

KAPITOLA 6

TVORBA UŽIVATELSKÉHO ROZHRAŇÍ

Praktické využití teorií osobitého designu

Bez kódu nemůže design v on-line prostředí existovat. HTML, CSS, JavaScript a další jazyky, frameworky, techniky a nejlepší postupy proměňují črty a umělecké výtvary v to, co dokáže prohlížeč vykreslit na obrazovku počítače. Tvorba jedinečných layoutů, které reagují na interakci uživatelů a poskytují jim požadovanou funkcionalitu, obsah a prožitek, se stala vášní nejednoho tvůrce webů. Musíte být proto schopni vytvořit a implementovat vaše výtvary takovým způsobem, jenž povede k zformování krásného rozhraní.

Tato kapitola se věnuje tomu, jak můžete pracovat s elementy jazyka HTML (jimž se jinak také říká značky) a mikroformáty na své cestě za tvorbou osobitých, bohatých layoutů. Dozvíte se o různých vlastnostech jazyka CSS, jež pomohou zrealizovat váš záměr, včetně toho, jak jednotlivými styly cílit na konkrétní média. Poté se přesunete k poskytování interaktivity pomocí JavaScriptu a frameworků, jako je třeba knihovna jQuery, abyste dosáhli funkčně bohatšího a silnějšího prožitku. A nakonec se dočtete o dějinách kódu a o tom, jak ovlivňuje prohlížeče a jak jej ladit, a o tom, jak budoucnost ovlivní vaši schopnost přidávat vašim stránkám na osobitosti a poskytnout tak uživatelům nejlepší možná rozhraní.

Zdrojový kód a mikroformáty

Všechny aspekty designu se točí okolo konceptu poskytování prožitku návštěvníkům. Ačkoliv design uživatelského rozhraní má stejný primární cíl (poskytnout něco použitelného), je důležité, aby vaše designy odrážely vaši tvrdou práci a proměnily veškeré možné teorie a fantastické myšlení do zrealizovaného řešení, které si uživatelé užijí. Neignorujte osobitost, jež poskytuje svět reálného kódování.

Osobitý, výrazný design pouze nesedí v zadní lavici, odkud poskytuje řadu teorií hlásajících nejlepší postupy designu. Předává vám myšlenky a příklady toho, jak věci vypadají na obrazovce, jak se zobrazí vaše obrázky, a co je důležitější, jak můžete (s trochou zkušenosti s kódováním) hloupý nebo méně citlivý layout (či naopak ten přehnaný) proměnit v něco krásného, elegantního a použitelného. Ideálně využijete výhod nabídnutých teorií; kód nabízí jedinou metodu, jak toho dosáhnout!



Existuje diskuze o tom, zda by designéři měli umět kódovat (nebo naopak jen poskytovat estetično). Protože nabízíte materiály pro web, zdá se, že je přirozené naučit se kódovat (abyste byli schopni využít vše, co toto médium nabízí). Nakonec přece jen platí, že když nemůžete použít úplně vše, proč se všechno učit.

Jak tedy může webu hned dodat na osobitosti (s pouhým použitím kódu)? Odpověď je nasnadě: Tvořte rozhraní pomocí jazyků HTML, CSS, JavaScript, a to přesně tak, jak určuje specifikace. Tato kapitola se bude věnovat každému z těchto jazyků. Podívá se na to, co nabízejí dobrému designérovi. Ukážeme si, jak tyto jazyky mohou pomoci dodat vašemu webu osobitost. Obrázek 6.1 ukazuje, že před zapojení těchto jazyků s jejich přesahem (na různé platformy), je možnost poskytnutí jedinečného vzhledu vašemu webu jen velmi omezená.

Vítejte na mém WML webu.

Protože tento web je optimalizovaný pro stará kapesní zařízení, jeho funkčnost je omezená. Nabídnout vám můžeme také plnohodnotný mobilní web určený pro chytré telefony (pokud jej máte).

