

Katedra informatiky  
Přírodovědecká fakulta  
Univerzita Palackého v Olomouci

# DIPLOMOVÁ PRÁCE

Srovnání frameworků pro multiplatformní mobilní vývoj



2016

Vedoucí práce: RNDr. Jan Konečný, Ph.D.

Bc. Stanislav Šejnoha

Studijní obor: Informatika, prezenční forma

## **Bibliografické údaje**

Autor: Bc. Stanislav Šejnoha  
Název práce: Srovnání frameworků pro multiplatformní mobilní vývoj  
Typ práce: diplomová práce  
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci  
Rok obhajoby: 2016  
Studijní obor: Informatika, prezenční forma  
Vedoucí práce: RNDr. Jan Konečný, Ph.D.  
Počet stran: 63  
Přílohy: 1 CD/DVD  
Jazyk práce: český

## **Bibliographic info**

Author: Bc. Stanislav Šejnoha  
Title: Comparison of frameworks for cross platform mobile development  
Thesis type: master thesis  
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc  
Year of defense: 2016  
Study field: Computer Science, full-time form  
Supervisor: RNDr. Jan Konečný, Ph.D.  
Page count: 63  
Supplements: 1 CD/DVD  
Thesis language: Czech

## Anotace

*Trh s mobilními zařízeními nyní většinou ovládají tři výrobci operačních systémů. V tomto segmentu je tak aktuálním trendem a zároveň logickým posunem multiplatformní vývoj aplikací na tato zařízení. Práce se zabývá porovnáním multiplatformního a nativního vývoje a především zpracováním přehledu společností, které nabízejí služby a nástroje pro multiplatformní mobilní vývoj. Text by měl pomoci vývojáři ke zvolení vhodné společnosti, vyhovující jeho konkrétním požadavkům. Z přehledu společností byly vybrány dvě konkrétní a na nich následně byla provedena důkladnější analýza. Frameworky Xamarin Forms a Apache Cordova, které tyto společnosti nabízejí, byly dále použity pro vývoj vzorové mobilní aplikace sloužící pro evidenci nalezených a ztracených zvířat. V závěru práce byly zhodnoceny jednotlivé aspekty vývoje aplikace pomocí těchto vybraných frameworků.*

## Synopsis

*The majority of market for mobile devices is now controlled by three manufacturers of operating systems. So in this segment the current trend and also the logical shift is the cross-platform application development for these devices. Thesis compares cross-platform and native development, creating survey of companies that offer services and tools for cross-platform mobile development. The text should help developers to select the appropriate company, complying with the specific requirements. From the survey of companies were selected two and subsequently detail analysis have been created. Frameworks Xamarin Forms and Apache Cordova, that these companies offer, were used for the development of a model mobile application which was made for recording of found and lost animals. In conclusion of thesis there were assessed various aspects of application development using those selected frameworks.*

**Klíčová slova:** mobilní vývoj; multiplatformní; Xamarin; Cordova

**Keywords:** mobile development; cross-platform; Xamarin; Cordova

Děkuji Janu Konečnému, za vedení této diplomové práce a Petru Mohylovi z firmy GMC Software Technology s.r.o., za její poutavé zadání a následné konzultace k jejímu řešení.

*Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.*

datum odevzdání práce

podpis autora

# Obsah

<b>1</b>	<b>Úvod</b>	<b>8</b>
<b>2</b>	<b>Vývoj nové aplikace</b>	<b>10</b>
2.1	Současné možnosti . . . . .	10
2.2	Výběr platformy . . . . .	10
2.2.1	Zastoupení platform . . . . .	10
2.2.2	Nativní vývoj pro Android . . . . .	10
2.2.3	Nativní vývoj pro iOS . . . . .	12
<b>3</b>	<b>Multiplatformní vývoj</b>	<b>13</b>
3.1	Trend multiplatformního vývoje . . . . .	13
3.2	Gartner Magic Quadrant . . . . .	15
3.2.1	Lídři . . . . .	15
3.2.1.1	Adobe Systems . . . . .	15
3.2.1.2	IBM . . . . .	17
3.2.1.3	Kony . . . . .	18
3.2.1.4	Salesforce.com . . . . .	19
3.2.2	Vyzyvatelé . . . . .	19
3.2.2.1	Microsoft . . . . .	20
3.2.2.2	Progress Software . . . . .	21
3.2.3	Vizionáři . . . . .	22
3.2.3.1	Appcelerator . . . . .	22
3.2.3.2	Backbase . . . . .	24
3.2.3.3	Pegasystems . . . . .	25
3.2.3.4	Sencha . . . . .	25
3.2.3.5	Xamarin . . . . .	26
3.2.4	Specializovaní hráči . . . . .	27
3.2.4.1	ClickSoftware . . . . .	27
3.2.4.2	DSI . . . . .	28
3.2.4.3	Embarcadero Technologies . . . . .	29
3.2.4.4	MicroStrategy . . . . .	30
3.2.4.5	Oracle . . . . .	30
3.2.4.6	Verivo Software . . . . .	31
3.2.4.7	Zebra Solutions . . . . .	32
3.3	Srovnání . . . . .	33
3.3.1	Informativní aplikace pro potenciální zákazníky . . . . .	34
3.3.2	Aplikace pro zadávání docházky zaměstnanců . . . . .	35
<b>4</b>	<b>Xamarin</b>	<b>37</b>
4.1	Historie . . . . .	37
4.2	Technologické možnosti . . . . .	37
4.3	Uživatelská základna . . . . .	38
4.4	Potenciální rozvoj . . . . .	39

<b>5</b>	<b>Apache Cordova</b>	<b>40</b>
5.1	Historie . . . . .	40
5.2	Technologické možnosti . . . . .	40
5.2.1	Architektura MVC . . . . .	40
5.2.2	AngularJS . . . . .	41
5.2.3	Ionic Framework . . . . .	42
5.2.4	Ripple emulátor . . . . .	43
5.3	Uživatelská základna . . . . .	43
5.4	Potenciální rozvoj . . . . .	44
<b>6</b>	<b>Tvorba testovací aplikace</b>	<b>45</b>
6.1	Téma . . . . .	45
6.2	Existující řešení . . . . .	45
6.3	Vývoj . . . . .	45
6.3.1	Potenciální problémy a jejich možná řešení . . . . .	46
6.3.2	Použité nástroje . . . . .	48
6.4	Vnitřní struktura aplikace . . . . .	49
6.5	Uživatelská dokumentace . . . . .	50
6.5.1	Hlavní obrazovka . . . . .	50
6.5.2	Ztracená a nalezená zvířata . . . . .	51
6.5.3	Nastavení . . . . .	53
6.6	Možná vylepšení aplikace . . . . .	54
6.7	Srovnání použitých frameworků . . . . .	55
6.7.1	Vzhled aplikace . . . . .	55
6.7.2	Přístup k funkcím telefonu . . . . .	57
6.7.3	Rychlost vývoje . . . . .	58
6.7.4	Vývojové nástroje . . . . .	58
6.7.5	Shrnutí . . . . .	59
	<b>Závěr</b>	<b>60</b>
	<b>Conclusions</b>	<b>61</b>
	<b>A Obsah přiloženého CD/DVD</b>	<b>62</b>
	<b>Literatura</b>	<b>63</b>

## Seznam obrázků

1	Podíl mobilních operačních systémů . . . . .	11
2	Magic Quadrant firmy Gartner pro rok 2015 . . . . .	16
3	Mobilní platforma Microsoft Azure . . . . .	21
4	Ukázka online nástrojů platformy Telerik . . . . .	23
5	Ukázka nástroje Appcelerator DashBoard . . . . .	24
6	Aplikace Kony Visualizer pro vývoj metodou drag&drop . . . . .	35
7	Princip fungování Xamarin Forms . . . . .	39
8	Princip fungování Apache Cordova . . . . .	41
9	Two Way Data-Binding v Apache Cordova . . . . .	42
10	Seznam zvířat a navigace v aplikaci Psí detektiv . . . . .	46
11	Prostředí Xamarin Test Cloud . . . . .	47
12	Vnitřní struktura aplikace ve frameworku Xamarin . . . . .	50
13	Vnitřní struktura aplikace ve frameworku Apache Cordova . . . . .	51
14	Hlavní obrazovka v aplikaci ČmuchaX . . . . .	52
15	Profil ztraceného zvířete v aplikaci ČmuchaX . . . . .	54
16	Vytvoření profilu zvířete v aplikaci ČmuchaX . . . . .	55
17	Nastavení v aplikaci ČmuchaC . . . . .	56

## Seznam tabulek

1	Nástroje pro multiplatformní vývoj . . . . .	34
---	--	----

## Seznam zdrojových kódů

1	Použití fotoaparátu pomocí pluginu v Apache Cordova . . . . .	57
2	Použití fotoaparátu pomocí rozhraní v Xamarin Forms . . . . .	58

# 1 Úvod

V dnešní době existuje velké množství výrobců mobilních zařízení, do popředí se ale dostalo jen několik operačních systémů, které na nich figurují. Napříč těmito systémy však nejsou uznávány žádné standardy, proto v základu nenabízejí možnost využít existující aplikace vyvinuté na konkurenční mobilní systémy. Samotné systémy jsou psány pomocí různých programovacích jazyků, a proto je tedy nutné psát i cílové aplikace v právě těchto jazycích. Kvůli těmto faktům se stává tvorba aplikace, která by měla pokrýt co největší množství potenciálních uživatelů, značně komplikovanou. Na první pohled se tak nabízí možnost vynalezení nástroje, jenž by dokázal vyvíjet aplikace pro všechny nejpoužívanější mobilní operační systémy. Této možnosti se tak chytlo velké množství firem a uživatel tak může vyvíjet aplikaci na více operačních systémů zároveň. Hlavním problémem tak zůstává výběr té nejuvhodnější varianty řešení.

Pomocí ve správném rozhodnutí výběru varianty může několik věcí. Může se jednat o existující srovnávací články na webových stránkách, komplexní analýzy nebo například testovací aplikace. I přesto, že se zdá tato nabídka vcelku bohatá, může mít potenciální vývojář problémy v orientaci mezi nabízenými řešeními. To je dáno především tím, že na ně není schopen nalézt komplexnější pohled. K němu se nejvíce blíží analýzy společnosti Gartner [1], která každoročně vydává takzvaný Magic Quadrant. Ani tyto zprávy však nejsou plně dostačující. Hlavní kritéria hodnocení existujících možností vývoje jsou spíše teoretického rázu, příliš tak nezkoumají technické nedostatky samotného vývoje, případně samotnou jeho rychlost. Je tomu tak zřejmě především kvůli časové náročnosti hlubšího zkoumání jednotlivých alternativ. Existují i projekty, které se zabývají čistě hodnocením samotného vývoje multiplatformních aplikací pomocí jednotlivých alternativ. Jedná se však často pouze o jednoduché aplikace, které uživatele spíše seznámí s konkrétními řešeními, ale samotnému výběru nikterak nepomohou.

Ve svém řešení se tak snažím zkombinovat všechny možné formy srovnání existujících možností multiplatformního vývoje mobilních aplikací. Ze začátku chci uživateli pomoci v rozhodování, zda se mu vůbec vyplatí vydat se formou multiplatformního vývoje. Pokud se následně pro takovýto vývoj rozhodne, je mu poskytnut základní přehled možných řešení, odvozený z aktuálního stavu trhu. Každá popisovaná nabízená možnost, dále také jako mobilní framework, je analyzována z několika pohledů. Těmi jsou například tržní postavení společnosti nabízející framework, technologické možnosti, jež řešení nabízí či historie společnosti samotné. Důležitým faktorem pak je konkrétní doporučení cílové skupiny vývojářů, pokud je dané řešení specifitěji zaměřeno. Vzhledem k tomu, že jednotlivé frameworky mají určité společné vlastnosti, jsou z nich vybrány dva zástupci, na kterých je provedena hlubší analýza. V ní je dbáno především na technologické možnosti, které jednotlivé alternativy nabízejí. Důležitým krokem této analýzy je tvorba plnohodnotné mobilní aplikace pomocí těchto frameworků. Ta je vytvořena zcela odděleně. Potenciální vývojář je pak z celkového srovnání vývoje těchto aplikací schopen získat přehled o kladech a záporech jednotlivých



řešení.

Vývoj mobilní aplikace a specifika srovnání jednotlivých alternativ vznikal ve spolupráci s firmou GMC Software Technology [2] a neziskovou organizací Moravskoslezský spolek pro ochranu zvířat [3].

## 2 Vývoj nové aplikace

Pro vývoj aplikace lze využít dvou různých stylů. V rámci těchto stylů pak také existuje více možných platforem a nástrojů.

### 2.1 Současné možnosti

Pokud má vývojář v plánu vytvořit aplikaci pro mobilní platformu, může se v zásadě vydat dvěma směry. Bohužel nebo možná z pohledu konkurence bohu-  
dík, je momentálně trh jak s telefony, tak s tablety (dále jako mobilní zařízení)  
rozdrcen do více společností. Člověk si tak může obstarat zařízení s různými  
operačními systémy. Z důvodu konkurenčních bojů však nejsou tyto systémy  
nijak standardizovány. Vývojářova tvorba tak může směřovat buď na využití ná-  
strojů a programovacích jazyků podporovaných přímo výrobcem daného operač-  
ního systému, nebo může sáhnout po druhém směru, jakým je použití některého  
multiplatformního nástroje. Ten mu umožní tvorbu na více operačních systémech  
současně. Nelze jednoznačně říci, který z těchto směrů je lepší, neboť záleží pře-  
devším na konkrétních požadavcích a specializaci vytvářené aplikace. Dalo by  
se však říci, vzhledem k aktuálnímu rozložení trhu, že multiplatformní vývoj  
je správným krokem a tvorba aplikací se bude v budoucnu ubírat právě tímto  
směrem.

### 2.2 Výběr platformy

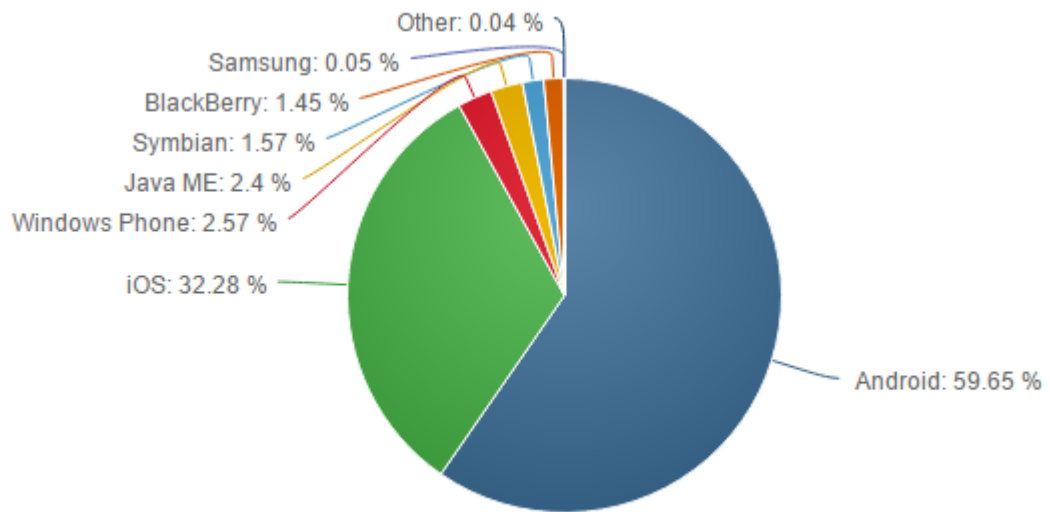
Před samotným vývojem je tedy zapotřebí si nejprve stanovit, pod jakou plat-  
formou se bude aplikace vyvíjet. K tomuto úkonu je dobré provést průzkum mezi  
možnými budoucími uživateli aplikace a následně stanovit největší procentuální  
zastoupení jednotlivých platforem mezi nimi. Tento krok je velmi důležitý, ne-  
boť, jak již bylo výše zmíněno, aplikace vyvíjené přímo na tyto platformy nejsou  
navzájem kompatibilní.

#### 2.2.1 Zastoupení platforem

Aktuální stav trhu nám může dosti napovědět o tom, jakou cílovou platformu  
zvolit. Jednou z možných analýz je analýza americké firmy Net Applications.  
Ta vychází čistě z podílu jednotlivých operačních systémů na prohlížení vybra-  
ných internetových stránek. Ukazuje nám, že téměř devadesáti procentní podíl  
na trhu platforem zabírají pouze dva z pěti průzkumem rozlišených operačních  
systémů (obr. 1). Těmito systémy jsou Android od společnosti Google a iOS od  
společnosti Apple.

#### 2.2.2 Nativní vývoj pro Android

Android jako takový vznikl ve stejnojmenné firmě v Kalifornii roku 2003. Již  
o dva roky později byla nevelká firma odkoupena internetovým gigantem, spo-



Obrázek 1: Podíl mobilních operačních systémů při prohlížení webů  
**Zdroj:** [www.netmarketshare.com](http://www.netmarketshare.com)

lečností Google. Až o další dva roky později byl ohlášen samotný operační systém určený pro chytré mobilní telefony. V roce 2008 pak byl na trh uvedeno první zařízení s tímto operačním systémem.

Společnost Google je autorem bezplatného balíku nástrojů Android Developer Tools. Ten tvoří SDK<sup>1</sup> pro každou verzi operačního systému Android, nástroje pro vývoj aplikací na tento systém v jazyce Java a také plugin pro vývojové prostředí Eclipse. Samotný plugin v tomto nástroji umožňuje efektivně zjednodušit psaní kódu aplikace a její následnou kompilaci a testování. Google taktéž podporuje nativní psaní aplikací v jazycích C a C++ pomocí vývojového balíčku Android Native Development Kit. Jako alternativu k pluginu pro Eclipse je možné využít software Android Studio, který je však stále distribuován ve stádiu beta verze. Android Studio je také uzpůsobeno k vývoji aplikací pro platformy Android Wear a Android TV.

Jak je z popisu zřejmé, tento styl vývoje je vhodný v případě, že se chce vývojář zaměřit čistě na platformu Android.

### Souhrn kladů:

- Android Developer kit, jak potvrzuje Allen Grant [4], je nejlepší volba pro programování v jazyce Java na operační systém Android.
- Platforma poskytuje okamžitý přístup k nejnovějším funkcím operačního systému z jeho vývojových nástrojů.

<sup>1</sup>SDK neboli Software development kit. Jedná se o sadu vývojářských nástrojů, umožňující vytváření konkrétních aplikací.

### **Souhrn záporů:**

- Vývoj pomocí těchto nástrojů není multiplatformní.

### **2.2.3 Nativní vývoj pro iOS**

První zmínka o iOS byla z úst zakladatele firmy Apple Steva Jobse roku 2005. V tu chvíli se však jednalo pouze o myšlenku chytrého mobilního telefonu bez jakýchkoliv konkrétních produktů. Původní vize byla taková, že na chytrém mobilním telefonu půjde spustit desktopový operační systém OS X, včetně aplikací, jenž na něj byly vytvořeny. To se roku 2007, kdy byl představen první iPhone, tak úplně nepotvrdilo. Operační systém běžel na stejném jádře, avšak byl tak rozdílný, že získal samostatný název iOS a jeho vývoj od té doby pokračuje odděleně. Ani aplikace navržené na operační systém OS X, nelze jednoduše spustit na mobilním systému. Apple ale nabízí nástroje, které mezi těmito systémy umožňují portaci aplikací.

Politika vývoje aplikací pro systém iOS obsahuje jedno základní pravidlo. Vývojář, jenž chce vyvíjet a distribuovat nativní iOS aplikace, musí mít dohodu s firmou Apple, neboť všechny aplikace musejí být distribuovány přes Apple App Store. Zároveň s tím vývojář potřebuje mít fyzicky dostupný stroj, na kterém je spuštěn operační systém OS X s vývojovým prostředím Xcode. Určitou alternativou je také možnost vývoje HTML5 aplikací, které lze spustit na iOS zařízených, nicméně Apple neposkytuje žádné nástroje určené pro tento druh vývoje.

Ani zde není těžké rozhodování pro zvolení právě tohoto nástroje. Je totiž vhodný přesně v tom případě, kdy chceme vyvíjet aplikace pouze pro zařízení s operačním systémem iOS.

### **Souhrn kladů:**

- Nástroje společnosti Apple nabízejí nejlepší možnou podporu pro iOS zařízení.
- Vývojářský balíček iOS SDK je jeden z nejpoblárnějších pro mobilní vývoj.

