19
2.3 TECHNOLOGIE A VYBAVENÍ PRO VÝVOJ MOBILNÍCH APLIKACÍ WINDOWS.....	20
2.3.1 <i>Windows SDK</i>	20
2.3.2 <i>Visual Studio</i>	20
2.3.3 <i>Programovací jazyky</i>	20
2.3.4 <i>Windows Phone Emulator</i>	21
2.3.5 <i>Windows App Certification Kit (WACK)</i>	21
2.3.6 <i>Vývojářský účet</i>	21
2.4 DESIGN APLIKACÍ.....	22
2.4.1 <i>Filosofie ovládání</i>	22
2.4.2 <i>Základní doporučení a možnosti nastavení</i>	23
2.5 VÝVOJ APLIKACÍ PRO WINDOWS PHONE	24
2.5.1 <i>Návrhový vzor MVVM a Data Binding</i>	24
2.5.2 <i>Stavy aplikace a jejich zachování</i>	25
2.5.3 <i>Asynchronní programování</i>	27
2.5.4 <i>Serializace a deserializace dat</i>	27
2.6 WINDOWS STORE	28
3. NÁVRH A VÝVOJ APLIKACE IP NETWORK CALCULATOR	29
3.1 INSTALACE APLIKACE VISUAL STUDIO 2015.....	29
3.1.1 <i>Základní prvky aplikace Visual Studio a jeho nástrojů</i>	30
3.2 PODPORA VIRTUALIZACE, ZAPNUTÍ HYPER-V A VÝVOJÁŘSKÉHO REŽIMU	32

3.3	REGISTRACE WINDOWS STORE DEVELOPER ACCOUNT.....	32
3.4	HLAVNÍ KOMPONENTY A VÝVOJ APLIKACE IP NETWORK CALCULATOR.....	34
3.5	VYTVORENÍ WINDOWS STORE BALÍČKU A OVĚŘENÍ POMOCÍ WACK.....	36
3.6	NAHRÁNÍ APLIKACE NA WINDOWS STORE	36
	ZÁVĚR	40
	ANOTACE	41
	SEZNAM ZKRATEK A POJMŮ	42
	SEZNAM LITERATURY	44
	SEZNAM OBRÁZKŮ A TABULEK	47
	SEZNAM PŘÍLOH	48
	PŘÍLOHY.....	49

ÚVOD

Elektronická zařízení a mobilní technologie patří k současnému životu mnoha z nás a bez chytrého mobilního telefonu či počítače bychom si uměli život už jen těžko představit. Vše se automatizuje, vzdáleně řídí i kontroluje, a tím nám tak usnadňuje v mnoha případech naši práci. Těmito technologiemi jsem byl vždy nadšen a byl to i jeden z důvodů, proč jsem se rozhodl studovat Podnikové informační systémy na Moravské vysoké škole v Olomouci. Informačním technologiím se věnuji již od útlého dětství a v současné době programuji pro odlišné operační systémy v různých programovacích jazycích. Díky tomu jsem měl příležitost seznámit se s vývojem aplikací a jejich optimalizací pro mobilní zařízení, jako jsou mobilní telefony a tablety, které velkou mírou mění zaběhlé standardy vývoje aplikací za posledních 30 let, kdy byly dostupné pouze počítače s velkými monitory (v porovnání s dnešními mobilními technologiemi). S příchodem dotykových obrazovek, prvních chytrých mobilních telefonů a malých laptopů (PDA) se zásadně změnil pohled na vývoj aplikací, kdy nebyla používána jen myš pro obsluhu, ale i stylusy, prsty a funkce senzorů zařízení daného telefonu. Pro použití těchto technik se musela přizpůsobit pracovní plocha i její aplikace. To umožnilo natáčet obrazovku, ovládat aplikaci pomocí prstů a stylusů u menších rozměrů obrazovek telefonů. Tyto zařízení automaticky mění velikost písma a odsazení prvků pro pohodlné ovládání jednou rukou.

Pro svou bakalářskou práci jsem si proto vybral téma – Tvorba aplikací pro chytré mobilní telefony se systémem Windows, protože mě vývoj aplikací pro mobilní platformu, jakožto vývojáře softwaru na platformě Microsoftu vždy zajímal, avšak doposud jsem vyvíjel pouze aplikace desktopové, tedy aplikace pro grafické rozhraní u běžných i výrobních počítačů. Historie a vývoj aplikací pro platformu Windows Mobile či pro Windows Phone je zajímavým tématem, které přináší nové poznatky, jež rozšiřují mou stávající znalost operačních systémů, zvláště jejich chování, a současného i budoucího trendu vývoje aplikací.

Cílem mé práce je seznámit čtenáře s problematikou vývoje aplikací pro chytré mobilní telefony založené na technologii Windows, která je velice zajímavá, ač z několika důvodů ne příliš úspěšná oproti své konkurenci.

První kapitola se věnuje porovnání současných mobilních operačních systémů, historii mobilních operačních systémů společnosti Microsoft a možnostmi vývoje pro tuto mobilní platformu.

Druhá kapitola je zaměřena na principy a metodiku vývoje aplikací pro Windows Store na platformách Windows Phone 8.1 a Windows 10 Mobile. Budou představeny základní prvky designu a filosofie ovládání.

Část věnovaná vývoji aplikací nastíní praktickou formou, jakým způsobem lze pro zařízení s Windows Mobile či Phone vyvíjet aplikace a jak je distribuovat pomocí oficiálního distribučního kanálu od společnosti Microsoft. Vytvoříme aplikaci IP Network Calculator, která by měla mít praktický přínos pro její uživatele, jednoduchý design a intuitivní ovládání.

1. Mobilní operační systémy

Mobilní zařízení s operačním systémem, ať už se jedná o Smartphony, PDA či jiné, mají obvykle procesor založený na architektuře ARM. Pozdější verze podporují i architekturu procesorů x86.

1.1 Porovnání mobilních operačních systémů

Výrobci chytrých mobilních telefonů je nespočet, ovšem rozdílná je situace na poli operačních systémů pro tyto telefony, kde se trh zaměřuje především na tři základní platformy, kterými jsou Android, iOS a Windows Mobile či Windows Phone.

Operační systém	Společnost	Prodeje 2016	Prodeje 2013	Distribuce Aplikací
Android	Google, Open Handset Alliance	87 %	79 %	Google Play
iOS	Apple Inc.	12 %	14,2 %	Apple App Store
Microsoft	Microsoft Corporation	1 %	3,3 %	Windows Store

Tabulka 1 – Porovnání mobilních operačních systémů (Zdroj: vlastní)

➤ Android – 2.2 milionu aplikací v Google Play | Programovací jazyk Java

Nejpočetnější skupina uživatelů má velký výběr telefonů všech značek i cenových relací. Možnosti nastavení systému jsou široké, stejně tak i možnosti dokoupení kompatibilních zařízení a příslušenství. Rovněž snadné napojení na služby od společnosti Google. Mezi nejčastější nevýhody patří nemožnost upgradovat verzi operačního systému na vyšší (některé značky umožňují), dále některé aplikace nejdu odinstalovat a v poslední řadě jde o pomalost systému a aplikací na zařízeních nižší cenové kategorie.

➤ iOS – 2 miliony aplikací v Apple App Store | Programovací jazyk Objective-C

Druhá nejpočetnější skupina uživatelů, která má jednoho osvědčeného výrobce telefonů dodávajícího i operační systém s názvem iOS. Společnost Apple vsází na jednoduchost a plynulost svého systému, který je napojen na Apple služby. Aktualizace jsou samozřejmostí. Nevýhody jsou vyšší pořizovací náklady a dražší příslušenství spolu s uzavřeností systému.

➤ Microsoft Mobile či Microsoft Phone – 0.7 milionu aplikací ve Windows Store | Programovací jazyk C++, C#, VB.NET, XAML

Operační systém, jemuž se věnuje tato práce, je nejméně využívaným operačním systémem z výše jmenovaných. Microsoft Windows Mobile či Phone se vyznačuje svou jednoduchostí a úvodní obrazovkou, jež se odlišuje od dvou předchozích. Tato úvodní obrazovka je uspořádána formou dlaždic. Ovládání je svou jednoduchostí vhodné pro začínající uživatele či starší ročníky, které nejsou zvyklé používat chytré dotykové telefony. Platforma Microsoft je uzavřená stejně jako je tomu i u iOS od Applu. Telefon se dá jednoduše napojit na služby Microsoftu a je velice podobný operačnímu systému Windows pro osobní počítače. Mezi hlavní nevýhody patří málo aplikací ve službě Windows Phone Store, jež slouží jako oficiální distribuční kanál pro aplikace určené této platformě, a omezený výběr mezi telefony, kde je

hlavním výrobcem převážně společnost Nokia, která byla oficiálně odkoupena společností Microsoft v roce 2014 se svými modely Lumia.¹

1.2 Historie mobilních operačních systémů na platformě Windows

Společnost Microsoft rozděluje ze začátku své operační systémy (dále jen OS) pro mobilní zařízení do několika edicí, a to na Smartphony (malá zařízení-telefony), Pocket PC či PDA (kapesní osobní počítače s větší obrazovkou a Wifi; z některých bylo možné i telefonovat – Pocket PC Phone Edition) a Portable Media Center (sloužila pro přehrávání hudby, videí a prohlížení fotek). Toto rozdělení bylo na drobné obměny používáno z historických důvodů vývoje hardwaru těchto zařízení, avšak u novějších Windows Phone zařízení se již toto dělení nevyskytuje.

1.2.1 Pocket PC 2000 (Rapier) a 2002 (Merlin)

První verze operačního systému pro malá zařízení byla uvedena na trh roku 2000 a byla založena na jádře Windows CE v.3 (Windows Embedded Compact, které bylo vyvinuto pro malé počítače – PDA). Svým vzhledem připomínalo operační systémy Windows 98 či 2000. Nabízelo jedno základní rozlišení (QVGA 320×240), podporu paměťových karet a podporu dotykové obrazovky. Dostupné byly aplikace firmy Microsoft jako např. Excel, Word, Internet Explorer (dále jen IE) či Media Player.

Nástupce s názvem Pocket PC 2002 (Windows Mobile 2002), byla vylepšená verze svého předchůdce (jak OS, tak i všech aplikací), jež byla vydána roku 2002 a svým vzhledem byla podobná nově vydanému OS Windows XP. Tento OS byl prvním, který byl určen pro zařízení s označením Smartphone a disponoval telefonním modulem. Přinesl MSN Messenger, ale na druhé straně postrádal dotykovou obrazovku.

Pro Pocket PC 2000 a 2002 je vedle Win32 API (zkratka pro Application Programming Interface) dostupný i nový .NET Compact Framework v.1 (jedná se o odnož .NET Frameworku pro mobilní zařízení; dále jen .NET CF) dodatečně instalovaný do RAM paměti těchto zařízení.² Pro provoz .NET CF je vyžadováno minimálně Windows CE v.4, ale pro Pocket PC 2000 a 2002 byla udělena výjimka.³

1.2.2 Windows Mobile 2003 (Ozone) a 2003 SE

Tento operační systém uveřejnili roku 2003 a běžel na jádře Windows CE v.4.2. Jedná se o první systém s oficiálním označením Windows Mobile (dále jen WM). Byl vydán celkem ve 4 edicích, Windows Mobile 2003 for Pocket PC – Premium, Professional, Phone a Windows Mobile 2003 for Smartphone. V této verzi byla vylepšena podpora pro webové technologie, také byly přidány technologie Bluetooth (pro bezdrátový přenos souborů) a Wifi. Verze obsahovala .NET Compact Framework v.1 SP1 pro Smartphony, jenž měl plnou podporu pro vývoj aplikací pro .NET CF.

¹ Srov. HILL, S., *Which smartphone OS wins 2016? Android Nougat vs. iOS 10 vs. Windows 10 Mobile*, <<http://www.digitaltrends.com/mobile/best-smartphone-os/>>.

² Srov. ATLANTIC, *Encyclopedia of information technology*, s. 725.

³ Srov. YAO, P., *Choosing a Windows Embedded API: Win32 vs. the .NET Compact Framework: Windows CE .NET*, <<https://msdn.microsoft.com/en-us/library/ms836774.aspx>>.

Verze 2003 SE (Second Edition) byla vydána roku 2004 a přinesla možnost zálohovat zařízení pomocí služby ActiveSync (bylo dostupné i pro Pocket PC 2002), dále přinesla podporu pro otáčení displeje (Portrait – na výšku a Landscape – na šířku) a další nová rozlišení, včetně podpory VGA rozlišení (640×480). Obsahovala .NET Compact Frameworku v.1 SP2.⁴

1.2.3 Windows Mobile 5 (Magneto)

Windows Mobile 5 vyšel roku 2005, běžel na Windows CE v.5 a byl prvním, v němž se použilo .NET Compact Frameworku v.1 SP3 a nové verze balíku Office, která byla pojmenována Microsoft Office Mobile. Přibyla lepší podpora pro Microsoft Exchange Server (služba pro emaily a zprávy), nová technologie GPS a vylepšená správa paměti RAM, díky níž se prodloužila výdrž baterie, a zlepšený proces ukládání.

1.2.4 Windows Mobile 6 (Cross Bow)

Windows Mobile ve verzi 6 byl uveden na trh roku 2007 a běžel na Windows CE v.5.2. Design se přibližoval právě vydanému desktopovému OS Windows Vista. Nabízen byl ve 3 edicích a to Classic, Standard a Professional. Nově přinesl službu Windows Life, vylepšenou podporu pro webové technologie a možnost VoIP (volání přes internet). WM6 měly předinstalován .NET Compact Framework v2 přímo v OS (operační systém).

V roce 2008 se objevila verze 6.1, která přinesla nový grafický design, vylepšené prostředí pro ovládání prsty a opravy chyb. Následujícího roku 2009 vyšly verze 6.5x, ty přinesly pouze aktualizované aplikace, opravy chyb, podporu nového HW (hardware) a úpravu prostředí pro lepší uživatelský komfort, ale také službu Windows Marketplace, jakožto oficiální zdroj pro stahování aplikací od MS (Microsoft).

S verzí Windows Mobile 6.5.5. skončilo označení Windows Mobile pro mobilní operační systém od firmy Microsoft a začalo se používat označení Windows Phone (dále jen WP).

1.2.5 Windows Phone 7 a 7.5 (Mango, Refresh, Tango) a 7.8

Windows Phone 7 byl prvním mobilním OS z rodiny nazývané „Phone“. Společnost Windows jej vydala roku 2010. Oproti předchozím verzím se ve velké míře změnilo grafické rozhraní, které dostalo název Modern UI (Metro UI) a bylo založeno na interaktivních dlaždicích a přizpůsobeno pro snadné ovládání prsty. Jádro systému běželo na Windows CE v.7 a nebylo zpětně kompatibilní s předchozími produkty. Jako odpověď na konkurenční distribuční kanály Apple App Store a Google Play přidal Microsoft svou verzi distribučního kanálu nazvanou Marketplace, představeného roku 2004. Kanál pro mobilní platformu byl uveden na trh roku 2009 s názvem Windows Marketplace for Mobile. Tato verze přinesla možnost napojení na sociální sítě, na nové technologie společnosti Microsoft i instalaci aplikací třetích stran. Tento OS měl dva zásadní nedostatky, prvním z nich byla nedostatečná podpora SD karet (nešly měnit SD karty nebo podpora zcela chyběla) a za další nedostatek považujeme chybějící českou lokalizaci, jež byla dostupná až od verze 7.5.

Windows Phone 7.5 byl představen roku 2011, přinesl s sebou podporu multitaskingu a novou verzi Internet Explorer 9, který podporoval stejné standardy jako jeho desktopová verze. Následoval update s názvem Refresh, který přinesl podporu LTE sítí a v roce 2012

⁴ Srov. LEE, W., *.NET Compact Framework pocket guide*, s. 1-7.

poslední update s názvem Tango, který snížil nároky systému na HW a přinesl rozšíření v podobě 23 nových lokalizací. Tento OS identifikuje sebe sama jako verzi 7.1 (verze SDK), avšak na trh byl uveden pod označením 7.5.

Aktualizace Windows Phone 7.8 vyšla roku 2013 (po oficiálním vydání WP 8) a přinesla nové možnosti grafického prostředí Metro, jako jsou různé velikosti i stavy interaktivních dlaždic a změny uzamykací obrazovky.

1.2.6 Windows Phone 8 (Apollo)

Windows Phone 8 je druhou generací řady Phone. Byl vydán roku 2012 a oproti svým předchůdcům, kteří běželi na jádrech řady Windows CE, byl WP 8 postaven na jádře Windows NT 6.2, stejně jako je tomu u desktopové verze OS Windows (dále jen OS W či W) 8. Nové jádro umožňuje WP 8 běh aplikací pro vícejádrové procesory, či rozlišení HD (1280×720), WXGA (1280×768), a dokonce i FullHD (Update 3). Díky novému jádru nejsou aplikace pro WP 8 zpětně kompatibilní s předešlými verzemi WP a starší verze WP nelze upgradovat na WP 8, přesto nabízí WP 8 zpětnou kompatibilitu pro aplikace z WP 7 (některé aplikace z WP 7 lze spouštět na WP 8). Pro WP 8 vyšlo několik Updateů, jež převážně opravovaly chyby, aktualizovaly aplikace a přidaly některá vylepšení.

S vydáním Windows 8 a Windows Phone 8 přišlo také nové prostředí pro programování aplikací s názvem Windows Runtime (WinRT, pro mobilní zařízení Windows Phone Runtime, dále jen WPRT)⁵. Založili jej na novém grafickém rozhraní Metro, to ale nebylo plně podporováno a převážná část aplikací byla založena na platformě Silverlight, jež se lišila od použité platformy v následujících verzích (hlavně verze XAML pro tvorbu UI rozhraní). Vývoj pomocí C++ a DirectX pro hry byl dostupný pomocí WPRT, avšak XAML Framework stále fungoval na technologii Silverlight. Pro toto prostředí byl stejného roku otevřen i nový distribuční kanál s názvem Windows Store. Windows store nahradil původní Windows Marketplace a byl určen primárně pro distribuci aplikací napsaných pomocí WinRT (Metro UI – design a principy). To ovšem neznamenalo, že by se nedaly dále vyvíjet aplikace pomocí Win32 API, kterým i ve standardním desktopovém OS W 8 nadále disponovaly Distribucí a možnostmi vývoje aplikací pro Windows Store se budou věnovat v dalších kapitolách podrobněji.⁶

1.2.7 Windows Phone 8.1 (Blue nebo Cryan)

WP 8.1 byl vydán roku 2014, jako třetí a zároveň poslední zástupce rodiny produktů Phone. Měl plnou podporu pro vývoj aplikací pomocí WinRT, ale zpětnou kompatibilitu pouze s WP 8. Některé telefony s tímto OS bylo možné upgradovat později na novější systém Windows 10 Mobile (dále jen W10M). Mezi hlavní novinky tohoto systému patřila hlasová inteligentní asistentka Cortana, vylepšení grafického designu, IE 11 a přidání dalších uživatelských funkcí, jako například notifikační centrum, lepší vyhledávání a mnohé další.⁷

⁵ Srov. MAYBERRY, M., *WinRT revealed*, s. 1-9.

⁶ Srov. SUBRAMANIAN, S., *Evolution of Windows Phone OS and the transition from Windows Mobile*, <<https://techglimpse.com/history-microsoft-windows-phone-os-wp/>>.

⁷ Srov. RUBINO, D., *Windows Phone 8.1 Review*, <<http://www.windowscentral.com/windows-phone-81-review>>.

