

# Úvod do webového vývoje pro mobilní zařízení

## V této kapitole:

- Co znamená mobilní?
- Proč na tom záleží?
- Výsadní postavení nativních aplikací je v ohrožení
- Pěkně od začátku
- Aplikace nestačí
- Poznámka k prostředí
- Vyhrňte si rukávy

Jste připraveni vstoupit do arény webového návrhářství? Jak již napovídá její název, kniha, kterou držíte v ruce, je věnována problematice navrhování webů pro mobilní zařízení – webů budoucnosti. Kniha vás provede celým procesem návrhu a vývoje webové aplikace pro mobilní zařízení, úplně od začátku. Nejprve probereme, co všechno byste měli při vývoji pro mobilní zařízení brát v potaz, a položíme společně základy aplikace. Na základy tvořené webovými standardy poté přidáme interaktivní funkcionalitu a nakonec aplikaci zabalíme do nativní obálky, aby ji bylo možné distribuovat pomocí různých tržišť aplikací. Kniha se zaměřuje na problematiku vývoje pro mobilní zařízení o rozměrech telefonu, ale většinu popsaných koncepcí lze využít i při vývoji pro jiná mobilní zařízení a kontexty, například pro tablety a netbooky.

Z technického hlediska budeme hovořit o stejných technologiích, na které jste při webovém vývoji zvyklí: Téměř veškerá probíraná látka se bude týkat práce s jazyky HTML, CSS a JavaScript. Z tohoto důvodu je také třeba, abyste disponovali alespoň základními znalostmi těchto technologií.

## Co znamená mobilní?

Nejprve si musíme ujasnit některé základní pojmy, abychom si mohli být jistí, že našemu textu správně porozumíte. Například byste se klidně mohli zeptat, co máme vlastně na mysli pojmem *mobilní*. Odpověď na tuto otázku zní, že mnoho různých věcí. Vývoj webů a webových aplikací určených pro mobilní zařízení se na první pohled nijak neliší od vývoje jaké-

hokoliv jiného webu či webové aplikace, takže by se mohlo zdát, že web pro mobilní zařízení pouze optimalizujeme. Ponoříme-li se však do této problematiky hlouběji, rychle přijdeme na to, že to tak jednoduché není.

Diskuze o webech pro mobilní zařízení mají tendenci sklouzávat k tématům o zařízeních a jejich funkcích – co umí poslední iPhone, nejnovější zařízení Android a co je tento týden nového v systému webOS. Změny v této oblasti se dějí téměř neustále, vývoj webů pro tento typ zařízení je proto opravdu vzrušující a nechat se strhnout diskuzí o technických požadavcích a řešeních pro mobilní zařízení je opravdu velmi snadné. Ve skutečnosti se však v podobných diskuzích ztrácí jedna velmi důležitá věc: Vývoj pro mobilní zařízení není o těchto zařízeních, ale o jejich uživateli. Velmi trefně to ve své knize *Designing the Mobile User Experience*<sup>1</sup> vystihla Barbara Ballardová:

„Mobilní je ve skutečnosti uživatel, a ne jeho zařízení nebo nějaká aplikace.“

V prvním řadě proto musíme vždy myslet na uživatele a teprve poté se zabývat technickými specifikacemi zařízení. Mobilita je něco víc než pouhá možnost opustit kancelář – jedná se o úplně jiný kontext, zcela specifický uživatelský prožitek. Lidé kupodivu používají mobilní aplikace, pokud jsou zrovna sami na cestách, *když jsou mobilní*, a je to především jejich všudypřítomná dosažitelnost, která činí mobilní aplikace tak užitečnými a současně značně ztěžuje jejich vývoj. Proto se při vývoji aplikací a webů pro mobilní zařízení musíme velmi důkladně zamyslet nad tím, komu je náš produkt vlastně určen a co jeho cílový spotřebitel vlastně chce a požaduje. Naším cílem pak musí být snaha o to, aby byla výsledná aplikace užitečná zejména v tomto kontextu. A i když se budeme v knize většinou zabývat především technickými problémy, nikdy při tom nesmíme zapomenout na definici Barbary Ballardové a vždy ji musíme brát v potaz.

## Proč na tom záleží?

Odhaduje se, že celkový počet smartphonů a dalších telefonů vybavených prohlížeči překoná v roce 2013 počet jejich stolních protějšků (1,82 miliardy oproti 1,78 miliardy)<sup>2</sup>. Spolehlivé statistiky o podílu využití prohlížečů pro mobilní zařízení při procházení Internetu je tradičně velmi obtížné získat, ale nezávisle na zdroji je obecný trend jasný. Podle webu StatCounter činí podíl využití prohlížečů pro mobily při procházení Internetu 4,36 %, a i když vám toto číslo možná připadá velmi malé, nezapomeňte na to, že v posledních dvou letech vzrostlo o 430 %. A to je pouhý začátek éry mobilního procházení Internetu. Již nikdy nebudeme trávit prací s mobilními telefony a dalšími mobilními zařízeními méně času než doposud.