### **Souhrn záporů:**

- Apple zaměřuje svůj vývoj mobilních aplikací pouze na operační systém iOS.
- Nutná potřeba programátorských dovedností, viz Jiří Vávru [5], v jazyce Objective-C a nástrojích Xcode IDE a iOS SDK.
- Veškeré nástroje společnosti Apple vyžadují ke svému spuštění MAC PC s operačním systémem OS X.

## 3 Multiplatformní vývoj

Před zvolením multiplatformního vývoje je vhodné uvést několik kladů a záporů, které tento styl provázejí. Multiplatformní vývoj je roztržštěn do mnoha různých technologií a směrů, proto se nabízí provést jejich analýzu, případně částečně využít již existujících analýz. Ty by se měli zabývat srovnáním nezávislých hodnocení těchto společností a tím, jaký využívaly a využívají marketing. Dále pak tím, jak velká základna uživatelů za nimi stojí, jaké možnosti mají nástroje těmito společnostmi vytvořené a především také, jaký je jejich potenciál pro budoucí rozvoj. Vyzdvihnutí kladů a záporů konkrétních řešení by uživateli měl sloužit pro rychlý přehled a usnadnit mu tak zvolení toho správného řešení.

### 3.1 Trend multiplatformního vývoje

Pokud jsem se jako vývojář rozhodl programovat multiplatformně, je jednou z nejdůležitějších věcí analýza trhu. Ta nám ukáže, jak aktuální nástroje pro tvorbu, tak různé možné technologie tvorby. Během této analýzy bychom se měli věnovat několika faktorům. Z pohledu společnosti bychom se měli zajímat jak o současnou podobu těchto faktorů, tak i o jejich historii.

Jednou z věcí, na kterou bychom se měli zaměřit při výběru adepta pro multiplatformní vývoj, je uživatelská základna a samotná velikost společnosti. Ať už co se jejího počtu zaměstnanců týče, tak geografického rozložení, které má. To nám mimo jiné usnadní řešení problému s rozmanitostí mobilních zařízení. V různých částech světa totiž jsou dostupné různé výrobky, testování frameworků tak bude komplexnější. Pokud je řešení, jenž chceme využít, distribuováno jako open-source<sup>2</sup>, měli bychom se zaměřit na počet uživatelů, kteří nad ním vyvíjejí. Podstatnou součástí dobré společnosti je také poskytování konzultací a schopnost reagovat na problémy a dotazy uživatelů na důležitých kanálech jako jsou diskuzní fóra a sociální sítě. Důležitá je také historie společnosti, ať už co se týká mobilního vývoje, nebo existence firmy samotné. Pokud je již firma nějakou dobu na trhu zavedena, můžeme také zkoumat, zda již byla schopna reagovat na rychlé změny trhu. Pakliže jsme schopni určit vizi, jakou zvolená firma má, je dobré analyzovat, zda s ní je schopná prorazit mezi ostatními hráči na trhu.

Z marketingového hlediska je vhodné zjistit, zda firma byla za poslední období v zisku či ztrátě, jaké množství produktů za období byla schopna prodat a zda výtěžek efektivně investuje do svého rozvoje. Díky referencím jednotlivých klientů můžeme získat informace o jednání společnosti a také informaci o rychlosti reakce na problémy, s nimiž se zákazníci setkali.

Neméně důležitý je styl vývoje a s ním spojený cílový segment trhu. Kvůli jeho rozmanitosti jsou společnosti nuceny nabízet možným zákazníkům vícero alternativ vývoje. Nežřídka se tak potkáme jak s možností tvorby pomocí webo-

---

<sup>2</sup>Jako open-source je označována distribuce, kdy se softwarem jsou získány také zdrojové kódy softwaru. Tyto kódy pak, dle pravidel licence, může uživatel měnit.

vých technologií, viz Earle Castledine [6], tak i s možností tvorby pomocí jiných specifických jazyků. Nad těmito styly také najdeme postavené sofistikované nástroje, které umožňují vývoj metodou drag&drop<sup>3</sup>. Výsledné multiplatformní aplikace lze rozdělit do dvou základních kategorií dle typu jejich budoucích uživatelů. Jedná se o aplikace typu Business-to-customer a Business-to-employee, tedy aplikace určené zákazníkům firmy nebo aplikace určené pro zaměstnance firmy. Některé ze společností, jež multiplatformní nástroje nabízejí, jsou výlučně zaměřeny na některý z těchto segmentů, avšak většina větších hráčů na trhu je nucena pokrývat oba segmenty. Podstatnou roli tak může hrát fakt, do jaké z těchto kategorií budou spadat aplikace, na jejichž tvorbu se chceme zaměřit. Pokud jsme se rozhodli vyvíjet složitější aplikaci, musíme zvolit metodu vývoje s psaním zdrojového kódu. Zde je důležitý faktor, zda můžeme u nalezené společnosti využít nějaké naše dosavadní znalosti. Jedním z příkladů je podpora programovacího jazyka, se kterým již máme nějaké zkušenosti. Důležitá je také možnost použití již předchystaných šablon aplikací pro urychlení jejich vývoje. V případě, že se bude jednat o vývoj pomocí webových technologií, je vhodné, aby cílená společnost podporovala psané i zavedené webové standardy.

Klíčovou roli v určování trendu zde sehrávají lídři trhu v oblasti vývoje aplikací. Jedná se jak o společnosti pro multiplatformní vývoj, tak samozřejmě ve velké míře také společnosti, které vytváří operační systémy samotné. Ty určují především technologie, které díky nim budou mít nějakou budoucnost. Míra možné předvídatelnosti použitelnosti nástrojů a technologií však není příliš vysoká. Jedná se o obor s krátkou historií a pokud se zaměříme čistě na multiplatformní vývoj, ten je na tom, co se týče stáří, ještě hůře. Dalo by se tak říci, že sázka na multiplatformní vývoj by mohla být trochu sázkou do loterie. Pokud ale přihlídneme k faktu, jak velký je potenciál tohoto přístupu k vývoji aplikací a jak obrovský je nárůst jeho oblíbenosti, neměli bychom ho rozhodně ztracovat.

S postupem času je zřejmé, že společnosti, které spolupracují s již zavedenými velkými hráči v oblasti mobilních a počítačových technologií, mají o mnoho větší šanci, že si na trhu vybojují schopnou pozici a nestanou se zapomenutým článkem postupného vývoje. Ukázkovou společností v tomto směru může být Xamarin. Ta se se svým postupným zdokonalováním dostala mezi lídry trhu a zaujala tak velké zavedené společnosti jako je například Microsoft. Pokud bychom hledali případ opačného směru, jednalo by se o společnosti, jež nebyly schopny aktivně zareagovat na změny v oblasti technologií. Šlo především o využívání cloudu, či automatické testování vytvořených aplikací na velkém portfoliu zařízení. Jako spolupráci je možné označit také schopnost využívat open-source technologie. Pokud je námi nalezená společnost schopná efektivně pracovat s dostupnými nástroji ostatních a reagovat na jejich novinky a změny, její postavení na trhu se tím značně upevní. Konečné slovo při rozhodování bude mít dozajista také cena. Zde je důležité zjistit, zda cenový model společnosti lze efektivně využít na veškeré aplikace, jež máme v plánu vyvíjet a zda cena, bude odpovídat nabízeným

---

<sup>3</sup>Jako metoda drag&drop označován vývoj, kdy není potřeba psát zdrojový kód a tvorba je možná pomocí grafického rozhraní nástroje.

službám. Služby, které firmy nabízejí, se velmi často neomezuji pouze na nástroje pro samotný vývoj, ale obstarávají celý životní cyklus aplikace. Proto může být důležitá i možnost využití takovýchto služeb, jako je vývoj v cloudu, testování aplikace, či různé back-end zázemí aplikací.

## 3.2 Gartner Magic Quadrant

Pokud bychom chtěli získat více informací o aktuální stavu trhu v oblasti multiplatformního vývoje, můžeme se zaměřit i na externí průzkumy. Společnost Gartner, Inc. je špička v oboru výzkumu a poradenství se zaměřením na informační technologie. Služby téměř čtyřicetileté americké firmy s více než sedmi tisíci zaměstnanci využívá v praxi přes deset tisíc firem z celého světa.

Jedním z témat, kterým se společnost Gartner zabývá, je také mobilní vývoj a jedním z výstupů analýzy trhu je také Magic Quadrant. Jedná se o grafické znázornění, kde jsou aktuální hráči na trhu splňující určité podmínky rozděleni mezi čtyři čtverce (obr. 2). Jejich umístění je zde určeno tím, jak splňují daná kritéria. Tato kritéria jsou rozdělena do dvou základních skupin. První z nich, označována jako výkonové možnosti, pokrývá možnosti služeb, životaschopnost společnosti, její cenotvorbu, schopnost reakce na trh, marketing a reference zákazníků. Druhá skupina kritérií je nazvána ucelenost vize a obsahuje schopnost porozumění trhu, strategii marketingu společnosti, prodejní strategii, produktovou strategii, podnikatelský plán, schopnost inovace a geografickou polohu. Jednotlivé čtverce pak jsou označeny jako lídři, vyzyvatelé, vizionáři a specializovaní hráči. V následujících kapitolách je popsáno několik vybraných společností, které se objevily v těchto čtvercích v letech 2013 až 2015.

Magic Quadrant společnosti Gartner vychází každoročně od roku 2013 a jeho součástí je také krátký přehled společností, které v něm figurují. Každoročně jsou také lehce měněny podmínky pro to, aby se konkrétní společnost do tohoto výběru dostala. To je dáno především postupným rozvojem trhu. Proto je například každoročně navyšován nutný minimální počet společnostmi uskutečněných projektů za poslední kalendářní rok.

### 3.2.1 Lídři

Aby byl hráč označen jako lídr, musí kromě samotné podpory multiplatformního vývoje také poskytovat služby napříč celým životním cyklem aplikace, podporovat více architektur a standardů a v neposlední řadě mít také silné partnery. Lídr musí nabídnout platformu, kterou je možné jednoduše zakoupit, používat a vylepšovat. Nesmí být profilově úzce zaměřen a musí poskytovat řešení jak pro segment B2C, tak B2E. Pro lídra je také důležité, aby měl již vybudovanou jistou pozici na trhu a projevoval tendence si ji dále udržet.

**3.2.1.1 Adobe Systems** Adobe je známý velkým množstvím nástrojů jak pro mobilní, tak i webový vývoj. Společnost byla založena roku 1982 a v roce



Obrázek 2: Magic Quadrant firmy Gartner pro rok 2015

**Zdroj:** Magic Quadrant [1]

2015 již měla dvanáct a půl tisíce zaměstnanců. Od počátku existence mobilních zařízení věřila v úspěch webových technologií v čele se svým řešením distribuce obsahu pomocí programu Flash Player. V roce 2011 však překvapila odkupem společnosti Nitobi, která stojí za webovým frameworkem PhoneGap, využívajícím technologie postavené na jazyce HTML5.

Díky skutečnosti, že firma Nitobi před uzavřením kontraktu darovala zdrojové kódy společnosti Apache Software Foundation<sup>4</sup>, vznikla oddělená větev projektu pod názvem Apache Cordova. Nyní se dá říci, že jádrem projektu je právě Apache Cordova a Adobe PhoneGap k němu přidává některé další služby a nástroje.

Framework PhoneGap je unikátní především v tom, že jej využívá velké množství níže uvedených konkurentů. Jak je ale zmíněno společností Gartner [7], i sám Adobe nabízí nad frameworkem PhoneGap platformu PhoneGap Enterprise. Ta obsahuje mnoho nástrojů, týkajících se celého životního cyklu aplikace a také integraci s jinými nástroji firmy Adobe, jako je například Adobe Analytics umožňující marketingové analýzy v reálném čase.

<sup>4</sup>Apache Software Foundation je nezisková organizace starající se o podporu softwarových projektů.



### Souhrn kladů:

- Jak multiplatformní, tak nativní vývoj v oddělených nástrojích.
- PhoneGap udává standard multiplatformního vývoje.
- PhoneGap má širokou škálu pluginů a velkou podporu.
- Zjednodušuje vývoj pomocí známých webových technologií HTML, CSS a JavaScript.
- Jádru Apache Cordova je open-source.

### Souhrn záporů:

- Technologie tvorby aplikací ve webových kontejnerech skýtá jistá omezení, ty lze částečně odstranit pomocí pluginů.

**3.2.1.2 IBM** Společnost IBM je na poli informačních technologií obrovský pojem. Svědčí o tom jak rok založení 1911, tak i roční příjmy v řádech miliard dolarů. Počet jejich zaměstnanců se pohybuje mezi čtyřmi sty až pěti sty tisíci napříč celým světem, což společnost řadí na druhé místo v počtu zaměstnanců v rámci celých spojených států amerických. Kromě vývoje softwaru se IBM zabývá například také vývojem hardwaru a výzkumem, za který získávají její vědci světově významné ceny. V červenci roku 2014 uzavřela společnost důležité partnerství v mobilní sféře s firmou Apple. V rámci tohoto kontraktu se IBM zavázalo k výrobě průmyslových mobilních aplikací exklusivně pro mobilní zařízení iPhone a iPad.

IBM nabízí své mobilní služby pro celý životní cyklus aplikace. Od představení, přes vývoj stylem drag&drop, integrované testy a různé vývojové nástroje, po možnost převezení aplikace z webové formy do nativní a naopak. IBM také umožňuje použití nástrojů třetích stran v každém z článků tohoto řetězce. Do tohoto postavení se IBM dostalo především svými velkými investicemi do webových a jim příbuzných standardů, zároveň má společnost na svém kontě více než čtyři tisíce patentů týkající se mobilních zařízení. Jak je zmíněno v analyzující zprávě [7], IBM své služby zařituje pod program Mobile First. Ten byl představen v roce 2013 a jeho hlavní součástí je platforma Worklight. Ta zahrnuje analýzu, testování, integraci, zabezpečení a management a může být použita jak pro vývoj webových, tak i multiplatformních aplikací. Aplikace mohou být vytvářeny jak formou drag&drop, tak psáním v programovacím jazyce HTML a JavaScript. Právě pro druhou možnost nabízí platforma nástroj MobileFirst Studio, jenž podporuje taktéž frameworky třetích stran, jako jsou Adobe PhoneGap či Sencha Touch. Společnost IBM se také velmi usilovně zapojuje do vývoje cloudových služeb.

Platformu MobileFirst je vhodné použít pro multiplatformní vývoj, s možností využití existujících dovedností v jazycích HTML5 a JavaScript či vývojovém prostředí Eclipse.

### Souhrn kladů:

- Worklight umožňuje využití znalostí vývoje webových a nativních aplikací.
- Worklight je součástí velkého balíku služeb MobileFirst zahrnující testování, cloud či bezpečnost.
- Řetězec řešení Worklight je otevřený a umožňuje využití prostředku třetích stran.
- IBM má silnou pozici na trhu.
- Společnost uzavřela velké partnerství s firmou Apple.

### Souhrn záporů:

- Společnosti IBM investuje velké množství prostředků do partnerského ekosystému. I přesto je počet vývojářů, kteří pracují s frameworkem Worklight, značně omezen.

**3.2.1.3 Kony** Firma Kony vznikla teprve v roce 2007 a nyní je již nadnárodní společností se sídlem v americkém Austinu. Ke konci roku 2015 měla již 350 zákazníků v čtyřiceti pěti různých státech. Mezi největší úspěchy firmy patří také výhry v soutěžích Mobile Star Award a CTIA MobIT's Awards v oblasti vývoje mobilních aplikací. Firma bývá také pravidelně uveřejněna v žebříčku pěti set nejrychleji se rozvíjejících firem ve Spojených státech amerických.

Ze zprávy společnosti Gartner [1] je zřejmé, že společnost Kony nabízí plnou podporu životního cyklu aplikace. Od jejího návrhu až po řízení managementu aplikace. Veškeré služby mohou být využity jak v reálných prostorech, tak v privátním či veřejném cloudu. Základní rozdělení platformy, jež Kony nabízí, je do dvou systémů. První z nich, Kony Visualizer, nabízí schopnosti návrhu aplikace. Ty jsou integrovány do vývojového prostředí a umožňují jak vývoj webových aplikací, tak nativních či multiplatformních postavených na webových technologiích HTML5 a JavaScript. Velkou výhodou nástroje je také kvalitně zpracovaný systém vývoje metodou drag&drop, který umožní tvorbu aplikace bez nutnosti psát zdrojový kód. Druhý systém, Kony Development Cloud, obsahuje vývojové nástroje Kony Studio. Ty sice nedovolují použití nástrojů třetích stran, nicméně umožňují vývoj jak nativních aplikací (pro iOS a Android), tak hybridních aplikací (na platformě PhoneGap).

Platformu Kony je vhodné použít pro projekty, u nichž se počítá s velkou komplexitou a dlouhou životností. Kony také nabízí silné back-end zázemí.

### Souhrn kladů:

- Díky zaměření na celý životní cyklus aplikace je Kony silným hráčem na trhu.

- Kony se rychle vyvíjí a nabízí garantovanou podporu pro nové funkce mobilních operačních systému.

#### **Souhrn záporů:**

- Firma provedla velké investice do managementu.
- Společnostem, které žádají rychlý vývoj aplikací, může připadat Kony až příliš složitý.

**3.2.1.4 Salesforce.com** Mezi ty nejlepší společnosti, které se jakkoliv dotýkají cloudových technologií, patří bezesporu Salesforce.com. Ta byla založena roku 1999 a k dnešnímu dni dosahuje ročních obrátů v miliardách amerických dolarů. Největší část z jejich příjmů plyne z produktu webového CRM, který je rozdělen mezi několik vzájemně spjatých služeb. Ty pak následně využívá přes sto tisíc zákazníků.

Salesforce je společnost, která již má své vydobyté místo na trhu s cloudovými aplikacemi. Na poli mobilních zařízení pak nabízí svou platformu Salesforce1 Platform. Ta umožňuje vývoj aplikací na několika úrovních. Pokud se jedná o CRM aplikaci, může zákazník využít již existujícího řešení od společnosti a to si s minimální nutností změny kódu upravit podle svých představ. Jako další úroveň se mu pak nabízí tvorba nativních aplikací pomocí SDK, jež poskytují sami výrobci platform. V nedávné době Salesforce představil také podporu pro wearables zařízení. Řešení společnosti lze využít na operačních systémech Android a iOS. Podpora systému Windows byla již oznámena, avšak její uvedení do ostrého provozu zatím neproběhlo.

Využití nabízený framework je více než vhodné v případě, že společnost používá některou ze služeb zmíněného CRM. Nejen z analýzy společnosti Gartner [8] lze vyčíst, že ostatní tvorba je na této platformě do jisté míry omezena.

#### **Souhrn kladů:**

- Platforma Salesforce1 umožňuje tvorbu aplikací na několika úrovních.
- Společnost je neustále na vzestupu.

#### **Souhrn záporů:**

- V některých odvětvích vysoká cena.
- Platforma dosud nepodporuje mobilní operační systém Windows.

### **3.2.2 Vyzyvatelé**

Mezi vyzyvatelé pak společnost Gartner řadí takové hráče, kteří si dokázali na trhu získat již velké množství zákazníků a mají dostatečné kapacity pro svůj další rozvoj. Musí mít taktéž jako společnost určitou historii, i když nemusí být

za každou cenu právě v oboru mobilního vývoje. Tak jako vizionáři, i vyzývatelé musí mít určitou vizi, jakým směrem se má jejich společnost ubírat.

**3.2.2.1 Microsoft** Firma Microsoft je další velké jméno v seznamu společností nabízejících multiplatformní řešení mobilních aplikací. Samotná firma vznikla již v roce 1975 a od té doby se, především díky jejich desktopovému operačnímu systému, dostala do podvědomí lidí po celém světě. Dnes má společnost přes sto dvacet tisíc zaměstnanců a průměrný roční obrat přesahující osmdesát šest miliard dolarů. Společnost se také angažuje v mobilním odvětví svými vlastními operačními systémy a v posledních letech také vlastními zařízeními.