Díky WinRT je možné vyvíjet aplikace pro Windows Store jak pro WP 8.1, tak i pro Windows 8 díky technologii Universal Apps, jež umožňuje sdílet většinu společného kódu pro obě platformy. Dřívější aplikace, které byly vyvinuty primárně pro WP 7.x nebo WP 8 na nové verzi s WP 8.1 běží bez větších problémů, ovšem aplikace vyvinuté pro WP 8.1 na starších WP nejsou podporovány.⁸

1.2.8 Windows 10 Mobile

Windows 10 Mobile je nástupcem WP 8.1 a i přesto, že nese označení Mobile, nejedná se o pokračování série Windows Mobile, ale pouze o označení, že je tato verze určena pro mobilní telefony a tablety (založené na architektuře ARM). Zveřejnili je roku 2016 a jedná se o nejnovější a zatím poslední mobilní operační systém od společnosti Microsoft, který vychází z Windows 10, avšak je upraven pro použití na mobilních zařízeních. Proto zde nejsou některé funkce z desktopových Windows 10 k dispozici, jelikož by na mobilních zařízeních postrádaly smysl. Windows 10 pro mobilní zařízení byl vydán ve dvou verzích, a to Mobile a Mobile Enterprise. Ve W10 i W10M byl přidán nový prohlížeč Edge (Project Spartan), jenž je nástupcem IE 11, a vylepšená verze inteligentní asistentky s názvem Cortana.⁹

Rodina systémů Windows 10 (jak pro osobní, tak i pro mobilní zařízení) používá novinku s názvem **Universal Windows Platform** (dále jen UWP) API, jedná se tedy o nové univerzální rozhraní pro programování aplikací, které je nástupcem či nadstavbou WinRT API (UWP je stejná technologie jako WinRT, ale o dost vylepšená). Umožňuje vytvářet univerzální aplikace napříč všemi platformami, jako jsou tablety, osobní počítače, či Xbox a další produkty z rodiny firmy Microsoft.¹⁰

1.3 Hardwarová výbava zařízení s mobilními Windows OS

První zařízení s OS Windows Pocket PC 2000–2002 bylo vybaveno procesorem běžícím na frekvenci 206 MHz s 32 (i 64) MB RAM s velikostí úložného prostoru 16 MB, dotykovým TFT displejem o velikosti 3.8 palce s rozlišením 320×240px, které bylo možné ovládat pomocí stylusu, hardwarových tlačítek i prstů. Například Compaq iPAQ H3600 (PDA), jenž vydržel okolo 14 hodin a byl stejně velký jako kalkulačka. Pro bezdrátovou komunikaci používal infračervený port a pro možnost telefonování bylo zapotřebí dokoupit speciální modul, jenž se instaloval na zadní stranu zařízení. Ve verzi H3870 byl dostupný BlueTooth, který fungoval Windows Pocket PC 2002. Na těchto zařízeních bylo možné přehrávání skladeb v MP3 formátu, stahovat a posílat e-maily, psát poznámky. Softwarová výbava též obsahovala kancelářské balíky Pocket Word, Excel a internetový prohlížeč Internet Explorer.

Zařízení s Windows Mobile 2003 a 2003 SE přinesly podporu GSM, Wifi, Bluetooth, SD karet a fotoaparátů pro pořizování fotek i videí (0.3-1.3 Mpx). Obrazovky měly podporu 65k barev (oproti 4k barvám u starších zařízení), např. HP iPAQ h6315, BenQ P50. Zařízení

⁸ Srov. LACKO, E., *Vývoj aplikací pro Windows 8.1 a Windows Phone*, s. 29-32, s. 274.

⁹ Srov. JANGLA, K., *Windows 10 revealed: the universal Windows operating system for PC, tablets, and Windows phone*, s. 1, s. 43-71.

¹⁰ Srov. WIGLEY, A., a NIXON, J., *Windows 10 - An Introduction to Building Windows Apps for Windows 10 Devices*, <<https://msdn.microsoft.com/en-us/magazine/dn973012.aspx>>.

Fujitsu Siemens Pocket LOOX 720 podporovalo VGA rozlišení (s natáčením obrazovky), procesor běžel na 520 MHz a měl dostupných 128 MB RAM paměti.

Windows Mobile 5 zařízení bylo obohaceno o podporu GPS. Zařízení s Windows Mobile 6.x bylo hardwarově vybaveno trochu lépe než její předchůdci (např. fotoaparát o rozlišení 3.2-8 Mpx) lepší výdrž baterie, displej 3.6 palců (WVGA 800×480px), procesor 1 GHz s 560 MB RAM. Též byla dostupná podpora akcelerometrů i dalších senzorů.

Zařízení, pro Windows Phone 7.x, vyžaduje procesor minimálně s 800MHz a 256 MB RAM, plynoucí z nároků kladených novým grafickým designem a možnostmi, které tento OS přináší. Toto období sebou přináší nový trend velkých displejů a výkonných telefonů. Windows Phone 8.x požaduje 4 GB místa a 512 MB RAM pro displeje WVGA a 1 GB RAM pro displeje vyššího rozlišení. Zařízení s Windows 10 Mobile jsou převážně telefony Microsoft Lumia, minimální požadavky jsou oproti předchozí verzi WP 8.1 navýšeny na 1 GB RAM a 8 GB místa na disku. Zařízení s Windows Phone mají také podporu pro DirectX a OpenGL pro podporu náročných 3D her a vedle výpočetního procesoru (převážně procesory Qualcomm Snapdragon), mají též procesor grafický.



Obrázek 1 – Historie UI u Windows mobilních zařízení (Zdroj: vlastní)

1.4 Historický přehled možností vývoje aplikací pro Windows mobilní zařízení

V této podkapitole si představíme možnosti vývoje pro mobilní zařízení založené na platformě Windows CE (od Pocket PC 2000 až WP 7.8), ta je 32 bitová a je založená na knihovně Win32, později se přidal i .NET Compact Framework.

Těž tu probereme i vývoj pro mobilní aplikace založené na Windows NT (od WP 8).

1.4.1 Windows Embedded Visual Tools

Windows Embedded Visual Tools 3.0 je sada nástrojů pro vývoj mobilních aplikací založených na Windows CE (Pocket PC 2000 a 2002). Tento balíček obsahuje vše pro vývoj, jako jsou překladače, nástroje pro ladění, dokumentace, grafické vývojové prostředí (dále jen IDE). SDK pro Pocket PC je již obsažen, ale další SDK je nutno doinstalovat. Obsahuje také grafické IDE pro eMbedded Visual Basic (eVB) a eMbedded Visual C++ (eVC).¹¹

Pro vývoj WM 2003 je potřeba eVC ve verzi 4 (pro vývoj aplikací založených na Win32, ALT nebo MFC) nebo Visual Studio .NET 2003 (pro vývoj v .NET Compact Framework).¹²

1.4.2 Microsoft Windows Visual Studio

Produkt Visual Studio (dále jen VS) od společnosti Microsoft nabízí kompletní IDE prostředí pro vývoj aplikací na operačním systému Windows a nově, i s omezenou funkcí, na OS Linux (Visual Studio Code) a OS macOS (OS X) od společnosti Apple Inc. (Visual Studio for Mac), které zatím nepodporuje vývoj aplikací pro Windows mobilní platformu. Pro OS Linux existuje projekt Mono (sada nástrojů pro .NET běžící pod Linux), v němž zatím též nelze vyvíjet aplikace pro Windows mobilní platformu. Jediným způsobem, jak lze pod OS X nebo Linuxem vyvíjet nové aplikace pro mobilní platformu Windows, je prozatím použití virtuálního stroje (WMWare, VirtualBox či Parallels) s Windows OS a nainstalovaným VS ve verzi, kterou potřebujete.

VS umožňuje nezávislé instalace svých různých verzí na jednom OS. V praxi to znamená, že je možné mít na jednom OS nainstalováno několik verzí produktu VS. Microsoft rozděluje VS do několika edic, dle ceny nebo nabízených služeb a nástrojů. Názvy edic se s postupem času mění, ale základní edice zůstávají pojmenovány stejně. Současné verze jsou Express (zdarma, neumožňuje instalovat pluginy, je ochuzeno o některé funkce, oproti verzi Professional, a zaměřeno pouze na určitou platformu - např. Web nebo Phone), Community (zdarma pro jednotlivce, pro organizace do 250 počítačů a obrátu do 1 milionu \$), a nakonec jsou zde placené Professional a Enterprise.

V roce 2003 bylo vydáno Microsoft Visual Studio .NET 2003, které jakožto první umožňovalo vývoj pro mobilní zařízení a nahradilo tak Windows Embedded Visual Tools, jehož komponenty do sebe zaintegrovalo, avšak také přidalo i další vylepšení a nástroje do jedné aplikace, která umožňovala vývoj pro různé platformy. VS nabízí ucelené prostředí pro vývoj aplikací v tzv. **native code** (C / C++ – kompiluje se přímo pro daný stroj/platformu)

¹¹ Srov. ATLANTIC, *Encyclopedia of information technology*, s. 721-725.

¹² Srov. LACKO, E., *Vývoj aplikací pre mobilné zariadenia*, s. 8-24.

nebo **managed code** (C# / Visual Basic .NET – kód se nejprve přeloží do **CIL** (programovací mezijazyk, dříve MSIL) a poté se spouští pomocí **CLR** (běhové prostředí), jež se stará o chod spuštěné aplikace, správu paměti atd.).¹³ Díky .NET Frameworku v.1.1 bylo možné programovat v programovacích jazycích Visual Basic a C#, spolu s daným SDK pro zařízení, pro které chcete vyvíjet své aplikace.¹⁴

Verze VS 2008 byla poslední, která umožňovala vývoj pro platformu od Pocket PC 2003 až po Windows Mobile 6 při nainstalovaném Windows Mobile 6 SDK. Bylo možné zde doinstalovat podporu pro vývoj WM 6.5 aplikací pomocí Windows Mobile 6.5 Developer Tool Kit. Tato verze umožňovala vývoj pro .NET CF ve verzích 2 a 3.5.¹⁵

V následující verzi VS 2010 je možné vyvíjet mobilní aplikace pro Windows Phone 7, platformy Windows Mobile a starší již nejsou podporovány. Pro vývoj mobilních aplikací byl zdarma vydán VS 2010 Express for Windows Phone, který nabízí kompletní vývojové prostředí pro vývoj WP 7 aplikací i s nástroji Windows Phone Emulator (dále jen WPE) pro emulaci prostředí WP zařízení a Expression Blend pro návrh grafického prostředí aplikace. Samotný vývoj probíhá v Silverlight for Windows Phone (C#) nebo XNA Game Studio 4, které slouží pro snadný vývoj her nejen pro WP. Expression Blend bylo poprvé ve VS 2008, ukládá do XAML formátu a také pro svůj chod vyžaduje 64 bitové Windows a technologii Hyper-V (další požadavky budou podrobněji popsány v následujících kapitolách).¹⁶

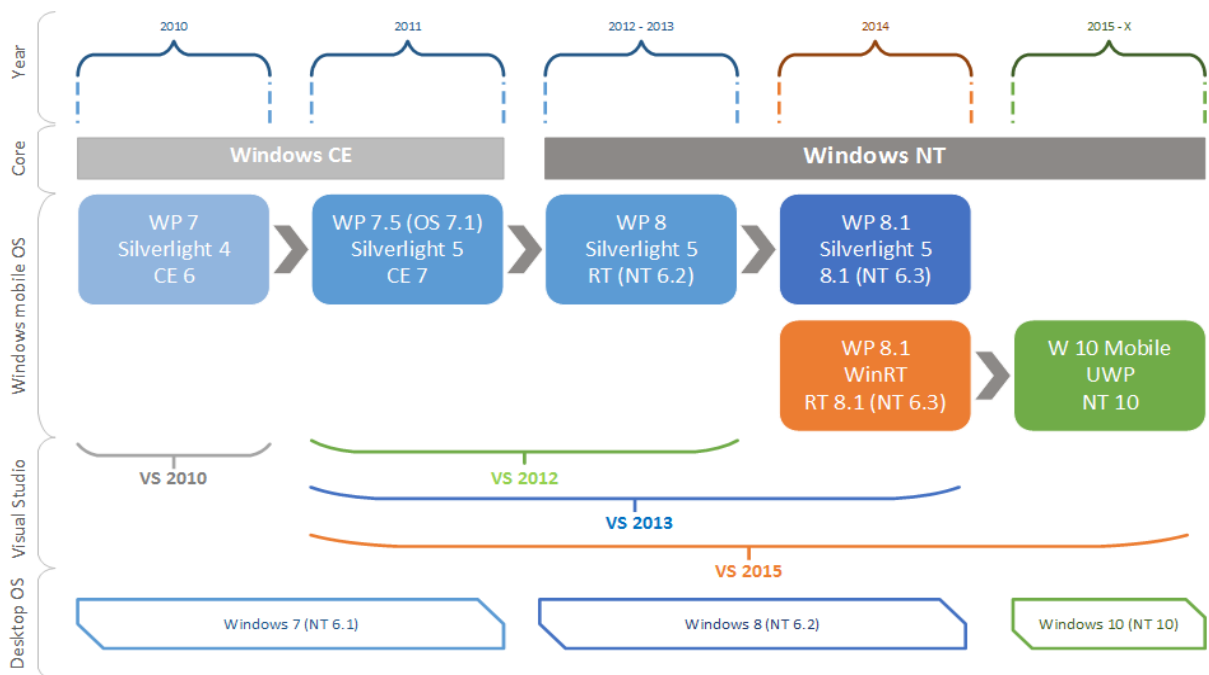
VS 2012 přineslo NuGet balíčkovací systém a pro vývoj mobilních aplikací byl zdarma vydán, stejně jako tomu bylo u VS2010, VS 2012 Express for Windows Phone. Přináší podporu WinRT a umožňuje vývoj pro WP 8 (aplikace funguje jen na WP 8) a 7.5 (aplikace funguje na WP 7.5 i WP 8) při nainstalovaném balíčku Windows Phone SDK 8.0. Toto Visual Studio vyžaduje OS Windows 8 Pro 64 bit.

¹³ Srov. GORDON, A., *The .NET and COM interoperability handbook*, s. 14-20.

¹⁴ Srov. LEE, W., *.NET Compact Framework pocket guide*, s. 11-13.

¹⁵ Srov. STRUYS, M., *Windows Mobile App Development: Part 1: Creating your first application*, <<https://www.codeproject.com/Articles/43357/Windows-Mobile-App-Development-Part-Creating-you>>.

¹⁶ Srov. WHITECHAPEL, A., a MCKENNA, S., *Windows Phone 8 development internals*, s. 6-18.



Obrázek 2 – Přehled Windows mobilní platformy od WP7 (Zdroj: vlastní)

VS 2013 umožňuje vývoj pro WP 8.1 i WP 8, je zde dostupný .NET Framework v4.5.1 a je zde nově pro vývoj mobilních aplikací na výběr mezi WinRT (Windows Store) a Silverlight aplikací (byla používána pro vývoj WP 7.x a WP 8 aplikací). VS 2013 Update 2 v sobě již obsahuje Windows Phone 8.1 SDK. Máme na výběr z tří druhů projektů pro Windows Store: Universal Apps (pro Windows 8.1 i WP 8.1), Windows Apps (W 8.1) a Windows Phone Apps (Windows Phone a Windows Phone Silverlight).

Projekt Universal Apps obsahuje 3 složky: Shared pro sdílení společného kódu, Windows Phone 8.1 (pro WP) a Windows 8.1 (desktop). Zpětnou kompatibilitu pro vývoj aplikací nabízí jen projekt Windows Phone Silverlight, kde se při jeho vytváření vybere platforma pro WP 8 nebo WP 8.1. Nově jsou také k dispozici šablony pro Windows Phone a to Blank, Hub, Pivot, WebView a HTML5. Verze Silverlight pro vývoj WP 8.1 není zpětně kompatibilní s WP 8 ani WP 7.5(7.1). A to z toho důvodu, že WP 8 využívá pro tvorbu UI starší verzi XAML Silverlight, která je jiná než u WP 8.1. Verze Silverlight pro WP 8 funguje i na WP 7.5. Projekty Windows Store (W8.1 a WP 8.1) nelze otevřít v starších verzích VS.¹⁷

VS 2015 přichází s novou technologií nazvanou Universal Windows Platform (UWP), jež umožňuje použití adaptabilního zobrazení a jednoho kódu napříč všemi platformami Windows. Pro vývoj i spuštění UWP aplikací je zapotřebí OS Windows 10 a Windows 10 SDK. Tuto verzi VS lze spustit i na starších verzích OS Windows (7 až 8.1), ale některé nové funkcionality pro vývoj UWP aplikací nebudou dostupné (aplikaci nelze spustit, pouze vzdáleně na Windows 10), všechny ostatní technologie poběží bez problémů. VS 2015 podporuje vývoj pro WP 8.1 stejně jako VS 2013, je zde též i Windows 8 – Windows Phone Silverlight pro WP 8 (běží i na WP 7.5) a WP 8.1.¹⁸

¹⁷ Srov. KUMAR, S., NAGARAJ, Lohith G., RAWAL, P., a ROHILLA, P., *Windows 10 development recipes: a problem-solution approach in HTML and JavaScript*, s. 1-14.

¹⁸ Srov. MOEMEKA, Edward, a MOEMEKA, Elizabeth, *Real world Windows 10 development*, s. 1-19.

1.5 Přechod na WinRT

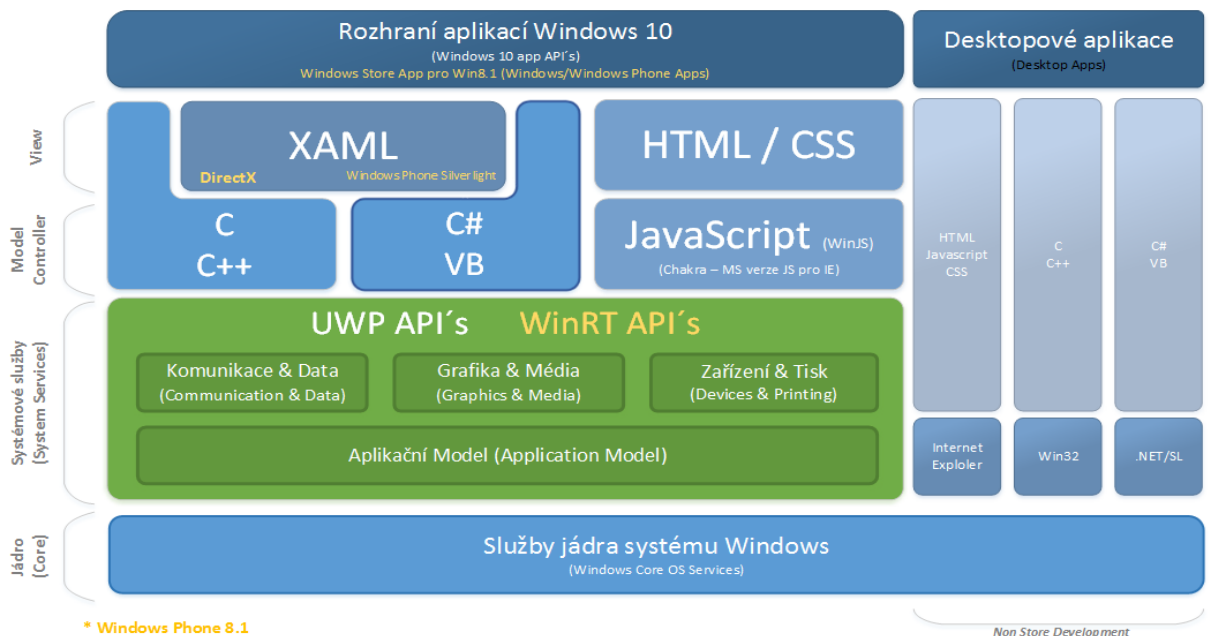
S příchodem WinRT se změnil zásadním způsobem i styl vývoje a filosofie ovládání aplikací pro Windows, kdy společnost MS udělala zásadní krok v rámci podpory zařízení s dotykovou obrazovkou. Zžití ovládání z desktopových OS, kde uživatel používal myš a klávesnici, se s příchodem Windows 8 přizpůsobilo pro použití i na tabletech a chytrých telefonech za použití re-designu pomocí tzv. dynamických dlaždic. Dlaždice usnadňovaly ovládání zařízení pomocí prstů, které jsou svou plochou méně přesné než myš, a nahradily tak klasickou nabídku aplikací, kterou známe například z Windows 7. S příchodem mobilních zařízení se změnil i styl chování aplikací, které jsou více intuitivní, přizpůsobivé, s novým rozhraním a velkým důrazem kladeným na rychlost a plynulost systému, jelikož přenosná zařízení nejsou tak výkonná jako pracovní PC a očekává se od nich dlouhá výdrž. Proto je potřeba přizpůsobit i aplikace a jejich způsob běhu. O této problematice dále pojednáváme v následující kapitole.