---

<sup>1</sup> Hoboken: Wiley, 2007

<sup>2</sup> <http://www.gartner.com/it/page.jsp?id=1278413>

Nevyhnutelně se proto na trhu budou objevovat neustále nová a výkonnější zařízení a všudypřítomné připojení k Internetu se stane normou. Současně se také bude velmi rychle měnit kontext, ve kterém budou tato zařízení používána. Počet vašich potencionálních zákazníků proto bude neustále narůstat a ignorování mobilního přístupu k Internetu jistě nebude tím nejšťastnějším obchodním rozhodnutím.

## Výsadní postavení nativních aplikací je v ohrožení

Při vývoji aplikace pro mobilní zařízení se nevyhneme rozhodnutí, zda zvolit nativní, či webovou aplikaci. Pojdme si nejprve pro pořádek oba tyto pojmy definovat. **Webové aplikace** jsou přístupné prostřednictvím internetového připojení a prohlížeče zařízení – jedná se v podstatě o weby, které nabízejí funkcionalitu aplikací. Takzvané **nativní aplikace** jsou vyvinuty specificky pro danou platformu – například Android nebo iOS – a nainstalovány v zařízení podobně, jako jsou nainstalovány aplikace ve stolním počítači. Aplikace tohoto typu jsou obvykle distribuovány prostřednictvím tržišť aplikací zaměřených na jednotlivé platformy. Nejznámějším tržištěm tohoto typu je služba AppStore společnosti Apple určená k distribuci aplikací pro iPhone a iPad.

Nyní se pojdme podívat na klady a zápory obou těchto typů aplikací. Nativní aplikace obecně vzato nabízejí ve srovnání s webovými aplikacemi kvalitnější uživatelský prožitek a tento rozdíl se ještě prohlubuje v zařízeních, která nedisponují příliš vysokým výkonem. Nativní aplikace jsou vyvinuty, optimalizovány a, což je nejdůležitější, kompilovány konkrétně pro zařízení a platformy, na kterých běží. V systému iOS to například znamená, že jsou naprogramovány v jazyce Objective-C, a v systému Android, že jsou naprogramovány v jazyce Java. Webové aplikace jsou naopak interpretovány – to znamená, že je jejich kód čten a překládán průběžně vykreslovacími stroji prohlížečů a jazyka JavaScript. V případě systémů iOS, Android, BlackBerry, Symbian a webOS vládne světu vykreslovacích strojů prohlížečů jádro WebKit, stejný stroj, na kterém jsou založeny také prohlížeče Safari a Chrome. V systému Windows Phone 7 se k interpretaci používá zvláštní verze stroje prohlížeče Internet Explorer 7, ačkoliv společnost Microsoft oznámila, že plánuje jeho nahrazení interpretačním strojem prohlížeče Internet Explorer 9. Díky této vrstvě navíc, která leží mezi jejich kódem a zařízením, webové aplikace nikdy nedosáhnou výkonu aplikací nativních, což může být problém, pokud aplikace pracuje s 3D grafikou ve vysokém rozlišení nebo provádí náročné výpočty. Pokud však není funkcionality aplikace tak složitá, webová aplikace si povede téměř stejně dobře jako aplikace nativní. Stále sice bude o něco pomalejší, ale na výsledném uživatelském prožitku se to téměř neprojeví.

Nutnost interpretace webových aplikací strojem také znamená, že je aplikace omezena schopnostmi příslušného stroje. Tam, kde mohou nativní aplikace přistupovat k úplné sadě metod

exponovaných operačním systémem, se musí webové aplikace omezit pouze na komunikaci, kterou umožňuje interpretační stroj. To znamená, že mohou používat pouze ta rozhraní API, která jim zpřístupní prohlížeč. V systému iOS mají nativní aplikace například přístup k celé sadě funkcí, které prohlížeč Mobile Safari nezpřístupňuje – k nabízeným oznámením, fotoaparátu, mikrofonu a kontaktům uživatele. To znamená, že nelze vyvinout webovou aplikaci, která by umožňovala odesílání fotografií do služeb, jako jsou služby Flickr nebo Facebook. Jednoduše to není možné. Zase tak úplně beznadějně to však není a i prohlížeče exponují množství různých funkcí: Rozhraní API Geolocation umožňuje zjistit zeměpisnou polohu zařízení (pokud je tato funkce povolena), rozhraní API DeviceOrientation zpřístupňuje údaje dodávané gyroskopem a akcelerometrem a pomocí rozhraní API Web Storage lze ukládat data mezi jednotlivými relacemi procházení. Přidáme-li ještě podporu zpracování audio- a videozáznamů jazyka HTML5, podporu gest na základě dotykových událostí prohlížeče, přechody a transformace jazyka CSS a podporu 3D grafiky na základě knihovny WebGL, je jasné, že možnosti webových aplikací opravdu zase tak omezené nejsou. Pravděpodobně se však vždy najde nějaká funkce, obvykle ta nejnovější a nejpokročilejší, kterou ve webové aplikaci opravdu využít nepůjde.