Roku 2008 byla firmou představena velká cloudová platforma s názvem Windows Azure. Ta poskytovala služby pro tvorbu a řízení aplikací a služeb. K dnešnímu dni již byla platforma přejmenována na Microsoft Azure a poskytuje služby již ve více než patnácti odvětvích. Do světa multiplatformního vývoje vstoupil Microsoft především svým balíkem nazvaným Azure Mobile Services. Ten nabízí back-end služby pro mobilní aplikace a jako celý systém Azure, je založen na cloudové platformě. Využívat jej mohou klientské aplikace jak na mobilních OS (iOS, Android a Windows), tak i mimo ně, pomocí HTML5 webového rozhraní. Dalším krokem Microsoftu pro multiplatformní vývoj je univerzálnost aplikací na Windows platformě. Uživatel si tak může zakoupit aplikaci například na Windows Phone a následně ji využívat i na desktopovém operačním systému Windows či na zařízení Xbox. Ve vývojovém prostředí Visual Studio, jež Microsoft vyvíjí, je umožněno vytvářet multiplatformní webové a hybridní aplikace, většinou však s využitím řešení třetích stran. Mezi tato řešení patří například tvorba pomocí frameworku firmy Xamarin. Novinkou pro rok 2014 pak je možnost využití vývojového prostředí Visual Studio také pro tvorbu multiplatformních aplikací ve standardu Apache Cordova v jazyce HTML5.

V dubnu roku 2015 byl na vývojové konferenci Build představen nový nástroj Visual Studio Code. Ten vychází ze svého velmi úspěšného většího bratra, avšak nově je ho možné spustit jak na operačním systému Windows, tak Linux či OS X. Toho je dosaženo především díky pojetí softwaru, neboť byl téhož roku vydán jako open-source pod MIT licencí. Jeho zdrojové kódy jsou tak volně přístupné na webu github.com. Nástroj se tak dostane mezi širší skupinu uživatelů a díky jeho modulové architektuře, kdy nové programovací jazyky lze instalovat pomocí různých rozšíření, bude mít jeho uvedení na trh vliv i na multiplatformní vývoj.

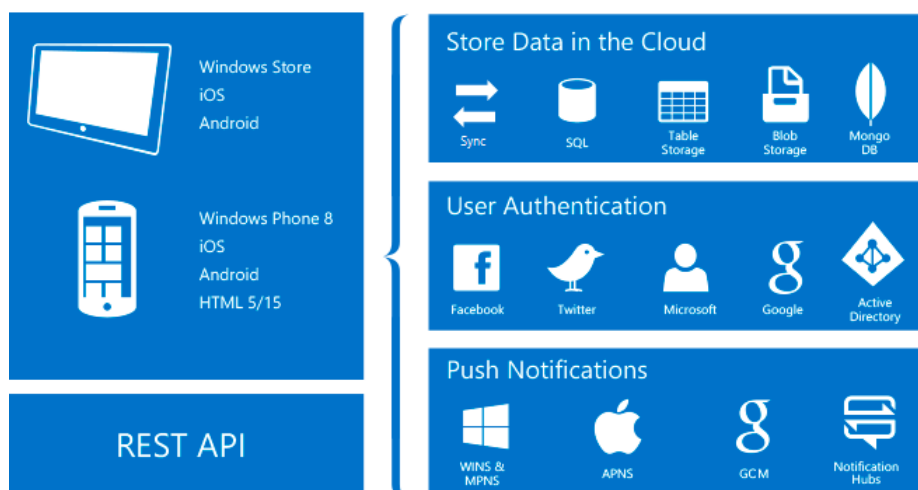
Využití platformy je vhodné především pro firmy, jež mají existující zkušenosti v nástroji Visual Studio či frameworku .NET, případně využívají cloudovou platformu Azure.

#### **Souhrn kladů:**

- Otevřené chování díky podpoře iOS a Android v Azure Mobile Services.
- Microsoft Azure je momentálně největší cloudová platforma (obr. 3).

## Souhrn záporů:

- Microsoft je stále zaměřen především na platformu Windows a nebude pro něj jednoduché expandovat mezi jiné.
- Microsoft již představil svou vizi, kterou se chce v příštích letech ubírat. Jak ale dodává analýza [7], zbývá mu ještě přesvědčit veřejnost, že jeho vize je ta správná.



Obrázek 3: Mobilní platforma Microsoft Azure

Zdroj: [azure.microsoft.com](http://azure.microsoft.com)

**3.2.2.2 Progress Software** Progress Software je americká softwarová firma založena v roce 1981. Počet jejich zaměstnanců se blíží jednomu tisíci a její roční příjmy se pohybují přes tři sta milionů dolarů. Produkty a služby, které Progress Software nabízí, využívá přes šedesát tisíc organizací ve sto čtyřiceti státech. Společnost se také chlubí tím, že v tomto počtu jejich zákazníků je devadesát procent společností zahrnutých ve Fortune 500<sup>5</sup>. Jedním z jejich největších projektů je také vlastní programovací jazyk OpenEdge ABL, vyvinutý primárně pro tvorbu business aplikací.

V prosinci roku 2014 firma oznámila dokončení akvizice bulharské společnosti Telerik. Ta se již od počátku své existence zabývala vývojem nástrojů a služeb pro multiplatformní mobilní vývoj se zaměřením primárně na Microsoft platformu. Využívá tak technologie Visual Basic, ASP.NET, WPF, Windows Forms či Silverlight. Jako alternativy podporuje však i jazyky Java či PHP. Zajímavostí, kterou našla analýza společnosti Gartner [1], je fakt, že za posledních deset let se firma Telerik pravidelně zvětšovala každým rokem o padesát procent. Samotná mobilní

<sup>5</sup>Fortune 500 je roční zpráva vydávaná magazínem Fortune. Pravidelně informuje své čtenáře o pěti stu největších amerických společností vzhledem k jejich ročním příjmům.

platforma Telerik Platform nabízí vývoj v HTML5 a JavaScript s možným cílovým použitím na mobilních operačních systémech iOS, Android, Windows Phone a BlackBerry (obr. 4). Platforma se skládá z nástrojů, frameworků a cloudových služeb, podporujících jak vývoj nativních, tak hybridních aplikací. K aplikaci lze vytvářet prototypy, testovat ji či analyzovat. Nejoblíbenějším produktem společnosti je Kendo UI, což je HTML5 framework postavený na oblíbeném frameworku jQuery. Ten je kompatibilní s nástroji Twitter Bootstrap či Google AngularJS. Telerik také podporuje tvorbu nativních aplikací skrz jeho knihovny a kompilátor NativeScript.

#### **Souhrn kladů:**

- Přístup Teleriku, při kterém své produkty a služby prodává výhradně přes internet, je velmi úspěšný.
- Řešení nabízí velké množství modulárních komponentů, frameworků a nástrojů, které podporují vývoj pro různé platformy.
- Platforma používá otevřené standardy jako jQuery, Bootstrap či AngularJS a samotný Kendo UI je taktéž open-source.

#### **Souhrn záporů:**

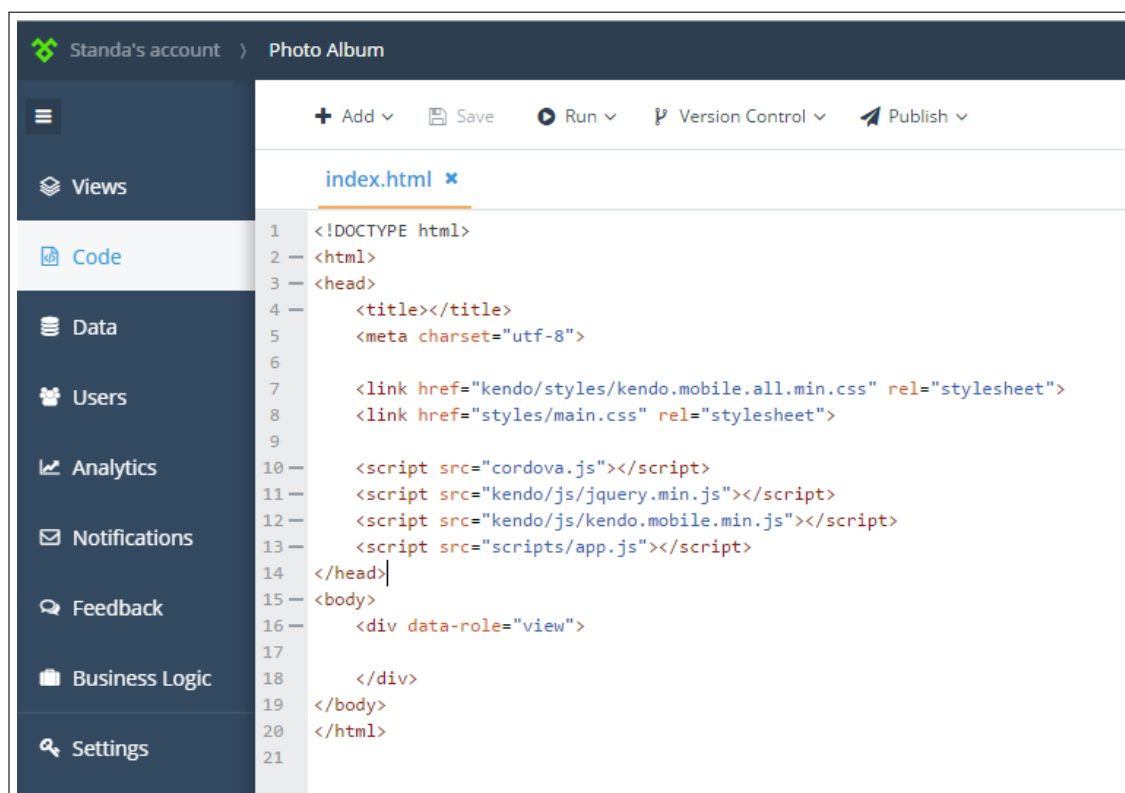
- V minulosti společnost nebyla zaměřena na mobilní vývoj.

### **3.2.3 Vizionáři**

Hráči takto označení musí mít jasnou vizi, jakým směrem se má jejich business ubírat a také mít jasno nad technickými prostředky, pomocí nichž chtějí těchto cílů dosáhnout. Mezi vyzyvatele se často nedostanou především kvůli chybějící firemní historii, nízkým příjmům či malé základně uživatelů. Do této kategorie tak mohou být zahrnuty i společnosti, které se na první pohled jeví jako velmi solidní, s velkou základnou uživatelů či firemní historií.

**3.2.3.1 Appcelerator** Společnost Appcelerator je, přesto že byla založena v roce 2006, průkopníkem na poli moderního multiplatformního vývoje. V roce 2015 již zastřešovala přes 600 000 vývojářů, čímž se po právu řadí mezi nejpoužívanější frameworky. V roce 2013 se Appcelerator pochlubil také tím, že aplikace, jež byly s pomocí jeho frameworku vytvořeny, se nacházejí na deseti procentech všech mobilních zařízeních.

Základ jejího úspěchu tvoří produkt Titanium, který jako open-source software nabízí tvorbu multiplatformních mobilních aplikací. Dále pak platforma Appcelerator Platform obsahující nástroje pro tvorbu a správu multiplatformních aplikací (obr. 5). Pomocí nástroje Appcelerator Studio je možné, na základě programovacího jazyka JavaScript, vytvářet nativní či hybridní aplikace. Ty jsou



Obrázek 4: Ukázka online nástrojů platformy Telerik

pak cíleně kompilovány do nativních Android a iOS aplikací, což zahrnuje překlady do jazyka Objective-C či Java.

Společnost také, dle analýzy [1], nabízí různé front-end a back-end služby týkající se vývoje, jako jsou nástroje pro marketingové analýzy či možnosti testování vytvořených aplikací na vícero zařízeních napříč jejich spektrem.

#### Souhrn kladů:

- Dobrá pozice na dynamickém a rychle se vyvíjejícím trhu.
- Možnost tvorby jak hybridních, tak nativních či webových aplikací.
- Appcelerator nabízí back-end služby, front-end nástroje a frameworky, díky nimž lze dosáhnout mnoha požadovaných scénářů.
- Společnost demonstrovala schopnost rychle se přizpůsobit měnícímu se trhu.

#### Souhrn záporů:

- Nutnost programovat pomocí jazyka JavaScript jak na straně klienta, tak na straně serveru.
- Vysoká cena.



Obrázek 5: Ukázka nástroje Appcelerator Dashboard

Zdroj: [appcelerator.com](http://appcelerator.com)

**3.2.3.2 Backbase** Společnost Backbase, založená v roce 2003, vznikla jako jedna z prvních společností, které pro vývoj svých aplikací využívala webový framework Ajax. Ten se dále dostával do podvědomí a s ním také společnost Backbase. Nyní je známa především z vývoje softwaru pro marketingové společnosti.

Jak je zmíněno ve zprávě společnosti Gartner [1], v nedávné době přidala Backbase do svého portfolia nabízených služeb také možnost multiplatformního vývoje. Ten je nabízen zároveň s dalšími nástroji a službami pod názvem Backbase Customer Experience Platform (CXP) a obsahuje velké množství služeb a nástrojů pro tvorbu aplikací nad všemi možnými kanály. Mimo jiné také umožňuje nejrozličnějším firemním uživatelům úpravu svých aplikací pomocí jednoduchého grafického uživatelského rozhraní, tedy bez nutné znalosti programování. Klientské aplikace jsou pak skládány z již předpřipravených widgetů. Takto vytvořená aplikace pak dokáže komunikovat s nakonfigurovaným webovým serverem, který společnost taktéž nabízí.

Nástroje a tvorba aplikací nijak nevybočuje z nabízených služeb společnosti Backbase a je jen její další nevyhnutelnou evolucí. Společnost je na trhu úzce specializována, díky tomu si zajistí jak stálé uživatele, tak přísun nových.

#### Souhrn kladů:

- Díky specializaci silný nástroj pro finanční klientelu.

#### Souhrn záporů:

- Specifický nástroj není vhodný pro tvorbu běžných mobilních aplikací.



**3.2.3.3 Pegasystems** Společnost Pegasystems se řadí svou velikostí mezi větší softwarové společnosti. Počet jejích zaměstnanců se odhaduje přes tři tisíce a její roční příjem je přes šest set milionů dolarů. Hlavní business společnosti se týká jejího produktového řešení CRM<sup>6</sup>, jež je řazené mezi pět světově největších systémů.

V roce 2015 proběhla akvizice společnosti Antenna Software, pomocí které Pegasystems zabředla do vod mobilního vývoje. Pegasystems podporuje vývoj webových, hybridních a nativních aplikací. Platforma může být využívána jak na vlastním serveru, tak v cloudu. Základem je hybridní kontejner, který je možný spustit na většině zařízení. Ten pak umožňuje například vzdálenou správu aplikací, spuštění více webových HTML5 aplikací či vícevláknové aplikace. Společnost nabízí také využití front-end nástrojů a frameworků třetích stran, jako je JQuery Mobile, Xamarin či Android SDK. Jedním z nástrojů, které lze využít, je Pega Designer Studio. To umožňuje tvorbu vizuálního vzhledu aplikace bez nutnosti psaní aplikačního kódu.

Pegasystems je atraktivní organizace, která se, dle analýzy [1], prezentuje především svým rychlým vývojem podle požadavků zákazníka. Současně je její tvorba stále více zaměřena na cloudové služby.

#### Souhrn kladů:

- Řešení společnosti využívá HTML5 architekturu.
- Společnost má vcelku velké online komunity vývojářů.
- Testovací nástroje jsou dostupné v cloudu.

#### Souhrn záporů:

- Pegasystems má malé zkušenosti s implementací B2C (Business to Customer).

**3.2.3.4 Sencha** Deset tisíc zákazníků, dva miliony vývojářů, přes sedm milionů stažení jejich produktu. To jsou klíčové statistiky, kterými se chlubí na svých webových stránkách společnost Sencha. Jedná se o úctyhodná čísla, neboť společnost jako taková existuje teprve od roku 2007. Kromě hlavní působnosti v Redwood City v Kalifornii nabízí také pobočky v Kanadě, Nizozemsku, Korei či Austrálii.

Sencha je také, mimo jiné, dobrý případ obchodníka, který je zaměřen v mobilním vývoji čistě na web-centric<sup>7</sup> přístup. Jak bylo zjištěno společností Gartner [7], Sencha má vynikající znalosti HTML5 a snaží se aktivně podílet na optimalizaci webových prohlížečů, především jazyka JavaScript. Její nástroj Sencha

---

<sup>6</sup>Customer relationship management - druh managementu jenž je orientovaný na zákazníky, značí databázové procesy zpracování a shromažďování informací o zákaznících.

<sup>7</sup>Jako web-centric je označován přístup, kdy je aplikace designována v první řadě pro web.

Architect umožňuje navrhnout, vyvíjet a publikovat mobilní či desktopovou aplikaci. Taktéž je zde dostupná možnost využití vyvíjeného pluginu do vývojového prostředí Eclipse. Dalším produktem, z významového pohledu tím nejdůležitějším, je Sencha Touch. Sencha Touch je webový Framework, umožňující vyvíjet mobilní aplikace pro operační systémy Android, iOS, Windows, Tizen a BlackBerry. Jeho velkým plusem je výborně zvládnutý přístup k nativním mobilním funkcím pomocí jazyka JavaScript. Framework obsahuje velké množství předpřipravených komponentů pro tvorbu uživatelského rozhraní aplikace. Tyto komponenty jsou následně převedeny do jejich nativní podoby vzhledem k zařízení, na němž se aplikace nachází. K této funkcionalitě využívá Sencha také jazyk Sass, jenž funguje jako nadstavba webového standardu CSS. Mezi další nabízené produkty pak patří například Sencha Space nabízející nástroje k zabezpečení dat a managementu aplikací.

Nabídku společnosti Sencha je vhodné využít v případě, že již existují nějaké zkušenosti v oblasti webových aplikací či jazyku JavaScript. Programovací model by měl být povědomý těm, jenž pracovali s nástroji Adobe Flex, WPF, Java Swing či s jinými objektově orientovanými platformami.

#### **Souhrn kladů:**

- Sencha je jedna z mála společností, které nabízí vývoj v HTML5 a přesto profesionální technickou podporu.
- Produkt Sencha Touch má silnou uživatelskou základnu.

#### **Souhrn záporů:**

- Sencha plně využívá HTML5 a JavaScript, což se může z části jevit i jako nevýhoda.

**3.2.3.5 Xamarin** Společnost zaměstnává přes 15 000 zaměstnanců a zahrnuje komunitu přes 650 000 vývojářů. Xamarin nabízí nástroje, komponenty a cloudové služby, které umožňují vývojovým týmům vytvářet nativní multiplatformní aplikace v jazyce C#. Ty jsou následně spustitelné na systémech Android, iOS, Windows, Windows Phone či Mac OS X. Xamarin nabízí rychlý vývoj, především díky možnosti využít existující dovednosti s frameworkem .NET. Samotné vývojové prostředí, viz [1], pak překládá jazyk C# do nativních kódů pro každou zvolenou cílovou platformu. Xamarin zde nabízí dvě alternativy. Prvním z nich je plugin do hojně využívaného vývojového prostředí Visual Studio, druhým pak vlastní vývojové prostředí Xamarin Studio.

Vývojáři mohou dosáhnout nativního UI a sdílení kódu mezi platformami pomocí různých metod. Mohou vytvářet oddělený kód uživatelského rozhraní v jazyce C# a využívat API specifikované na konkrétní platformu, nebo mohou využít komponenty Xamarin.Forms, jež se pak do zvolené platformy přeloží. Užitečným nástrojem pro usnadnění vývoje je také Xamarin Test Cloud. Ten umožňuje automatické testování napříč mnoha zařízeními a verzemi operačních

systémů.

V roce 2014 Xamarin uzavřel partnerství s velkými společnostmi, jako je IBM, SAP či Oracle. Pokračuje také ve vylepšování svých produktů. Xamarin Test Cloud tak k tomuto roku podporuje více než 1 600 zařízení a operačních systémů. Organizace také představila službu Xamarin Insights, která se stará o monitorování pádů a chyb uživatelsky vytvořených aplikací. Společnost by bylo vhodné uvést spíše v rámci organizace Microsoft, jež ji v lednu roku 2016 odkoupila. Odděleně se tak nachází kvůli více směrům, které Microsoft pro multiplatformní vývoj nabízí.

Xamarin je vhodné využít pro vývojáře, kteří mají již existující zkušenosti s jazykem C# a chtějí vyvíjet aplikace, jež mohou být přeloženy na více operačních systémů.

#### **Souhrn kladů:**

- Xamarin cílí na neobsazené místo na trhu – C# vývojáře, kteří chtějí své zkušenosti přenést do mobilního vývoje.
- Xamarin nabízí jak své vlastní IDE, tak plugin do Visual Studia.
- Xamarin si vytvořil na trhu velmi silné partnery.

#### **Souhrn záporů:**

- Neexistuje možnost využít v tomto řešení jiné programovací znalosti než v jazyce C#.