2. Principy vývoje aplikací

V této kapitole se budeme podrobněji zabývat principy designu a vývoje aplikací pro Windows Phone od verze 8.1 až po nejnovější Windows 10 Mobile. Pro vývoj budeme používat jen Visual Studio 2015 ve verzi Community běžící na operačním systému Windows 10, ve kterém lze vyvíjet mobilní aplikace pro Windows 10 UWP i WP 7.5 až 8.1. U první kapitoly (2.1) se zaměříme na životní cyklus aplikací od 7.5 až k Windows 10. Jako programovací jazyk jsme zvolili C# a XAML pro tvorbu UI rozhraní, pro tuto kombinaci budeme demonstrovat všechny ukázky částí kódů, názvů tříd i metod.

2.1 Architektura

Vývoj mobilních aplikací se odvíjí od architektury systému, kterou si popíšeme v této kapitole. Vývoj pro Windows Phone 8.1 probíhal trochu odlišným způsobem než pro současné Windows 10 mobile, v nichž vývoj univerzálních aplikací dospěl již ke zdárnému konci. OS Windows 8.1 využíval pro své aplikace WinRT API, které je asynchronní (náročné operace se vykonávají na pozadí, zatímco aplikace běží a nemusí na tyto procesy čekat a blokovat tím běh aplikace). S WP 8.1 přišly nové typy projektů s názvem Universal Windows 8.1, které umožňovaly sdílet až 90 % společného kódu pro Windows 8.1 a Windows Phone 8.1 aplikace ve společném adresáři nazvaném Shared („Sdílený“), zbytek aplikace, převážně UI rozdíly a ovládání, byly řešeny samostatně pro daný typ zařízení (W8.1 nebo WP 8.1).



Obrázek 3 – Architektura OS Windows 8.1 a 10¹⁹

¹⁹ Srov. MOEMEKA, Edward, a MOEMEKA, Elizabeth, *Real world Windows 10 development*, s. 16.

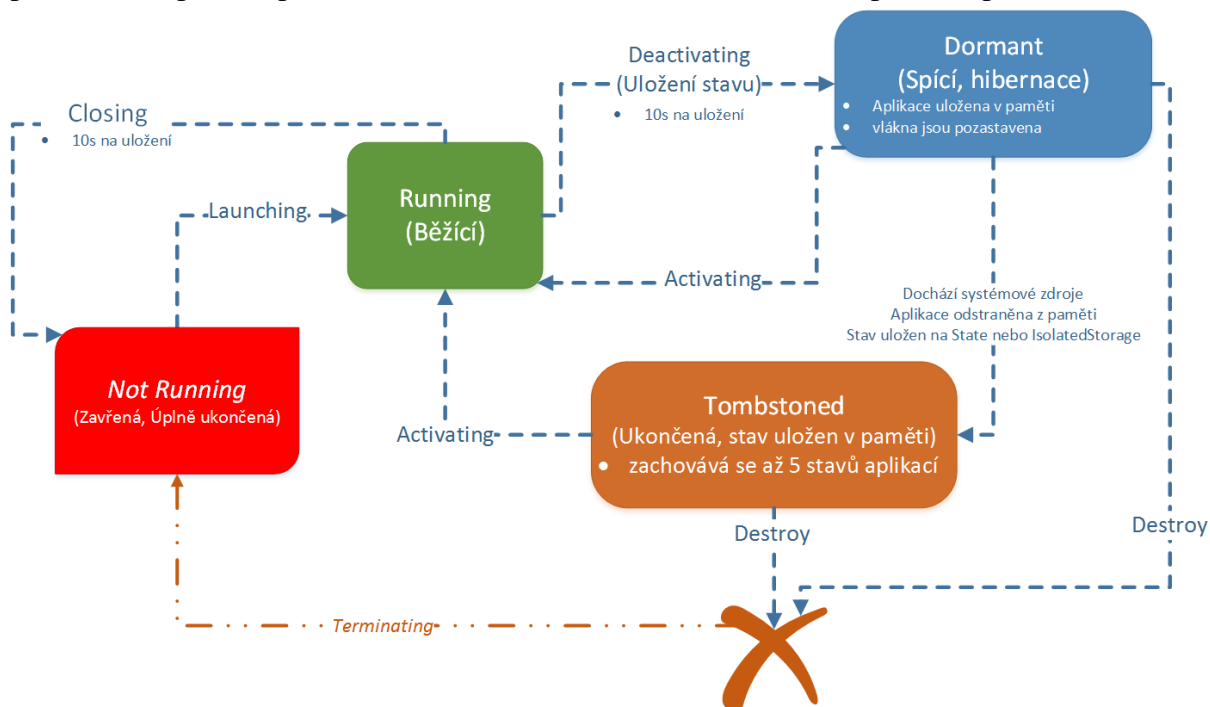
S příchodem Windows 10 se změnil i přístup k vývoji aplikací. WinRT byl nahrazen svým rozšířením a vylepšením, nazvaným UWP, které přináší myšlenku programovat univerzální aplikace napříč všemi platformami Windows a které má adaptabilní přizpůsobování prvků pro velikost obrazovky a typu zařízení s tím, že je možné sdílet 100 % kódu, jelikož není projekt rozdělen na 3 části, jako tomu bylo při vývoji u 8.1 platformy, ale máme pouze jeden projekt a programujeme jedno UI a aplikační logiku společnou pro všechny typy zařízení, o zbytek se nám postará UWP API.

2.2 Životní cyklus aplikace

Životním cyklem aplikace rozumíme stádia, jimiž si každá aplikace prochází během svého spuštění, a stavy, jež mohou nastat během jejího používání. U mobilních zařízení je způsob řízení životního cyklu jiný než u klasických desktopových aplikací. Aplikace se nemohou překrývat, ale pouze mezi sebou sdílet část obrazovky.

2.2.1 Windows Phone 7.5 až 8

V životním cyklu pro WP 7.5 přibyl oproti předcházející verzi nový stav Dormant. Všechny stavy, v nichž se může nacházet aplikace pro zařízení s WP 7.5, až WP 8 jsou znázorněny na obrázku 4. Prvním krokem je spuštění aplikace (launching), kdy uživatel vybere aplikaci, kterou chce spustit. Aplikace má 5 s na zobrazení úvodní obrazovky a 20 s na to, aby byla plně provozu schopná. Pokud nesplní tyto požadavky, není schopná projít certifikačním procesem pro aplikace ve Windows Store (Marketplace pro WP 7.x).



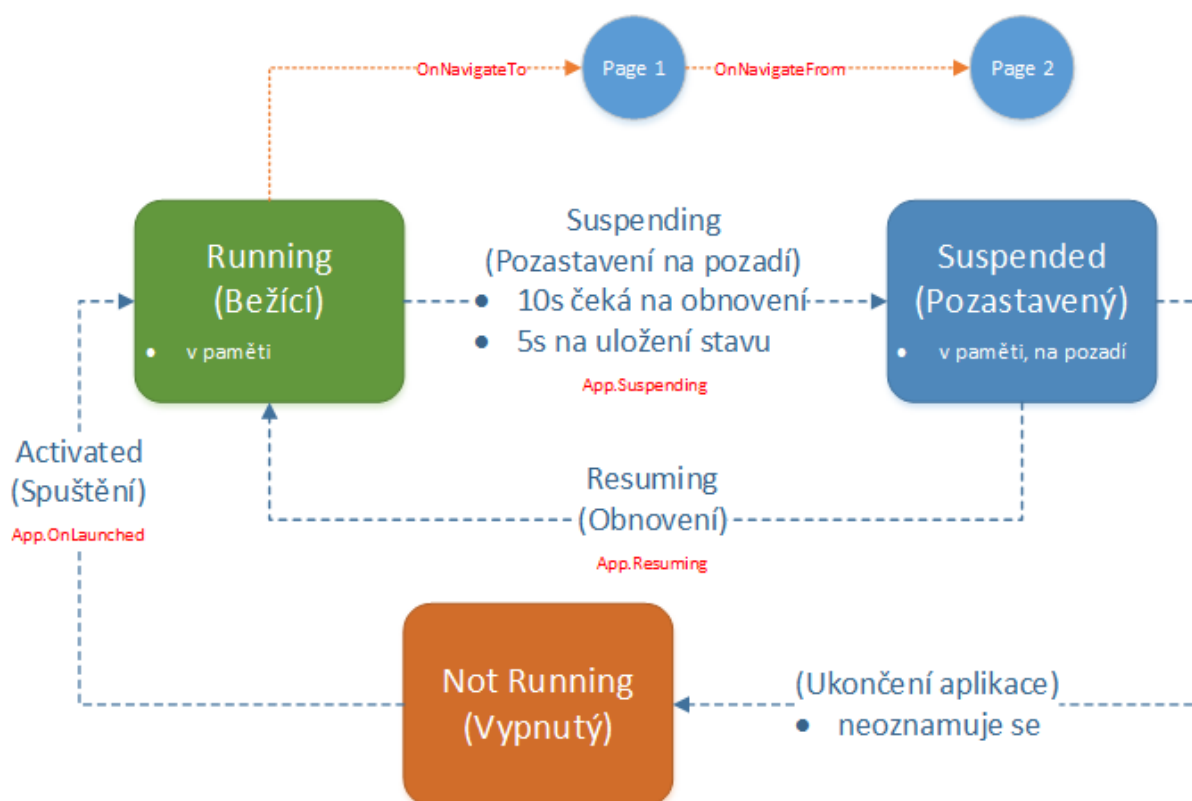
Obrázek 4 – Životní cyklus aplikace od Windows Phone 7.5 až 8²⁰

²⁰ Srov. LACKO, E., *Vývoj aplikací pro Windows Phone 7*, s. 6.

Při kliknutí na tlačítko start nebo přepnutí na jinou aplikaci se stav naší aplikace musí do 10 s uložit a přechází do stavu Dormant, v němž je aplikace stále v paměti, avšak její vlákna jsou pozastavena. Pokud si to OS vyžádá pro nedostatek paměti, aplikace může být ukončena (stav Tombstoned). Stav Tombstoned aplikaci odebere z paměti zařízení, ale zároveň je do ní uložen speciální objekt. Tento objekt uchovává informace o posledním stavu aplikace před jejím odebráním. Díky tomu může být aplikace na vyžádání znovu obnovena. Systém si takto dokáže zapamatovat pouze pět posledních objektů, jeden ke každé ukončené aplikaci, jež byla ve stavu Tombstoned odebrána z paměti. Celkového uzavření aplikace docílíme kliknutím na tlačítko „zpět“ na úvodní stránce aplikace. Tímto se aplikace vypne, je odstraněna z paměti a již neběží.²¹

2.2.2 Windows Phone 8.1 a novější

Prvním stavem je stav Spuštěný (Running), jež může nastat novým spuštěním aplikace nebo přechodem ze stavu Suspended (Pozastavený). Aplikace může být ukončena, pokud se nachází ve stavu Suspended a systém potřebuje volnou operační paměť pro další procesy. Při přesunutí stavu aplikace do stavu Pozastaveného čeká aplikace ještě 10 s, zda nebude obnovena. Pokud není, uloží si data a přechází do stavu Suspended.



Obrázek 5 – Životní cyklus aplikace od Windows Phone 8.1²²

²¹ Srov. LACKO, E., *Vývoj aplikací pro Windows Phone 7*, s. 6-7.

²² Srov. LACKO, E., *Vývoj aplikací pro Windows 8.1 a Windows Phone*, s. 74.

V tomto stavu zůstává v paměti na pozadí a aplikace nemůže vykonávat žádné činnosti a čeká na interakci uživatele na Obnovení (Resuming), které načte všechna data a aplikace je ihned k dispozici díky rychlému načtení z paměti. Posledním stavem je ukončení aplikace, které si systém řídí sám. Uživatel může též pomocí uživatelských gest systému aplikaci vypnout (vyvolá se stav Suspended pro uložení stavu aplikace), ovšem aplikace samotná nesmí mít žádný ovládací prvek pro její ukončení. Pokud se aplikace ukončí sama od sebe či nereaguje na gesta, považuje to systém Windows za chybu (pád, havárii). Havárii aplikace systém řeší odesláním informací na server Microsoftu, který poté upozorňuje vývojáře o vzniklé chybě pro její budoucí odstranění.²³

2.3 Technologie a vybavení pro vývoj mobilních aplikací Windows

2.3.1 Windows SDK

Windows SDK, též Windows Software Development Kit, je sada vývojových nástrojů, knihoven, ukázek a návodů pro vývoj aplikací na určité platformě. Pro vývoj mobilních aplikací budeme potřebovat např. Windows Phone SDK 8.1 nebo 10 (číslo verze obvykle demonstruje, pro jakou verzi je SDK určeno, ale obsahuje i podporu pro starší verze).²⁴ SDK se dá stáhnout ze stránek společnosti Microsoft dostupné na WWW: <<https://developer.microsoft.com/en-us/windows/downloads/sdk-archive>>.

2.3.2 Visual Studio

Visual Studio je vývojové prostředí (IDE) nabízející kompletní sadu nástrojů pro vývoj aplikací nejen pro platformu Windows (webové, konzolové, grafické a další). Obsahuje editor kódu, debugger, designer GUI rozhraní pro tvorbu formulářů i grafických prvků aplikace, též umožňuje instalovat rozšíření. Rozsáhlejší popis a historie verzí byly popsány v předcházející kapitole. Pro vývoj aplikací se musí v OS W10 povolit tzv. „Developer mode“ v sekci:

Nastavení » Pro vývojáře » Používat vývojářské funkce » Vývojářský režim (Developer Mode).

2.3.3 Programovací jazyky

Visual Studio ve verzi 2015 umožňuje vývoj aplikací v těchto programovacích jazycích: C, C++, C++/CLI (Visual C++), VB.NET, C#, F#, XML/XSLT, HTML/XHTML, JavaScript a CSS. Pro vývoj mobilních aplikací je k dispozici omezenější množství programovacích jazyků. Pro vývoj her pro mobilní zařízení slouží vývoj v DirectX prostředí spolu s programovacím jazykem C++.

Návrh grafického prostředí se provádí ve VS pomocí prostředí Blend for Visual Studio. Pro samotné programování se využívá editor kódu a debugovací nástroje.

²³ Srov. LACKO, E., *Vývoj aplikací pro Windows 8.1 a Windows Phone*, s 73-77.

²⁴ Srov. LALONDE, L., a TOTZKE, David R., *Windows Phone 8 recipes: a problem-solution approach*, s. 1-3.

Grafické prostředí (UI)	Programovací jazyk	SDK	Platforma (API)	Projekt (*více typů projektů)	Lze spustit (*nemusí běžet korektně)
HTML, CSS	JavaScript	10	UWP	*(Universal Windows)	W10M
XAML	C# / VB	10	UWP	*(Universal Windows)	W10M
XAML, DirectX	C++	10	UWP	*(Universal Windows)	W10M
DirectX	C++	10	UWP	*(Universal Windows)	W10M
HTML, CSS	JavaScript	8.1	WinRT	*(Windows Phone)	WP 8.1, W10M
XAML	C# / VB	8.1	WinRT	*(Windows Phone)	WP 8.1, W10M
XAML, DirectX	C++	8.1	WinRT	*(Windows Phone)	WP 8.1, W10M
DirectX	C++	8.1	WinRT	*(Windows Phone)	WP 8.1, W10M
XAML	C# / VB	8.1	Silverlight	*(Windows Phone Silverlight)	WP 8.1, W10M
XAML, DirectX	C++	8.1	Silverlight	DirectX and XAML App (Windows Phone Silverlight 8.1)	WP 8.1, W10M
XAML	C# / VB	8.0	Silverlight	*(Windows Phone Silverlight)	WP 7.5, WP 8, *WP8.1, *W10M
DirectX	C++	8.0	WinRT	DirectX App (Windows Phone 8.0)	WP 7.5, WP 8, *WP8.1, *W10M
XAML, DirectX	C++	8.0	WinRT	DirectX and XAML App (Windows Phone 8.0)	WP 7.5, WP 8, *WP8.1, *W10M

Tabulka 2 – Možnosti vývoje Windows Store aplikací ve VS 2015 (Zdroj: vlastní)

2.3.4 Windows Phone Emulator

Windows Phone Emulator je aplikace, která poskytuje virtualizované rozhraní se simulací zařízení WP jak po SW, tak i HW stránce pro testování těchto aplikací. Lze jej spouštět u WP projektů ve VS.

Windows Phone Emulator je součástí Windows Phone SDK, a nově je SDK součástí aplikace Visual Studio od verze 2015 (SDK 10). Pro jeho zprovoznění je potřeba 64bitový operační systém Windows 10 Pro či vyšší, 4 GB operační paměti a podpora pro SLAT (Second Level Address Translation), DEP (Hardware-based Data Execution Prevention) a Hyper-V (procesor s podporou hardwarové virtualizace – Intel VT nebo AMD-V), která se povoluje v nastavení BIOS a nastavení operačního systému. Windows Phone 8 Emulator vyžaduje VS 2012 a novější, se staršími verzemi VS není zpětně kompatibilní.²⁵

2.3.5 Windows App Certification Kit (WACK)

Windows App Certification Kit je součástí Windows SDK, ovšem jedná se o samostatnou aplikaci, jež slouží pro otestování vyvinuté aplikace před získáním certifikace pro Windows Store (lze testovat Windows Store a Windows Desktop aplikace). Tedy testuje všechny důležité požadavky kladené na Windows aplikaci, aby mohla být nahrána a dále distribuována ve Windows Store.²⁶

2.3.6 Vývojářský účet

S příchodem Windows 8.1 se rozšířilo i množství typů účtů (doménových a lokálních) o online Microsoft účet (dříve LiveID), který poskytuje připojení ke všem službám Microsoftu. Microsoft Visual Studio také umožňuje přihlášení pomocí Microsoft účtu. Přihlášení do aplikace VS pomocí Microsoft účtu přidává nové možnosti, jako je např. synchronizace nastavení prostředí aplikace a licencí VS mezi jednotlivými zařízeními, Team Foundation (nástroje pro vývoj a správu týmových projektů) či MSDN (předplacené služby MS) účtu atd.