Když jsme se tedy shodli na tom, že jsou nativní aplikace nejlepší, proč se vlastně zabýváme vývojem aplikací webových?

## Nevýhody nativních aplikací

Jednou z hlavních nevýhod nativních aplikací je fragmentace trhu. Protože jsou nativní aplikace určeny pro konkrétní platformy, je třeba při vývoji nativní aplikace nejprve nalézt odpověď na následující otázku: Pro které platformy bude vlastně aplikace určena? Měli bychom ji začít distribuovat nejprve prostřednictvím služby App Store společnosti Apple, nebo raději prostřednictvím služby Android Market? A co systémy BlackBerry a Windows Phone 7? Přitom je třeba brát v potaz, že pro každou podporovanou platformu bude nutné aplikaci zcela přeprogramovat. V ideálním případě bychom vytvořili nativní aplikaci pro všechny tyto platformy, ale ve skutečnosti to nejde a musíme zvolit, na které platformy aplikaci omezíme – nebo, lépe řečeno, které uživatele o aplikaci ošidíme. V případě webové aplikace však tento problém zcela odpadá – pokud je zařízení vybaveno internetovým prohlížečem, jedinou aplikaci s jedinou verzí zdrojového kódu lze spouštět na všech platformách. Problematika fragmentace trhu se sice v omezené míře vztahuje i na prohlížeče, a tedy i na webové aplikace, ale webovým vývojářům jsou s ní obvykle velmi dobře obeznámeni a rozdíly mezi jednotlivými platformami jsou v tomto případě pouze velmi malé.

Dalším problémem nativních aplikací je rychlost vývoje. Jako profesionální webovým vývojářům již máte s vývojem a správou webových aplikací jistě bohaté zkušenosti. Osvojení úplně nové sady vývojových nástrojů nebo najmutí programátora, který disponuje příslušnými znalostmi,

může být časově či finančně velmi náročné, nemluvě o úsilí, které při tom musíte vynaložit. Proti tomu se dá namítnout, že taková argumentace je čistě ekonomická a nebere v potaz spokojenost cílových uživatelů. Proto je třeba oba argumenty pečlivě zvážit – rozhodnutí mezi webovou a nativní aplikací je vždy současně volbou mezi co nejlepším uživatelským prožitkem a pohodlným rychlým vývojem pro všechny platformy najednou. Zprostředkování co nejlepšího uživatelského prožitku je samozřejmě při vývoji aplikace prvořadým cílem, ale aplikace, kterou opravdu vyvinete, bude na druhou stranu pro uživatele jistě větším přínosem než aplikace, která nakonec nikdy nespátří světlo světa.

Stejně dilema v poslední době řešil management několika špičkových vývojářských společností a nakonec se rozhodl pro webovou cestu. Společnost 37signals, autor mnoha podnikových webových aplikací, včetně aplikací Basecamp a Highrise, se při vývoji verze aplikace Basecamp určené pro mobilní zařízení nativní cestě zcela vyhnula:

„Nakonec jsme došli k závěru, že bychom se měli držet toho, v čem jsme nejlepší: webových aplikací. Technologie používané k jejich vývoji známe velmi dobře, máme skvělé vývojové prostředí, zavedené pracovní postupy, dokonalou kontrolu nad celým vývojovým cyklem a na vývoji tohoto druhu aplikací se může podílet každý náš zaměstnanec.

...

S jazyky HTML, CSS a JavaScript pracujeme již po léta každý den. Výsledky, kterých dosáhneme při vývoji pro stolní počítače, můžeme využít při vývoji pro mobilní zařízení a výsledky, kterých dosáhneme při vývoji pro mobilní zařízení, zase můžeme využít při vývoji pro stolní počítače. Jedná se o dokonalý koloběh.“<sup>3</sup>

Pro společnost 37signals nepředstavovaly problém vynaložené náklady či prostředky; její management se jednoduše rozhodl, že webová aplikace bude větším přínosem pro větší počet uživatelů a že její vývoj na základě zažitých technologií zajišťuje lepší kontrolu nad celou aplikací.

K podobnému závěru došla i společnost Netflix. Její aplikace určená pro konzoli PlayStation 3 je kompletně naprogramována pomocí webových technologií a umožňuje vývojářům simultánní testování a aktualizace:

„Naším hlavním cílem je neustálé experimentování s technologiemi, funkcemi a zážitky, které nabízíme svým uživatelům. Testujeme každý nový nápad, abychom mohli posoudit jeho dopad na své zákazníky.