[Novinky](#)

[Produkty](#)

[Služby](#)

[Kontakt](#)

[O nás](#)

Obrázek 6.1: Před érou chytrých telefonů znala mobilní zařízení jazyk WML, který nebyl osobitý ani trochu

Ještě než se vrhneme na schopnosti jazyka HTML (a než se dostaneme k dalším užitečným jazykům), stojí za zmínku kompatibilita (více informací ohledně kompatibility webdesignu viz kapitola 4). Ačkoliv HTML4 a XHTML 1.0 jsou podporovány velmi slušně, určitá zařízení a webové prohlížeče poskytují jen malou podporu HTML5 (poslední verzi v tobě tvorby knihy) nebo preferují omezenější mobilní profily či WML (jazyk, kterého by se kvůli jeho omezením měli designéři skutečně bát).

Pokud hledáte nějaký konformní příjemný jazyk použitelný na více platformách, pak ze svého seznamu vyškrtněte XHTML 1.1. Není kompatibilní s Internet Explorerem verze 6 až 8. A dejte si také pozor na používání HTML5 a CSS3. Pamatujte si, že ne všechna zařízení nebo prohlížeče jsou si rovnocenné. Pro účely osobitého designu je klíčová viditelnost (pokud lidé nemohou váš web vidět, kvůli problémům s jeho vykreslením či chybám, pak budou trpět), a designy tak mohou fungovat, pouze pokud nabídnete dobré a bezpečné řešení. To mějte při práci vždy na mysli.



S moderními standardy, jako jsou HTML5 a CSS3, již není nutné ladit váš kód pro konkrétní vykreslovací jádra prohlížečů. Pro CSS3 nabízejí výrobci prohlížečů řadu předpon. HTML5 často vyžaduje přidávání dalšího kódu a skriptů, aby vše fungovalo. Ale i když dostat vše pod kontrolu může být složité, výsledný krásný design může stát za to.

Elementy určují osobitost

Ačkoliv strukturální jazyky, jako je HTML, nemusíte považovat za něco, co přispěje osobitosti designu, jejich role je důležitější, než si většina lidí uvědomuje. CSS má moc kreslit na plátno vašeho webu vizuální parádičky běžně spojované s designem, ale struktura webu říká hodně o vztahu a vlivu obsahu na stránce. Nezapomínejte na důležitost strukturálních jazyků v osobitém designu. Můžete objevit, že za určitých okolností může o všem rozhodovat právě kód.

Jedním z nejpřehlíženějších míst osobitého designu je záhlaví stránky (hlavička dokumentu). Zvažte například titulek stránky (`title`), jenž je jedním z více elementů záhlaví (`head`). V prohlížeči má tento element význam v dodání jedinečného a snadno čitelného názvu okna. Obrázek 6.2 ukazuje příklad identifikace stránky mezi řadou jiných záložek nebo oken. Takové konvence nemusí vypadat jako důležité, ale dokážou zvýšit viditelnost a snížit množství času, které musí uživatel strávit orientací při práci s více záložkami či okny. Titulek volte uvážlivě. Je třeba, aby byl popisný a ne příliš dlouhý.



Obrázek 6.2: Krátký a jedinečný titulek zastupuje stránku na záložce nebo v okně prohlížeče

Element `title` není jediným prvkem záhlaví, který hraje roli v odlišení vašeho webu. Elementy `link` a `style` mohou osobitosti a vzhledu stránky pomoci také tím, že odkazují na kód jazyka CSS, který je nutný k vykreslení jedinečnosti layoutu. Navíc se k nim přidává ještě i element `script`, jenž odkazuje na připojený kód jazyka JavaScript (a ten může ovlivnit vizuální stránku věci, pokud je k tomu určený). Zdrojový kód vašeho webu má dopad na vizuální stránku věci sám o sobě. Například si vezměte *carousel* (neboli kolotoč) knihovny jQuery.

Pomocí něj můžete poskytnout rotující snímkovou prezentací (slideshow), jejíž na kódu založená interaktivita dokonce dodává obsahu animace.