### **3.2.4 Specializovaní hráči**

Hráči označení jako specializovaní jsou tací, kteří nesplňují některá základní kritéria. Těmi může být například schopnost nabízet své služby po celém světě, nebo schopnost pokrytí všech možných vývojových scénářů, jenž může zákazník vyžadovat. Tyto společnosti jsou zde uvedeny především pro takové zákazníky, kteří potřebují najít řešení pro konkrétní problém, případně již využívají některá další řešení těchto firem.

**3.2.4.1 ClickSoftware** Společnost ClickSoftware je lídrem v oblasti řešení řízení pracovní síly v technickém průmyslu. Založena byla roku 1979 v Izraeli a mezi její nejlepší reference patří například firmy Hewlett-Packard, Bosch či Xerox. Firma má také podepsaná cenná partnerství s firmami IBM, SAP či Salesforce.com.

Díky silné pozici na trhu je společnost již nyní schopna nabídnout funkční řešení aplikací pro komunikaci či financování. Tyto aplikace se nacházejí ve svém vlastním obchodě s názvem ClickAppStore, založeném v roce 2012. Kromě balíku existujících řešení umožňuje také tvorbu kompletně nových aplikací. Takové

klientské aplikace jsou založeny na HTML5 hybridní architektuře. Jak zmiňuje analýza společnosti Gartner [8], ClickSoftware byla jedna z prvních společností, která začala nabízet hostingové služby založené na cloud technologiích. Proto není divu, že veškeré produkty, ať zakoupené přímo od společnosti nebo uživatelsky vytvořené, mohou být dostupné také v cloudu.

#### **Souhrn kladů:**

- Společnost nabízí platformu zahrnující nástroje, frameworky, obchod aplikací a rozsáhlé knihovny, což umožňuje rychlý vývoj v oblasti mobilních aplikací.
- ClickSoftware pokračuje v investování do inovací a vývoje rozdílných rysů svých platforem.
- Organizace má silné geografické pokrytí a rostoucí síť partnerů, včetně úspěšného partnerství se společnostmi SAP, salesforce.com a IBM.
- Jako jedni z prvních zahrnuli do svých aplikačních řešení také podporu wearables.

#### **Souhrn záporů:**

- Mobilní nástroje jsou nyní nejlépe použitelné pouze v souvislosti s řešením pro správu pracovní síly.
- Řešení společnosti jsou slabá v některých částech životního cyklu aplikace jako je například management.

**3.2.4.2 DSI** Společnost DSI, neboli Data Systems International, je americký výrobce počítačového softwaru založený roku 1979. Během let jejího působení vyprofilovala firma svou pozici směrem k vývoji softwaru, který se jakkoliv pojí s přepravním segmentem. Jedním z jejich nejaktuálnějších koncových produktů je jejich vlastní mobilní platforma. Platforma umožňuje rychlý vývoj pomocí drag&drop technologie, tedy bez nutnosti znalosti jakéhokoliv programovacího jazyka a konceptu cílové platformy. Takto vytvořená aplikace lze spustit jako nativní na zařízeních s operačními systémy Windows, iOS, Android a BlackBerry. Mezi další nabízené služby patří například distribuce takto vytvořených aplikací do veřejných obchodů s aplikacemi. Odtud jsou aplikace dále získávány koncovými zákazníky. DSI má také vyvinutý vlastní systém, pomocí něhož mohou být vytvořené mobilní aplikace použity ve formě webových stránek. V roce 2014 společnost svou mobilní platformu rozšířila mimo jiné o cloudové služby.

Firma neustále zvyšuje počet svých zaměstnanců a to jak vlastním náborem, tak i pomocí akvizic jiných firem. Poslední taková akvizice se udála v roce 2015, kdy se ke společnosti přidala firma RareWire LLC zabývající se právě vývojem mobilních aplikací. Jak dodává společnost Gartner [1], řešení společnosti DSI je vhodné pro tvorbu aplikací především ve vztahu B2E (Business to Employee),

které se jakkoliv týkají dodavatelského segmentu.

#### **Souhrn kladů:**

- DSI má velké zkušenosti s tvorbou mobilních aplikací pro výrobní a dodavatelský segment.
- Společnost má více než 1 100 zákazníků ve více než padesáti státech.

#### **Souhrn záporů:**

- Nástroje společnosti nejsou koncipovány pro tvorbu aplikací ve vztahu B2C (Business to Client).

**3.2.4.3 Embarcadero Technologies** I přes mládí firmy založené v roce 1993 se Embarcadero řadí mezi nejlepší světové společnosti vytvářející počítačový software. O této pozici také svědčí jak narůstající počet zaměstnanců, kterých je již přes pět set, tak její roční příjem pohybující se přes sto milionů dolarů. Společnost má komunitu tří miliónů vývojářů a více než pěti tisíc zákazníků. Mezi nejznámější produkty společnosti patří databázové nástroje jako Rapid SQL či InterBase, ale také RAD Studio, vývojářský nástroj pro psaní v programovacích jazycích Delphi a C++.

Právě produkt RAD Studio získal v roce 2013, ve verzi XE5, podporu pro vývoj mobilních aplikací. Ty lze psát jak v jazycích C++ a Delphi, tak pomocí webových technologií HTML a JavaScript. Výsledné nativní aplikace mohou být použity na mobilních operačních systémech iOS, Android či Windows a taktéž jako desktopové aplikace na OS Windows a MAC OS X. RAD Studio umožňuje kompilaci do kódu stroje bez nutnosti virtuálního stroje či jiné mezivrstvy. V rámci obohacování nástrojů o mobilní vývoj vznikl také další produkt společnosti a to nástroj FireUI. Ten umožňuje jednotnou tvorbu uživatelského rozhraní pro všechny podporované platformy včetně chytrých hodinek.

Je zřejmé, že mobilní směr je pro Embarcadero vizí do budoucnosti, neboť, jak udává zpráva z roku 2014 [7], společnost produkuje také další nástroj s názvem AppMethod. Ten je často prezentovaný jako nástupce RAD Studia zaměřený především na tvorbu Android aplikací a jejich následnou distribuci skrz veřejný obchod s aplikacemi. Embarcadero je vhodné použít pro firmy, jež mají zkušenosti s jazyky C++ či Pascal.

#### **Souhrn kladů:**

- V dostupných nástrojích je možný vývoj aplikace spustitelné jak z mobilních OS, tak z desktopových OS.
- Prodejce má existující základnu podnikových vývojářů.
- Řešení společnosti umožňují rychlou kompilaci a vývoj.

### Souhrn záporů:

- Embarcadero nenabízí úplný životní cyklus aplikace, jako například backend služby a cloudové testování.

**3.2.4.4 MicroStrategy** Pokud bychom hledali společnost, jejíž náplní je analýza firemních dat, zajisté bychom velmi rychle narazili na firmu MicroStrategy. Ta za dobu své existence získala již přes tři tisíce zaměstnanců po celém světě a také tak velké klienty jako jsou Starbucks či Facebook. Její hlavní náplní je poskytovat nástroje business intelligence<sup>8</sup>, mobilní software a také cloudové služby.

První zmínka společnosti o mobilních zařízeních vznikla v roce 2010, kdy byl představen nový business intelligence software na zařízení iPhone a iPad. Zároveň s tím byl představen také produkt MicroStrategy Mobile, sloužící pro přístup k firemním datům z mobilních zařízení. Multiplatformního vývoje společnost dosahuje produkcí vizuálních nástrojů na vytvoření a konfiguraci uživatelského rozhraní, s možností optimalizace vyvíjené aplikace pro různé velikosti a tvary displejů. Klientská část aplikace je pak zajištěna pomocí nativního kontejneru a widgetů pro jednotlivé operační systémy. MicroStrategy taktéž nabízí například služby v oblasti správy uživatelů, zabezpečení či synchronizaci a uložení dat.

Služby této společnosti je vhodné využít při potřebě rychlé tvorby aplikace zabývající se business intelligence.

### Souhrn kladů:

- Nástroje umožňují tvorbu hezkého jednotného uživatelského rozhraní, které na pohled vypadá jako nativní.
- Aplikace lze vytvářet bez nutné znalosti jakéhokoliv programovacího jazyka.

### Souhrn záporů:

- Vizualizace aplikace je limitována a nedovolí významnější přizpůsobení.
- MicroStrategy není příliš známé svou mobilní platformou.
- Mobilní platforma je zaměřena především do oblasti business intelligence.

**3.2.4.5 Oracle** Korporace Oracle je další kandidát, pohybující se již dlouhou dobu mezi největšími společnostmi v oblasti informačních technologií. S miliardovými obraty a s počtem zaměstnanců převyšujícím sto třicet pět tisíc, je jednou z největších společností v tomto přehledu. Její specializací je vývoj a distribuce databázových technologií, firemního softwaru a cloudových inženýrských

---

<sup>8</sup>Pojem business intelligence, zkráceně také jako BI, popisuje množinu technik a nástrojů, které slouží k získávání a transformaci dat pro potřeby obchodních analýz.

systémů. Mezi její nejznámější produkty patří, kromě Oracle databáze, také databáze MySQL.

Firma Oracle vstoupila do mobilního odvětví svou službou Oracle Mobile Cloud Service a poté představením plánu na další služby týkající se mobilního vývoje. Tyto cloudové služby jsou od svého založení neustále rozšiřovány a vylepšovány. V roce 2013 také vznikl Oracle BI Mobile App Designer, jako součást balíku BI Foundation Suite. Tento designer umožňuje tvorbu analytických mobilních aplikací bez potřeby psát programový kód. Důležité potvrzení směru, kterým se korporace vydává, přišlo ke konci roku 2015, kdy bylo oznámeno silné partnerství se společnostmi Xamarin a Sencha. Z tohoto partnerství chtějí všichni jeho účastníci vytěžit lepší vzájemnou kooperaci jejich řešení a to jak v odvětví vývoje aplikací, tak i v testování výsledných aplikací. Druhému bodu jdou naproti propojením služeb Xamarin Test Cloud a Oracle Developer Cloud. Ze strany společnosti Sencha pak přichází zkušenosti v oblasti tvorby aplikací pomocí webových technologií.

Řešení firmy Oracle je vhodné především pro organizace, které mají již nějaké zkušenosti se softwarem firmy Oracle a chtějí využít znalostí jazyka Java.

#### **Souhrn kladů:**

- Firma má velké dosavadní výsledky v oblasti prodeje softwaru.
- Oracle ADF Mobile nabízí hybridní vývojové prostředí podporující jazyky Java a JavaScript.
- V poslední době Oracle uzavřel silná partnerství.

#### **Souhrn záporů:**

- Řešení společnosti Oracle neobsahuje back-end služby.

**3.2.4.6 Verivo Software** Firma Verivo je jednou z menších softwarových společností v tomto přehledu. Co se týče počtu jejich zaměstnanců, pohybuje se kolem jedné stovky. Při jejím vzniku v roce 1998 byla hlavním cílem společnosti tvorba aplikací pro přenosné zařízení PalmPilot. V roce 2012 firma oznámila, že disponuje prostředky od investorů ve výši sedmnácti milionů dolarů. Tyto prostředky se rozhodla využít k přeorientování své business strategie z vývoje a prodeje aplikací na nabízení mobilní platformy. V témže roce společnost také založila svou jedinou pobočku nacházející se mimo spojené státy americké, v anglickém Londýně.

V historii byla firma Verivo známá svou aplikací AppStudio. Ta sloužila pro vývoj metodou drag&drop a měla stovky zákazníků po celém světě. Tento směr vývoje však ukončila a nyní se zaměřuje na svou platformu Akula, založenou na jazyce Java. Ta nezahrnuje žádné front-end nástroje a namísto toho je navržena k využití frameworků a nástrojů třetích stran.

Jak dodává společnost Gartner [7], organizace by měly zvážit využití platformy Akula, pokud mají v plánu svou vyvíjenou multiplatformní aplikaci připojit k podnikovému systému. Pokud chtějí použít pro vývoj metodu drag&drop, měly by využít stále podporovanou aplikaci AppStudio.

#### Souhrn kladů:

- Společnost ukázala, že se nebojí reagovat na změny trhu.
- Společnost ukázala, že je schopná nabídnout služby velkým organizacím.

#### Souhrn záporů:

- Firma prošla velkou změnou a dosud není jisté, zda tento krok byl správným směrem.
- Platforma dosud nepodporuje mobilní operační systém Windows.

**3.2.4.7 Zebra Solutions** Řešení RhoMobile Suite je založeno na frameworku Rhodes, dostupným jako open-source pod licencí MIT. Nyní je celé řešení pod taktovkou světové společnosti Zebra Technologies, která se zaměřuje především na technologie tiskáren čárových kódů a osobních identifikací. K září roku 2014 zaměstnávala firma přes sedm tisíc zaměstnanců a její roční příjem činil milion dolarů. K mobilním technologiím však dospěla až akvizicí společnosti Motorola Solutions v roce 2014. Motorola Solutions vznikla z dříve velmi známé společnosti Motorola, založené v roce 1928. Známa byla jak svými mobilními telefony, tak širokou telekomunikační sítí, kterou vlastnila.

Jádrem společnosti v oblasti mobilního vývoje je tedy momentálně balík služeb a nástrojů RhoMobile Suite. Jeho vznik se datuje do roku 2008, kdy byl Adamem Blumem založen framework Rhodes. Již v roce 2009 získal tento projekt důležité ocenění „Best Start Up Company“ a i díky tomu byl celý start up<sup>9</sup> koupen společností Motorola Solutions. Celé řešení obsahuje kromě jiného také produkt RhoStudio. Ten, jako rozšíření nástroje Eclipse, slouží pro tvorbu multiplatformních mobilních aplikací. Dále pod RhoMobile spadá také nástroj RhoElements, HTML5 framework a kontejner pro nativní a hybridní mobilní aplikace, jenž byl díky Zebra Technologies obohacen také o podporu Windows Mobile a Windows CE systémů. Dále nástroj RhoConnect umožňující offline synchronizaci dat a back-end integraci služeb na mobilním zařízení a také nástroj RhoHub – cloudová služba umožňující vývoj nativních aplikací bez nutnosti jednotlivých SDK samotných operačních systémů.

Dle analýzy společnosti Gartner [1], je vhodné celé řešení RhoMobile Suite použít v případě, že se potřebujeme starat o velké množství dat, potřebujeme back-end zázemí či máme široký rozptyl v oblasti cílového hardware.

---

<sup>9</sup>Jako start up je označována firma, jenž je na počátku svého vzniku. Často také disponuje cizími prostředky.



### Souhrn kladů:

- RhoElements podporuje velkou škálu mobilních zařízení.
- Zebra Technologies má díky Motorola Solutions velké zkušenosti v oblastech maloobchodu, výroby, inženýrských sítí a dalších významných prostředích.

### Souhrn záporů:

- Životní cyklus vývoje mobilní aplikace není zcela úplný, chybí například podpora fáze prvotního návrhu aplikace, automatické testování či povývojová analýza.
- Neustálé přesuny původní firmy pod jiné vedení mohou působit na celé řešení spíše negativně.

## 3.3 Srovnání

V tabulce 1 lze vidět, že všechny frameworky uvedené v tomto přehledu, nabízejí tvorbu aplikací pro dvě nejpoužívanější platformy iOS a Android. Ne všechny však podporují mobilní operační systém Windows. Proto je vhodné zvážit, jak velké procento potenciálních uživatelů bude tento operační systém využívat. Ve formě vývoje převládají webové technologie, konkrétně HTML5, většinou v kombinaci se skriptovacím jazykem JavaScript. Druhou, často použitou technologií, je drag&drop, v něm však často narazíme na větší omezenost vývoje. Další programovací technologie, tedy jazyky C# a Java, jsou pak v přehledu zastoupeny jednotlivě.

Při specifikaci možného segmentu aplikací, jež mají být ve frameworku vyvíjeny, je zde kromě všeobecných aplikací silně zastoupen podnikový segment. Je tomu tak z důvodu velké poptávky ze stran firem na jednoduché nástroje pro portování interních aplikací či jednoduchých aplikací pro zákazníky do mobilní podoby. Specializované společnosti nabízejí i tvorbu multiplatformních mobilních aplikací čistě pro segment, který je hlavním zájmem jejich businessu. Jedná se především o rozšíření jejich dosud nabízených služeb do sféry mobilních zařízení.

Co se týče předposledního sloupce tabulky, obsahuje rychlý přehled, jak dlouho existuje firma a také kolik let se zabývá mobilním vývojem. Druhý údaj je zde především kvůli tomu, že hodně větších firem nabízí řešení pro multiplatformní vývoj od doby, kdy koupili menší společnost tímto vývojem se zabývajících. Počty zaměstnanců v tabulce pak mohou posloužit jako ukazatel velikosti společnosti.

Pokud přistoupíme k samému srovnání, je téměř nemožné určit absolutního vítěze či poraženého. Takové určení záleží na individuálních potřebách vývojáře a na dalších mnoha faktorech popsanych výše. Pro konkrétnější představu je níže uvedeno několik vzorových scénářů s typy, jaká společnost se k daným podmínkám nejvíce hodí.

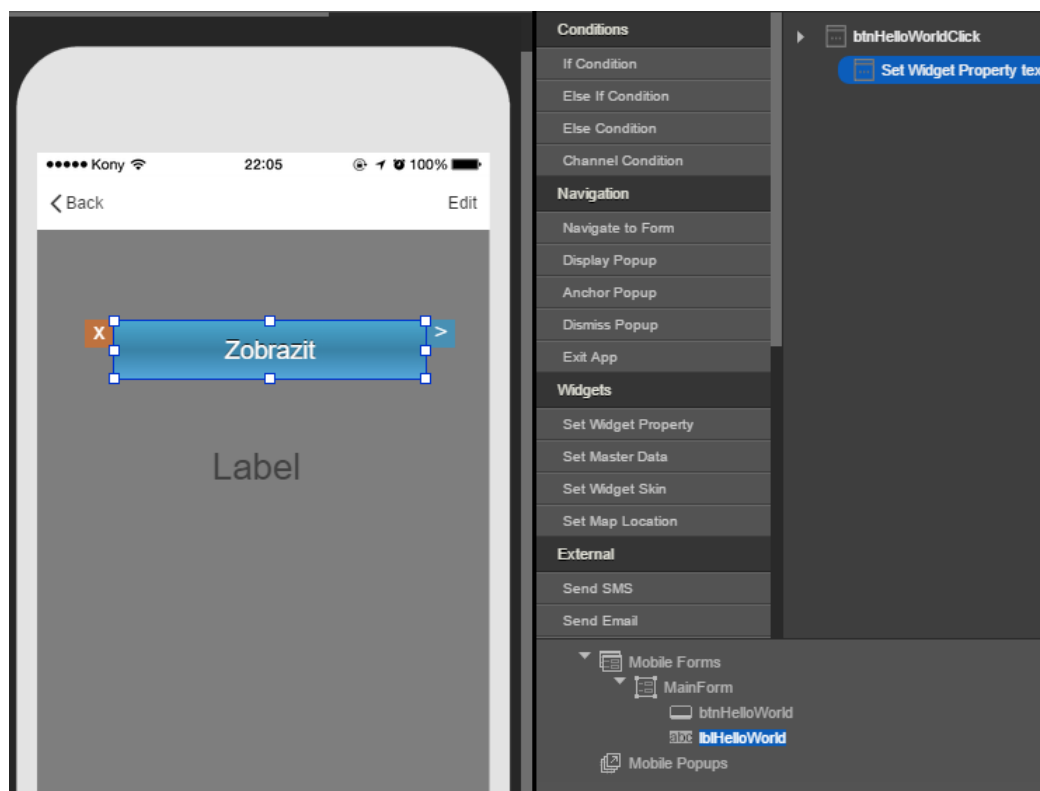
Společnost	Platformy			Vývoj	Zaměření	Zkuš.	Zam.
	iOS	And.	Win.				
Adobe PhoneGap	ano	ano	ano	HTML5	obecné	34/7	500
Appcelerator	ano	ano	ano	HTML5	obecné	9/9	500
Backbase	ano	ano	ano	D&D	marketing	13/3	500
ClickSoftware	ano	ano	ano	D&D	pracovní síla	37/3	1000
DSI	ano	ano	ano	D&D	transport	37/3	500
Embarcadero	ano	ano	ano	HTML5,D&D	obecné	23/4	500
IBM	ano	ano	ano	HMTL5	obecné	104/3	10000+
Kony	ano	ano	ano	HTML5,D&D	obecné	9/9	5000
MicroStrategy	ano	ano	ne	D&D	podnikové	27/3	5000
Zebra Techno.	ano	ano	ano	HTML5	podnikové	87/8	10000+
Oracle ADF Mob.	ano	ano	ne	HTML5,Java	podnikové	38/3	10000+
Pegasystems	ano	ano	ne	D&D	podnikové	33/19	5000
Sencha Touch	ano	ano	ano	HTML5	obecné	9/6	200
Progress Software	ano	ano	ano	HTML5,D&D	obecné	14/5	5000
Xamarin	ano	ano	ano	Csharp	obecné	5/5	500

Tabulka 1: Nástroje pro multiplatformní vývoj

### 3.3.1 Informativní aplikace pro potenciální zákazníky

Představme si, že jsme firma, která má vlastní webové stránky. Ku příležitosti vydání nového produktu na sebe chceme upozornit mobilní aplikací, jež o něm bude informovat. Taková aplikace se bude skládat pouze ze statických stránek. Ty budou obsahovat různé texty, obrázky a tlačítka, pomocí kterých půjde přecházet mezi různými obrazovkami aplikace. Aplikace nebude využívat žádnou databázi, ani nebude přistupovat k žádným funkcím telefonu, jako jsou například notifikace či fotoaparát. Naše firma není primárně zaměřená na vývoj aplikací a její business spočívá v jiném odvětví.