...

---

3 Jason Fried v blogu Signal vs. Noise, 1. února 2001 (<http://37signals.com/svn/posts/2761-launch-basecamp-mobile>)

Zde se projevuje síla jazyka HTML5. Technologii doručujeme zákazníkům ze svých serverů při každém spuštění aplikace. To znamená, že můžeme její funkce neustále aktualizovat, testovat a vylepšovat. Při testování funkcionality aplikace v konzoli PS3 jsme již provedli několik pokusů a na dalších v současné době pracujeme. Naši zákazníci nemusí při každé úpravě softwaru provádět jeho ruční aktualizace, změny se jednoduše „dějí samy“.<sup>4</sup>

Fragmentace trhu způsobená odlišnostmi jednotlivých platforem představuje vážný problém dokonce i pro společnost Facebook, která disponuje celou armádou programátorů (a která vyvinula *nejúspěšnější aplikaci pro iPhone všech dob*), a i tato společnost proto založila svou strategii přítomnosti na trhu mobilních zařízení na webových standardech.<sup>5</sup>

Webové aplikace pro mobily nabízí ve srovnání s nativními aplikacemi několik výhod, a i přesto, že čelí některým problémům návrhu, vývoje a nasazení, představují výkonné víceplatformové řešení, které je škálovatelné a ekonomicky dostupné.



### ROZHRANÍ API SPOJUJÍ NÁRODY

Ačkoliv se společnost 37signals rozhodla nativnímu vývoji interně vyhýbat, ve službě App Store společnosti Apple je v současné době pro její webovou aplikaci Basecamp k dispozici nejméně deset nativních klientů. Komplexní rozhraní API, které aplikace nabízí, umožňuje vývojářům třetích stran vyvíjet nativní aplikace, které s aplikací Basecamp spolupracují, a současně zaručuje, že společnost 37signals neztratí úplnou kontrolu nad typem operací, které tyto aplikace provádějí s daty uživatelů. Dobře navržené rozhraní API umožňuje, aby na vámi položených základech vyvíjeli své vlastní aplikace i uživatelé, kteří někdy přijdou se zcela originálními nápady.

## Pěkně od začátku

„Ale my *potřebujeme* aplikaci pro iPhone.“ Dobře, nic proti tomu, ale nativní aplikace pro různé platformy není všechno. Při volbě typu aplikace byste se neměli řídit pouze imperativem „ale má ji přece každý“. V první řadě je třeba zvážit, zda je vůbec vhodné vyvíjet aplikaci pro mobilní zařízení (ať už nativní nebo webovou). Vlastníte-li řetězec kaváren se stovkou poboček po celé zemi, bude zřejmě aplikace pro mobilní zařízení, která pomůže vašim zákazníkům vyhledat vaši nejbližší kavárnu, zcela na místě. Pokud však potřebujete vytvořit reprezentaci malého bistra za rohem, asi bude lepší zvolit nějakou jednodušší alternativu.

4 John Ciancutti v blogu The Netflix „Tech“ Blog, 3. prosince 2010 (<http://techblog.netflix.com/2010/12/why-we-choose-html5-for-user.html>)

5 <http://networkeffect.allthingsd.com/20110125/facebook-sets-mobile-sights-on-html5/>

Opravdu lidé potřebují to, co nabízíme? Proč by měli naši aplikaci používat, zrovna když jsou mobilní? Kde ji budou používat? Jaké výhody nám to přináší z obchodního hlediska?

Dobrým způsobem, jak získat odpovědi na tyto otázky, je analýza informací, které již máte k dispozici. Podívejte se na aktuální statistiky svého webu: Kolik návštěvníků jej navštěvuje ve svých mobilních zařízeních? Jaká zařízení používají? Jaký obsah vašeho webu je zajímavé nejvíce? Tato data vám pomohou udělat si představu o tom, co lidé v mobilním kontextu na vašem webu vlastně hledají. Data budou samozřejmě ovlivněna omezeními vašeho webu, takže ať už z nich vyvodíte jakékoliv závěry, měly by být pouze jedním z podnětů ke stanovení konečného rozhodnutí.

A když žádná taková data k dispozici nemáte? Vždy se můžete na to, co potřebujete vědět, zkusit uživatele zeptat. Při průzkumech tohoto typu obvykle zjistíte, že uživatelé chtějí v podstatě cokoli, co můžete nabídnout, a to pokud možno co nejrychleji. Jste květinář? Budou chtít květiny – *hned*. Máte kavárnu? Budou chtít kávu – *také hned*. Ať už nabízáte jakékoliv produkty či služby, vytvoříte-li aplikaci, která tyto požadavky splňuje, uživatelé ji určitě oceňují (a vy vyděláte peníze).