²⁵ Srov. SOFTWARE, F., *Pro Windows phone app development*, s. 2-5.

²⁶ Srov. MOEMEKA, Edward, a MOEMEKA, Elizabeth, *Real world Windows 10 development*, s. 576-577.

Funkční Microsoft účet je vyžadován pro registraci k vývojářskému účtu. Vývojářský účet se totiž následně přidruží k Vašemu Microsoft účtu.²⁷ Tento účet umožňuje vyvíjet a publikovat aplikace na Windows Store (placené i zdarma). Zaregistrovat si vývojářský účet lze na oficiálních stránkách pro vývojáře Microsoftu dostupné na WWW: <<https://developer.microsoft.com/cs-cz/store/register>>. Vytvoření vývojářského účtu si popíšeme v nacházejících kapitolách. Pro vlastní vývoj a testování aplikací není zapotřebí vytvářet vývojářský účet, jelikož ten slouží hlavně pro publikaci aplikací ve Windows Store.²⁸

2.4 Design aplikací

Design aplikací musí být přizpůsoben funkčnímu ovládní pro dosah prstů, na telefonech i tabletech. Oproti ovládní počítače pomocí myši a klávesnice se jedná o značný rozdíl, kdy musíme brát v úvahu větší plochu prstů pro dotyk na tlačítka, která nesmí být hned vedle sebe, ale nabídnout dostatečný rozestup. Stejně tak lišta na rolování nabídky a další prvky musí být vhodně a prakticky rozestavěny pro tento styl ovládní. O pravidlech a doporučeních při tvorbě designu aplikací bude pojednávat tato kapitola.

2.4.1 Filosofie ovládní

Primárním vstupem ovládní chytrých telefonů s dotykovou obrazovkou je dotek či doteky prstů. Pro tento styl ovládní musí být efektivně navržená aplikace, u níž budou v maximální možné míře bez většího omezení uživatelského komfortu rozmístěny ovládací prvky i samotný obsah aplikace pro pohodlné ovládní pomocí jedné či obou rukou. Pro řešení tohoto problému existuje doporučení, kam na obrazovce telefonu či tabletu umístit aktivní prvky.

Dalším ovlivňujícím faktorem je velikost zařízení a jeho obrazovky, jemuž se musí zobrazovaný formát aplikace přizpůsobit v dostatečné míře, aniž by byl omezen komfort uživatele a ten si nemusel například přibližovat obrazovku, jelikož nepřečte text či se mu nedaří strefit prstem na ovládací prvek. O to se stará automaticky tzv. Scaling (škálování či responzivní design, který je dlouhé roky používán na webových stránkách), který si automaticky přizpůsobuje prvky, tak aby byly co nejlépe čitelné na všech typech zařízení.

Oba zmíněné principy, tedy způsob ovládní i responzivní design, musí brát ohled na velikost zařízení a jeho rozlišení obrazovky, dále také dálku, ze které se na něj standardně uživatel dívá, a v neposlední řadě i na natočení displeje na výšku (**Portrait**) či šířku (**Landscape**). Též si můžeme aplikaci přichytit pouze na část obrazovky (flexibilní zobrazení), přičemž je minimální šířka okna nastavena na 500px (lze i 320px pro podporu aplikací z Windows 8).²⁹

Tyto techniky se dají nastavit pomocí komponenty VisualStateManager (VSM) obsahující adaptabilní spouštěče, které si může programátor upravit k obrazu svému pro různá rozlišení nebo pro různé typy zařízení. U aplikací napsaných pomocí HTML5/JavaScript se responzivní

²⁷ Srov. MICROSOFT, *Časté otázky k programu pro vývojáře*, <<https://developer.microsoft.com/cs-cz/store/register/faq>>.

²⁸ Srov. MOEMEKA, Edward, a MOEMEKA, Elizabeth, *Real world Windows 10 development*, s. 565-568.

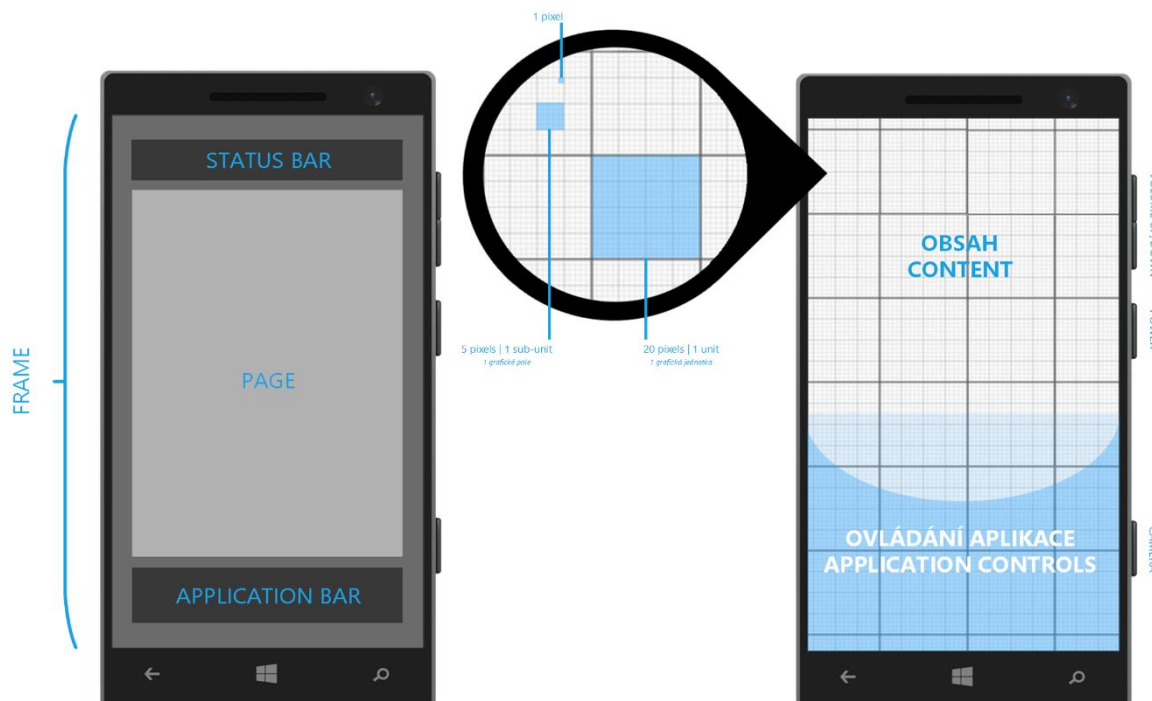
²⁹ Srov. LACKO, L., *Vývoj aplikací pro Windows 8.1 a Windows Phone*, s. 24-27, s. 61-67.

design nastavuje pomocí CSS @media. Téměř stejného výsledku se dá též docílit pomocí RelativePanel nebo SplitView komponent. Pro ovládací prvky je dostupná komponenta Pointer API, jež přizpůsobuje ovládací prvky pro použití dotyků. Zařízení Windows počítají s automatickým škálováním aplikace a to na 100 %, 140 % a 180 %, u projektu ve složce Assets jsou obrázky, a hlavně loga v těchto velikostech pro ikony aplikace, určené pro dynamické dlaždice i úvodní obrazovku.

2.4.2 Základní doporučení a možnosti nastavení

Pro mobilní zařízení jsou v základu nastavena dvě barevná zobrazení. Tmavé (Dark), které má tmavé pozadí a světlé písmo. Světlé (Light), jež má tmavé písmo a světlé pozadí. Mezi těmito zobrazeními se dá přepínat v nastavení telefonu, situace je však odlišná pro Desktopové aplikace, kde barevnou šablonu vybírá návrhář aplikace, neboť musí do aplikace přepínání barevných šablon zakomponovat.

Design aplikací rozlišuje takzvaný typografický styl, jež určuje hierarchii písma (převážně typografický font Segoe) a jeho velikost pro nadpis, podnadpis, prostý text a mnohé další. Dalším pojmem je grafická jednotka (unit), jež se skládá ze čtverce o velikosti 20×20px (pixelů), ten je rozdělen na grafická pole (sub-unit), která mají velikost 5×5px a jsou 4×obsažena v jedné grafické jednotce. Pomocí grafických jednotek určujeme doporučené odsazení prvků aplikace od okrajů či sebe samých, díky tomu je aplikace přehledná a dá se snadno ovládat. Odsazení prvků aplikace se neprovádí orámováním pomocí čar či boxů, ale pouze prázdným místem, které je rozloženo na hierarchické virtuální mřížce.³⁰



Obrázek 6 – Grafické rozložení prvků a Windows Phone Emulator (Zdroj: vlastní)

³⁰ Srov. LACKO, E., *Vývoj aplikací pro Windows 8.1 a Windows Phone*, s. 56-61.

2.5 Vývoj aplikací pro Windows Phone

Design a potřebné nástroje pro vývoj jsme rozebrali v předchozích kapitolách. V této kapitole představíme aplikační model pro vývoj moderních WinRT aplikací a uskutečnime jednoduchý náhled do hlavních částí prvků aplikační logiky.

2.5.1 Návrhový vzor MVVM a Data Binding

Aplikační logika je jednou z důležitých věcí při vývoji aplikací, díky ní můžeme oddělit potřebné části od sebe, jednak pro zvýšení kvality, tak i logicky zpřehlednit výsledný kód pro budoucí úpravy či vylepšení. Windows Phone používá pro tyto účely návrhový vzor MVVM (Model – View – ViewModel).³¹

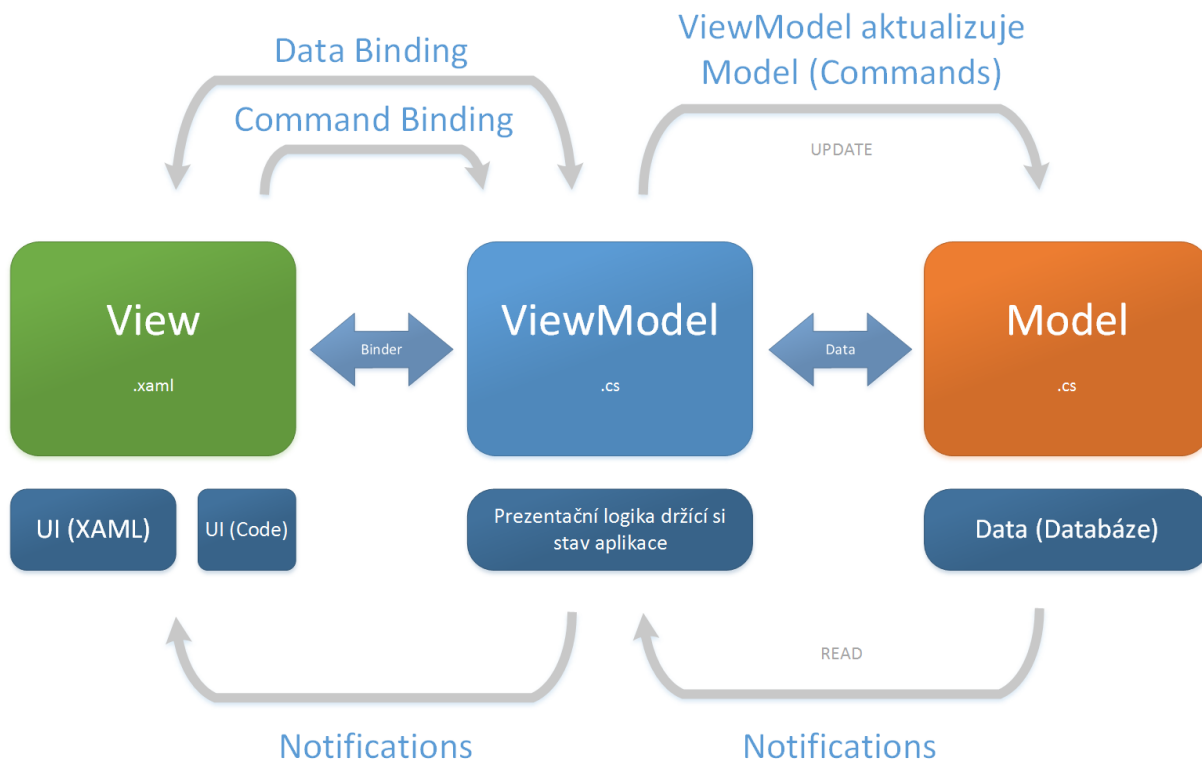
Tento mechanismus využívá propojení 3 vrstev: pro komunikaci mezi View (dále V) a ViewModel (dále VM) se používá tzv. Data Bindingu (Vázání dat), který umožňuje propojit prezenční logiku s uživatelským rozhraním (UI) pro flexibilní synchronizaci dat, jež prováže prvky mezi sebou a o vzájemných změnách je vždy druhá strana informována a může tak reagovat na danou tzv. událost (**Event**). Na straně UI je zastoupen XAML, který identifikuje Data Binding pomocí **{Binding}** identifikátoru. **DataTemplate** je datová šablona pro jeden prvek z kolekce.

```
<TextBlock Text="{Binding Date, StringFormat= dd/MM/yyyy}" />
```

Notifikace o změnách obstarávají tzv. události (**Eventy**) pomocí použití třídy **ObservableCollection<T>** ve ViewModel obstarávající přenos vlastností z VM do V, který implementuje interface **INotifyCollectionChanged** a **INotifyPropertyChanged** (interface lze implementovat i nezávisle na třídě **ObservableCollection**). Model samotný obsahuje Data aplikace a základní metody pro práci s nimi, například Entity pro ORM, XML datový soubor atd.

³¹ Srov. LEE, H., a CHUVYROV, E., *Beginning Windows phone app development*, s. 38-39.

K Data Bindingu se též váže pojem Converter (či Value Converter), který slouží pro změnu či převod vstupního formátu hodnot do výstupních hodnot, které jsou zpracovány dle požadovaných pravidel předtím, než jsou bindovány. Stačí implementovat interface **IValueConverter**.



Obrázek 7 – Vzor MVVM (Zdroj: vlastní)

2.5.2 Stav aplikace a jejich zachování

Stavy aplikace vycházejí z životního cyklu aplikace. Tyto stavy jsou obsluhovány v UI Code části aplikace, která je schována v XAML souboru ***.xaml.cs**. Stavy aplikace jsou vyvolány při přechodu na novou stránku, spuštění jiné aplikace, navrácení se zpět k aplikaci či předchozí stránce v aplikaci. Navigace mezi stránkami lze obsluhovat hned několika metodami. Je důležité poznamenat, že se stránky při každém zobrazení vytvářejí zcela znovu. V praxi to znamená, že můžeme tyto stavy obsloužit při opouštění stránky (SaveState) či návratu na předchozí (LoadState). A hodnoty, např. vyplněné hodnoty či nastavení, znovu načíst. Tento princip vychází z úspory baterie, flexibilního využívání paměti a svižnosti aplikace u Modern UI zařízení, která si nekešují data a neudržují si všechny stránky dané aplikace v paměti. V následujících odstavcích si vysvětlíme základní prvky pro navigaci a uchování stavu aplikace, jež nám tato platforma nabízí.

Třída **Frame (PhoneApplicationFrame)** slouží jako základní prvek obsahu, který podporuje navigaci a udržuje její historii. Obsahuje též instance všech **Page (PhoneApplicationPage)** prvků a metodu **Navigate**. Frame vyplňuje celou oblast obrazovky telefonu a zpřístupňuje tím plochu pro stránky (dále jen Page), kterých může být vytvořeno libovolné množství. Avšak prvek Frame by měl být jen jeden (ve WP8 bylo možné vytvořit pro jednu aplikaci pouze jeden Frame, zatímco od WP8.1 jich lze vytvořit více a pro každý Frame

si lze uchovávat zvlášť samostatnou navigační historii). Frame se vytváří v metodě **OnLaunched** (zahájení, start) v App.xaml.cs.

Každá **Page** obsahuje metodu **OnNavigatedTo**, jež je vyvolána před samotným zobrazením stránky a slouží pro načtení předávaných hodnot z předchozí stránky atd. Lze použít též metodu **OnNavigatedFrom**, která je vyvolána před opuštěním stránky a používá se pro uložení stavu na stávající stránce.

Akce životního cyklu aplikace jsou umístěny v **App.xaml.cs** a jedná se o akce jako **OnLaunched** při spuštění aplikace (načtení dat), **OnSuspending** při přesouvání aplikace na pozadí (ukládání dat) nebo při stisknutí jednoho ze tří HW (u novějších zařízení jsou tlačítka součástí OS) tlačítek (Back key, Main menu, Search), **OnResuming** (načítání dat) při obnovování stavu aplikace ze stavu Suspended.

Předávání parametrů mezi jednotlivými stránkami obstarává také metoda **Frame.Navigate(typeof(stranka), parametry)**, která umožňuje přesunutí z jedné stránky na druhou a přenáší i parametry. Zařízení s Windows Phone obsahuje HW tlačítko zpět (šipka ukazující vlevo, Back key), které nás vždy vrátí na předchozí aplikaci, ukončí stávající či nás přesune do seznamu aplikací Windows Phone. Jedná se tedy o navigaci mezi aplikacemi, nikoli mezi stránkami této aplikace. Pokud chcete tuto funkčnost změnit, musíte ji přepsat pomocí **HardwareButtons_BackPressed** event metody. Všechny tyto stavy aplikace lze nasimulovat pomocí VS a Windows Phone Emulatoru.

Visual studio nabízí několik templatů (předpřipravených šablon) pro snadnější psaní aplikací, jako je **HubApp**, **SplitApp**, **GridApp** či **PivotApp** (ve kterých je obsažena třída **NavigationHelper**, **SuspensionManager** a další) a také úplně prázdnou šablonu aplikace **BlankApp**. **NavigationHelper** slouží pro snadnou obsluhu navigace mezi stránkami a je umístěn v projektu ve složce **Common**. Přidává rozšíření v podobě přepsání **Back key** a obsahuje metody **SaveState** a **LoadState** pro zapamatování si stavů prvků na dané stránce. Projekt **BlankApp** jej neobsahuje, ale lze jej jednoduše dodat do projektu přidáním nové položky, pomocí **BasicPage**, která tuto funkcionalitu obsahuje (při přidání **BasicPage** do daného projektu VS uživatele vyzve, zda chcete doplnit do složky **Common** i závislosti, jež **BasicPage** vyžaduje), na výběr je též přidání nové položky s názvem **BlankPage**, ale ta obsahuje pouze prázdnou stránku bez jakýchkoliv pomocných tříd.

Uložení stavu aplikace obstarává třída **Windows.Storage.ApplicationData** přes složky umístěné v **Isolated Storage** (Izolované úložiště). Úložiště je rozděleno na dva typy a to na: **Folder**, kam se dají ukládat libovolné soubory, a **Settings**, které slouží pro uložení typu klíč hodnota (**Dictionary**). Další speciální typ úložiště je **Roaming**, kdy se hodnoty ukládají i do cloudového úložiště **OneDrive** (dříve **SkyDrive**), pokud je má uživatel aktivované. Posledním typem úložiště je **Temp**, které slouží pro dočasné uložení dat pro jeden běh aplikace, po znovuzapnutí aplikace se data z minulého běhu smažou.³²

³² Srov. TABOR, B., *Part 12 - Understanding the App's Lifecycle and Managing State*, <<https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-12-Understanding-the-App-s-Lifecycle-and-Managing-State>>.