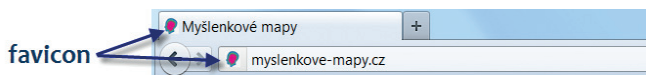
Ačkoliv použití CSS nebo JavaScriptu s odpovídajícími značkami je obvyklým příkladem vlivu HTML na váš design, stojí za to se zamyslet nad prohlížečem – ten kódu ve výchozím stavu dodává základní vizuální styl (pokud není přítomen kód CSS). Toto výchozí stylování dodává základní úroveň osobitosti, z níž můžete jednak něco vytěžit nebo ji přepsat. Poskytuje solidní základ stylování s garantovanou osobitostí (jediným problémem je fakt, že každý prohlížeč má své vlastní výchozí hodnoty, které vedou k resetování CSS a v tomto případě i k několika vykreslovacím chybám).



TIP

I když značky pro styly a skripty vám opravdu dávají možnost vložit do stránky zajímavé objekty, můžete především oddělit styly, strukturu a chování od sebe, abyste mohli využít ukládání do vyrovnávací paměti (čímž ušetříte množství přenesených dat) a zjednodušit si údržbu.

Mimo použití těchto elementů, abyste váš layout trochu odlišili, vám hlavička dokumentu HTML umožňuje také odkázat na ikonu webu (*favicon* jakožto zkratka pro *favourite icon*) a ikony Apple Touch, které poskytují při zatřídění odkazu na web mezi oblíbené položky prohlížeče malý obrázek, jímž se odkaz odliší od ostatních odkazů (viz obrázek 6.3). Pokud jste zatím (z hlediska osobitosti) nad hlavičkou nepřemýšleli, nyní je čas na prozkoumání skvělých možností, které nabízí. Malé detaily vašemu webu dodají dojem dotaženosti a větší vizuální osobitosti. Pokud nejsou vtíravé, pak se vyplatí!



Obrázek 6.3: Ikony jsou malé, ale oblíbeným položkám nebo záložkám dodají na osobitosti (což je užitečné)



ODKAZ

Abyste se dozvěděli více o specifických funkcích konkrétních elementů jazyka HTML, navštivte adresu <http://meiert.com/en/indices/html-elements/> nebo se podívejte do volně dostupné reference HTML například na adrese <http://www.jakpsatweb.cz/html/>.

Na rozdíl od záhlaví je tělo dokumentu místem, v němž se nachází hlavní část viditelné, přirozené a strukturální osobitosti. HTML nabízí řadu elementů mají-

cích vliv na estetiku webu (dokud jejich styl nepřepíšete). Každou značku v tělu dokumentu lze stylovat. K přepsání výchozích stylů prohlížeče použijte CSS.

Jak jsou tyto elementy osobité? V některých případech, jako je třeba značka `div`, elementy existují pouze k označení bloku obsahu nebo značek. Nadpisy mají větší velikost písma, aby vypadaly důležitěji. Některé elementy používají odsazení a značení, jako je tomu třeba u seznamů nebo tabulek, a na některé se vztahuje základní formátování (například na elementy `b`, `i`, `a`, `strong`, `em`, `ins`, `del` a `q`), které vyzdvihuje jejich jedinečnou roli a důležitost na stránce. Tabulka 6.1 ukazuje, jak jsou elementy vizuálně interpretovány (ve výchozím stavu), ale mimo toho mají také určitý sémantický a kontextuální význam.

Tabulka 6.1: Výchozí stylistické odlišnosti elementů jazyka HTML

Vizuální odlišnost	Ovlivněné elementy
Výchozí text	<code>p</code> , <code>div</code> , <code>abbr</code> , <code>acronym</code> , <code>dfn</code> , <code>span</code> , <code>form</code> , <code>label</code> a <code>legend</code>
Změna fontu	<code>pre</code> , <code>code</code> , <code>samp</code> , <code>kbd</code> a <code>tt</code>
Velikost textu	<code>h1</code> , <code>h2</code> , <code>h3</code> , <code>h4</code> , <code>h5</code> a <code>h6</code>
Tučný text	Nadpisy (<code>h1</code> až <code>h6</code>), <code>strong</code> a <code>b</code>
Kurziva	<code>em</code> , <code>var</code> , <code>i</code> a <code>cite</code>
Text s čárou	<code>del</code> , <code>ins</code> , <code>a</code>
Odsazený text	<code>dl</code> (<code>dd</code>), <code>ol</code> (<code>li</code>) a <code>blockquote</code>
Jiné efekty	<code>a</code> , <code>hr</code> , <code>br</code> , <code>bdo</code> , <code>q</code> , <code>sub</code> , <code>sup</code> , <code>button</code> , <code>fieldset</code> , <code>input</code> , <code>select</code> , <code>textarea</code> , <code>table</code> , <code>frame</code> a <code>iframe</code>
Nic	<code>area</code> , <code>img</code> , <code>map</code> , <code>object</code> a <code>param</code>