Kvůli poslednímu faktu v našem vzorovém scénáři se nám počet potenciálních společností pro multiplatformní vývoj značně redukuje. Nemáme vývojáře, kteří by uměli psát programový kód, proto je nutné se zaměřit na společnosti, u kterých lze vytvářet aplikace pomocí technologie drag&drop. Jelikož chceme vytvářet obecnou aplikaci, výběr se nám zužuje mezi společnostmi Embarcadero, Kony a Progress Software. Firma Embarcadero je vhodná spíše na větší projekty a upřednostní ji ti, jež mají s jejími produkty už nějakou zkušenost, případně využívají některé jejich serverové služby. Kvůli těmto faktům a jejímu cenovému modelu se pro náš scénář Embarcadero příliš nehodí. Progress Software a jejich platforma Telerik obsahuje mnoho nástrojů, ale bez znalosti JavaScriptového jazyka se nám opět nepodaří vytvořit plnohodnotnou aplikaci. Zůstává tedy poslední možnost a to Kony. Díky velkému množství dostupných widgetů a nástroji Vizualizer, jsme schopni navrhnout naši cílovou aplikaci s minimální znalostí kódování (obr. 6). Proto je právě tato volba tou správnou.



Obrázek 6: Aplikace Kony Visualizer pro vývoj metodou drag&drop

### 3.3.2 Aplikace pro zadávání docházky zaměstnanců

V dalším scénáři jsme opět firma, avšak nyní již vývojářská. Aktuálně máme novou poptávku, jejíž zadáním je tvorba aplikace pro správu docházky zaměstnanců firmy. Firma má existující systém docházky, který nyní aktivně využívá, avšak její vedení se nebrání kompletně novému systému.

Vzhledem k tomu, že se tentokrát jedná o vývojářskou firmu, jsou naše možnosti oproti předchozímu scénáři znatelně větší. V první řadě je důležité zjistit, jaké technologie by bylo nejjednodušší použít. K tomu je zapotřebí si nejprve určit naše firemní technologické možnosti. Po tomto kroku je také důležité prozkoumat aktuální stav systému docházky u zákazníka. Pokud bychom díky našim možnostem a zjištěnému stavu ve firmě byli sjednoceni vývojovou platformou, mohl by se nám výběr frameworku značně zúžit.

Výběr firmy si představíme opět na omezeném vzorku společností, prezentovaném naší tabulkou. Pokud bychom takovouto aktuální tabulku neměli, je vhodné ji v nějakém stavu, pomocí průzkumu trhu, sestavit. Kvůli tomu, že jsme vývojářská firma a lze předpokládat, že ve vybraném frameworku budeme vyvíjet i nadále, je více než vhodné vyloučit veškeré frameworky, které mají jiné než obecné zaměření. Mezi takto selektovanými společnostmi pak bude dozajista dalším důležitým krokem výběr jazyka, který pro vývoj podporují. Pokud nám z dřívější analýzy vplynuly konkrétní technologie, zkusíme se jim naším výbě-

rem co nejvíce přiblížit. Pokud tomu tak není, zbývají nám tři směry, kterými se můžeme vydat.

První z nich je styl vývoje použitý v předchozím scénáři, tedy drag&drop. Díky našim možnostem, které nám umožňují využít i částečného vývoje pomocí jazyka HTML, se dá kromě výše vybraného frameworku Kony, využít také konkurenční řešení společnosti Embarcadero či Progress Software. Všechny tyto společnosti nabízejí tvorbu uživatelského rozhraní aplikace pomocí systému drag&drop, což má například tu výhodu, že tvorba aplikace může být škálovatelná na její design a aplikační logiku. Pokud nám tento směr vyhovuje, je vhodné vybírat z těchto tří společností.

Dalším možným řešením je použití programovacího jazyka C#, tedy frameworku Xamarin. Díky tomuto frameworku budeme vytvářet hybridní aplikace, jež budou při kompilaci převedeny do nativního kódu cílového operačního systému. Xamarin je rapidně rostoucí firma odkoupena v únoru roku 2016 gigantem Microsoft. I přesto, že Microsoft vyvíjí svůj vlastní mobilní operační systém Windows, dá se očekávat, že akvizice společnosti Xamarin nebude v ohledu multiplatformního vývoje nijak naškodu. Lze tomu tak soudit i kvůli politice, kterou Microsoft během posledního roku razí. Xamarin již v minulosti dokázal značné schopnosti rychlé reakce na změny na trhu a i díky jednomu a půl miliónu vývojářů, kteří ho využívají, je pro tento scénář vhodnou volbou.

Posledním směrem, kterým se ve vývoji můžeme vydat, je využití čistě HTML5 technologií. Tím lze výběr zúžit mezi Adobe PhoneGap, Appcelerator, IBM a Sencha Touch. Z tohoto seznamu lze vypíchnout řešení Adobe PhoneGap, respektive jeho open-source variantu Apache Cordova. Právě díky povaze open-source a tedy i faktu, že je zdarma, čelí vysoké oblibě a naše firma tak ušetřené peníze může využít například pro zdokonalování samotné vyvíjené aplikace. Je však důležité zjistit, zda nebudeme potřebovat využít i některé doplňkové služby pro správu aplikace či databáze. V tom případě by bylo vhodné využít některou ze zbylých platforem, které nabízejí služby pro celý životní cyklus aplikace.

## 4 Xamarin

Ze všech společností, které jsme doposud představili, je nyní vybrána jedna konkrétní pro podrobnější analýzu. Tou je právě společnost Xamarin především díky způsobu fungování frameworku, který nabízí. Ten jde svou vlastní cestou a mezi zbylými řešeními se na první pohled vyjímá.

### 4.1 Historie

Za vznikem společnosti Xamarin stojí především dva milníky v historii jejich současných vlastníků. První z nich byl vznik společnosti Ximian. Ta byla založena mexickým programátorem Miguelem de Icaza a jeho americkým přítelem Natem Friedmanem. Miguel je nyní známý především svou významnou rolí při tvorbě unixového prostředí GNOME, i proto firma Ximian pracovala především na tvorbě business aplikací pro toto prostředí. Jedním z jejích největších cílů však bylo vytvořit open-source sadu nástrojů, které umožní unixovým uživatelům pracovat s programovacím jazykem C#, tedy jazykem, jež vynalezla konkurenční společnost Microsoft. Pomocí těchto nástrojů pak chtěli vyvíjet aplikace v prostředí kompatibilním s .NET Framework. Tento projekt byl nazván Mono.

Především právě díky projektu Mono se pak odehrál druhý významný milník v historii společnosti Xamarin. V dubnu roku 2003 prohlásil americký softwarový gigant Novell, že společnost Ximian kupuje. Ta tak získala nové finanční zdroje a možnosti pro vývoj svých projektů a v roce 2004 oznámila první překladač v jazyce C# verze 1.0.

Samotný vznik společnosti Xamarin pak zapříčinila akvizice Novellu firmou Attachmate. V tomtéž roce tak Miguel de Icaza oznámil, že projekt Mono bude vyvíjen nově vzniklou společností Xamarin a Novell jí přenechá část svých licenčních práv. V této chvíli již společnost nabízela také vývojové rozhraní MonoDevelop a rozšíření Mono for Android pro tvorbu mobilních aplikací. Velkým krokem pak bylo oznámení o platformě Xamarin ve verzi 2.0. Ta přinesla kromě nástupce MonoDevelop Xamarin Studio také rozšíření pro tvorbu na platformě Xamarin (Mono) pro Visual Studio a také komplexní tvorbu aplikací pro platformy Android, iOS a Microsoft.

### 4.2 Technologické možnosti

Samotný Xamarin v základu umožňuje tvorbu jak multiplatformních, tak i nativních aplikací. Nativní aplikace je možné vyvíjet pomocí dostupných knihoven a hodí se v případě, kdy víme, že náš vývoj bude zaměřen primárně na jednu platformu. Další scénář využití může být takový, kdy chceme pro každou z platform vyvíjet specifický vzhled aplikace. Oproti vývojovým nástrojům, jež jsou dodány přímo výrobcí operačních systémů, spočívá hlavní výhoda Xamarinu v možnosti využití programovacího jazyka C# a prostředí .NET. S využitím tohoto faktu tak lze vyvíjet aplikace pro platformy iOS, Android a Windows. Poslední jmenovaný však podporuje psaní v jazyce C# a platformu .NET již přímo od výrobce.

Nad těmito knihovnamy je pak z části postaven i multiplatformní vývoj.

Mezi hráče v soubojích o oblíbený multiplatformní nástroj se Xamarin dostal se svým řešením Xamarin Forms. Jeho podstata, což je podstata i samotného multiplatformního vývoje, spočívá ve využití co největšího množství programového kódu napříč všemi cílovými platformami. Jak udává Charles Petzold [9], Xamarin Forms využívají dvě různá řešení pro sdílení programového kódu (obr. 7). Hlavní rys, ve kterém se tato dvě řešení liší, spočívá v různém provedení zápisu platformně specifického kódu. Takový kód se týká především komunikace s funkcemi zařízení, jako je například fotoaparát či GPS lokátor.

Prvním systémem, který využívají Xamarin Forms, se nazývá Shared Project. Jeho stavebním kamenem je využití jednoho projektu, ve kterém bude veškerý programový kód. Tento kód se následně zkompile do strojového kódu cílové platformy. Pokud potřebujeme psát programový kód specifický pro platformu, je nutné tento kód oddělit pomocí určitých direktiv, ty jsou ve své deklaraci označeny cílovou platformou. Projekt, jenž je psán formou sdíleného projektu, není možné kvůli systému direktiv zkompile do DLL knihovny a tu pak použít v úplně jiném projektu.

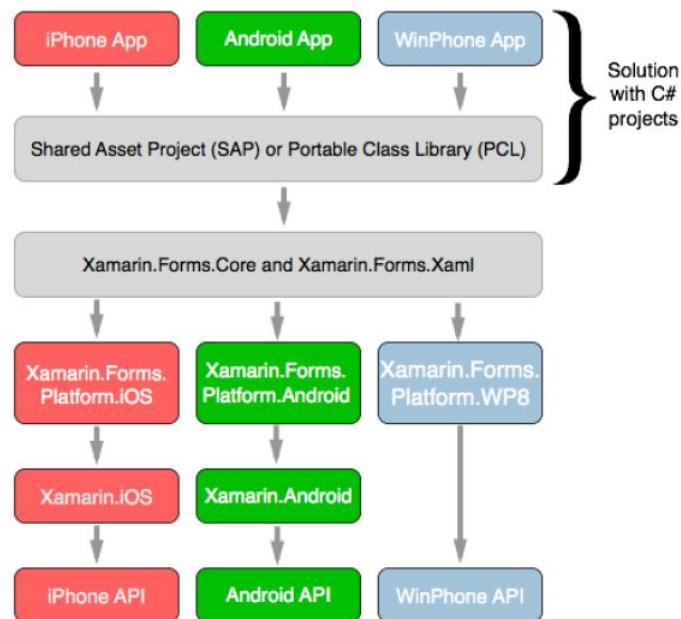
Pokud vidíte v řešení pomocí sdíleného projektu dosti nevýhod, nabízí Xamarin Forms také další řešení nazvané Portable Class Library. I toto řešení obsahuje jeden projekt, v němž je obsažen kód pro všechny vyvíjené platformy. Tento projekt však neobsahuje takový kód, který je psán specificky pro konkrétní platformy. Projekt je pak následně zkompile do knihovny, která je připojena k ostatním platformně specifickým projektům. Kód pro určitou platformu je pak vytvářen následujícím stylem. V hlavním projektu je deklarováno jedinečné rozhraní obsahující metody, s nimiž bude možné v tomto projektu pracovat. Toto rozhraní je následně dle potřeby implementováno v jednotlivých projektech platformem. Projekt, jenž je napsán tímto způsobem, je možné následně zkompile do DLL knihovny a tu dále používat.

### 4.3 Uživatelská základna

Xamarin jako takový hrdě hlásí, že aktuální stav vývojářů, kteří ho používají nebo alespoň jednou použili, se blíží k jeden a půl miliónu. Mezi tyto vývojáře se řadí jak lidé, jež vyvíjejí multiplatformní aplikace pomocí Xamarin Forms, tak také lidé, kteří vyvíjejí nativní aplikace pomocí knihoven Xamarin.Android či Xamarin.iOS.

Vývojáři mohou přispívat také do diskuzního fóra na webových stránkách společnosti Xamarin. Ke konci března roku 2016 toto fórum obsahovalo přes čtyřicet šest tisíc otázek na téma Xamarin Forms, z toho jich je kolem čtrnácti procent dosud nezodpovězených. Zmíněné procento tak ukazuje, že si společnost drží stabilní základnu uživatelů, kteří jsou ochotni reagovat a poskytovat rady méně zkušeným začínajícím uživatelům.

Dalším důležitým faktorem jsou reference zákazníků. Xamarin má mezi svými referencemi, jež využívají jeho technologie, velké hráče jako je Honeywell, U.S.



Obrázek 7: Princip fungování Xamarin Forms

Zdroj: [azurevietnam.net](http://azurevietnam.net)

AIR FORCE či Bosch.

#### 4.4 Potenciální rozvoj

Díky akvizici společností Microsoft se role Xamarinu na trhu mobilních frameworků značně posílila. Dalo by se očekávat, že se tím Xamarin více zaměří na zlepšování interakce s produkty společnosti Microsoft a také zdokonalí překlad Xamarin Forms do nativní aplikace spustitelné na mobilních systémech Windows.

Dalším krokem společnosti bude zřejmě co nejtěsnější spojení mezi výrobci mobilních operačních systémů Android a iOS. Tento krok je vítaný především kvůli silné závislosti na právě jimi vydaných novinkách a obecných změnách. Xamarin již v dnešní době získává přístup k takovýmto změnám dříve než obecná veřejnost. I přesto ale stále musí počítat s tím, že s každou změnou v operačních systémech může nastat mnoho chyb v aplikacích, jež byly pomocí Xamarin Forms vytvořeny. Ty bude následně potřeba najít a v co nejkratší době odstranit.

Očekávat lze také postupné zvyšování počtu volně dostupných rozšíření frameworku. Tím se znatelně zvýší možnosti a rychlost vývoje.

## 5 Apache Cordova

Jako druhá společnost vybraná pro hlubší analýzu zde figuruje Apache Cordova. Mezi zbylými zaujme především tím, že nad jejím řešením je postavené velké množství ostatních frameworků představených v přehledu. Je tak velká pravděpodobnost, že uživatel právě s Apache Cordova přijde při multiplatformním vývoji do styku.

### 5.1 Historie

Historie tohoto frameworku začíná v roce 2009, kdy byl představen na konferenci iPhoneDevCamp v San Franciscu jako projekt s názvem PhoneGap. V této době byl vyvíjen softwarovou firmou Nitobi a téměř okamžitě si získal přízeň jak uživatelů, tak i velkých hráčů jako je například Apple. Po necelých dvou letech došlo k akvizici Nitobi společností Adobe. Zároveň s tím, především pro zachování otevřenosti frameworku, byl vytvořen projekt Apache Cordova. Původní projekt PhoneGap nadále pokračoval pod svým jménem, nicméně nebyl již dostupný zdarma. Rozdíl mezi těmito řešeními pak tvoří především doplňkové služby, které PhoneGap nabízí.

### 5.2 Technologické možnosti

Apache Cordova je typickým zástupcem směru vývoje, kdy jsou používány webové technologie. Princip fungování aplikace pak spočívá v tom, že na mobilním zařízení je použit nativní kontejner pro zobrazení uživatelského rozhraní (obr. 8). Jedná se o ten stejný kontejner, pomocí kterého zařízení zobrazují webové stránky. Výsledné aplikace lze přeložit do mobilních operačních systémů iOS, Android, BlackBerry, Windows Phone, Palm WebOS, Bada a Symbian.

#### 5.2.1 Architektura MVC

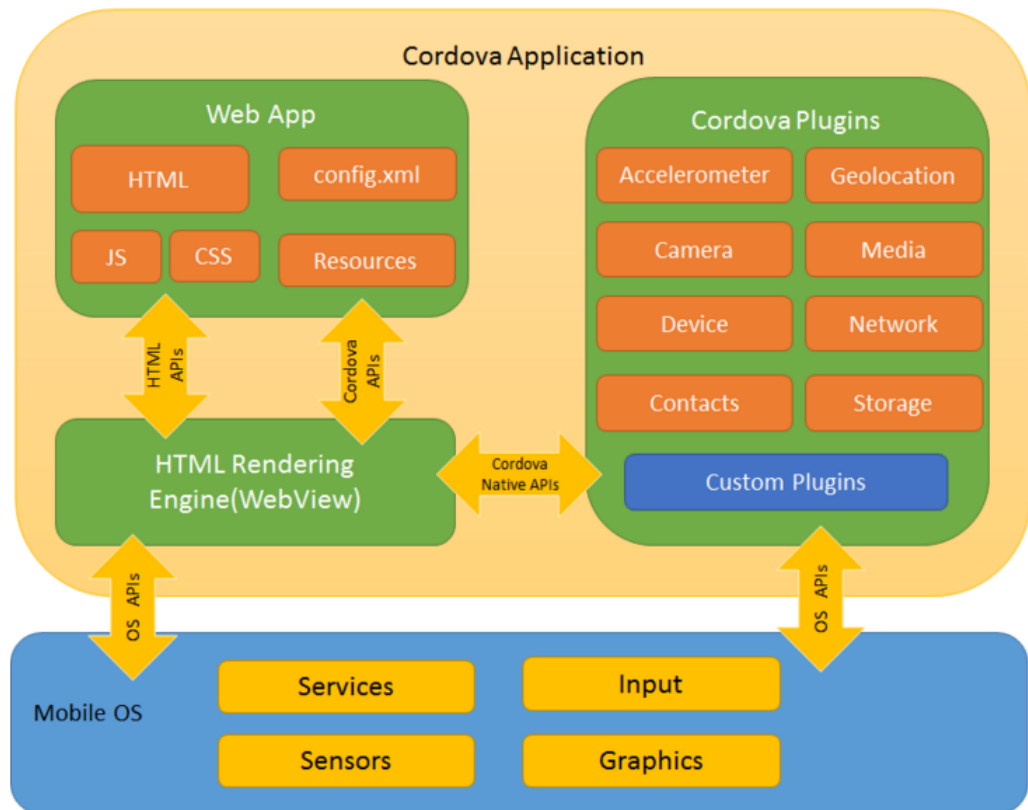
Základním problémem při vývoji ne-primitivních webových stránek je míchání prezentační a logické vrstvy. To způsobuje nepřehlednost kódu a s tím spojené například složitější hledání a odstranění problémů. Řešením je využití architektury MVC. Architektura byla poprvé popsána v roce 1979 a její první použití je zaznamenáno v programovacím jazyce Smalltalk, vyvíjeným firmou Xerox. Nyní je architektura hojně využívána především ve webovém vývoji. Jejím základem je zmiňované oddělení aplikační logiky od prezentační vrstvy. Toho je dosaženo využitím tří různých druhů komponent a to modelů, pohledů a kontrolérů.