Windows.Storage.ApplicationData.Current.*		
Název	Typ	Popis
Local Settings	Lokální hodnoty aplikace	lokální nastavení aplikace (klíč ↔ hodnota)
Local Folder	Lokální soubory aplikace	lokální složka pro soubory aplikace (libovolné soubory)
Roaming Settings	Cloudové hodnoty aplikace	roaming nastavení aplikace (klíč ↔ hodnota)
Roaming Folder	Cloudové soubory aplikace	roaming složka pro soubory aplikace (libovolné soubory)
Temporary Folder	Uložení dočasných souborů	temp aplikace

Tabulka 3 – Typy úložišť pro Windows Phone aplikace (Zdroj: vlastní)

Uložení stavu relace (jednoho běhu aplikace, Session či dočasných údajů) obstarává třída SuspensionManager s pomocí NavigationHelper. SuspensionManager je třída typu helper, jež obstarává řízení běhu aplikace a usnadňuje tak práci s dočasnými údaji aplikace během přecházení mezi těmito stavy životního cyklu aplikace. SuspensionManager obsahuje metodu SessionState sloužící pro ukládání a práci s dočasnými údaji typu Session.³³

2.5.3 Asynchronní programování

Asynchronní zpracování můžeme chápat jako proces, který vykonává nějakou složitější či dlouhotrvající úlohu na pozadí (kupříkladu načítání dat, odesílání či přijímání údajů prostřednictvím Internetu), aniž by brzdil či zmrazil aplikaci tak, aby byla po tuto dobu nečinná. To znamená, ačkoli aplikace vykonává načítání dat, přesto je po tuto dobu schopná reagovat na uživatelské podněty. Po dokončení své úlohy notifikuje aplikaci o této události a vykoná se další posloupnost příkazů na tuto akci navazujících. Podle doporučení Microsoftu by každá akce trvající déle než 50 ms měla být naprogramovaná jako asynchronní. Metoda, která je označena modifikátorem **async**, určuje, že tato metoda zavolá asynchronní metody pomocí operátoru **await** (oddělené vlákno). Await může být vyžádán (volán) pouze v metodě, která má nastaven **async** modifikátor. Asynchronní metody mají obvykle název metody **Async()** a vrací návratový typ typu **Task**, **Task<T>** nebo **void**.³⁴

2.5.4 Serializace a deserializace dat

Serializace (serialization) je přenos objektu do datového streamu, jako je soubor, databáze nebo paměť. Deserializace (deserialization) je „vytáhnutí“ dat a vytvoření objektu z DB, paměti nebo souboru.³⁵

³³ Srov. BUDDY, J., a LALONDE, L., *Pro XAML with C#: from design to deployment on WPF, Windows Store, and Windows Phone*, s. 172-176.

³⁴ Srov. TABOR, B., *Part 19 - Understanding async and Awaitable Tasks*, <<https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-19-Understanding-async-and-Awaitable-Tasks>>.

³⁵ Srov. TABOR, B., *Part 22 - Storing and Retrieving Serialized Data*, <<https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-22-Storing-and-Retrieving-Serialized-Data>>.

2.6 Windows Store

Windows Store je, jak už bylo řečeno, oficiálním distribučním kanálem pro Windows Store Aplikace od OS Windows 8 (2012). Pro mobilní zařízení je určen Windows Phone Store. Od listopadu 2016 umožňuje převod klasických desktopových aplikací (.NET nebo Win32) na Windows Store pomocí aplikace Desktop Bridge, ta konvertuje stávající aplikace na platformu UWP. Starší projekty VS pro WP lze převést na novější platformu UWP pomocí návodů uvedených na oficiálních stránkách společnosti Microsoft.

Aplikace se začaly dělit na Windows Desktop Apps (napsané ve Win32/COM API či .NET Frameworku) a nové Windows Store Apps, které běží pouze na Windows 8/Windows Phone 8 a novějších, které jsou napsané pomocí Windows Runtime. Windows Store je dostupný pro oba typy aplikací. Co se týče desktopového OS, pro mobilní OS jsou dostupné pouze Windows Phone Store aplikace a UWP aplikace.

Pro Windows Store je dostupný WinRT API od WP 8.1, který nahrazuje Win32 v oblasti vývoje aplikací pro mobilní zařízení od společnosti Microsoft Windows. S příchodem Windows 10 Mobile, bylo nahrazeno WinRT za UWP (UWP je založeno na WinRT, ale je plně univerzální). Windows 10 Mobile nepodporují aplikace vyvinuté ve Win32/.NET. Ale aplikace napsané v C++/CX mají přístup do Win32 API, jelikož jsou součástí UWP (Windows 10 implementují Win32 API pomocí knihovny Windowsapp.lib).

Aplikace musí splňovat několik požadavků, aby mohla být publikována přes distribuční kanál Windows Store. Aplikace musí běžet v tzv. Sandboxech (izolované/ohraňované místo pro běh aplikace s přidělenými zdroji pro zajištění větší bezpečnosti)³⁶, nesmí mít žádný ovládací prvek na její ukončení, musí mít nezávadný obsah, splňovat bezpečnostní kritéria a mít nějakou smysluplnou přínosnost. Jeden z prvků, jenž si MS klade za cíl, je distribuovat aplikace s jednotným rozhraním pro ovládání, na které je uživatel zvyklý, a umožnit co nejlepší využití zařízení s ohledem na uživatelův komfort.

Pro samotnou publikaci je zapotřebí vytvoření Vývojářského účtu a po nahrání musí aplikace projít schvalovacím procesem pro publikaci na Windows Store. Windows Store má sice jeden název, ale zatímco pro UWP aplikace od Windows 10 nabízí jeden produkt pro všechny zařízení od Microsoftu, je situace pro starší produkty odlišná, jelikož ty jsou zaměřeny (napsány) speciálně pro dané zařízení a aplikace jsou tak zvlášť distribuovány ve verzi Windows Store určených pro Xbox, Windows Phone, Windows a další.

Pro firmy existuje odnož Windows Store for Business, kde je možné šířit aplikace pouze v rámci firmy, a to i bez připojení k internetu díky offline licencím, které spravují administrátoři organizace pro jejich jednotlivé aplikace. Stačí mít u Microsoftu zřízeny účty pro Azure AD (Microsoft Azure Active Directory), přiřazené k dané organizaci, a vytvořit si Privátní Windows Store.³⁷

³⁶ Srov. SZOSTAK, T., *Windows Phone 8 Application Development Essentials: A practical guide to creating a Windows Phone 8 application using C#, XAML, and MVVM*, s. 72.

³⁷ Srov. MICROSOFT, *Windows Store for Business overview*, <<https://technet.microsoft.com/itpro/windows/manage/windows-store-for-business-overview#Supported-markets>>.

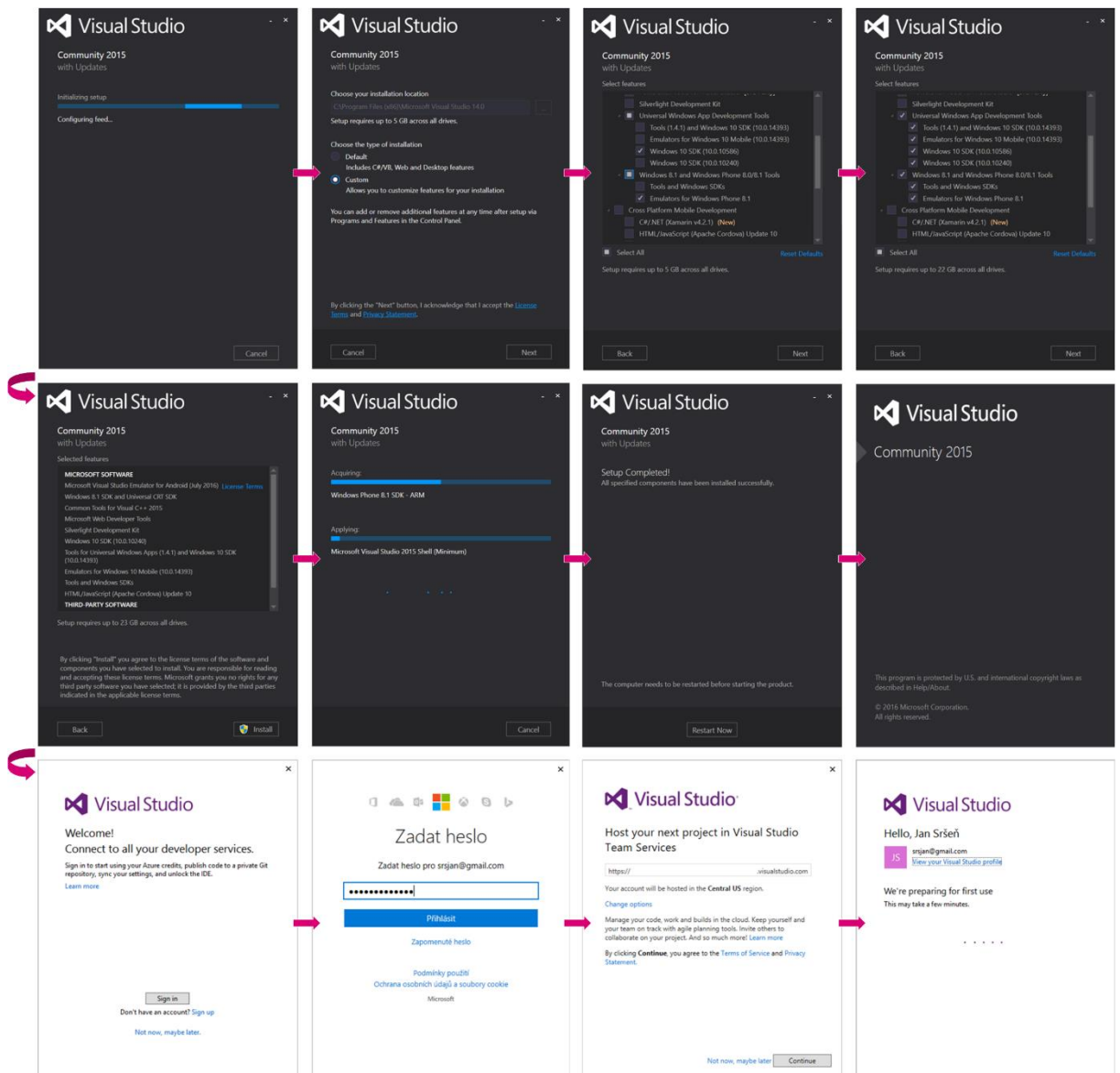
3. Návrh a vývoj aplikace IP Network Calculator

Vývoj aplikace IP Network Calculator bude probíhat v IDE prostředí Visual Studio verze 2015 Community, nainstalovaném na OS Windows 10 Pro, a následná aplikace bude demonstrována na mobilním telefonu Microsoft Lumia 535 s operačním systémem Microsoft Windows 10 Mobile. Pro vývoj použijeme programovací jazyk C# a pro návrh prezenčního rozhraní jazyk XAML. Výsledná aplikace bude napsána pomocí technologie UWP, jež byla uvedena společně s platformou Windows 10.

3.1 Instalace aplikace Visual Studio 2015

Pro vývoj aplikací určených pro Windows Store a použití na zařízeních s Windows Phone 8.1 a Windows 10 Mobile je potřeba použít MS Visual Studio 2015 Community, starší verze aplikace VS nepodporují vývoj UWP. Vývoj UWP (Universal Windows Platform) plně podporuje jen OS Windows 10, starší verze OS Windows též umožňují vyvíjet aplikace pro UWP, nejdou zde však spustit. VS Community lze stáhnout z oficiálních stránek, standardně se stáhne jen instalační soubor pro online instalaci aplikace VS (online instalace z internetu), ale můžeme rovněž stáhnout i ISO soubor (7.2 GB – offline instalace), který obsahuje základní komponenty pro vývoj. Je možné jej stáhnout na WWW: <<https://www.visualstudio.com/downloads/>>. Samotná instalace trvá od 40 minut (Default – velikost instalace je cca 8 GB) až do 2 hodin (Custom) podle výběru typu instalace, kdy máme na výběr mezi klasickou a volitelnou instalací. Volitelná (Custom) instalace umožňuje výběr jednotlivých prvků instalace aplikace VS i instalace aplikací třetích stran. Tento typ instalace musíme zvolit a vybrat emulátor pro Windows 10 Mobile, pokud chceme vyvíjet pro mobilní platformu, jelikož v základu se instaluje pouze Windows 10 SDK bez emulátorů pro mobilní vývoj v UWP (W10M) a instaluje se emulátor jen pro WP 8.1 (celková velikost instalace je cca 32 GB). Emulátor lze doinstalovat i samostatně.

Při instalaci aplikace VS 2015 na novém PC a otevření projektu z jiné revize aplikace VS 2015 je někdy potřeba doinstalovat chybějící verzi Windows 10 SDK specifické revize, na níž Vás aplikace VS upozorní.

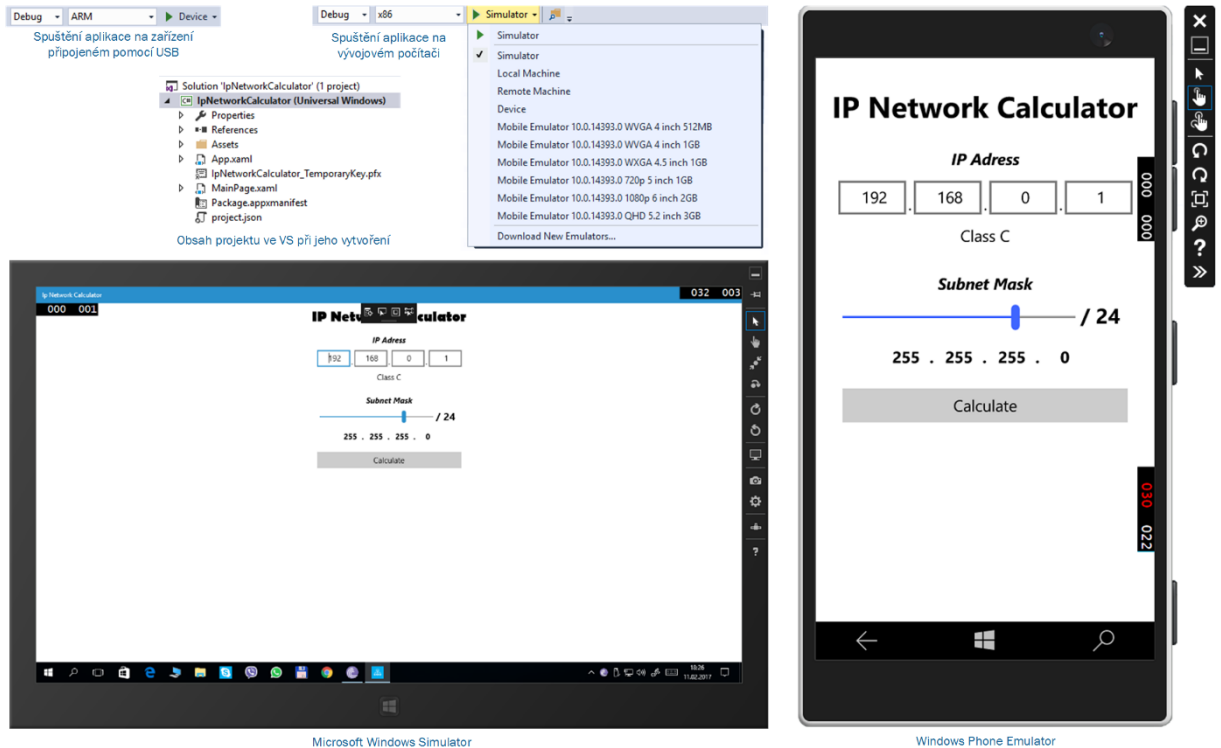


Obrázek 8 – Instalace aplikace Visual Studio 2015 (Zdroj: vlastní)

3.1.1 Základní prvky aplikace Visual Studio a jeho nástrojů

Aplikaci lze spustit ve VS několika způsoby, dle typu projektu. První a tou nejrychlejší možností je spustit projekt pomocí volby Local Machine, tedy spustit aplikaci přímo jako program v systému Windows. Volba Simulator (Microsoft Windows Simulator (dále jen MWS) je obdobou Windows Phone Emulatoru od 4“ do 6“ (rozlišení v palcích) s tím rozdílem, že simuluje prostředí Tabletů od 7“ do 27“) spustí v novém okně váš systém znovu i s aplikací, kterou jste si přáli spustit. Tato možnost skrývá jednu nepříjemnost, Simulator nenabízí tlačítko X pro jeho vypnutí, jako je tomu u WPE. Pokud simulátor MWS vypnete v prostředí pomocí nabídky **Start » Vypnout**, může se vám kompletně vypnout celý počítač, musí se proto použít volba **Start » Odpojit**. Windows Phone Emulator nabízí hned několik variant dle rozlišení a přidělené operační paměti. Pro spuštění dané aplikace v Emulatoru je potřeba nainstalovat aplikaci VS s podporou Windows Phone Emulatoru, mít povolen Vývojářský režim a zapnutou technologii Hyper-V.

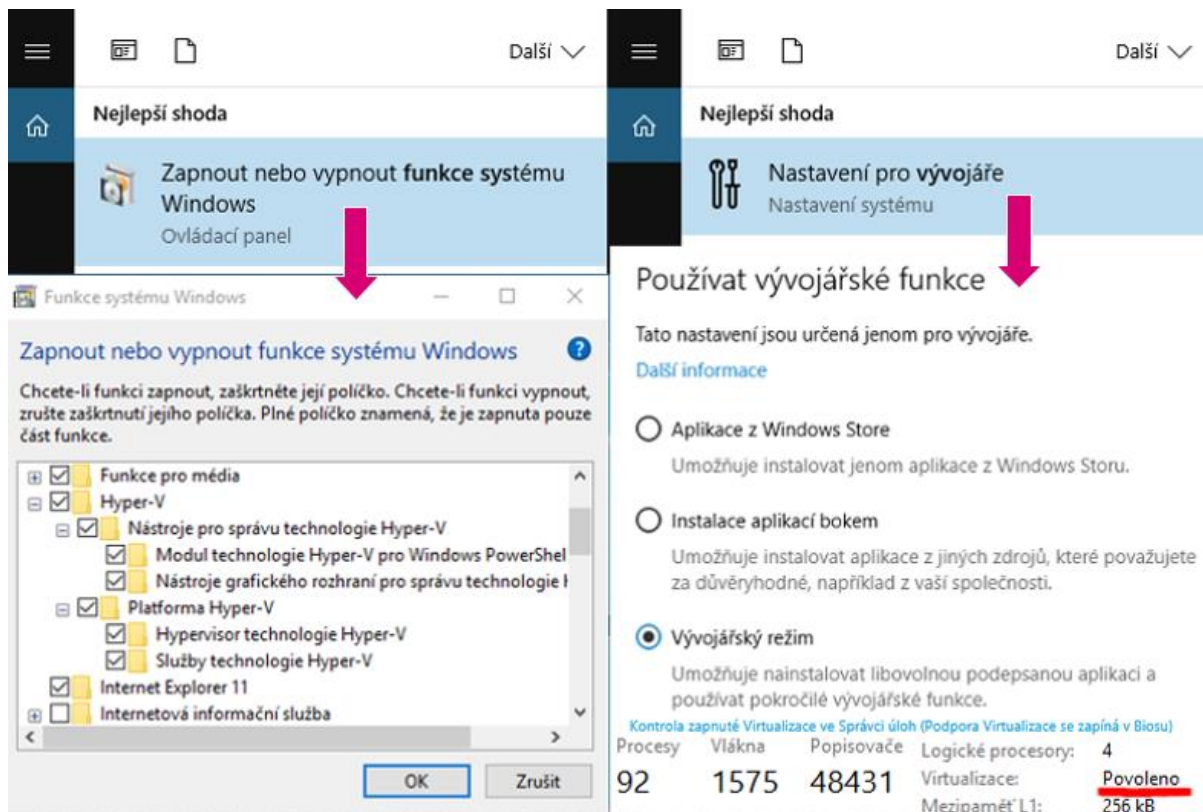
Aplikaci lze též spustit přímo na daném mobilním zařízení, které je potřeba mít připojené pomocí USB kabelu: v nabídce spuštění aplikace vybereme položku Device a typ platformy ARM. Telefon je nutné zaregistrovat pro vývoj aplikací pomocí nástroje Windows Phone Developer Registration, jenž vyžaduje přihlášení k MS Developer účtu, také je potřeba mít připojený telefon pomocí USB kabelu k PC a zároveň musí být telefon či zařízení odemknuto (odemčená zamykací obrazovka telefonu).