### EFEKT TRŽIŠTĚ APLIKACÍ

Úspěch služby App Store společnosti Apple nelze ignorovat: Prezentace aplikace v tak populární službě, jakou je obchod App Store společnosti Apple, představuje nepopíratelnou marketingovou výhodu a zobrazení ikony uprostřed domovské obrazovky zařízení uživatele zajistí takovou propagaci aplikace, jaké přidáním záložky do oblíbených položek nikdy nedocílíte. Obchody s aplikacemi navíc zcela přesně definují způsob zpeněžení aplikace: Přitahují zákazníky a tito zákazníci v nich utrácejí peníze. My sice vyvineme aplikaci webovou, to však v žádném případě neznamená, že tento potencionálně lukrativní způsob distribuce vynecháme. Zde se uplatní hybrid nativní a webové aplikace, o kterém začneme podrobně hovořit v kapitole 7 – to už ale předbýváme, takže zpátky k tématu.

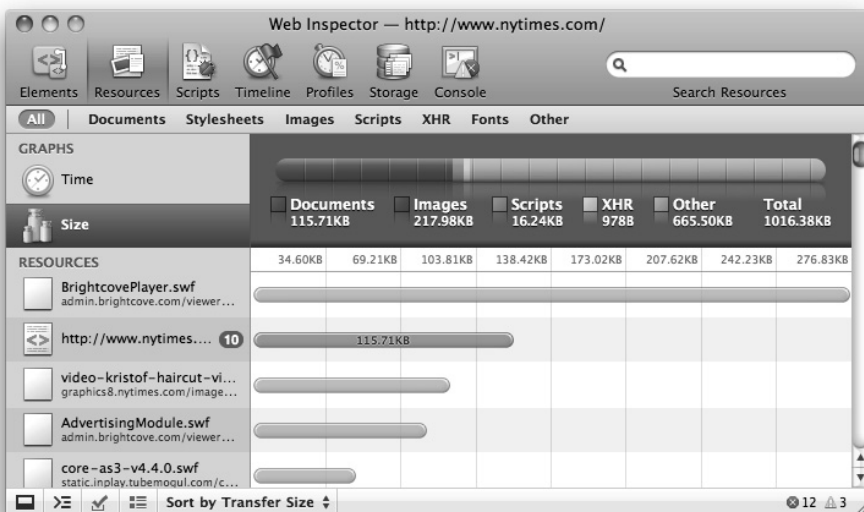
## Aplikace nestačí

Největším argumentem ve prospěch vývoje aplikací pro mobilní zařízení na základě webových technologií je fakt, že bychom alespoň část práce, kterou tento úkol obnáší, stejně museli odvést. Uživatelé oprávněně očekávají, že web, který již máte, bude fungovat i v jejich mobilních zařízeních. A při optimalizaci webu pro mobilní zařízení bychom neměli spoléhat na pravděpodobný typ nebo schopnosti těchto zařízení, protože naše předpoklady se nevyhnutelně projeví jako mylné (tento princip platí v případě Internetu v podstatě obecně). Z toho je zřejmé, že nativní aplikace tento problém nijak neřeší.

O tom, že je navrhování aplikací pro mobilní zařízení především o kontextu, jsme již hovořili. Podstatným faktorem je však i *rychlost zprovoznění aplikace*. Naším cílem je nejenom nabídnout uživatelům to, co chtějí, ale také nabídnout jim to co nejrychleji. Dokonalá nativní aplikace splní svůj účel pouze v případě, že si ji uživatelé nainstalují. Chtít po uživatelích, aby navštívili stránky tržiště a na nich si aplikaci stáhli (což obvykle chvíli trvá), může být trochu moc – zejména když jsou již na cestách a spoléhají pouze na připojení pomocí sítí mobilních operátorů. Vytvoření verze našeho webu určené pro mobilní zařízení je proto nutné, ať už nějakou nativní aplikaci nabízíme, či nikoliv. Takže co s tím?

## První možnost: Nic

První možností je neudělat nic – rozhodně se nejedná o možnost, která by měla být opomenuta. Nové typy smartphonů již umožňují velmi snadnou navigaci i na těch nejsložitějších stránkách. Domovská stránka novin *The New York Times* například nabízí obrovské množství informací o široké škále témat. Z technického hlediska však načtení této stránky není zadarmo – k načtení domovské stránky webu <http://nytimes.com/> je třeba stáhnout téměř 1 megabajt dat; viz analýza v nástroji Web Inspector prohlížeče Chrome, kterou si můžete prohlédnout na obrázku 1.1.