The image shows a web form titled "Lots of options" with the following components:

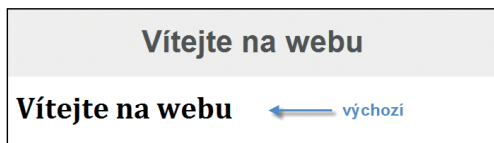
- A checked checkbox labeled "Check me quick!".
- Two radio buttons, both labeled "Which Option?".
- A text input field containing "This is a text box".
- A dropdown menu with the text "Select an Option".
- A button labeled "This is a button".
- A file upload button labeled "Choose File" with the text "No file chosen" next to it.
- A text area containing the text "This text box needs filling in by a user!".

Obrázek 6.4: Existuje mnoho komponent formuláře a každá z nich má svůj vlastní vizuální styl a funkci

Mimo síly elementů běžného textu nebo obrázků, které poskytují určitou úroveň osobitosti na stránce, jsou zde formulářové elementy HTML, jež používají snadno rozeznatelnou osobitost. Ačkoliv komponenty uživatelského rozhraní, jako jsou textové bloky, rozevírací nabídky a tlačítka, jsou snadno rozeznatelné, mohou si také podržet jedinečné stylování (s omezeními), čímž uživatelům poskytnou něco interaktivního, takže se může vyplatit držet se zavedených trendů designu (jak ukazuje obrázek 6.4).

Některé elementy přirozeně vypadají a chovají se osobitě (jako třeba `strong` nebo `em`). Jeden speciálně, element `hr` pro vodorovnou čáru, slouží k tvorbě vizuálního oddělení obsahu. Vyváženým použitím těchto typů elementů (přičemž je třeba, abyste byli opatrní, jak je externě stylujete pomocí CSS) čerpáte z teorie stojící za tvorbou osobitějšího textu stránky.

Rychlým a snadným způsobem, jak zjistit přirozené stylování elementů (a kolik je jim ve výchozím stavu přiděleno důrazu), je experimentování s kódem. Nakonec budete chtít použít CSS kdykoliv to bude možné, abyste svému webu dodali jedinečný vzhled, jež vyžaduje (jazyk HTML je navržen ke strukturování textu, nikoliv k jeho stylování). Černobílý text jen se základním stylováním vypadá primitivně, s čímž u návštěvníků cenu za vizuální estetiku nevyhrajete (viz obrázek 6.5), ale pochopení, jak jej používat, stojí za to.



Obrázek 6.5: Výchozí stylování není nic zvláštního, ale jisté náznaky osobitosti v něm nalezneme

Když vytváříte webové stránky zcela od nuly, zvažte využití výhod výchozího formátování těchto značek (jako třeba ponechání nadpisů větším písmem nebo ponechání podtržení u odkazů) nebo použití CSS k přidání vylepšení, například když při použití značky `kbd` k označení klávesové zkratky jí můžete přidat obrázek na pozadí, aby vypadala skutečně jako klávesa na klávesnici. Nebo například značka `strong` dostane formátování ve formě tučného řezu písma.

Protože se ale tato kniha zaměřuje na design, musíte znát význam a použití elementů jazyka HTML. Na toto téma najdete řadu knih a návodů. Jako designér si

musíte všimnout každému elementu, jenž se na stránce objeví. Nevšimnout si je příliš velké riziko. Například, i když slovu dodáte zdůraznění (ale resetujete veškerou vizuální stránku), stále má vliv na osobitost webu. Cílem kódování designu je zobrazit váš obsah nejlepším možným způsobem. Ujistěte se o tom.