Obrázek 9 – Základní prvky aplikace VS a jeho nástrojů (Zdroj: vlastní)

3.2 Podpora Virtualizace, zapnutí Hyper-V a Vývojářského režimu

Jak bylo uvedeno v kapitole 2.3.4, pro vývoj aplikací pro platformu Windows Phone a použití Windows Phone Emulátoru musí být zapnut Vývojářský režim, Hyper-V a v BIOSu PC povolena Virtualizace (procesor ji musí podporovat).



Obrázek 10 – Zapnutí Vývojářského režimu a povolení Virtualizace (Zdroj: vlastní)

3.3 Registrace Windows Store Developer Account

Registrace Windows Store Developer Account je dostupná na WWW: <<https://developer.microsoft.com/cs-cz/store/register>>, kde po kliknutí na odkaz „Zaregistrovat se“ budete přeměrováni na stránku pro přihlášení do Microsoft Developer Centra pomocí LiveID účtu. Pokud jej zatím nemáte, najdete zde i odkaz na vytvoření nového MS účtu. My jsme se přihlásili pomocí svého LiveID účtu, který je zaregistrován na soukromou Gmail adresu. Po přihlášení máte na výběr mezi Individuálním účtem za (365 CZK) či účtem pro Společnosti (1720 CZK). Platba by měla být jednorázová a není potřeba účet po uplynutí nějaké doby znovu obnovovat (i když je v uživatelském panelu Microsoft Účtu napsáno **Roční registrace vývojáře ve Windows a Windows Phone v jiné záložce je uvedeno Vyprší dne 1. 2. 2067**). My jsme si vybrali Individuální účet a zaplatili danou částku pomocí debetní karty (opsáním čísla karty a dalších potřebných údajů). Po uplynutí krátké doby bylo z účtu strženo 442 CZK (365 CZK + daň 77 CZK) a Developer Account byl po zadání přihlašovacích údajů a potvrzení pomocí emailu úspěšně aktivován.

Microsoft Prostředky pro vývojáře

STORE Zaregistrovat se Vývoj aplikací pro Windows Publikování Finanční zúčtování Propagace Prostředky

Zaregistrujte se jako vývojář aplikací

S vývojářským účtem můžete odesílat aplikace a doplňky na weby marketplace Microsoftu, jako je Windows Store, Office Store, Azure Marketplace a další partnerské weby.

Jste připraveni začít?

Kliknutím dále si můžete zaregistrovat osobní/studentský účet nebo účet společnosti. Požadujeme vás, abyste zadali svoje kontaktní informace, zveřejníme zobrazovaný název vydavatele a přidali aplikační poplatky pro jednorázovou registrační poplatek - Visa/MasterCard. Poplatek spoled to platit nebo propagační kód. (Poznámka: Propagační platěcí karty nejsou akceptovány.)

Účet jednotlivců stojí přibližně 19 USD a účty společnosti stojí přibližně 99 USD (platěné částky se kři podle vaší země nebo oblasti). Jde o jednorázový registrační poplatek a není nutné žádné obnovení předplatného.

Poznámka: Musíte se přihlásit pomocí účtu Microsoft, který dříve přiložit se svému vývojářskému účtu. Pokud účet Microsoft ještě nemáte, můžete ho získat, když zahájíte proces registrace.

Zaregistrovat se

Máte otázky? Přetáhněte si odpovědi na registraci dříve.

Registrace - Informace o účtu

Země/oblast účtu

Česká republika

Vyberte zemi nebo oblast, kde žijete nebo kde půlíte vaše společnost. Po zadání informací pro váš účet v této oblasti moct tento údaj změnit. Odpovědi na otázky týkající se podporovaných zemí nebo oblastí najdete v registračních datech.

Typ účtu

Nevíte, který typ účtu vybrat? Další informace

Po vyplnění informací o účtu není možné změnit typ účtu. Uvedená cena je jednorázový registrační poplatek a není potřeba žádné obnovení.

Individuální 365,00 CZK
 Získáte možnost vyvíjet a prodávat aplikace, doplňky a služby jako jednotlivec, student nebo občanské sdružení.

Společnost 1 720,00 CZK
 Získáte možnost vyvíjet a prodávat aplikace, doplňky a služby pod svým regionálním uzávaným a registrovaným obchodním názvem. Získáte přístup k pokročilým analytickým funkcím a dalším možnostem aplikací.

Zobrazovaný název vydavatele

Jan Sršeň

Znáte-li už několik náhod aplikací, doplňků nebo služeb pod svým jedinečným zobrazovaným názvem vydavatele.

Registrace - Platba

Informace o účtu ✓

Platba

Kontrola

Zadejte prosím své platební údaje. Dokud nedokončíte registraci na další stránce, nebudeme vám účtovat žádné platby. Toto je jednorázový registrační poplatek a není potřeba žádné obnovení.

Promo kód

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

Pokud máte propagační kód, zadejte ho sem, abyste pokryli náklady na registraci.

Fakturace

Zvolte způsob platby

Kreditní/debetní karta
 PayPal

Přidat informace o platbě

Typ karty

VISA

Číslo karty

Údaj zadejte bez pomlček a mezer

Jméno na kartě

CVV

_____ [Co je to?](#)

Datum ukončení platnosti

MM / RRRR

_____ / _____

Nastavení účtu

Nastavení účtu

Podrobnosti o účtu Uplatnit kód

Základní informace

Typ účtu Individuální

Stav účtu Aktivní

Stav prověření Autorizovaný

ID vydavatele

ID prodejce 29

ID vydavatele Windows CN=86776497-

ID vydavatele Windows Phonu 415a65c8-6429

Finanční údaje

Pokud publikujete placené aplikace, doplňky nebo služby, poskytněte nebo aktualizujte své finanční údaje.

Vyplnit účet Přidat

Daňový profil Přidat

Stav blokování výplaty Vypnuto

Platby vám budoume posílat, jakmile budete mít nárok na výplatu. Další informace

Kontaktní informace Aktualizovat

Veřejné informace

Zobrazovaný název vydavatele Jan Sršeň

Kontaktní informace prodejce

Název Jan Sršeň

E-mail

Telefon

Adresa

Česká republika

Zařízení

Odemknuté Windows Phony

Podívejte se na zařízení Windows Phone 8 a Windows Phone 8.1, která se dají použít pro testování balíčků aplikací před odesláním do Store.

Vývojářské konzoly Xbox One

Umožňuje zobrazování a správu vašich konzol Xbox One, u kterých je povoleno použití pro vývoj. Můžete aktivovat a odebrat konzoly, které je možné používat pro testování her a aplikací před odesláním do Store.

Registrace - Kontrola

Informace o účtu

Jan Sršeň

Česká republika

Poplatky

Cena za registraci 365,00 CZK

Odhadovaná daň 77,00 CZK

Odhadovaný součet 442,00 CZK

Smlouva s vývojářem aplikace

Smlouva s vývojářem aplikací obsahuje podmínky vztahu mezi vámi a společností Microsoft, protože se týká vašeho používání webu Windows Store a řídicího panelu. Přetáhněte si prosím Smlouvu s vývojářem aplikací.

Až se na webu Windows Dev Center zaregistrujete jako vývojář, budete dostávat informace, tipy a nabídky týkající se vývoje aplikací pro Windows, jakož i dalších produktů a služeb Microsoftu. Odebrat našich e-mailů můžete kdykoli zrušit.

Souhlasím s podmínkami a ujednáními Smlouvy s vývojářem aplikací, kterou navrhla společnost Microsoft.

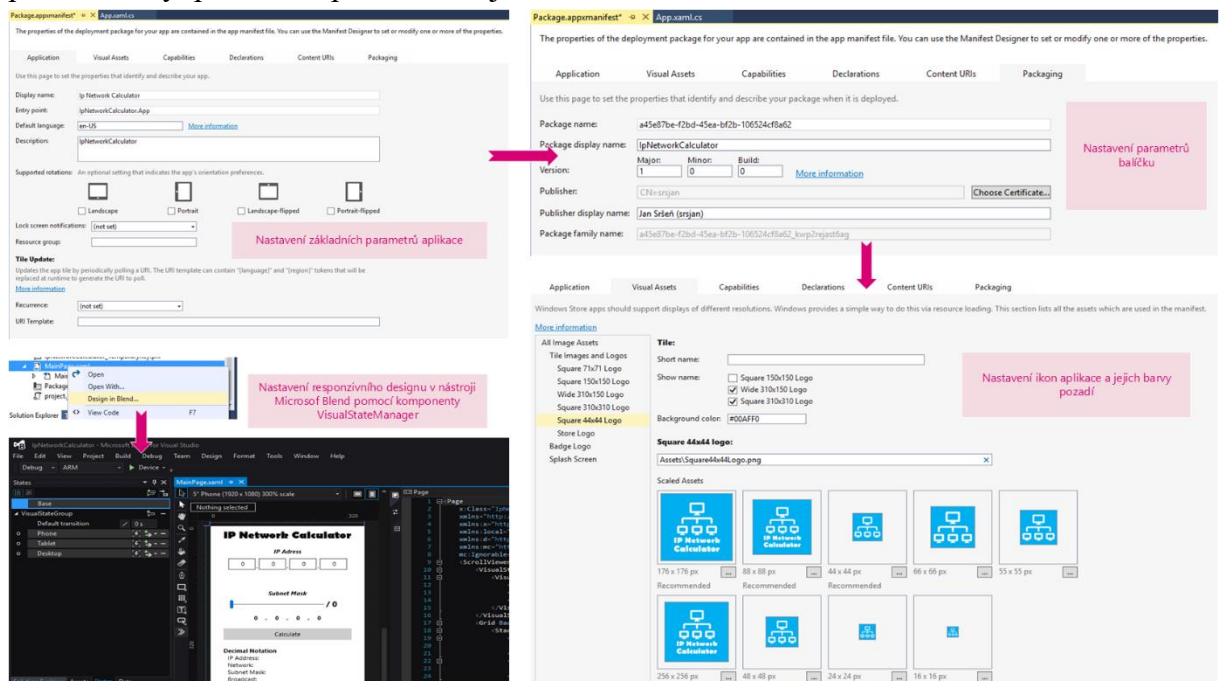
Zpět Dokončit

Obrázek 11 – Registrace Windows Developer Account (Zdroj: vlastní)

3.4 Hlavní komponenty a vývoj aplikace IP Network Calculator

Ve VS 2015 vytvoříme nový projekt v jazyce C# a vybereme šablonu UWP (**Visual C# » Windows » Universal » Blank App**). Po úspěšném vytvoření projektu otevřeme k editaci soubor Package.appxmanifest. Tento soubor obsahuje několik záložek, z nichž první se jmenuje Application. Application slouží k nastavení jména aplikace, voleb zobrazení a natočení aplikace pro zařízení, které tuto funkčnost umožňují. Další záložka Visual Assets umožňuje nastavení ikon a ostatních grafických prvků, které aplikace využívá pro svou prezentaci v systému a spouštění aplikace. Poslední zajímavé záložky, jež bychom ještě rádi zmínili, jsou Capabilities, tedy schopnosti (použití Lokace, Internetu, Webkamery atd.), které daná aplikace umí (využívá), a záložka Packaging sloužící k nastavení verze aplikace, certifikátu a názvu pro nahrání na Windows Store.

Aplikace využívá několika prvků jazyka XAML, a to **StackPanel**, který zajišťuje nastavení zarovnání do horizontální či vertikální polohy. O responzivní design se stará komponenta **VisualStateManager**. Ta přidává ke každé své položce tzv. trigger (**AdaptiveTrigger**), jenž reaguje při změně velikosti okna na definovanou hodnotu velikosti okna, při které je pak možné změnit různá nastavení, jako je velikost okna, změna zarovnání atd. Komponenta Visual StateManager se přidá pomocí nástroje Blend v záložce States (**MainPage.xaml » Design in Blend . . . » States**). Při zadávání vstupních dat používá aplikace komponenty: **TextBox** pro zadání obsahu a **Slider** pro nastavení hodnoty v určitém povoleném rozmezí. Při výpočtu se používá komponenty **Button**, jež vyvolá funkci pro načtení vstupních dat, provede výpočty a dané hodnoty vykreslí do komponent **TextBlock**. Pro použití aplikace na mobilních zařízeních je přidána komponenta **ScrollView**, která se stará o zobrazení posuvné lišty, pokud se aplikace nevejde celá na obrazovku.



Obrázek 12 – Nastavení aplikace a přidání responzivního designu (Zdroj: vlastní)

Hlavní výpočty jsou pro svou složitost umístěny ve složce Helpers, kde se nachází z převážné části statické třídy pro výpočty i převody IP adresy a jejích součástí.

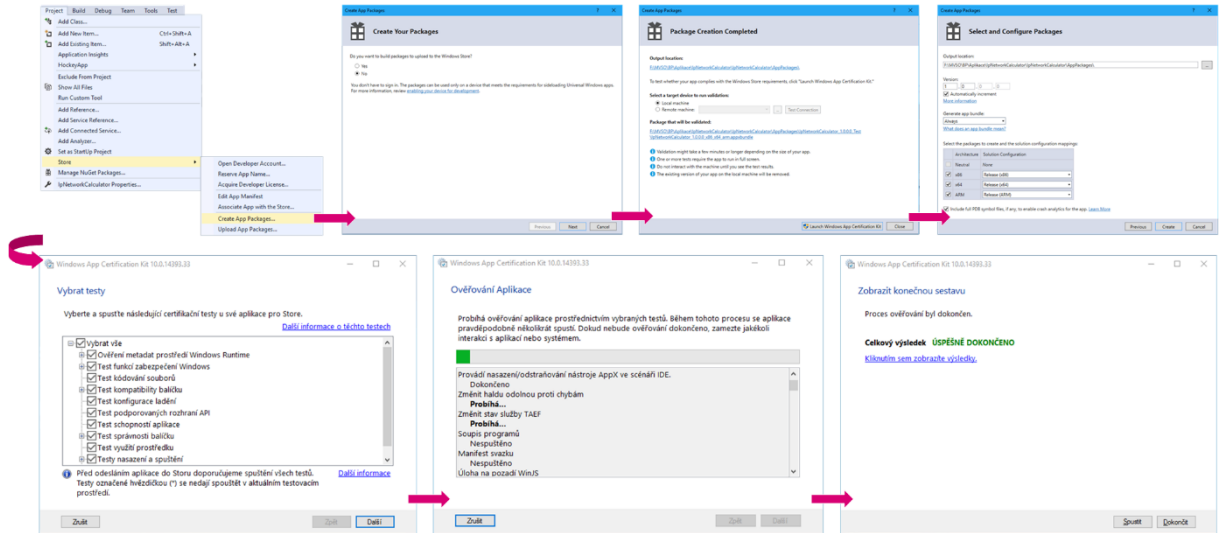
Aplikace užívá pouze jedné stránky a to MainPage, která je rozdělena na část grafickou (*.xaml - XAML) a část logickou (*.xaml.cs – C#). Aplikační logika využívá třídy ze složky Helpers v **MainPage.xaml.cs**, kde je definováno chování všech ovládacích prvků aplikace. Tyto ovládací prvky byly přidány v **MainPage.xaml** pomocí nástroje Visual Studio Blend a výše popsaných komponent jazyka XAML.

Visual Studio obsahuje základní verzi nástroje Blend pro editaci všech xaml souborů. Jedinou výjimku, kdy se musí zapnout plné prostředí Blend, vidíme při přidání komponenty pro responzivní design **VisualStateManager** na Obrázku 12. Přidali jsme 3 triggrы nazvané Phone, Tablet a Desktop. Tablet má nastavenou minimální šířku pro spuštění na 680 px (změní velikost písma nadpisu na 36), Desktop pak 1024 px (změní velikost písma nadpisu na 72 a zároveň položky výpočtů vedle sebe).

V záložce Assets souboru Package.appxmanifest je potřeba přidat všechny povinné ikony a Loga, které Windows Store vyžaduje. Aplikace VS si povinné prvky sama hlídá. Ikony vhodné velikosti nahrajeme pomocí záložky Assets a nepřidáváme obrázky přímým vložením do složky Assets v projektu VS. Všechny ikony vytvoříme ve vektorovém grafickém editoru Inkscape.

3.5 Vytvoření Windows Store balíčku a ověření pomocí WACK

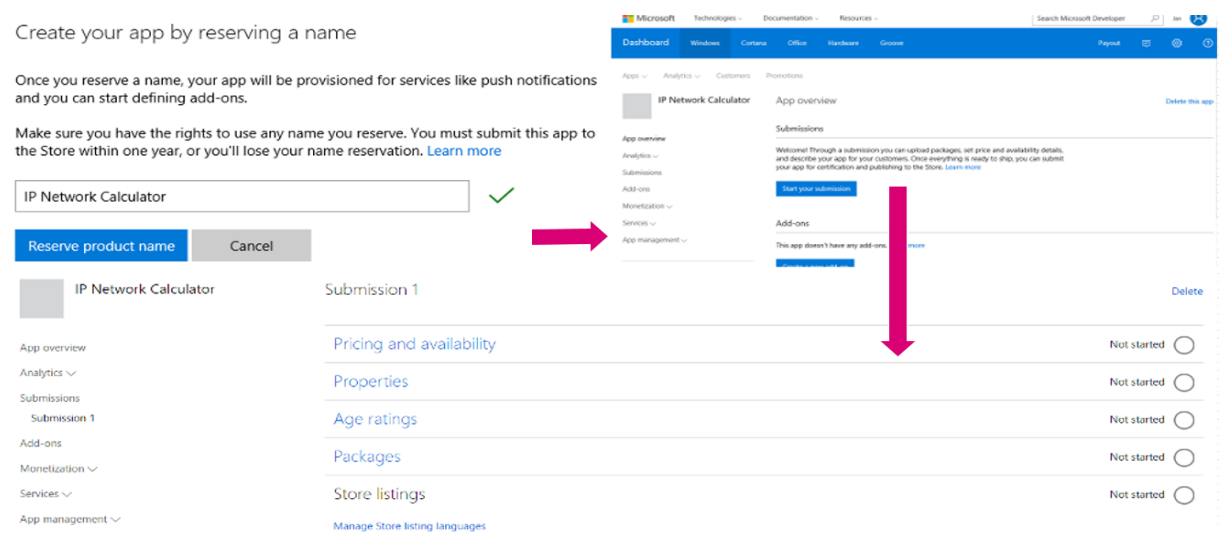
Po úspěšném napsání aplikace a jejím otestování nás čeká další krok v podobě vytvoření Windows Store balíčku, který je potřeba pro otestování naší aplikace pomocí Windows App Certification Kit, který ověří, zda daná aplikace splňuje všechny požadavky pro nahrání na Microsoft Windows Store. Po vytvoření balíčku se systém automaticky zeptá, zda jej chcete ověřit pomocí WACK.