Model je chápán jako část aplikace, jež obsahuje veškerou aplikační logiku. Model neví nic o výstupu aplikace a částečně neví, odkud mu data přišla. Pokud aplikace využívá ORM<sup>10</sup>, jednotlivé modely odpovídají jednotlivým tabulkám.

---

<sup>10</sup>ORM je zkratka sousloví objektově relační mapování. Jedná se o označení programovací techniky, která zajišťuje konverzi dat mezi relační databází a programovacím jazykem





Obrázek 8: Princip fungování Apache Cordova

Zdroj: [www.itnose.net](http://www.itnose.net)

Pokud tomu tak není, lze vytvořit modely jako manažery, kteří pak nadále pracují s jednotlivými tabulkami v databázi.

Druhý komponent – pohled, zobrazuje výstup uživateli. V mobilním vývoji pomocí webových technologií se tak jedná o HTML stránku, která obsahuje určité specifické tagy značkovacího jazyka. Samotné pohledy obsahují minimální množství logiky, omezené především na jednoduché podmínky. Pohled neví a nijak neřeší, odkud mu přišla jakákoliv data.

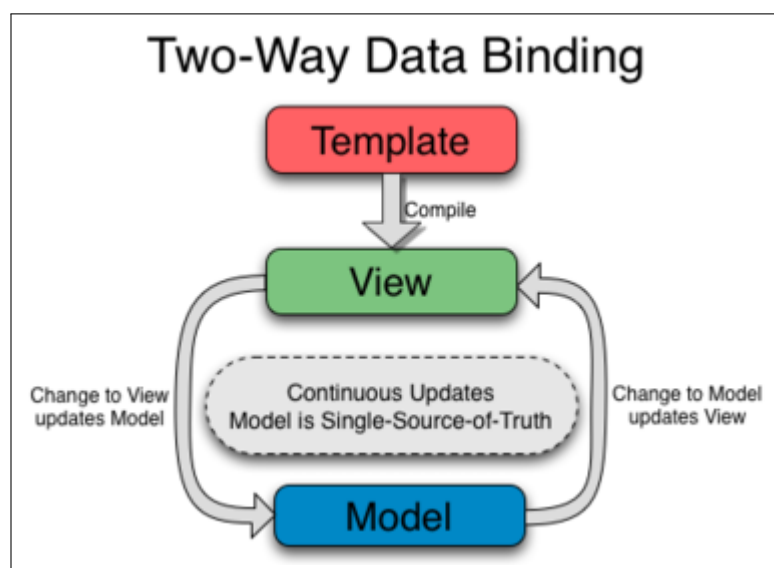
Posledním komponentou je kontrolér. Ten slouží jako prostředník mezi modelem a pohledem. Obsahuje také téměř veškerou aplikační logiku konkrétního modelu. Často používaným scénářem je, že každý pohled má svůj vlastní kontrolér.

### 5.2.2 AngularJS

Samotný Apache Cordova lze díky javascriptovému založení rozšířit o další technologie, které zjednodušují vývoj samotných aplikací. Při prvním pohledu na styl tvorby webových aplikací pomocí programovacího jazyka JavaScript napadne začínajícího programátora jeden základní nedostatek pro moderní programování.

Jazyk je to totiž spíše imperativní než deklarativní. Mnoho věcí se zde tak řeší opakovaně manuálně, i když by mohli být řešeny deklarativně, například webovým prohlížečem. Pokud člověk hledá javascriptový webový framework, který tento problém z určité části odstraňuje, a láká ho vyzkoušet architekturu MVC popsanou výše, nabízí se hojně využívaný framework AngularJS. Jedná se o volně dostupný framework pod licencí MIT, který byl založen v roce 2009 firmou Google. Hlavním lákadlem technologie se stal dobře řešený Two Way Data-Binding, neboli systém, jakým jsou předávána data mezi modelem a pohledem (obr. 9). Jak název napovídá, synchronizace je řešena dvoucestně. Během první cesty je uživatelský vstup, což může být například kliknutí na tlačítko nebo zápis textu do pole, přenesen do návazného modelu. Během druhé cesty je pak tento vstup z modelu distribuován do všech návazných pohledů.

Dalším důležitým přínosem do webového vývoje je ze strany AngularJS implementace návrhového vzoru Dependency Injection. Ten řeší závislosti mezi jednotlivými komponentami programu, tyto komponenty jsou nazývány jako služby. Framework podle parametrů kontroléru sám rozpozná, s čím bude v rámci kontroléru potřeba pracovat a podle toho vytvoří automaticky potřebné závislosti. Tímto stylem zápisu lze při testování také na první pohled rozpoznat, kde jsou využívány jaké služby.



Obrázek 9: Two Way Data-Binding v Apache Cordova

Zdroj: [docs.angularjs.com](https://docs.angularjs.com)

### 5.2.3 Ionic Framework

Když už jsme si představili architekturu MVC a framework AngularJS, který ji používá, zbývá uvést poslední část, která nám představené technologie spojí s námi použitým frameworkem Apache Cordova. Právě tímto posledním článkem řetězce je rozšiřující JavaScriptový framework Ionic. Jedná se o open-source

SDK pro tvorbu hybridních mobilních aplikací, postavené na výše představeném AngularJS frameworku. Právě díky jemu dokáže využívat hardwarovou akceleraci zařízení a tím je pro mobilní vývoj více než vhodný. Ionic ve svém balíčku mimo jiné nabízí také předpřipravené šablony aplikací. Ty obsahují vlastní Ionic komponenty, které se používají pomocí direktiv vytvořených ve frameworku AngularJS. Zajímavou součástí Ionic frameworku je také volně dostupná sada ikon obsahující přes šest stovek různých kusů ikon laděných pro konkrétní mobilní operační systémy.

Ionic je pouze jeden z mnoha mobilních frameworků, avšak právě díky využití AngularJS se těší velké oblibě a zřejmě ho čeká ještě dlouhá budoucnost. Mezi jeho alternativy lze řadit například Sencha Touch, jQuery Mobile či Kendo UI.

#### 5.2.4 Ripple emulátor

Stavebním kamenem vývoje mobilních aplikací je jejich testování a ladění na konkrétních zařízeních. U mobilního vývoje je tato část snad nejdůležitější, neboť každé zařízení se chová přece jen trochu jinak. Pokud k tomu přičteme ještě fakt, že mobilní aplikaci vyvíjíme multiplatformně, dostáváme se do obrovského kolotoče operačních systémů, jejich různých verzí a také různých implementací těchto verzí. Abychom mohli zamezit chybám, nezbyvá nám nic jiného, než námi vyvíjenou aplikaci otestovat na co největším počtu zařízení.

Emulátor Ripple vznikl jako menší projekt firmy Apache a jeho cílem je nabídnout co největší množství testovacích scénářů pro mobilní HTML5 aplikace. Díky tomu, že jeho funkčnost je závislá na jádru prohlížeče Chrome, spočívá jeho instalace pouze v přidání rozšíření do internetového prohlížeče. Kvůli této strategii je použití velmi snadné a především, oproti emulátorům dodávaných výrobci operačních systémů, nesrovnatelně méně náročné na výpočetní výkon. Pro ještě větší zjednodušení umožňuje Apache Cordova spouštět vyvíjenou aplikaci v tomto emulátoru přímo z vývojového prostředí. Není tak nutná předchozí instalace samotného webového rozšíření.

Přesto, že se emulátor nachází stále ve verzi beta, obsahuje již nyní téměř čtyři desítky různých zařízení, které dokáže emulovat. Každé zařízení se, kromě v operačním systému, liší především svým rozlišením displeje. Z nabídky lze tak vybírat od WVGA rozlišení až po FullHD rozlišení se systémy Windows, Android, BlackBerry či iOS. Kromě možnosti změny orientace displeje můžeme také emulovat základní hardwarové funkcionality mobilního zařízení. Mezi ně patří akcelerometr umožňující nám pohybem myši simulovat naklonění zařízení, či například simulace existujícího připojení k internetové síti a GPS lokátoru.

### 5.3 Uživatelská základna

Vzhledem k faktu, že v případě Apache Cordova se jedná o open-source projekt, je těžké zjistit konkrétní počty vývojářů a aplikací, které tento framework využívají. Poslední čísla z května 2015 mluví o tom, že na tomto frameworku je postaveno zhruba pět a půl procenta všech aplikací, jež jsou dostupné v obchodě Android

Play. Apache Cordova nevlastní ani diskuzní fórum, avšak na stránkách odkazuje na známou technickou poradnu [stackoverflow.com](http://stackoverflow.com). Na této poradně je k březnu 2016 napsáno čtyřicet tři tisíc dotazů týkajících se tohoto frameworku. Z toho jich je čtyřicet šest procent nezodpovězených. Toto procento je poměrně veliké a je dáno zřejmě právě nevýdělečností celého frameworku, kdy si vývojáři nemohou dovolit placené konzultanty, kteří by se o dotazy zákazníků staraly. Z přehledu příspěvků však lze vyčíst trend neustálého zvyšování počtu odpovědí.

## 5.4 Potenciální rozvoj

Díky své otevřenosti je projekt velmi oblíbený a drží si stabilně silnou pozici na trhu. Framework je vzájemně propojený s projektem PhoneGap, pod záštitou softwarového giganta, firmy Adobe. Nedá se tak zřejmě očekávat další rozvoj co se partnerů týče. Je důležité, aby v co největší míře framework nadále podporoval veškeré usnesené či zaběhnuté standardy v jazycích HTML5 a JavaScript. Pokud by tomu tak nebylo a vzhledem ke konkurenci, která panuje, by existence frameworku neměla dlouhého trvání.

Kritizovanou částí frameworku je jeho výkon. Často se tak stává, že uživatelé některých zařízení, především těch s operačním systémem Android, mají problém s plynulostí na tomto frameworku postavených aplikací. Lze tak očekávat, že změny ve frameworku se budou v nejbližší budoucnosti týkat právě odstranění tohoto problému.

## 6 Tvorba testovací aplikace

Aby bylo možné zvolené frameworky porovnat, je vhodné využít praktického testování pomocí vývoje vzorové aplikace. Aplikace by měla využít co nejvíce možností, které frameworky nabízejí, jako jsou šablony či různá rozšíření. Z finančního hlediska není možnost plně otestovat většinu doplňkových služeb, při tvorbě aplikace a v následném porovnání frameworku tak na ně nebude brán příliš zřetel.

### 6.1 Téma

Hlavní cíl mobilní aplikace, jež je vyvíjena v rámci této diplomové práce, je pomoc se ztrátou zvířete jak jeho majiteli, tak i zvířeti samotnému. Pokud přihlídneme k tomu, že v takovýchto případech jde hlavně o čas, bylo při vývoji aplikace bráno zřetel především na rychlost, jednoduchost a intuitivitu ovládání. Aplikace jako taková je propojena, skrz webový server, s databází. Tato databáze pak obsahuje veškeré informace o ztracených nebo nalezených zvířatech zadané jednotlivými uživateli aplikace. Záznamy může uživatel s využitím aplikace prohlížet a případně na ně zareagovat pomocí uvedeného kontaktu. Základním stavebním kamenem aplikace je také zjednodušení vyhledávání pomocí spolupráce s GPS lokátory, které jsou obsaženy téměř ve všech moderních mobilních zařízeních.

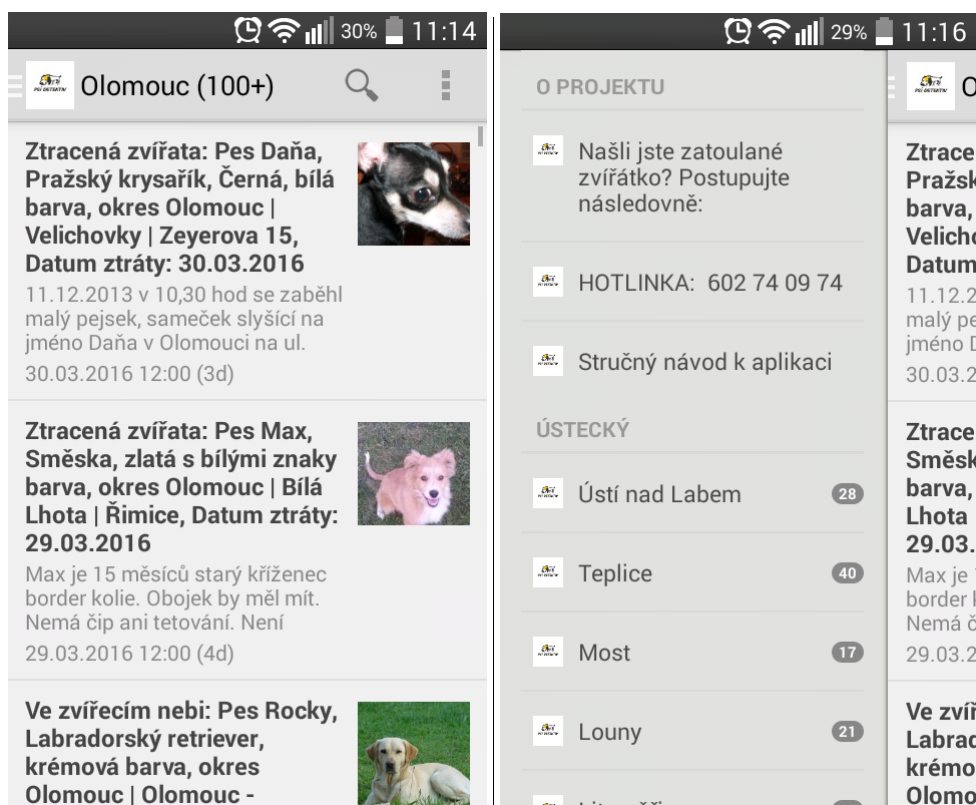
### 6.2 Existující řešení

Jediným existujícím řešením zabývajícím se obdobnou tematikou je aplikace Psí detektiv. Aplikace je dostupná pouze pro operační systém Android a slouží jako prostředek k zobrazení databáze ztracených psů a koček, ze stránek psidetektiv.cz. Samotná aplikace má dosti omezené možnosti. Pokud pomineme samotnou neexistenci aplikace na konkurenční platformy, je hlavním nedostatkem fakt, že z aplikace není možné přidávat nové záznamy o ztrátách či nálezech zvířete. Pro tyto případy je zde dostupný pouze textový návod, jak by měl uživatel v případě ztráty či nálezu postupovat.

Základní ovládání aplikace se odvíjí od rozdělení všech záznamů do krajů a okresů (obr. 10). Mezi nimi pak lze pomocí bočního menu přecházet. Po kliknutí na konkrétní oblast je zobrazen textový seznam, ve kterém jsou pod sebou jednotlivé záznamy o psech a kočkách. Tyto záznamy nelze filtrovat pomocí konkrétního parametru, dostupné je pouze obecné fulltextové vyhledávání. Aplikace má ke dni 3.4.2016 na Google Play Store hodnocení čtyř bodů z pěti a počet instalací mezi pěti sty až tisícem.

### 6.3 Vývoj

Aplikace komunikuje během své činnosti se vzdáleným API serverem. Díky tomu jsou synchronizovány databáze aplikace napříč všemi jejími uživateli. Vzhledem



Obrázek 10: Seznam zvířat a navigace v aplikaci Psí detektiv

k tomu, že se tvorba API serveru tématem značně liší s touto diplomovou prací, byla jeho tvorba přenechána firmě GMC Software Technology[2]. Samotná aplikace je již připravena na připojení k takto fungujícímu serveru. Ze serveru pak budou zaslány i notifikace s informacemi o nových záznamech. Plnohodnotné funkčnosti aplikace tak bude dosaženo až s dokončením vývoje API serveru a jeho následným připojením.

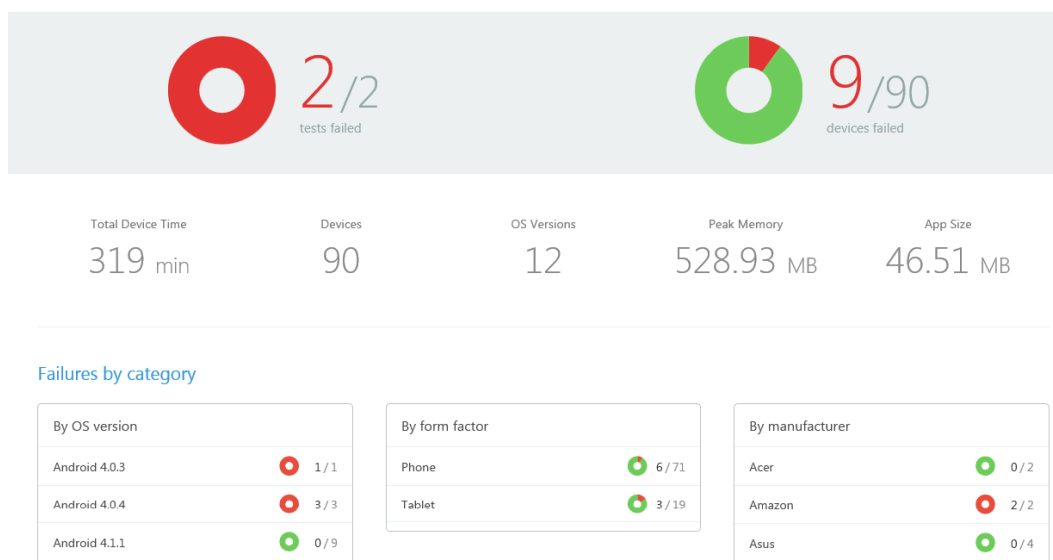
### 6.3.1 Potenciální problémy a jejich možná řešení

**Problém testování.** Jak již bylo řečeno v dřívějších kapitolách, největším úskalím vývoje aplikace je velká rozmanitost modifikací operačních systémů napříč mobilními zařízeními. Vytvářená aplikace byla testována na třech zařízeních tří různých značek. Pro rozšíření aplikace mezi koncové uživatele je toto číslo velmi nízké.

**Možná řešení.** Jednou možností řešení tohoto problému je předčasné uvolnění aplikace mezi omezenou skupinu lidí, kteří by sloužili jako testeři. Pokud by tuto skupinu tvořili budoucí uživatelé, jež nejsou komunitně nijak propojeni a nemají s vývojem samotným nic společného, je zapotřebí brát zřetel na možné riziko velké chybovosti aplikace. To by mohlo, kvůli nespokojenosti uživatelů, do budoucna poškodit reputaci aplikace. Vhodnější je tedy vybrat skupinu testovacích

uživatelů cíleně a mít nad ní co největší kontrolu.

Další možností, jak odstranit problém s nekompatibilitou, je využití nástrojů, které nabízejí jako doplňkové služby společnosti, pomocí nichž byla aplikace vyvíjena. Jedním z takovýchto nástrojů je Xamarin Test Cloud společnosti Xamarin. Systém testování v takovýchto nástrojích je většinou rozdělen do dvou etap. V první z nich je zapotřebí vytvořit testovací scénář. Zde například Xamarin nabízí nástroj Test Recorder, jenž po spuštění ukládá uživatelské interakce s aplikací na připojeném zařízení. Pokud nemáme k dispozici žádné potřebné zařízení, je nástroj schopný získat tyto informace i z emulátoru. Po vytvoření takového scénáře je zapotřebí nahrát výsledná data do webového rozhraní Test Cloud (obr. 11). Poté bude spuštěna druhá část, během níž je daný scénář opakován na reálných zařízeních. Počet těchto zařízení u společnosti Xamarin čítá již přes dva tisíce unikátních kusů. Kvůli tomuto číslu a pokročilosti služby si však Xamarin může dovolit říci vysokou částku za přístup k tomuto nástroji. Ceny tak začínají na sto dolarech za hodinový přístup denně.



Obrázek 11: Prostředí Xamarin Test Cloud

**Problém postupného vývoje zařízení.** Kvůli rychlému rozvoji mobilních zařízení a vysoké konkurenci mezi jejími výrobci jsou výrobci nuceni neustále zdokonalovat své operační systémy. Tím jsou uvolňovány jejich nové verze, což, i přesto že výrobci bývá zaručována zpětná kompatibilita aplikací, dělá vrásky na čele nejednomu vývojáři.