Při označování obsahu vašeho webu zvažte, zda používáte značky k tomu účelu určené. Některé elementy se přirozeně k tvorbě osobitých objektů hodí více než jiné. Například byste nevyměnili element `img` (označující místo pro obrázek, na který odkazuje) za element `span` (určený k seskupování řádkových elementů). Zdrojového kódu si navíc všimají vyhledávače. Sémanticky relevantním zdrojovým kódem zajistíte, že web bude možné snadněji vyhledat (web složený pouze z obrázků nebude možné kvalitně optimalizovat pro vyhledávače).



Pokud chcete něco stylisticky formátovat kurzivou, nepoužívejte pouze značku `i` (protože jde o značku zavrženou, jež byla nahrazena lepší variantou v CSS). Vizuální vzhled vašeho webu musí obsah vašeho webu doplňovat. I když použitím zavržených značek dosáhnete odlišení (značka `i` vede k formátování kurzivou), je lepší kódovat správně podle specifikace.

Zde je seznam některých osvědčených postupů (vizualizace stylů elementů):

- V seznamech používejte odrážky nebo číslování (mimo těch použitých jako navigační nabídky)
- Odkazy odlište jinou barvou než běžný text, aby je bylo možné snáze rozpoznat
- Používejte tečkované podtržení k naznačení efektů překreslení (u efektů pro jiné než hypertextové odkazy)
- Vyhýbejte se značkám (zavrženým a jiným) s nízkou podporou ze strany prohlížečů
- Správně používejte značky `sub` a `sup` (často se na ně zapomíná)

A nakonec: HTML nabízí určité možnosti základního stylování, které je důležité v situacích, kdy nelze do vykreslení stránky zapojit CSS (například v případě čteček obrazovky nebo textových prohlížečů). Proto věnujte trochu pozornosti případům, kdy se vaše stránky zobrazí „nahé“. Mnoho designérů se zaměřuje pouze na to, jaký vliv na vizuální stránku věci bude mít použití CSS a skriptů, ale i čisté HTML hraje svou roli v estetice.

Kouzlo s mikroformáty

Jako designéři strávíte spoustu času přidělováním stylů CSS do vašich layoutů, abyste vytvořili rozhraní vašeho webu, ale musíte také pamatovat na to, že obsah má stále nad stylem navrch. Kód označující váš obsah popisuje jeho účel a význam ve vztahu k dalším objektům na stránce, a i když nejde o vizuální stránku věci, stále má ohromný význam na osobitost webu. V jádru věci HTML slouží k dodání kontextuálního významu vašemu obsahu.

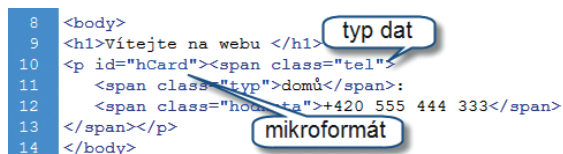
HTML může pomocí vašeho oblíbeného prohlížeče zajistit některé základní stylistické aspekty v případě nutnosti tvorby webu bez CSS a z důvodů použitelnosti, ale k dispozici máte také další nástroje, jež mohou zvýšit hodnotu obsahu a pomocí pasážím textu získat zaslouženou pozornost. Pokud chcete vylepšit význam svého webu, pak je nejlepší k tomu využít strukturu kódu.

Vrstvy vaší struktury zahrnují následující složky:

- Obsah (textová data tvořící rozhraní)
- HTML (kód definující účel textu)
- Mikroformáty (názvy určující význam)

Populární metodou, jak vylepšit sémantickou a kontextuální hodnotu, jsou *účelové mikroformáty*. Poskytují řadu názvových konvencí, které použijete na element, abyste vyzdvihli jejich speciální význam, jež (řekněme) má konkrétní odstavec vzhledem k okolním elementům. Mikroformáty a tato „metadata“ můžete použít k odkázání na události, umístění a identity. Prohlížeče a sociální sítě si mohou ve skutečnosti tyto užitečné informace vytáhnout a něco s nimi provést (třeba vytvořit položku adresáře kontaktů). Obrázek 6.6 ukazuje fungování základních mikroformátů – mCard poskytující telefonní číslo.

```
8 <body>
9 <h1>Vítejte na webu </h1>
10 <p id="hCard"><span class="tel">
11 <span class="typ">domů</span>:
12 <span class="home tel">+420 555 444 333</span>
13 </span></p>
14 </body>
```