**Obrázek 1.1** Nástroj Web Inspector prohlížeče Chrome odhaluje skutečnou cenu, kterou platíme za procházení plnohodnotných stránek



Pokrytí sítěmi 3. generace je v současnosti sice již velmi kvalitní, ale stále po uživateli chceme, aby čekali čtyři nebo pět sekund, než budou s naším webem moci začít interagovat. To není zrovna mobilní ani příjemné. Pokud bychom se tedy již pro tuto cestu nejmenšího odporu opravdu rozhodli, museli bychom vytvořit velmi dobře strukturovaný a implementovaný web, který by tyto negativní dopady minimalizoval. Použití správných vývojových postupů se při procházení webu v mobilních zařízeních projeví ještě více než při jejich procházení v prohlížečích stolních počítačů. Lehká rozložení založená na jazyce CSS, návrh typu content-out a zaměření na přístupnost a použitelnost – tyto faktory jsou ještě důležitější, pokud má uživatel k dispozici méně času, menší obrazovku a pomalejší připojení k Internetu.

## Druhá možnost: Transformace

Na pomoc přichází návrh typu responsive design. Tedy alespoň něco na ten způsob. Pokud jste z nějakého důvodu ještě nečetli klíčový článek, který o tomto tématu napsal Ethan Marcotte, silně vám doporučujeme, abyste si našli čas a přečetli si jej.<sup>6</sup> Pojem „responsive design“ reprezentuje zvláštní umění využití dotazů na média jazyka CSS, dynamických mřížek a dynamických obrázků k přizpůsobení webu rozlišení obrazovky zařízení (nebo okna prohlížeče) uživatele a jeho navrzení tak, aby nabízel co nejlepší možné rozložení pro jakékoli rozlišení.

Jedná se o geniálně jednoduchou techniku, která stojí za pozornost. Dotazy na média jsou rozšířením známých atributů typů médií, které se již dlouho používají k implementaci tiskových šablon stylů na stránkách HTML. Rozdíl spočívá v tom, že místo toho, abychom pro tato pravidla CSS zavedli jednotný kontext, můžeme se dotazovat (odtud také „dotazy na média“) na fyzické vlastnosti zařízení uživatele. To znamená, že můžeme doručovat jednotlivé styly šablon (nebo jejich části) pouze do zařízení, která splňují daná kritéria. V níže uvedeném příkladu říkáme: „Soubor **mobile.css** načti, pouze pokud je šířka zobrazované oblasti menší nebo rovna 480 pixelům.“

```
<link rel="stylesheet" type="text/css" media="screen and (max-width: 480px)"
href="mobile.css" />
```

Při tom samozřejmě nejsme omezeni pouze na zjišťování šířky obrazovky zařízení. Z široké škály funkcí a vlastností podporovaných ve specifikaci dotazů na média se v dotazech na média můžeme ptát mimo jiné na následující vlastnosti zařízení:

- výšku a šířku obrazovky (viz výše uvedený příklad),
- rozlišení obrazovky,
- orientaci zařízení,
- poměr stran obrazovky zařízení.

6 <http://www.alistapart.com/articles/responsive-web-design/>

Opravdová síla dotazů na média však spočívá v možnosti jejich kombinace do složitějších pravidel. Potřebujete nabídnout specifické styly pro zařízení s vysokým rozlišením obrazovky ve vodorovné orientaci? Žádný problém:

```
<link rel="stylesheet" type="text/css"
  media="screen and (min-resolution: 300dpi) and (orientation: landscape)"
  href="mobile.css" />
```

Precizní a elegantní, že? Pomocí tohoto přístupu tedy můžeme zobrazovat dokonalý web pro mobilní zařízení, parádní web pro stolní počítače a cokoli mezi tím – to vše s využitím jediného zdrojového kódu.

Tak jednoduché to však bohužel není. I návrh typu responsive design má svá omezení. V určitém smyslu je to něco podobného, jako když nalícíte prase. Díky přeformátování rozložení může být vzhled webu v mobilním zařízení přijatelnější, ale v podstatě je to jenom klam. Existuje dokonce i tvrzení, že návrh typu responsive design je ve skutečnosti horší než nedělat nic, protože nutí uživatele stahovat prostředky, které nikdy ani nemusí zobrazit. Některým z těchto problémů však lze při pečlivé přípravě předejít.

Zprvče web vyvíjejte vždy *nejprve pro mobilní zařízení*. Ačkoliv mají vývojáři tendence považovat dotazy na média pouze za prostředek, pomocí něhož lze ještě trochu vylepšit web, který je již v podstatě hotový, ve skutečnosti byste měli postupovat přesně opačně. Začněte vždy jednoduše a teprve poté postupně přidávejte složitější funkcionalitu určenou pro stolní počítače:

```
<link rel="stylesheet"
  media="screen and (min-width: 939px)" href="desktop.css" />
```

S touto koncepcí jste se již jistě setkali, jedná se o tzv. postupné vylepšování. Odesílání stejného kódu jazyka HTML do každého zařízení se sice nevyhneme, budeme-li však své stránky konstruovat a své styly šablon doručovat opatrně, zajistíme našim uživatelům co nejpříjemnější práci s naším webem a budeme jim prezentovat především obsah, který je zajímá. Cílem je minimalizace ztráty výkonu při zachování stejných funkcí, které má web při procházení ve stolním počítači.