Obrázek 13 – Vytvoření Windows Store balíčku a ověření aplikace pomocí WACK (Zdroj: vlastní)

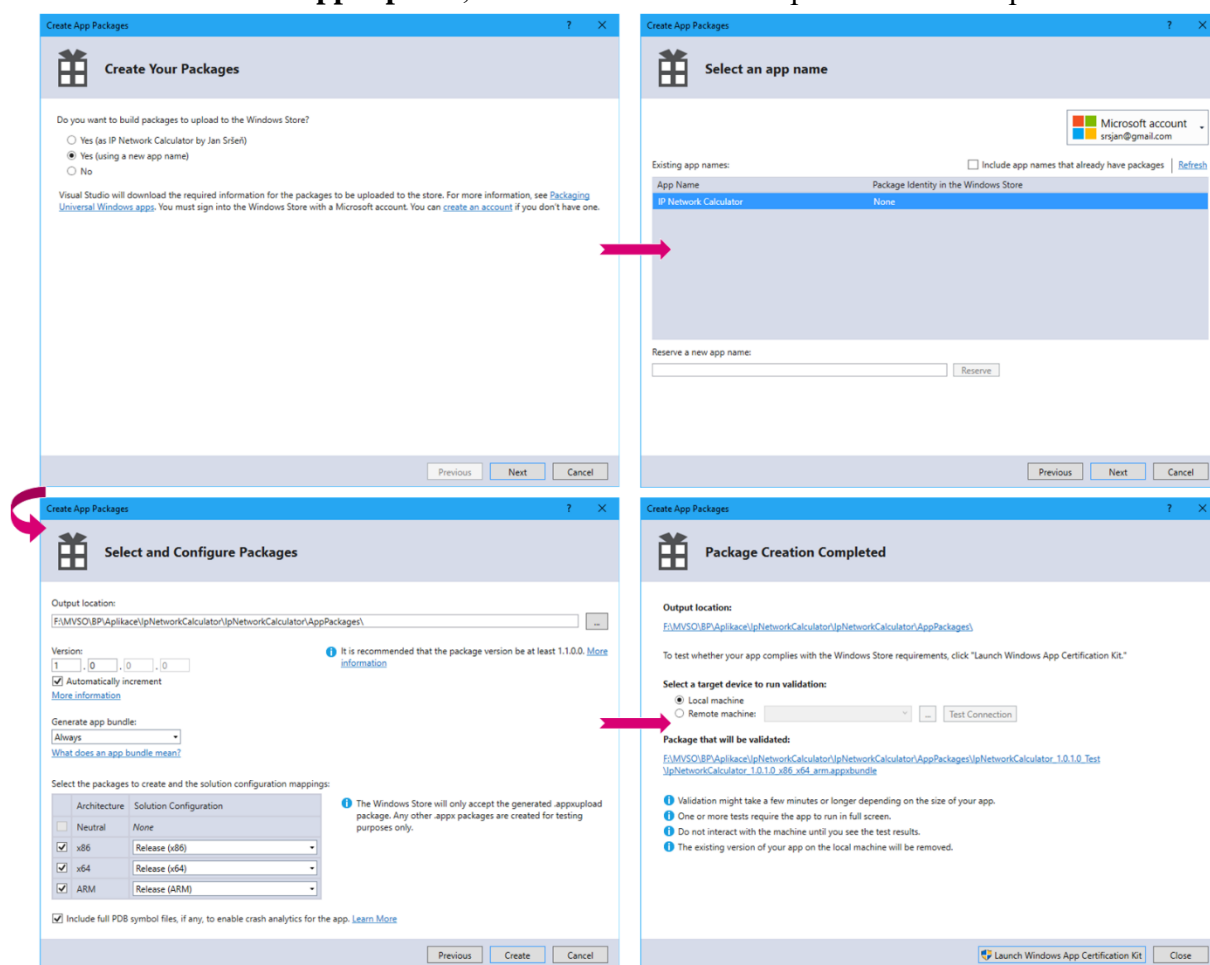
3.6 Nahrání aplikace na Windows Store

Registraci názvu aplikace lze zarezervovat i před tím, než máme samotnou aplikaci hotovou. Stačí si úspěšně zaregistrovat vývojářský účet a hned vám bude nabídnuta možnost rezervovat si jméno aplikace. Jméno aplikace lze ověřit, nesmí se samozřejmě shodovat s již rezervovaným nebo použitým názvem. Rezervovat jméno aplikace lze na WWW: <<https://developer.microsoft.com/en-us/dashboard/Application/New>>.



Obrázek 14 – Registrace názvu aplikace ve Windows Store (Zdroj: vlastní)

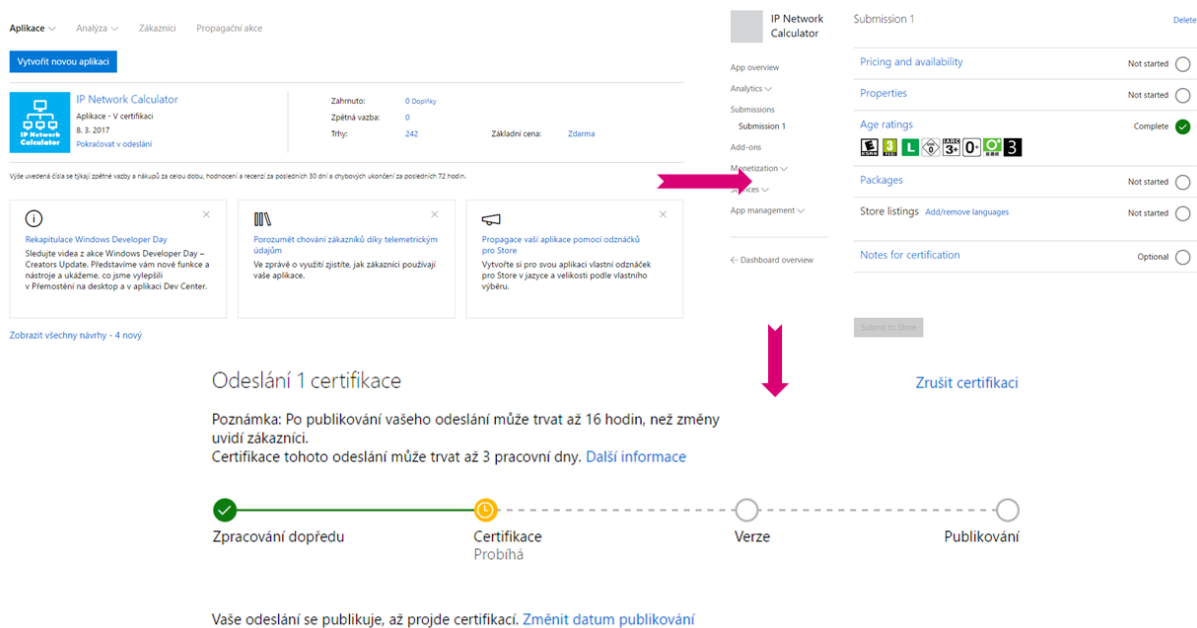
Po otestování aplikace ve VS a jeho nástrojích, otestování Store balíčku pomocí WACK, můžeme nyní vytvořit App Store balíček určený pro nahrání na Windows Store, tzv. produkční nebo Release verzi aplikace. Postup je téměř totožný jako v kapitole 3.5 jen s tím rozdílem, že vybereme možnost sestavit balíčky pro nahrání na Windows Store. Pokud jste aplikaci již jednou takto sestavili, nabídne vám VS možnost sestavit balíček znova pro konkrétní název aplikace, který již byl asociován s názvem aplikace zaregistrovaným na serveru Windows Store. My vybereme sestavit s použitím nového jména aplikace a v dalším kroku vám VS nabídne vybrat jeden z již registrovaných názvů (registrovaných na Store) nebo zaregistrovat pro aplikaci nový název. V posledním kroku potvrdíme, pro jaké platformy je aplikace určena, kam se balíček nahraje a o jakou verzi aplikace se jedná. Po potvrzení se vytvoří balíček obsahující kód pro všechny vybrané platformy, určený pro nahrání na Windows Store s koncovkou ***bundle.appxupload**, zobrazí se též možnost aplikaci otestovat pomocí WACK.



Obrázek 15 – Vytvoření balíčku pro nahrání na Windows Store (Zdroj: vlastní)

Druhým krokem po vytvoření balíčku je přihlásit se na svůj Windows Developer účet, vybrat aplikaci, pro kterou jsme si rezervovali jméno a asociovali jsme ji s balíčkem pro Store ve VS. Dále musíme vyplnit všechny potřebné údaje u položek, které obsahuje odkaz Submission (či Odeslání), jako je věková dostupnost, nezávadnost aplikace, dostupné jazyky, popis a obrázky aplikace pro Windows Store. A hlavně nahrát balíček (*bundle.appxupload) na web v záložce Packages (Balíčky); po nahrání balíčku se vám zpřístupní možnost u Store

listings (Popisy Storu), kde se automaticky přidá defaultní jazyk aplikace a je též možné přidat i další jazyky. Další políčka pro vyplnění jsou popis aplikace a nahrání ikon pro různé platformy.



Obrázek 16 – Nahrání aplikace na Windows Store a Publikování (Zdroj: vlastní)

Po nahrání aplikace na Windows Store je potřeba u aplikace vyplnit URL odkaz Privacy Policy (zásady ochrany osobních údajů) i v případě, že aplikace nepoužívá nic ze záložky Capabilities, neodesílá žádná data ani neshromažďuje informace o uživateli. Též je dobré vyplnit klíčová slova, podle kterých budoucí uživatel danou aplikaci snadněji ve vyhledávání Windows Store najde. Pokud je již vše řádně vyplněno, stačí stisknout v záložce Submission pod seznamem všech požadavků na aplikaci tlačítko Submit to Store (Odeslat do Storu). Po tom, co aplikaci odešleme do Storu, musíme počkat 2-3 pracovní dny na její schválení. Aplikace se buď úspěšně schválí a po schválení se do 24 hodin objeví ve Windows Store, nebo vám přijde upozornění s odůvodněním, proč nebyla schválena. Přesto i v prvním úspěšném případě přijdou vývojáři další pokyny s postupem na e-mail.

Jakmile dojde ke schválení, stačí jednoduše otevřít Windows Store a napsat do vyhledávání Ip Network Calculator. Na tomto odkazu jsou o aplikaci uvedeny již všechny podrobnosti. A dále po kliknutí na tlačítko Získat se aplikace automaticky nainstaluje. Aplikace je dostupná pro architektury procesorů ARM, x86 a x64 a k dispozici pouze pro OS založený na W10. Pro mobilní platformu i normální zařízení je Windows Store téměř totožný.

Windows Store též najdete online a při vyhledání jakékoliv aplikace se otevře jak webová stránka, tak i aplikace Store v daném zařízení.

Aplikace Ip Network Calculator je dostupná na WWW: <https://www.microsoft.com/cs-cz/store/p/ip-network-calculator/9npnkmxg4ft7>

Your submission for your app IP Network Calculator passed certification.

It will take up to 24 hours for your submission to be visible to customers in the Store.

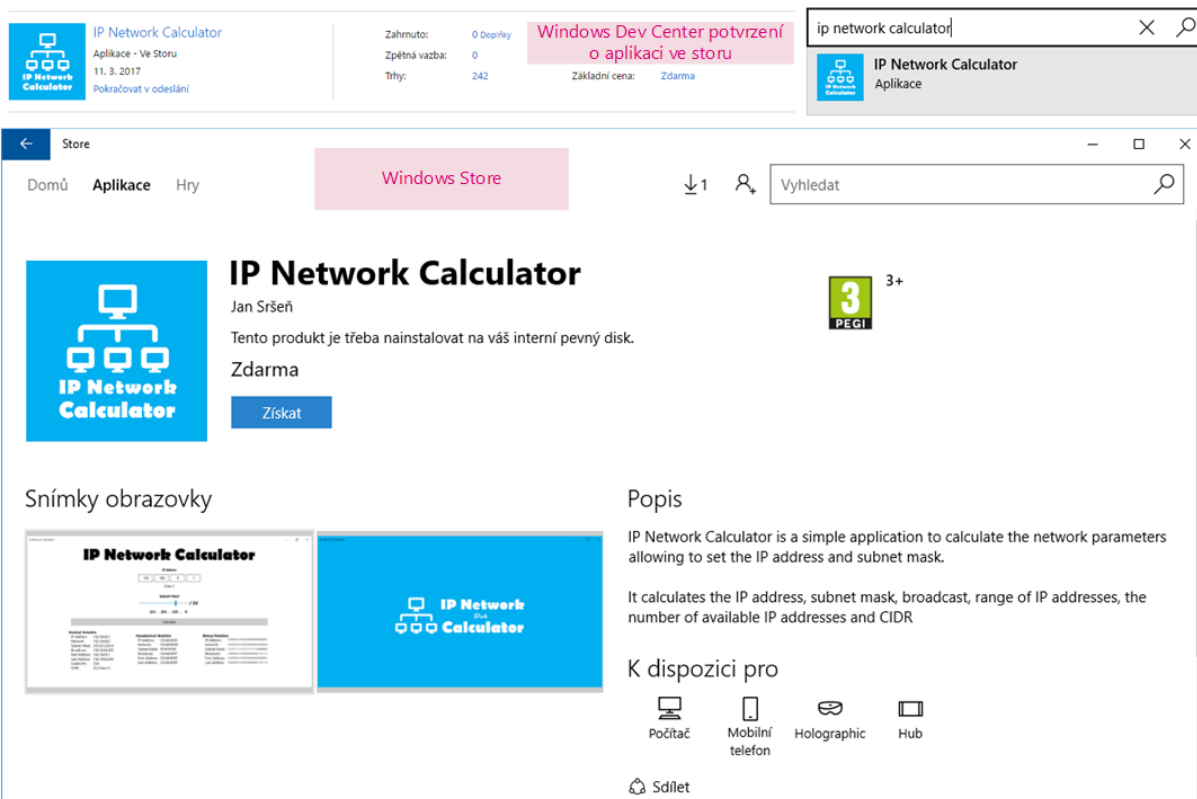
Thank you,
The Windows Store Team

Your submission for your app IP Network Calculator needs some attention.

We reviewed your submission for your app and found something that could use your attention. Please look over our comments in your [report](#) and resubmit the submission.

You'll need your Microsoft account information to access this report.

Thank you,
The Windows Store Team



Obrázek 17 – Schválená aplikace (Zdroj: vlastní)

Update (aktualizace) aplikace se provádí jednoduše. Stačí upravit aplikaci, vygenerovat pro ni balíček pro App Store, který má vyšší číslo než předcházející verze (VS samo inkrementuje verze aplikace – např.: z verze 1.0.0 na 1.0.1) a u Submission kliknout na tlačítko Update. Otevře se okno s aktuální verzí aplikace a všemi jejími vyplněnými položkami (Submission se automaticky inkrementuje). Poté stačí kliknout na odkaz Package a nahrát novou verzi, popřípadě doopravit i některé další potřebné informace v záložkách. Následně se akce potvrdí tlačítkem Submit to Store a nová verze se pošle znovu na schválení, přičemž aktuální schválená verze (Submission 1) bude na Windows Store dostupná do té doby, než ji nahradí novější (Submission 2). U starších verzí aplikace Vás Windows Store upozorní na jejich přebytečnost, jelikož není možné, aby si uživatel stáhnul starší verzi před aktuálně dostupnou.

ZÁVĚR

Cílem práce bylo seznámit čtenáře s problematikou vývoje aplikací pro chytré mobilní telefony založené na technologii Windows. Tuto problematiku jsme rozdělili na několik teoretických částí a jednu praktickou část s ukázkou vývoje, a to především distribucí aplikace s názvem Ip Network Calculator, jež demonstruje námi nabyté znalosti v této problematice. Informace získané během psaní této práce byly především ze zahraničních zdrojů v anglickém jazyce.

Teoretickou část jsme rozdělili na dvě podkapitoly. První část seznamuje čtenáře se současnou situací na poli mobilních operačních systémů a s podrobnou historií verzí i možností vývoje mobilních operačních systémů od společnosti Microsoft. Druhá část se již podrobněji věnuje základům vývoje, požadavkům na vývoj a seznámení s architekturou posledních verzí mobilního OS od verze 7.5 až po současnou verzi 10.

Napříč bakalářskou prací se především prakticky zaměřujeme na poslední technologii s názvem UWP, která přinesla velké možnosti vývoje aplikací skrze celou rodinu produktů Microsoft, a to právě proto, že v době psaní naší práce byla uvolněna právě ona nová technologie UWP, kterou jsme se následně rozhodli použít pro demonstraci aplikace, jež je napsaná tak, aby ji bylo možné používat jak na mobilních zařízeních, tak i na pracovních stanicích. Aplikace demonstruje jednoduchost a základní principy vývoje i distribuce aplikací pomocí této platformy. Pro vývoj aplikace jsme zvolili programovací jazyk C# pro psaní výpočtů a značkovací jazyk XAML pro prvky a design aplikace.

Do budoucna bychom chtěli aplikaci rozšířit o nové možnosti výpočtů, vylepšit design a umožnit uživateli větší komfort pro práci s touto aplikací.

I když je mobilní platforma Windows v současnosti jednou z nejméně používaných, přináší zajímavé možnosti vývoje pro psaní rychlých aplikací s velkým výběrem pro zaplnění chybějících aplikací na oficiálním distribučním kanálu společnosti Microsoft. Největším problémem je, alespoň z našeho pohledu, častá změna principů vývoje aplikací a nemožnost přenášet aplikace mezi novými verzemi systému, které musí programátor složitě přepisovat a uživatel dlouho čekat na uvolnění aplikace za předpokladu, že ji vůbec někdo napíše či si uživatel koupí nový telefon.

Hlavním přínosem této práce je získání nových znalostí a seznámení se s různými možnostmi vývoje, a v neposlední řadě distribuce aplikací pro Windows Store. Osvojení si základních dovedností procesu vývoje aplikací a možností, které přináší.

ANOTACE

Příjmení a jméno autora:	Sršeň Jan
Instituce:	Moravská vysoká škola Olomouc, o. p. s.
Název práce v českém jazyce:	Tvorba aplikací pro chytré mobilní telefony se systémem Windows
Název práce v anglickém jazyce:	Development of Applications for Windows Based Smartphones
Vedoucí práce:	Mgr. Jiří Martinů
Počet stran:	52
Počet příloh:	2
Rok obhajoby:	2017

Klíčová slova v české jazyce

vývoj aplikací, Windows Phone, Windows Store, XAML, Visual Studio

Klíčová slova v anglickém jazyce

application development, Windows Phone, Windows Store, XAML, Visual Studio

Anotace v českém jazyce

Bakalářská práce je zaměřena na oblast vývoje aplikací pro mobilní zařízení s operačním systémem Windows. Uvádí historii těchto operačních systémů i zařízení, na kterých byly tyto mobilní operační systémy postupem času provozovány. Představuje prvky potřebné pro vývoj a testování plnohodnotných aplikací, které lze publikovat přes oficiální distribuční kanál Microsoft Store. Práce představuje základní filosofii ovládání a designu aplikací spolu s aplikací programových principů nutných pro flexibilní chod aplikace. Samotná aplikace je implementována pomocí vývojového nástroje Microsoft Visual Studio a shlukuje nabyté informace a znalosti o vývoji těchto aplikací v systému pro výpočet parametrů sítě, který vznikl jako demonstrace praktické ukázky, která je plně funkční.