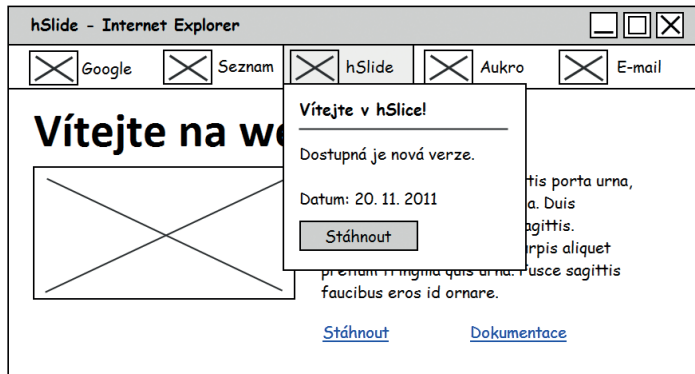


Obrázek 6.6: Mikroformáty přidáte k existujícímu kódu HTML, abyste uvedli, o jaký typ dat jde

V případě oborů jako jsou informační architektura a design je potřeba efektivně strukturovat vaše cenné prostředky (třeba obsahu) součástí bitvy, které designéři

čelí neustále. I když vlastnictví kódu, jenž jen stěží ovlivňuje vizuální stránku layoutu, může vzhledem k designu vypadat chudě, pamatujte ale, že prožitek stojí na maximalizaci efektivity kódu a vizuální podoby, takže mikroformáty byste měli vítat s otevřenou náručí (zvláště protože, když jsou podporované, návštěvníkům poskytují další funkcionalitu).

Skvělým příkladem podtrhujícím přínos mikroformátů je použití hSlice, proprietární funkce zabudované v prohlížečích společnosti Microsoft (Internet Explorer 8 a novějších verzích). Jen díky trošce chytrého kódu mohou návštěvníci ukládat a dynamicky vytažovat části obsahu (tzv. oblasti Web Slice), jež pak automaticky zobrazují aktuální obsah dané oblasti v malé vysouvací nabídce přímo v prohlížeči. Obsah oblasti se automaticky obnovuje, aniž by uživatel musel stránku, z níž oblast pochází, navštívit. To má osobitou hodnotu, protože výslednou akcí je to, že uživatel se zaměří jen na obsah dané oblasti stránky (jak ukazuje obrázek 6.7). Existují další skvělé mikroformáty, některé zajímavější, jiné méně. Jejich použití ve svých projektech každopádně zvažte. Mohou být cenným artiklem s jistou užžitnou hodnotou pro uživatele prohlížečů.



Obrázek 6.7: hSlice, mikroformát Microsoftu, umí vytvořit dynamickou a zajímavou záložku

Výhody těchto konvencí kódu překračují pouze vizuální odlišení poskytované samotným CSS. Mikroformáty mohou velmi úspěšně pomáhat paměti (hCard), vyznačovat licence (`rel="license"`), vzbudit dojem minimalismu (hSlice), a dokonce pomoci s úkoly, jako je optimalizace pro vyhledávače (`rel="nofollow"`). Je to jako nejenom říci, že odstavec je na stránce, ale také říci, že odstavec obsahuje určitý typ obsahu, jenž má vztah k něčemu, co lidé shledávají jako důležité a užitečné (a prohlížeče to mohou využít).

Zjistíte, že mikroformáty mají dobrou podporu. Ve vyhledávacích už začíná mít takový vliv. Sociální sítě a některé prohlížeče jsou na tom podobně (i když to není případ rovných příležitostí). Kromě toho, že mikroformáty přidávají význam obsahu vašich stránek, je zde místo také pro použití tradiční metadat jako těch, které najdete v záhlaví dokumentu nebo ve formátovaných souborech RDF (dokument jazyka XML modelující a popisující informace, aby vyjádřil význam objektů) s definovaným indexem.