## Třetí možnost: Navždy sám

Obvyklejším řešením je vytvoření úplně odděleného webu určeného pro mobilní uživatele. Podobné weby jsou obvykle součástí subdomény `m.` nebo `mobile.` (nebo obou). Tato možnost nám zajistí, že se budeme moci věnovat pouze těm potřebám našich uživatelů, které mají v mobilním kontextu. Perfektní.

Tedy skoro. I tento přístup má svá negativa. Oddělení samostatné verze webu určené pro mobilní zařízení obvykle předpokládá, že máme k dispozici nějaký detekční mechanismus,

na jehož základě můžeme zjistit, kterým zařízením doručíme kterou z verzí našeho webu – jednu verzi budeme například doručovat uživatelům iPhone, jinou uživatelům prohlížeče Firefox atd. Z teoretického hlediska to zní skvěle, ale ve skutečnosti je tento typ detekce (detekce prohlížeče na základě informací, které o sobě poskytuje při zasílání požadavků serveru) velmi nespolehlivý. Řetězec uživatelského agenta lze v mnoha prohlížečích snadno změnit (a velmi často se to i dělá, aby bylo možné popisovanou techniku obejít). A dokonce i kdyby se nám podařilo na našem webu v doméně `.mobi` implementovat spolehlivou funkcionalitu detekce aktuální generace mobilních prohlížečů, mohli bychom tímto přístupem způsobit nepředvídatelné problémy uživatelům zařízení, která neznáme. Je to otázka volby: Chceme uživatele nutit, aby používali web, který jsme pro ně vybrali?

Řešení je velmi jednoduché: Nejlepší je nabídnout uživatelům odkaz na standardní verzi webu (a obráceně) a poté jejich rozhodnutí respektovat. Na navádění uživatelů na verzi webu určenou pro mobilní zařízení není nic špatného, ale neměli bychom je nijak dále omezovat a bránit jim v přístupu na stránky, které mohou normálně navštěvovat.

Dobrym příkladem správné implementace tohoto chování jsou weby služby Facebook. Tyto weby řeší důvody, které vás mohou vést k umožnění přepínání mezi standardním webem a webem určeným pro mobilní zařízení. Služba Facebook nabízí dva weby určené pro mobilní zařízení: `web.touch.facebook.com`, na který jsou směřováni uživatelé mobilních zařízení s dotykovou obrazovkou, a `web.m.facebook.com`, určený uživatelům, kteří dotykovou obrazovku nemají. Oba weby umožňují provádění normálních úkonů a interakcí, které jsou uživatelé ve službě Facebook zvyklí provádět: mohou číst zprávy a odpovídat na ně, zveřejňovat aktualizace stavů a prohlížet si aktivitu uživatelů na zdech, což je základní funkce služby. Verze webu určené pro mobilní zařízení však neumožňují provádět *všechny* úkony, které umožňují provádět jejich standardní klony, například nahrávat fotografie nebo upravovat uživatelský profil. Pokud uživatelé úkon, který na verzi webu určené pro mobilní zařízení provést nelze (nebo se provádí na standardní verzi snáze), provést potřebují, mají na obou verzích webu služby určených pro mobilní zařízení k dispozici šikovné odkazy (v zápatí stránek), pomocí nichž mohou snadno přejít na standardní verzi webu. Klíčové je vždy umožnit uživatelům snadný přístup na plnohodnotnou verzi webu a neochudit je o žádnou jeho funkcionalitu. Oddělujte, ale neizolujte.

## Poznámka k prostředí

Při průzkumu technologií, které lze využít při vývoji webových aplikací pro mobilní zařízení, bezpochyby narazíte na víceplatformová vývojová prostředí. Mezi nejdůležitější z nich patří

projekty Sencha Touch<sup>7</sup> a jQuery Mobile<sup>8</sup>. Problematikou jejich implementace se zde zabývat nebudeme, ale obě dvě stojí za rychlé shrnutí, abyste si mohli udělat představu o tom, zda opravdu splňují to, co slibují.