Anotace v anglickém jazyce

Bachelor thesis is focused on the development of applications for mobile devices running on Windows OS. It presents the history of these operating systems and devices on which these mobile operating systems over time were operated. It represents the elements needed for development and testing full-featured applications that can be published via the official Microsoft Store distribution channel. Work is the basic philosophy of control and design applications with applications programming principles necessary for the flexible operation of the application. The application itself is implemented using development tools of Microsoft Visual Studio and coalesces acquired informations and knowledges about the development of these applications in the system for calculating the parameters of the network, which was created as a demonstration of practical example, which is fully functional.

SEZNAM ZKRATEK A POJMŮ

- .NET (.NET Framework)** – základní platforma knihoven pro běh aplikací a jejich vývoj v C++
- .NET CF (.NET Compact Framework)** – je .NET Framework pro mobilní zařízení
- ALT** – Active Template Library, poskytuje šablony tříd a konstruktů pro snadnější vývoj v C++
- API** – Application Programming Interface (rozhraní pro programování aplikací)
- ARM** – rodina procesorů s nízkou energetickou náročností určených do malých zařízení
- BIOS (Basic Input-Output System)** – je SW uložený na základní desce PC v paměti Flash a slouží pro nastavení základních funkcí HW a poskytuje základní rozhraní pro tento HW
- C++** – programovací OOP jazyk
- CIL (Common Intermediate Language)** – programovací jazyk nezávislý na procesoru a platformě
- CLI (Common Language Infrastructure)** – je specifikace společného mezikódu (CIL) a popis běhového prostředí (CLR)
- CLR (Common Language Runtime)** – běhové prostředí pro aplikace napsané v .NET jazycích
- eVB** – eMbedded Visual Basic, vývojové prostředí pro Visual Basic od MS (zakomponován do VS)
- eVC** – eMbedded Visual C++, vývojové prostředí pro C, C++ od MS (zakomponován do VS)
- GB** – (GigaByte) Gigabajt je jednotka velikosti, 1 GB = 10⁹ B (Bajt – Byte)
- HW** – Hardware (fyzické vybavení počítače – klávesnice, myš, monitor, procesor)
- IDE** – vývojové prostředí, ve kterém se programují a ladí aplikace
- IE** – Internet Explorer (webový prohlížeč os společnosti Microsoft)
- Managed Code** – ze zdrojového kódu (C#, VB.NET) se přeloží do tzv. mezikódů (CIL, tento mezikód je definován pomocí CLI), který je možné spouštět pomocí běhového prostředí (CLR)
- MFC** – Microsoft Foundation Class, OOP knihovna postavená nad Win32 pro vývoj v C++
- MS** – Microsoft (Microsoft Corporation)
- MWS (Microsoft Windows Simulator)** – emulátor Windows tabletu
- Native Code** – kompiluje se přímo pro danou platformu ze zdrojového kódu (C, C++) do strojového kódu (daný typ procesoru)
- OOP** – Object-oriented programming (Objektově Orientované programování)
- OS** – Operační systém (základní softwarové vybavení počítače, hlavní řídicí program)
- RAM** – operační paměť počítače
- SDK** – software development kit, sada vývojových nástrojů pro vývoj aplikací
- Silverlight** – vývojový nástroj od MS pro tvorbu webových a UI prostředí
- SP (Service Pack)** – je balíček obsahující opravy, aktualizace a vylepšení
- Stylus** – pero pro ovládání dotykové obrazovky
- SW** – Software (všechny systémové aplikace a programy)
- UI** – User Interface (uživatelské prostředí)
- UWP** – Universal Windows Platform API
- VS** – Visual Studio, integrované vývojové prostředí od společnosti Microsoft

W – Windows (operační systém od společnosti Microsoft)

Win32 (WinAPI) – je API pro 32bitové systémy, běžící pod Windows

Windows CE – Windows Embedded Compact je OS pro malá zařízení

Windows Store – oficiální obchod pro distribuci aplikací napsaných hlavně ve WinRT od společnosti Microsoft. Windows Store nahradil v roce 2008, Windows Marketplace, který sloužil k distribuci desktopových a později i mobilních aplikací.

WinRT (Windows Runtime) – je API pro vývoj aplikací založených na Metro UI

WM – Windows Mobile

WP – Windows Phone (WP číslo verze), např. WP 7.x znamená všechny verze WP 7 (WP 7, WP 7.5, 7.8)

WPE (Windows Phone Emulator) – emulátor Windows Phone (či Windows 10 Mobile) zařízení

WPRT (Windows Phone Runtime) – verze WinRT určená pro použití na chytrých telefonech

x86 – rodina procesorů s 32 bitovou architekturou, převážně určená pro použití v PC

XAML – deklarativní značkovací jazyk pro definici UI ve WinRT aplikacích, obdoba HTML a XML

SEZNAM LITERATURY

➤ Bibliografie

ATLANTIC. *Encyclopedia of information technology*. New Delhi: Atlantic, 2007, 816 s. ISBN 9788126907526.

BUDDY, James a Lori LALONDE. *Pro XAML with C#: from design to deployment on WPF, Windows Store, and Windows Phone*. New York: Apress, 2015, 269 s. ISBN 978-1-4302-6776-8.

GORDON, Alan. *The .NET and COM interoperability handbook*. Upper Saddle River, N.J.: Prentice Hall PTR, 2003, 720 s. ISBN 013046130X.

JANGLA, Kinnary. *Windows 10 revealed: the universal Windows operating system for PC, tablets, and Windows phone*. New York, N.Y.: Apress, 2015, 96 s. Expert's voice in Windows. ISBN 978-1-4842-0687-4.

KUMAR, Senthil, Lohith Goudagere NAGARAJ, Pathik RAWAL a Pryank ROHILLA. *Windows 10 development recipes: a problem-solution approach in HTML and JavaScript*. Berkeley, CA: Apress, 2016, 373 s. ISBN 1484207211.

LACKO, Ľuboslav. *Vývoj aplikací pro Windows 8.1 a Windows Phone*. Brno: Computer Press, 2014, 330 s. ISBN 978-80-251-3822-9.

LACKO, Ľuboslav. *Vývoj aplikácií pre Windows Phone 7* [online]. Praha: Microsoft, 2011, 84 s. [cit. 2017-01-19]. Dostupné z: <https://technet.microsoft.com/cs-cz/dd727769.aspx>

LACKO, Ľuboslav. *Vývoj aplikácií pre mobilné zariadenia* [online]. Praha: Microsoft, 2003, 121 s. [cit. 2017-01-19]. Dostupné z: http://download.microsoft.com/download/8/6/c/86c09926-affc-4e14-bec0-3c45cd989436/vyvoj_aplikacii_pre_mobilne_zariadenia.pdf

LALONDE, Lori a David R. TOTZKE. *Windows Phone 8 recipes: a problem-solution approach*. New York, N.Y.: Apress, 2013, 428 s. ISBN 978-1-4302-5902-2.

LEE, Henry a Eugene CHUVYROV. *Beginning Windows phone app development*. 3rd ed. New York, N.Y.: Apress, 2012, 548 s. ISBN 1430241349

LEE, Wei-Meng. *.NET Compact Framework pocket guide*. Sebastopol, CA: O'Reilly, 2004, 111 s. ISBN 0596007574.

MAYBERRY, Michael. *WinRT revealed*. Berkeley, CA: Apress, 2012, 84 s. ISBN 9781430245841.

MOEMEKA, Edward a Elizabeth MOEMEKA. *Real world Windows 10 development*. 2nd. New York: Apress, 2015, 607 s. ISBN 978-1484214503.

SOFTWARE, Falafel. *Pro Windows phone app development*. 3rd ed. New York, N.Y.: Apress, 2013, 560 s. ISBN 9781430247821.

SZOSTAK, Tomasz. *Windows Phone 8 Application Development Essentials: A practical guide to creating a Windows Phone 8 application using C#, XAML, and MVVM*. Birmingham: Packt Publishing, Limited, 2013, 118 s. ISBN 9781849696760.

WHITECHAPEL, Andrew a Sean MCKENNA. *Windows Phone 8 development internals*. Microsoft: Microsoft Press, 2013, 1044 s. ISBN 0735676232.

➤ **Internetové zdroje**

HILL, Simon. *Which smartphone OS wins 2016? Android Nougat vs. iOS 10 vs. Windows 10 Mobile*. In: Digital Trends [online]. 2016 [cit. 2016-12-22]. Dostupné z: <http://www.digitaltrends.com/mobile/best-smartphone-os/>

MICROSOFT. *Časté otázky k programu pro vývojáře*. In: Microsoft Developer [online]. 2016 [cit. 2017-01-08]. Dostupné z: <https://developer.microsoft.com/cs-cz/store/register/faq>

MICROSOFT. *Windows Store for Business overview*. Microsoft [online]. Microsoft: Microsoft, 2017 [cit. 2017-03-14]. Dostupné z: <https://technet.microsoft.com/itpro/windows/manage/windows-store-for-business-overview#Supported-markets>

RUBINO, Daniel. *Windows Phone 8.1 Review*. In: Windowscentral [online]. 2014 [cit. 2016-12-22]. Dostupné z: <http://www.windowscentral.com/windows-phone-81-review>

STRUYS, Maarten. *Windows Mobile App Development: Part 1: Creating your first application*. In: Code PROJECT [online]. 2009 [cit. 2016-12-25]. Dostupné z: <https://www.codeproject.com/Articles/43357/Windows-Mobile-App-Development-Part-Creating-you>

SUBRAMANIAN, Shiva. *Evolution of Windows Phone OS and the transition from Windows Mobile*. In: Techglimpse [online]. 2013 [cit. 2016-12-18]. Dostupné z: <https://techglimpse.com/history-microsoft-windows-phone-os-wp/>

TABOR, Bob. Part 12 - Understanding the App's Lifecycle and Managing State. In: *Channel 9: Windows Phone 8.1 Development for Absolute Beginners* [online]. Microsoft: Microsoft, 2014 [cit. 2017-01-30]. Dostupné z: <https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-12-Understanding-the-App-s-Lifecycle-and-Managing-State>

TABOR, Bob. Part 19 - Understanding async and Awaitable Tasks. In: *Channel 9: Windows Phone 8.1 Development for Absolute Beginners* [online]. Microsoft: Microsoft, 2014 [cit. 2017-01-25]. Dostupné z: <https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-19-Understanding-async-and-Awaitable-Tasks>

TABOR, Bob. Part 22 - Storing and Retrieving Serialized Data. In: *Channel 9: Windows Phone 8.1 Development for Absolute Beginners* [online]. Microsoft: Microsoft, 2014 [cit. 2017-01-24]. Dostupné z: <https://channel9.msdn.com/Series/Windows-Phone-8-1-Development-for-Absolute-Beginners/Part-22-Storing-and-Retrieving-Serialized-Data>

WIGLEY, Andy a Jerry NIXON. *Windows 10 - An Introduction to Building Windows Apps for Windows 10 Devices*. In: MSDN Microsoft [online]. Microsoft, 2015 [cit. 2016-12-22]. Dostupné z: <https://msdn.microsoft.com/en-us/magazine/dn973012.aspx>

YAO, Paul. *Choosing a Windows Embedded API: Win32 vs. the .NET Compact Framework: Windows CE .NET*. In: MSDN Microsoft [online]. 2002 [cit. 2016-12-23]. Dostupné z: <https://msdn.microsoft.com/en-us/library/ms836774.aspx>

SEZNAM OBRÁZKŮ A TABULEK

➤ **Tabulky**

Tabulka 1 – Porovnání mobilních operačních systémů	7
Tabulka 2 – Možnosti vývoje Windows Store aplikací ve VS 2015	21
Tabulka 3 – Typy úložišť pro Windows Phone aplikace	27

➤ **Obrázky**

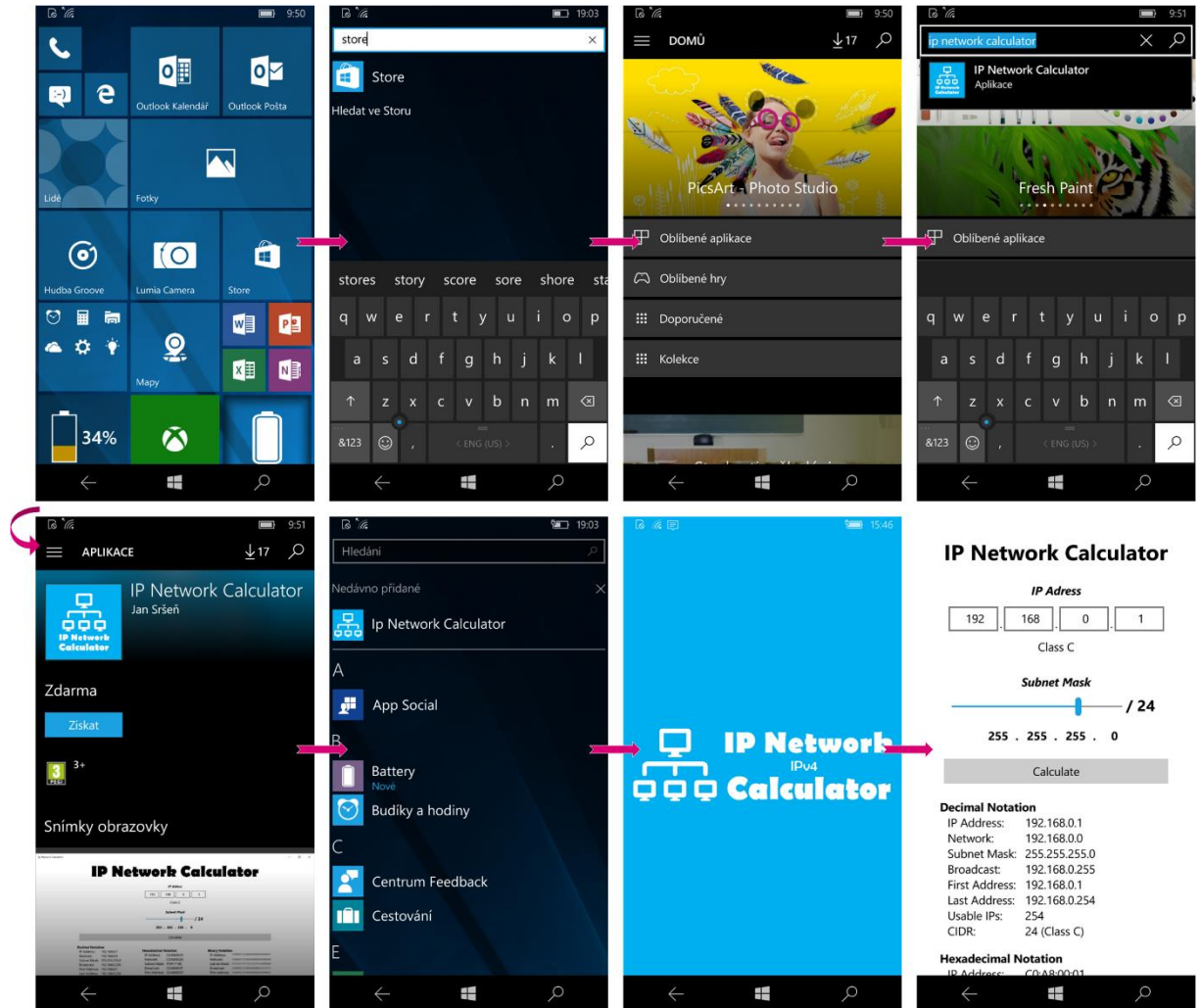
Obrázek 1 – Historie UI u Windows mobilních zařízení	12
Obrázek 2 – Přehled Windows mobilní platformy od WP7	15
Obrázek 3 – Architektura OS Windows 8.1 a 10.....	17
Obrázek 4 – Životní cyklus aplikace od Windows Phone 7.5 až 8.....	18
Obrázek 5 – Životní cyklus aplikace od Windows Phone 8.1	19
Obrázek 6 – Grafické rozložení prvků a Windows Phone Emulator	23
Obrázek 7 – Vzor MVVM	25
Obrázek 8 – Instalace aplikace Visual Studio 2015	30
Obrázek 9 – Základní prvky aplikace VS a jeho nástrojů	31
Obrázek 10 – Zapnutí Vývojářského režimu a povolení Virtualizace	32
Obrázek 11 – Registrace Windows Developer Account	33
Obrázek 12 – Nastavení aplikace a přidání responzivního designu	34
Obrázek 13 – Vytvoření Windows Store balíčku a ověření aplikace pomocí WACK	36
Obrázek 14 – Registrace názvu aplikace ve Windows Store	36
Obrázek 15 – Vytvoření balíčku pro nahrání na Windows Store	37
Obrázek 16 – Nahrání aplikace na Windows Store a Publikování	38
Obrázek 17 – Schválená aplikace	39

SEZNAM PŘÍLOH

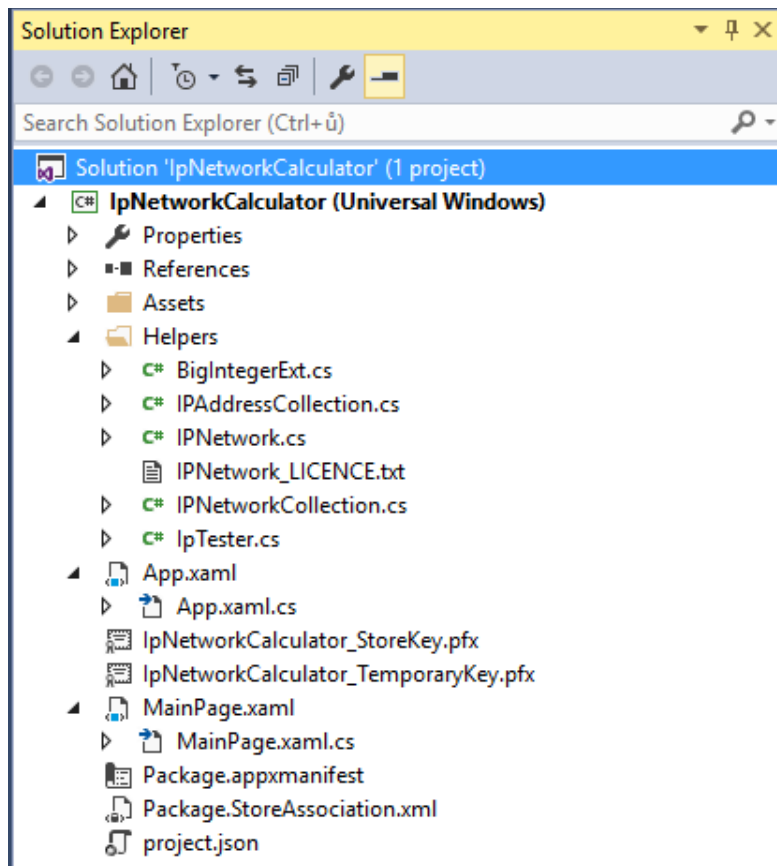
Příloha 1 – Instalace aplikace Ip Network Calculator na zařízení Microsoft Lumia 535	.49
Příloha 2 – Struktura projektu IpNetworkCalculator	50
Příloha 3 – Microsoft Blend for Visual Studio 2015	50
CD s obsahem souborů použitých při práci	

PŘÍLOHY

Příloha 1 – Instalace aplikace Ip Network Calculator na zařízení Microsoft Lumia 535



Příloha 2 – Struktura projektu IpNetworkCalculator



Příloha 3 – Microsoft Blend for Visual Studio 2015