TIP

Tak jako v případě jakéhokoli typu popisných dat, i v tomto případě je důležité, abyste svou pozici nezneužívali. Metaznačku pro klíčová slova začaly vyhledávače ignorovat kvůli jejímu zneužívání. Snažte se, aby se totéž nestalo také metadatům RDF nebo mikroformátům.

Kód hraje v designu mocnou roli, a ačkoliv funguje pod kapotou, mikroformáty jej nepřehlídí. Pokud vám dokážou poskytnout lepší pozici ve vyhledávacích, pak bude možné váš obsah snáze nalézt (v případě návštěvníků, kteří využijí vyhledávač). Jestliže některé z nich funkčně použijete v akci, pak mohou zvýšit produktivitu a docílit snazší komunikace s návštěvníky. Maximalizace vašeho HTML a stavu sémantiky posílí efektivitu vaší práce a to je to, oč tu běží. Osobití designéři nemohou takový silný infrastrukturní efekt opomenout.

Podprahový (neinvasivní) kód funguje pod povrchem a pokládá základy vašeho vizuálního úsilí. Pokud mezi sebou vyvážíte použití elementů a jim přiděleného stylu, pak omezíte ztráty (nutnost načítání dodatečného kódu) a vytvoříte elegantní a osobitý výsledek s rychlou odezvou. A tím můžete získat náskok před konkurencí a zajistíte, že stránky budou dobře fungovat a reagovat na úrovni prohlížeče.



ODKAZ

Více se o mikroformátech dozvíte ve specifikacích na adrese <http://microformats.org>. Více se o hSlide, proprietárním řešení společnosti Microsoft, dozvíte na adrese [http://msdn.microsoft.com/en-us/library/cc304073\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/cc304073(v=VS.85).aspx).

Kaskáda s elegancí

HTML poskytuje osobitosti základní úroveň, kterou nelze ignorovat, ale v případě většiny webových designérů se to hlavní ohledně estetiky a vizuálního layoutu

dodává pomocí CSS. Protože průměrný web bude na stránce obsahovat celou řadu umístěných objektů, potřebujete při tvorbě webových stránek rozumět selektorům, vlastnostem a hodnotám, které definují, jak se obsah na obrazovce vykreslí.

Mezi mnoha funkcemi, které CSS designérům nabízí, najdete vlastnosti určené k dodání trochy prázdného prostoru okolo objektů, k umísťování objektů na stránce a k zvýraznění oblastí okolo objektů. Použití je můžete také k upravení seznamů a tabulek, k dodání odlišného vzhledu textu, a dokonce i ke změně podoby kurzoru. Je to spousta zábavy. Obrázek 6.8 ukazuje několik věcí, které dokážete vytvořit pomocí CSS (ve vztahu k textu).

Můžete změnit typ písma

nebo jeho velikost

či dokonce jeho barvu pomocí stylů CSS

Obrázek 6.8: CSS můžete použít ke změně fontů, velikosti písma a jeho barvy

Ideální metoda, jak webu dodat tolik osobitosti, kolik je jen možné, spočívá v oddělení struktury webu (HTML) a stylů (CSS). Pro to existuje celá řada důvodů, ale faktem je, že pokud někdo chce web vidět bez vizuálních doplňků, pak si web může s trochou snahy takto prohlédnout. Ačkoliv vy budete chtít, aby uživatelé viděli váš obsah v nejelegantnějším a nejkreativnějším stavu, jako vždy platí, že výsledný prožitek kontroluje právě uživatel.

Pro designéry je CSS ekvivalentem švýcarského armádního nože. Všem těm, kdo vstoupí na váš web, poskytuje možnost na míru šitého prožitku (na rozdíl od HTML) bez trapného pocitu nezajímavosti. A i když nejde o perfektní jazyk, CSS poskytuje vysokou úroveň kontroly nad layoutem, barvami, typografií a vzhledem objektů zobrazených ve webovém prohlížeči (takže tento jazyk považujte za něco, co musíte mít).