Obě tato prostředí jsou v podstatě prostředními jazyka JavaScript. Aplikace vyvinuté v prostředí Sencha Touch se zcela spoléhají na to, že budou jejich cílová zařízení vybavena kvalitním strojem jazyka JavaScript. Že tomu tak v praxi často není, však již jistě víte. Spustíte-li aplikaci naprogramovanou v prostředí Sencha Touch bez nainstalovaného stroje jazyka JavaScript, bude výsledkem prázdná skořápka, která nic nedělá. Pro prostředí jQuery Mobile pak využívá o něco flexibilnější přístup postupného vylepšování; aplikace vyvinuté v tomto prostředí se sestavují v čistém jazyce HTML a jejich pokročilá funkcionalita je teprve následně přidávána na tento základ. Jedná se o další výměnu „něco za něco“ – jak už vám asi začíná docházet, svět vývoje pro mobilní zařízení je výměn tohoto typu plný. Pro prostředí Sencha Touch využívá koncepci „pouze jazyk JavaScript“ proto, že zaručuje lepší výkon – logika aplikace je sestavena čistě v jazyce JavaScript a není naroubována na model DOM. Bez ohledu na zvolený přístup byste však o těchto prostředích měli vědět především to, že obě implementují obrovské množství funkcí, které replikují chování a funkcionalitu nativních aplikací, a že pokud všechny tyto funkce nevyužijete, zbytečně zatěžujete prostředky zařízení.

To nás přivádí k další koncepci, kterou je třeba brát v potaz při návrhu webových aplikací pro mobilní zařízení: koncepci **tajemného valu**. Tajemný val je název teorie, která říká, že čím více se roboti blíží lidem, tím pravděpodobněji budou lidé jejich vzhledem znechuceni. Tuto myšlenku pak lze aplikovat i na uživatelská rozhraní, a měli byste ji proto brát v potaz i ve chvíli, kdy začnete s navrhováním vzhledu (a chování) aplikace.

Pokud něco vypadá jako kachna, kváká jako kachna, ale není to kachna, nemohou uživatelé vědět, jak s tím mají zacházet. Replikací vzhledu a funkcionality nativních aplikací vývojová prostředí nastavují určitou laťku, které však již z podstaty nelze dosáhnout. Jak se s tím tedy vyrovnat? Odpověď je jednoduchá: S omezeními se jednoduše smiřte. Není žádný důvod předstírat, že aplikace přesahuje rámec možností běžného rozhraní prohlížeče. Zaměření aplikace na mobilní zařízení není kletba, ale naopak příležitost k vlastní invenci. Verze webu `twitter.com` určená pro mobilní zařízení se vůbec nesnaží vypadat jako kterákoliv z nativních aplikací Twitter. Pouze dělá to, co od ní uživatelé očekávají: zprostředkovává jim co nejrychleji co největší množství informací.

---

<sup>7</sup> <http://www.sencha.com/products/touch/>

<sup>8</sup> <http://jquerymobile.com/>

## Vyhrňte si rukávy

Tohle povídání je sice zajímavé, ale neměla by kniha ve skutečnosti být o *vývoji* webových aplikací? Máte pravdu, měla. A i když je rozbor důvodů, které nás vedly ke zvolení konkrétní strategie, jistě zajímavý, něco skutečně tvořit je mnohem zajímavější. Takže co budeme vlastně tvořit? Bylo nebylo a velmi zajímavý klient nás oslovil s požadavkem na vytvoření verze svého populárního webu StarTrackr určené pro mobilní zařízení. StarTrackr je web zaměřený na sledování celebrit, který umožňuje přidávat záznamy o spatření celebrit „v terénu“, což je funkcionalita jako stvořená pro web pro mobilní zařízení.<sup>9</sup> Pojdme si tedy zopakovat, z jakých možností máme při řešení tohoto úkolu na výběr: 1. Neuděláme nic (to se asi našemu klientovi líbit nebude); 2. vytvoříme web s návrhem zaměřeným na mobilní zařízení; nebo 3. vytvoříme oddělenou (ale příbuznou) verzi webu určenou pro mobilní zařízení. Volba toho pravého postupu pak záleží především na tom, čeho chceme (nebo chce spíše náš klient) dosáhnout. Klient vyžaduje, aby uživatelé mohli na webu StarTrackr pro mobilní zařízení provádět následující úkony:

- zobrazovat blízka místa spatření celebrit a různá spatření, ke kterým došlo na těchto místech,
- vyhledávat své oblíbené celebrity a prohlížet si záznamy o jejich nedávných spatřeních,
- přidávat nová spatření.

Když se nad touto sadou požadovaných funkcí zamyslíme, je jasné, že pouhý web nebude stačit a že budeme muset vyvinout *webovou aplikaci*. Jednoduše řečeno: weby slouží k prohlížení informací, zatímco webové aplikace slouží k provádění úkonů. Porozumění tomuto rozdílu a vhodnosti popisovaných technik pro tyto dvě alternativy je velmi důležité. Profesionální make-up toho zakryje hodně, ale chcete-li uživatelům zprostředkovat přesvědčivý a do kontextu zasazený uživatelský prožitek – (a to jistě chcete), bude třeba vyvinout samostatnou aplikaci pro mobilní zařízení.

Dejme se tedy do toho.

---

<sup>9</sup> Web StarTrack je bohužel vymyšlený, takže budete muset své bulvární potřeby ukojit jinde.