**Možná řešení.** Pokud bychom vyvíjeli aplikace nativně, můžeme se přece jen na výrobce mírně spolehnout, neboť i on před uvolněním novější verze operačního systému provádí testování funkčnosti dříve vytvořených aplikací. Díky tomu má

výhodu způsob řešení vývoje, jež nabízí například společnost Xamarin. V tomto způsobu na konci vývoje nahráváme do mobilního zařízení aplikaci, která se chová jako nativní a nejedná se tak pouze o HTML5 kontejner, jako v případě vývoje ve frameworku Apache Cordova. V něm jsme odkázáni na způsob překladač zařízením samotným, respektive na možnosti vestavěného internetového prohlížeče, jenž může být často zastaralý. U obou stylů vývoje však musíme počítat s tím, že jsme schopni se dostat od stavu, kdy námi vytvořená aplikace není schopna po aktualizaci operačního systému komunikovat s vestavěnými funkcemi telefonu. V takovémto případě jsme často odkázáni na rychlost řešení problému ze strany distributora námi používaného frameworku či překladače. Obecně lze těmto druhům problémům předejít také obdobným stylem, jakým je možné řešit problém výše, tedy využitím testovacích zařízení.

### 6.3.2 Použité nástroje

**Xamarin Forms.** Pro vývoj aplikace ve frameworku Xamarin bylo použito vývojové prostředí **Microsoft Visual Studio ve verzi 2013**. Tato verze již při instalaci umožňuje využít možnosti instalace pluginu Xamarin. Ten přidá schopnost překladač do nativních aplikací a také základní šablony projektů pro tvorbu v Xamarin Forms, Xamarin.iOS či Xamarin.Android. Vzhledem k tomu, že k překladač do nativních aplikací operačního systému iOS je zapotřebí vývojové prostředí Xcode, umožňuje Xamarin využití vzdáleného připojení k zařízení, které má právě aplikaci Xcode nainstalovanou. U starších verzí systému OS X bylo nutné na daný MAC PC nainstalovat aplikaci **Xamarin Build Host**. V novějších verzích se Xamarin dokáže spojit s takovýmto zařízením již jen pomocí vestavěných funkcí systému OS X. Jedinou podmínkou vývoje na systém iOS je tedy mít zařízení, které je dostupné na stejné počítačové síti. Samotný překlad pak funguje rychle a bezobslužně z vývojového prostředí Visual Studia.

**Apache Cordova.** Pro vývoj aplikace pomocí frameworku Apache Cordova bylo využito taktéž vývojového prostředí **Microsoft Visual Studio**, avšak **ve verzi 2015**. Důvodem byla podpora vývoje pomocí tohoto frameworku až s příchodem této verze. Vývoj aplikací Apache Cordova ve Visual Studiu je opět možné povolit již během instalace tohoto softwaru. Díky tomu vývojové prostředí získá schopnost překladač těchto aplikací, šablony pomocí nichž lze aplikace vytvářet a také nainstalovaný Ripple emulátor, pomocí něhož lze velmi svižně vyvíjené aplikace testovat. Pro použití rozšiřujícího frameworku Ionic bylo zapotřebí nainstalovat také NuGet balíček, který v sobě zahrnuje obsáhlé šablony, jež jsou schopny široce pokrýt možné scénáře cílových aplikací.

Vzhledem k tomu, že aplikace byla většinu času testována s lokální databází ve formátu SQLite, byl k návrhu a správě těchto databází využit nástroj



**SqliteBrowser.** Tento nástroj je dostupný zdarma a obsahuje rychlé prostředí pro práci s tímto druhem databází. Jeho plusem je také možnost použití na všech nejrozšířenějších platformách.

## 6.4 Vnitřní struktura aplikace

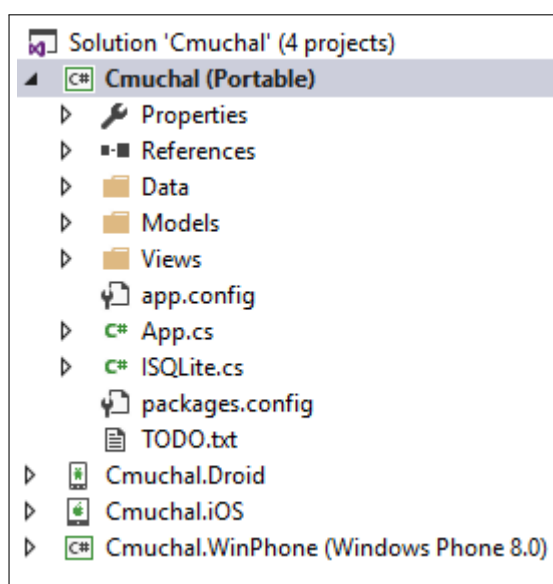
Solution, které obsahuje zdrojové kódy aplikace Čmucha ve frameworku Xamarin Forms, je rozdělena do čtyř projektů (obr. 12). Prvním z těchto projektů je projekt s názvem Cmucha. Ten obsahuje definici uživatelského rozhraní aplikace a také veškerou aplikační logiku sdílenou všemi platformami. Tento projekt je dále rozdělen do tří složek podle způsobu použití tříd, které obsahují. Složka Data obsahuje definici třídy, pomocí níž lze provádět dotazy ve formátu JSON na API server obsahující databázi aplikace. Složka Models pak obsahuje definice tříd sloužících jako modely dat přenášených mezi aplikací a databází. Poslední složka Views pak obsahuje definice jednotlivých obrazovek aplikace, včetně jejich vzhledu. Mimo tyto složky nalezneme ještě třídu App.cs obsahující pouze nastavení počáteční obrazovky aplikace a třídu ISQLite sloužící jako definice rozhraní pro připojení k lokální SQLite databázi.

Dalšími projekty v této Solution jsou Cmucha.Droid, Cmucha.iOS a Cmucha.WinPhone. Jak názvy napovídají, jedná se o projekty, jež slouží k sestavení aplikace pro konkrétní cílové platformy. Tyto projekty mají ve svých referencích sdílený projekt Cmucha popsán výše. Kromě zdrojových souborů, jako jsou obrázky či lokální databáze, obsahují projekty také implementace rozhraní definovaných ve sdíleném projektu. Dalšími důležitými soubory jsou pak ty s názvem app.config, neboť obsahují definice nastavení aplikace pro cílové platformy. Mezi tato nastavení patří například způsob spuštění aplikace či stanovení práv na funkce telefonu, které bude aplikace využívat.

Práce s frameworkem Apache Cordova není takto striktně rozdělena cílovou platformou. Jediný projekt patřící do Solution obsahuje šest složek (obr. 13). Zajímavé z nich jsou – merges obsahující platformně specifické javascripty, plugins obsahující veškeré potřebné zdrojové soubory k použitým pluginům, resources obsahující mimo jiné také sadu ikon aplikace pro jednotlivé platformy a pak www, asi nejdůležitější složka obsahující zdrojové kódy celé aplikace. V projektu samotném existuje také hlavní konfigurační soubor config.xml obsahující informace o aplikaci, jako je například typ spuštění či definice práv k užívání funkcí telefonu. Ve složce www dále najdeme odděleně složky se zdrojovými soubory kaskádových stylů, javascriptů a pak také složku templates obsahující definice pohledů pro jednotlivé obrazovky aplikace. Uvnitř složky www je nutné také definovat soubor index.html, který je zavolán při spuštění aplikace. Obsahuje tak především definice všech importů a skriptů které aplikace používá. Javascriptové soubory jsou dále rozděleny podle jejich funkcionality. Soubor app.js je zapotřebí pro samotné spuštění aplikace a jeho součástí je prvotní inicializace konstant stateProvider a urlrouteProvider. Tyto provideři slouží, mimo jiné, také pro definici jednotlivých vztahů mezi pohledem a jeho kontrolérem a také k navigaci mezi

obrazovkami aplikace v rámci zdrojových kódů. Dalším důležitým souborem je `controller.js`, který obsahuje definici všech kontrolérů, které se v aplikaci nacházejí. Posledním, především uživatelsky definovaným skript souborem, je `services.js`. V rámci něj je definováno propojení na interní databázi a také definice JSON dotazů pro komunikaci s API serverem, na němž se nachází vzdálená sdílená databáze.

Veškerá struktura obou aplikací je ve zdrojovém kódu detailně komentována. Tyto komentáře tak lze, s využitím vestavěných nástrojů IDE, použít jako podklady pro vygenerování programátorské dokumentace.



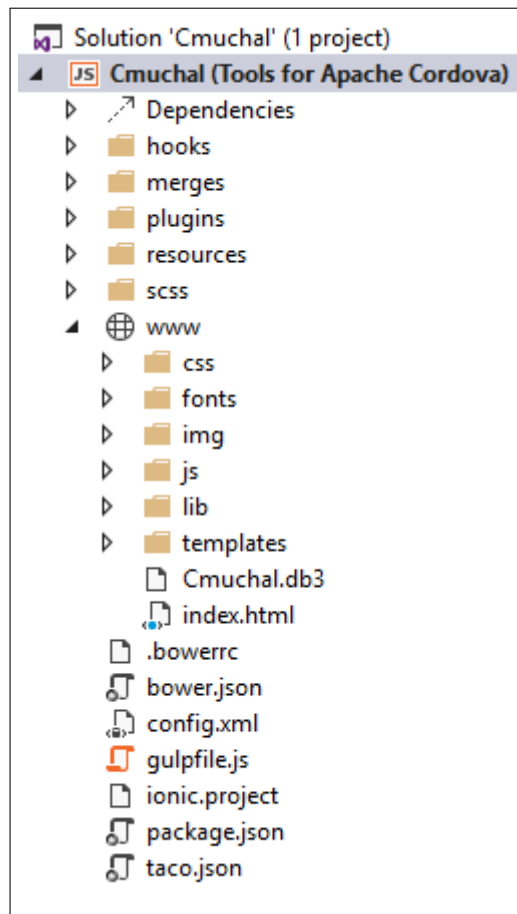
Obrázek 12: Vnitřní struktura aplikace ve frameworku Xamarin

## 6.5 Uživatelská dokumentace

Aplikace Čmuchal zabývající se ztracenými a nalezenými zvířaty funguje v rámci několika obrazovek. Tyto obrazovky se liší v závislosti na zvolené vývojářské platformě. Z tohoto důvodu může být mírně odlišné i samotné ovládání aplikace. V takovém případě jsou pak tyto rozdíly promítnuty i do této dokumentace. Pro přehlednost bude v následujících podkapitolách označena aplikace, jež byla vyvíjena ve frameworku Xamarin, jako ČmuchalX a aplikace vyvíjená ve frameworku Apache Cordova jako ČmuchalC. Díky nepřipojenému API serveru mohou popisovat některé části dokumentace nefungující části aplikace.

### 6.5.1 Hlavní obrazovka

V případě aplikace ČmuchalX se po spuštění dostaneme na základní menu, které bude sloužit dále jako rozcestník pro pohyb mezi obrazovkami aplikace. Toto menu je složeno ze čtyř dlaždic a jednoho samostatného tlačítka pro nastavení



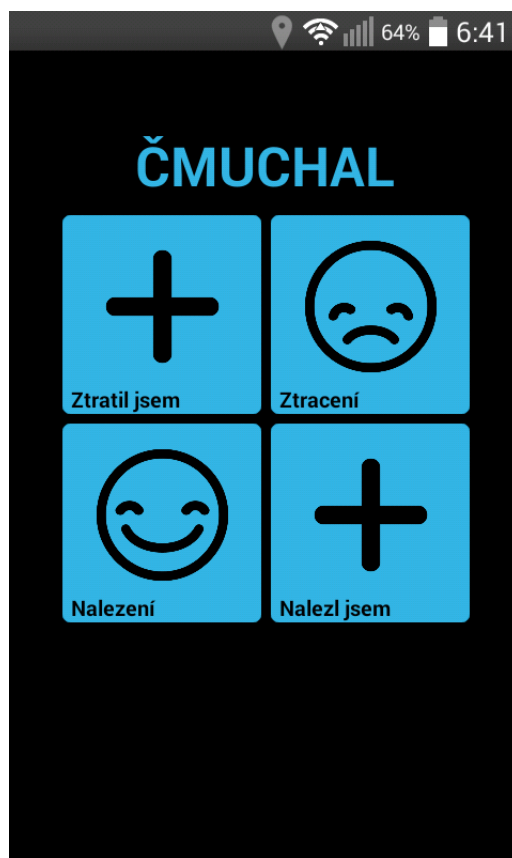
Obrázek 13: Vnitřní struktura aplikace ve frameworku Apache Cordova

aplikace (obr. 14). Dlaždice samotné po klepnutí přenesou uživatele na konkrétní obrazovky aplikace. Na hlavní obrazovku se uživatel může odkudkoli dostat vrácením se pomocí navigace zobrazené v horní části každé následující obrazovky.

U aplikace ČmuchalC, je základní pohyb mezi obrazovkami implementován pomocí karet, jak je tomu známé například z novodobých internetových prohlížečů. Přepínání mezi těmito kartami je dostupné v horní části obrazovky a kromě názvu jsou karty označené také ikonkou, která by uživateli měla pomoci k rychlejší orientaci v aplikaci. Po spuštění je zde uživateli ihned zobrazena karta se ztracenými zvířaty, bez nutnosti další interakce.

### 6.5.2 Ztracená a nalezená zvířata

Pro intuitivnější ovládání mají dvě základní obrazovky celé aplikace obdobný vzhled i chování. Základem je seznam všech zvířat, který je možné filtrovat a řadit. Každé z těchto zvířat pak má svůj profil obsahující fotku zvířete, jeho jméno a popis a také základní informace o uživateli, jenž zvířeti profil vytvořil (obr. 15). Mezi těmito profily jednotlivých zvířat může uživatel přecházet pomocí přejetí zleva doprava, či zpětně pomocí přejetí zprava doleva. K filtrování a řazení



Obrázek 14: Hlavní obrazovka v aplikaci ČmuchaX

tohoto seznamu slouží tlačítko s ikonou filtru nacházející se v případě aplikace ČmuchaX v horním pravém rohu obrazovky a v případě ČmuchaC v dolním středu obrazovky. Po stisku tohoto tlačítka může uživatel nejprve změnit kritérium, podle kterého chce, aby byla zvířata v seznamu seřazena. Implicitně po spuštění aplikace bude vždy vybráno řazení podle data vložení profilu od nejnovějšího. V další části pak lze, pokud uživatel hledá konkrétní zvíře, blíže specifikovat zobrazené profily. Všechna pole týkající se filtrace jsou volitelná.

V případě, že chceme do některé z těchto dvou kategorií přidat vlastní profil zvířete, slouží k tomu tlačítko s ikonou plusu umístěné u aplikace ČmuchaX v horním pravém rohu a u ČmuchaC v dolním levém rohu obrazovky. K této funkci lze také v případě ČmuchaX použít zkratku, jež je umístěna jako dlaždice na hlavní obrazovce aplikace. Samotné vytváření profilu je pak následně rozděleno do tří částí (obr. 16). Mezi těmito částmi je možné procházet pomocí karet v horní části obrazovky, případně pomocí tlačítek na konci každé z karet. Při vytváření je důležité, aby uživatel vyplnil co nejvíce polí kvůli následné korektní identifikaci zvířete jiným uživatelem. Na první kartě, jež je označena jako Zvíře, je možné vyplnit základní informace o zvířeti, které jsme našli či ztratili. Po přechodu na druhou kartu je uživatel vyzván k vložení fotografie daného zvířete. Na výběr, jak fotografii vložit, má ze dvou možností. Při zvolení možnosti Vyfotit

je nutné, aby zařízení, na němž je aplikace spuštěna, mělo fotoaparát a funkční aplikaci, jež ho ovládá. Tato aplikace bude následně spuštěna a uživatel zde tak má možnost vyfotit zvíře, jemuž je profil vytvářen. Tuto možnost lze využít i v případě, že se jedná o ztracené zvíře a uživatel má dostupnou jeho tištěnou fotografii. Druhou možností vložení obrázku ke zvířeti je pak výběr z galerie telefonu. K této funkci je opět využito vestavěné aplikace telefonu. Výsledná vyfocená či zvolená fotografie bude následně na kartě zobrazena. Třetí a poslední kartou pak je Místo a kontakt. Zde je velmi důležité, aby uživatel co nejpřesněji vyplnil pole pro popis místa, kde bylo zvíře nalezeno či ztraceno. Dalším důležitým polem je pole Souřadnice, do nějž lze, v případě, že je v zařízení dostupný GPS lokační modul, zapsat stiskem tlačítka GPS souřadnice převzaté z tohoto lokačního modulu. Na kartě je ještě důležité vyplnit alespoň některá z polí v sekci Kontakt. Zde je potřeba brát zřetel na to, že informace, včetně telefonního čísla, budou zobrazeny všem uživatelům aplikace. Poslední informaci, kterou musí uživatel vyplnit, je Heslo. Heslo slouží pro následné odstranění profilu zvířete a je tak nutné, aby si ho uživatel zapamatoval. Celý vyplněný profil je následně tlačítkem Přidat uložen a zobrazen v daném seznamu zvířat.

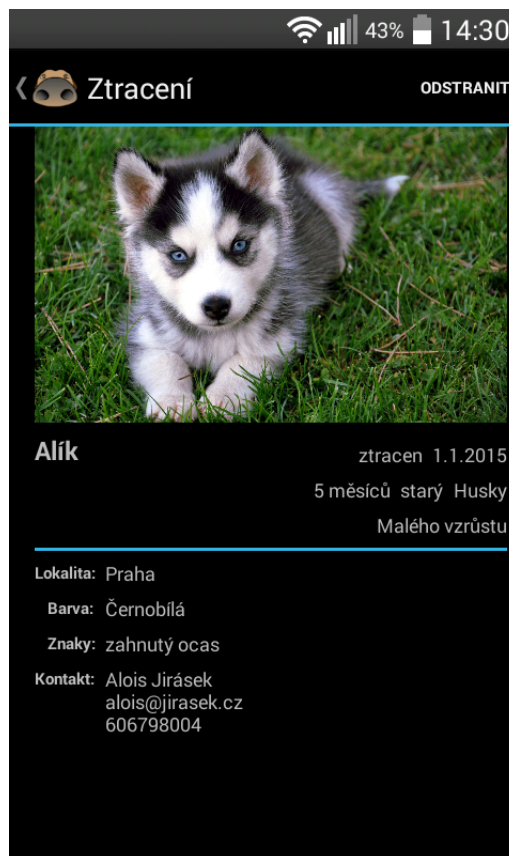
Pokud uživatel chce některé ze zvířat, jemuž vytvořil profil, odstranit ze seznamu, může tak učinit tlačítkem s ikonou křížku. To se nachází u aplikace ČmuchalC v dolním pravém rohu obrazovky a u aplikace ČmuchalX v horním pravém rohu obrazovky. Po stisku je nutné do zobrazeného okna vyplnit heslo, jež bylo uživatelem zadáno při vkládání zvířete do seznamu. Pokud se tato hesla shodují, je následně ze seznamu profil odstraněn.

### 6.5.3 Nastavení

Do nastavení aplikace se lze dostat dvěma způsoby v závislosti na verzi aplikace. Pokud se jedná o verzi ČmuchalX, je nastavení dostupné z hlavní obrazovky, po stisku tlačítka s ozubeným kolečkem, v horním pravém rohu. U ČmuchalC je pak nastavení dostupné jako třetí karta v aplikaci.

Nastavení obsahuje dvě oddělené sekce (obr. 17). První z nich se týká uživatelsky vytvořených profilů. Jedná se o předpřipravené profily zvířat, které lze využít jako šablonu pro následné vkládání do seznamu ztracených zvířat. Funkce je zde především kvůli tomu, aby bylo možné přidat ztracené zvíře do seznamu za co možná nejkratší dobu. Proto je výhodné si spravovat a aktualizovat profily v nastavení a v případě ztráty tohoto zvířete pak pouze vybrat některý z těchto profilů. Stiskem tlačítka Přidat je umožněno vytvořit nový profil. V tomto případě se jedná o mírně zestručněnou verzi přidávání zvířete dostupnou ze seznamu zvířat. Po vyplnění a uložení takového profilu je možné jej editovat klepnutím na něj v jejich seznamu.

Další sekce nastavení se týká systému GPS. Pokud jím není zařízení uživatele vybaveno, nastavení je možné ignorovat. V opačném případě je dostupná nejprve možnost vypnout či zapnout upozorňování. Jedná se o push-up notifikace, které informují o nové ztrátě či nálezů psa pomocí notifikace v mobilním zařízení. Kvůli možnému budoucímu velkému počtu ztrát či nálezů je možné pomocí



Obrázek 15: Profil ztraceného zvířete v aplikaci ČmuchaIX

posuvníku s kilometry určit vzdálenost, do jaké budou zobrazovány takovéto informace o nových zvířatech. Dalším možným nastavením je, zda uživatel chce, aby se daný okruh kilometrů počítal vždy od jedné, předem nastavené lokace, nebo zda bude jako střed okruhu brána vždy aktuální pozice uživatele.

## 6.6 Možná vylepšení aplikace

Jedním z možných budoucích vylepšení aplikace je umožnění vložení více fotografií k přidávanému zvířeti. Nyní má uživatel možnost vložit pouze jednu fotografii u nově vytvářeného profilu zvířete. Je možné, obzvláště v případě nalezených zvířat, že by přidání více fotografií a tedy vytvoření galerie vkládaného zvířete usnadnilo identifikaci zvířete ostatním uživatelům. Galerie by byla dostupná po kliku na fotografii zvířete v přehledu zvířat.

Pokud by se počet uživatelů zvětšil na určitou míru, je pravděpodobné, že by byl zapotřebí nějaký systém schvalování a případná administrace příspěvků. Řešení by bylo možné pomocí webové aplikace, případně pomocí rozšíření mobilní aplikace. Sofistikovanějším řešením se jeví webová aplikace, která by umožňovala jednoduchou správu potenciálních i aktuálně platných příspěvků.

Jako rozšíření se také nabízí napojení na stávající databázi ztracených zvíř-

Obrázek 16: Vytvoření profilu zvířete v aplikaci ČmuchalX

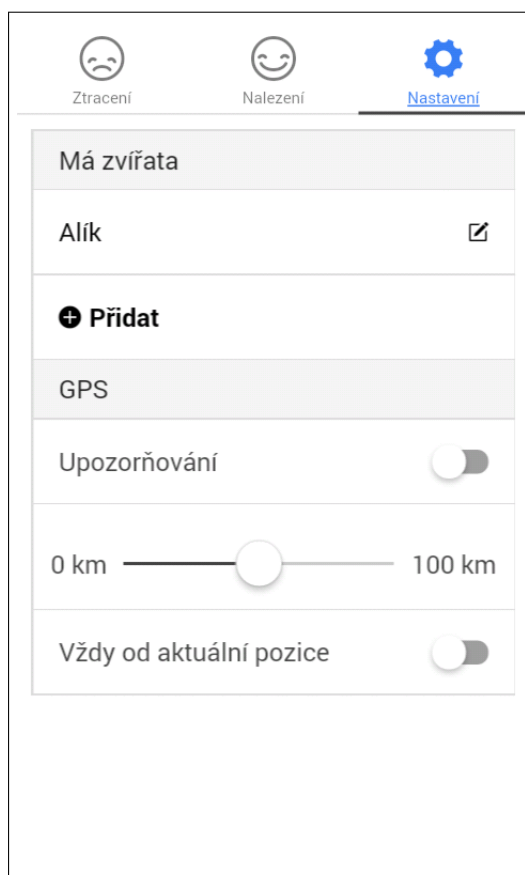
řat, jako má například server PsiDetektiv.cz. Z této databáze by bylo možné po vzájemné spolupráci čerpat obsažená data, případně i v opačném směru posílat nová data, která uživatelé skrz mobilní aplikaci zadají. Toto rozšíření by zřejmě vyžadovalo úpravy stávajících řešení na obou stranách.

## 6.7 Srovnání použitých frameworků

Při vývoji testovací aplikace byly použity frameworky Apache Cordova a Xamarin Forms. Oba z nich jsou popsány výše, zde tak bude následovat jejich přímé porovnání pomocí mých názorů a zkušeností získaných při vývoji aplikace.

### 6.7.1 Vzhled aplikace

Co se týče možností, které frameworky nabízejí pro vzhled aplikace, je dozajista více omezující Xamarin Forms. Je to dáno především překladem do nativní formy aplikace, tedy určitým omezením právě výrobcem operačního systému. Xamarin Forms také později reflektuje nové verze jednotlivých API operačních systémů,



Obrázek 17: Nastavení v aplikaci ČmuchaC

což vede jak k občasným problémům s nevhodným zobrazením na novější verzi operačního systému, tak také k opožděné možnosti využití vzhledových novinek ve verzích.

Apache Cordova je, co se týče designu aplikace, přece jen mírně zvýhodněn. Tyto výhody plynou především opět z architektury frameworku. Výsledná aplikace využívá k vykreslení a spuštění vestavěný internetový prohlížeč v zařízení, čímž umožní téměř jakkoliv využít možností jazyků které podporuje. Nejsme tak omezeni nativním vzhledem a jsme schopni využít široké možnosti kombinace kaskádových stylů a javascriptu. Důležitou roli zde hraje také fakt, že díky použití velmi rozšířených technologií existuje velké množství předpřipravených šablon. Pomocí nich je pak tvorba designu aplikace značně rychlejší a méně náročná. Použití vestavěného prohlížeče však může být i nevýhodou. Vzhledem k tomu, že uživateli ani vývojáři není dána možnost, v jakém z dostupných internetových prohlížečů je možné aplikaci vykreslit, nastává zde problém s různými verzemi prohlížečů na různých zařízeních. Kvůli tomu je nutné testovat výslednou aplikaci na více zařízeních, což ovlivní jak čas, tak cenu vývoje.



## 6.7.2 Přístup k funkcím telefonu

Pro vývoj vzorové aplikace pro hlášení ztracených a nalezených zvířat je potřeba využít některé vestavěné funkce telefonu. Jedná se o funkce fotoaparátu a GPS lokátoru. To podporují oba použité frameworky, i když každý svým trochu odlišným způsobem. Pro Apache Cordova je zapotřebí se spolehnout na vytvořené pluginy, tedy vlastně kusy předpřipraveného javascriptového kódu. Ty mohou být uloženy a nativně podporovány v jádře frameworku, nebo případně mohou být vytvořeny kýmkoliv jiným. V druhém případě je lze následně načíst jak z lokálního, tak GIT úložiště. Seznam pluginů v jádře čítá v aktuální verzi frameworku dvacet čtyři pluginů včetně těch potřebných pro zde vyvíjenou aplikaci, tedy Camera a Geolocation. Ke staženým pluginům lze nalézt dokumentaci na webu a jejich použití je intuitivní a rychlé. Co se týče pluginů, které uloženy v jádře nejsou, jejich počet přesahuje tisíc kusů a jejich galerii je možné procházet na stránkách <http://cordova.apache.org/plugins>.

Pokud nechceme psát vlastní řešení přístupu k potřebným funkcím telefonu, nabízí Xamarin Forms obdobnou funkcionalitu pomocí pluginů. Ty se však, především kvůli překladu do nativního kódu, mírně liší od Apache Cordova ve formě použití. Pokud je zapotřebí využít například funkci fotoaparátu, je možné nalézt v galerii NuGet balíčků několik hotových řešení a tato řešení následně aplikovat. Princip fungování těchto balíčků napříč cílenými platformami, do nichž je aplikace překládána, je vytvoření rozhraní ve sdíleném projektu a následná implementace tohoto rozhraní v projektech pro konkrétní platformy. Řešení je to taktéž rychlé, avšak oproti javascriptu více náchylné k chybovosti.

```
1  \$.scope.takePicture = function () { //funkce volaná stiskem tlačí  
    tka z template  
2      Camera.getPicture({ //volání funkce z pluginu  
3          quality: 75,  
4          targetWidth: 320,  
5          targetHeight: 320,  
6          saveToPhotoAlbum: false,  
7          correctOrientation: true,  
8      }).then(function (imageURI) {  
9          var photo = document.getElementById('photo'); //předání  
            fotografie do template  
10         photo.style.display = 'block';  
11         photo.src = imageURI;  
12     }, function (err) { //zpracování chyby předané z pluginu  
13         Error(err);  
14     });  
15 }
```

Zdrojový kód 1: Použití fotoaparátu pomocí pluginu v Apache Cordova

```

1 private async Task TakePicture() {
2     if (_mediaPicker != null)
3         return;
4     IDevice device = Resolver.Resolve<IDevice>();
5     _mediaPicker = DependencyService.Get<IMediaPicker>() ?? device.
        MediaPlayer; //získání instance dle platformy
6     _imageSource = null;
7     await this._mediaPicker.TakePhotoAsync(new
        CameraMediaStorageOptions { //volání funkce z implementované
            ho rozhraní
8         DefaultCamera = CameraDevice.Rear, MaxPixelDimension = 400 }) //
            dodatečné parametry
9         .ContinueWith(t => {
10            if (t.IsFaulted)
11                Error(t.Exception.InnerException.ToString());
12            else if(t.IsCompleted){
13                MediaFile mediaFile = t.Result;
14                _imageSource = ImageSource.FromStream(() => mediaFile.
                    Source); //předání přijaté fotografie
15                return mediaFile;
16            }
17            return null;
18        }, _scheduler); //použití task scheduleru
19    }

```

Zdrojový kód 2: Použití fotoaparátu pomocí rozhraní v Xamarin Forms

### 6.7.3 Rychlost vývoje

Co se týče rychlosti tvorby aplikace, je dozajista velmi důležité, jaké jsou zkušenosti v programovacím jazyce, ve kterém je nutné ve zvoleném frameworku psát. Vzhledem k této okolnosti má převahu Apache Cordova, neboť znalost jazyků HTML, Javascript a CSS je přeci jen rozšířenější, než jazyka C#. Z mého pohledu programátora, který se věnuje o dosti více jazyku C# a s jazykem JavaScript má malé zkušenosti, je pak výsledek srovnání rychlosti vývoje o to větším překvapením. Tvorba základní kostry aplikace je značně rychlejší díky použití jazyka HTML a také díky širokým možnostem využití již předpřipravených šablon. Využití funkcí telefonu je rychlejší opět u Apache Cordova, a to kvůli kombinaci využití pluginů a nutnosti spravovat pouze jeden kód, který není platformně specifický. Plusové body na stranu Xamarin Forms zaručeně přináší vestavěná podpora objektů a tříd v jazyce C#. To však lze, díky externím knihovnám, simulovat i v jazyce JavaScript.

### 6.7.4 Vývojové nástroje

Oba použité frameworky nabízejí rozšíření do velmi oblíbeného IDE Visual Studio. Lze tak porovnat samotné schopnosti pluginů v rámci tohoto vývojového nástroje.

Rozšíření pro framework Xamarin nabízí, kromě základních šablon aplikací a knihoven, také možnost připojení vzdáleného počítače MAC s nainstalovaným vývojovým rozhraním Xcode. Díky tomu zajišťuje schopnost kompilace aplikací pro operační systém iOS. Tato kompilace probíhá formou bezdrátového Wi-fi spojení, kdy obě pracující zařízení musí být připojena ke stejné síti. I když toto řešení zní hezky, je dosti problémové. Je to dáno především nutnou kompatibilitou mnoha faktorů. Komunikovat zde totiž musí různé verze nástrojů Visual Studio, Xamarin iOS, Xamarin, MAC OS a Xcode. Díky tomu tak často nastává scénář, kdy vše funguje dokonale, ale při jakékoliv aktualizaci některého z těchto nástrojů vše rázem fungovat přestane. Vzhledem k tomu, že kompilace na MAC zařízení je pro iOS aplikace nutnost, lze označit řešení jako vhodné, avšak jeho nestabilita může mnohé vývojáře od frameworku Xamarin odradit.

Plugin pro Apache Cordova nabízí obdobnou funkcionalitu pomocí nástroje *remotebuild*. Tento nástroj je zapotřebí nainstalovat na zařízení MAC. To pak musí být dostupné ve stejné síti jako zařízení, na kterém je vyvíjená mobilní aplikace. Po instalaci a nakonfigurování serveru *remotebuild* probíhá kompilace z IDE Visual Studio svižně a také, oproti řešení frameworku Xamarin, vcelku průhledně. Uživatel je zde totiž na obou stranách informován o aktuálním dění pomocí logů v konzolích. Mírné problémy mohou nastat v případě nízké výpočetní rychlosti zařízení MAC, avšak vhodnou konfigurací serveru jim lze předejít. Řešení je taktéž citlivé na verze jednotlivých nástrojů, při zkoušení ale působí robustněji než řešení frameworku Xamarin.

### 6.7.5 Shrnutí

Z praktického srovnání řešení vývoje mobilních aplikací pomocí Apache Cordova a Xamarin Forms není možné určit jednoznačného vítěze. Každý z nich je vhodnější na určité scénáře. Pokud vývojář klade důraz na nízké náklady samotného vývoje aplikace, bude ním zřejmě upřednostněn právě Apache Cordova. Je tomu tak díky jeho cenové politice a také schopnosti rychlého vývoje. Pokud vývojář klade důraz na doplňkové služby jakou jsou pomocné testovací nástroje či cloudové databáze, je vhodnější využít řešení Xamarin Forms. Hlavním rozhodovacím kritériem zřejmě i přesto zůstane oblíbenost konkrétního programovacího jazyka vývojářem. V té by mělo převažovat řešení Apache Cordova, které je postavené na často využívaných webových technologiích.

Při tvorbě testovacích aplikací byla dohodnuta jednotlivými stranami, kterých se tvorba aplikace týká, jejich priorita. Pro nasazení mezi koncové zákazníky bude sloužit aplikace ve frameworku Xamarin. Výběr byl zvolen především kvůli širším znalostem s jazykem C# v rámci zúčastněných stran. Dokončení a testování aplikace tak bude probíhat především v tomto frameworku.

## Závěr

Pokud člověk ví, co chce na mobilní zařízení vyvíjet, je teprve na počátku dlouhého rozhodovacího procesu, jakým stylem a v čem mobilní aplikaci vyvíjet. Kvůli velké konkurenci na trhu mu rozhodování nijak neulehčuje ani široká možnost volby. Zároveň je toto rozhodnutí pro budoucnost klíčové, neboť trh je již značně nasycený a každoročně několik hráčů svou činnou v tomto odvětví vzdá. Při nevhodné volbě tak může nastat scénář, kdy se uživatel musí učit opakovaně různé technologie.

Na počátku je nejprve nutné zhodnotit, zda je v konkrétním případě pro uživatele vůbec vhodné vydat se směrem multiplatformního vývoje. Existují důvody, proč není vhodné vyvíjet multiplatformně. Avšak dá se očekávat, že v budoucnu se tomuto směru nevyvaruje žádný z mobilních vývojářů. Pokud se uživatel rozhodne pro multiplatformní vývoj, potřebuje nejprve provést analýzu trhu pro zjištění aktuálních vývojových možností. V tomto směru není lehké začínat s čistým štítem. Taková věc je totiž časově velmi náročná, ať už kvůli velkému množství aspektů analýzy, nebo kvůli její časové náročnosti a technologické komplikovanosti. Nabízí se tak použití již existujících řešení. Takovýchto analýz existuje celá řada. Jejich největším problémem je však nekomplexnost a rychlé zestárnutí poskytnutých informací.

Zde představené řešení srovnání frameworků, které nabízejí multiplatformní vývoj, tak sází především na komplexnost. Pro řešení problému výběru by nemělo poskytnout pouze přehled aktuálního stavu trhu, ale mělo by i poradit, jak takovýto stav trhu co nejlépe analyzovat. Pokud si tedy vývojář bude v budoucnu chtít vybrat multiplatformní vývoj, tak díky tomuto obsahu získá důležité rady, jak při takovém výběru postupovat. Důležitým bodem představení aktuálních řešení jsou jejich technické možnosti. Kvůli tomu byla zvolena cesta tvorby funkčně a vizuálně identických mobilních aplikací pomocí dvou vybraných zástupců frameworků. Výsledky srovnání těchto dvou možných řešení pak poskytují ucelený přehled kladů a záporů těchto dvou směrů. Srovnání samotné však nedokázalo v rámci multiplatformního vývoje určit jednoznačného vítěze. To je dáno především možnými preferencemi jednotlivých vývojářů.

Jedním z výsledků analýzy je celková malá rozdílnost existujících řešení. Ty jsou tak postaveny na dvou až třech totožných technologiích a samotná přidaná hodnota frameworku se pak většinou týká doplňkových služeb. Zajímavým výsledkem celého srovnání je pak především fakt, že i přes preference určitého programovacího jazyka vývojářem nemusí být řešení, které jako jediné tento programovací jazyk využívá, zaručeně tím nejlepším možným.

## Conclusions

If a man knows, what he wants to develop for a mobile device, he is only on the very beginning of a long decision-making process – deciding what style and how to develop his mobile application. Due to high competition his decision is not facilitated by wide possibilities in the market. At the same time this decision is crucial for the future as the market is already highly saturated and every year several players leave the market. When inappropriate choice is made then a scenario arise where the user must repeatedly learn various new technologies.

At the beginning it is necessary to assess whether in a specific case it is suitable to go towards cross-platform development. There exist number of reasons why it is not appropriate using cross-platform development. However, it is expected that in the future this direction will not be avoided by any of the mobile applications developers. If user opts for cross-platform development, he first needs to analyze the market to determine the actual development possibilities. In this respect, it is not easy to start with clean slate. Such a thing is very time consuming, whether due a large number of aspects of analysis, or because the time demands and technology complexity. One of the options can be using existing analysis. There is a high number of these analyzes. Their biggest problem is the incompleteness and the rapid aging of the information provided.

Here introduced complex solution compares frameworks that offer cross-platform development. To solve the problem of selection there should not be only an overview of the current state of the market, but solution should also advise on how such market situation is best analyzed. If the developer in the future will want to choose a cross-platform development, thanks to this content he acquires important tips on how to do such a selection. The important point of presentation of current solutions are their technical capabilities. Because of this two functional and visually identical mobile applications were created using two selected representative frameworks. The result of the comparison of these two possible solutions is a comprehensive overview of the pros and cons of these two directions. Comparison itself, however, failed in the context of cross-platform development to determine a clear winner. This is primarily due to possible preferences of individual developers.

One result of the analysis is existence of small disparity between solutions. They are based on two or three identical technologies and the added value of framework lies mostly with ancillary services. Interesting result of the comparison is the fact that despite the popularity of certain programming language by the developer, solution offered by this programming language do not have to be the best possible.

## A Obsah přiloženého CD/DVD

Součástí diplomové práce je CD s následující strukturou.

### **bin/**

Adresář obsahující soubory mobilní aplikace. Tu je možné nainstalovat na reálné mobilní zařízení s OS Android. Pro ostatní platformy je nutné využít emulátoru.

### **doc/**

Adresář obsahující text práce ve formátu PDF a všechny soubory potřebné pro vygenerování PDF dokumentu textu v archivu ZIP.

### **src/**

Adresář obsahující zdrojové kódy mobilních aplikací.

### **readme.txt**

Textový soubor obsahující informace pro spuštění mobilní aplikace.

## Literatura

- [1] Baker, Van L., Ray Valdes, Adrian Leow, Jason Wrong a Richard MARSHALL. Magic Quadrant for Mobile Application Development Platforms. Gartner, Inc [online]. 2015 [cit. 2016-04-21]. Dostupné z: <http://adobe.ly/1yhE4nA>
- [2] GMC Software Technology [online]. 2013 [cit. 2016-05-07]. Dostupné z: <http://gmchk.cz/>
- [3] Moravskoslezský spolek pro ochranu zvířat [online]. 2012 [cit. 2016-05-07]. Dostupné z: <http://pomahamezviratum.cz/>
- [4] Allen, Grant. Android 4: průvodce programováním mobilních aplikací. 1. vyd. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6.
- [5] Vávru, Jiří. iPhone: vývoj aplikací. 1. vyd. Praha: Grada, 2012. Průvodce (Grada). ISBN 978-80-247-4457-5.
- [6] Castledine, Earle, Myles Eftors a Max Wheeler. Vytváříme mobilní web a aplikace pro chytré telefony a tablety. 1. vyd. Brno: Computer Press, 2013. ISBN 978-80-251-3763-5.
- [7] Baker, Van L., Ray Valdes, Gordon Van Huizen, Jason Wrong a Richard Marshall. Magic Quadrant for Mobile Application Development Platforms. Gartner, Inc [online]. 2014 [cit. 2016-04-21]. Dostupné z: <http://adobe.ly/1yhE4nA>
- [8] Finley, Ian, Van L. Baker, Ken Farmlee, David Mitchell Smith, Ray Valdes a Gordon Van Huizen. Magic Quadrant for Mobile Application Development Platforms. Gartner, Inc [online]. 2013, 34 [cit. 2016-04-21]. Dostupné z: <http://bit.ly/1pl5BkY>
- [9] Petzold, Charles. Creating Mobile Apps with Xamarin.Forms: Cross-platform C# programming for iOS, Android, and Windows [online]. 1. vyd. Redmond, Washington 98052-6399: Microsoft Press, 2016 [cit. 2016-04-21]. ISBN 978-1-5093-0297-0. Dostupné z: <http://aka.ms/xamebook>