TIP

Pamatujte na to, že i když můžete provést řadu změn ve vizuálním vzhledu obsahu, je důležité si každé vyladění obhájit v ohledu jeho vlivu na jiné objekty na stránce. Využitím principů a prvků designu si pomůžete, ale jako v jiných případech i zde platí, že je třeba se spolehnout hlavně na zdravý rozum a logiku.

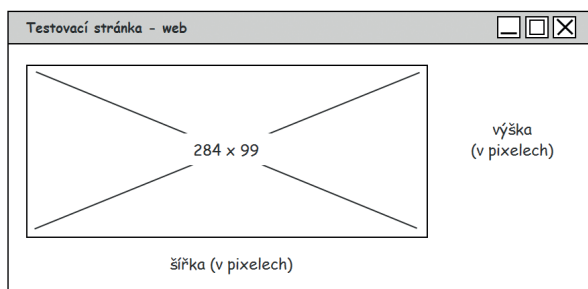
Podobně jako v případě HTML (jež se neustále vyvíjí) i jazyk CSS již ušel dlouhou cestu. CSS3, nový bezvadný standard, jenž je postupně implementován do

prohlížečů, přináší celou řadu dalších aspektů designu, které můžete kontrolovat. Jakmile bude tento standard široce zavedený, budete schopni vytvářet ještě složitější, elegantnější designy spoléhající se méně na obrázky k zajištění požadovaného vizuálního dojmu (k dosažení určitých efektů). Problémem je samozřejmě zprovoznění ve všech prohlížečích.

Jazyk CSS je skvělý zvláště díky své progresivní povaze. V případě jazyka HTML se jednotlivé prvky do specifikace přijímají a jiné z ní zase vylučují. Každá podoba CSS však přidává k těm předchozím další funkce. Takže se nic neztrácí, ale naopak se přidává. Protože jazyk CSS je založený na prezentaci, každá vlastnost (správně použitá) může vašemu webu přidat na osobitosti (takže je všechny musíte znát).

Správné stylování pomocí CSS

Jaký vliv mají všechny tyto vlastnosti na osobitost? Na to se podíváme nyní. Je důležité vědět, které nástroje máte k dispozici pro prezentaci obsahu. Možnost kontroly nad prezentací vašeho rozhraní je omezená (mimo stylistických efektů poskytnutých CSS), pokud ovšem nepoužíváte nástroje třetích stran, jak je třeba Adobe Flash (jenž umožňuje tvorbu vlastních objektů). Vzhledem k tomu, že úroveň podpory ze strany starších prohlížečů se liší, implementace nemusí být perfektní.



Obrázek 6.9: Vlastnosti `width` a `height` udávají absolutní rozměr prostoru, který bude objekt obývat

Prvními vlastnostmi, které stojí za pozornost, jsou ty, jež nastavují velikost nebo rozměry každého objektu na stránce. Protože velikost je důležitý faktor (chcete rozhodovat o tom, které objekty budou výraznější), schopnost upravit výšku a šířku elementů (viz obrázek 6.9) nesmíte podhodnotit. Vlastnosti `width` a `height` pro nastavení šířky a výšky lze doplnit užitečnými předponami `max-`

a `min-` (například `max-width`), jež mohou nastavit maximální nebo minimální možnou hodnotu šířky či výšky. Čím menší je objekt, tím méně je výrazný, takže se rozměrům a velikostem věnujte pozorně (pokud něco umístíte příliš blízko sebe, může dojít k přetečení a váš obsah by mohl být oříznut).

Pokud obsah přeteče hranice objektu proti vašemu záměru (po nastavení rozměrů objektu), pak máte k dispozici vlastnost `overflow`, pomocí níž přetečený obsah buď skryjete, nebo v elementu zobrazíte posuvník. (CSS3 nabízí také vlastnosti `overflow-x` a `overflow-y` pro jednotlivé osy. Tyto vlastnosti jsou také dobře podporované.) Pokud se chcete vrhnout do dobrodružství, pak vám vlastnost `clip` umožní vyříznutí díry v objektu, takže vše, co se vyskytuje za objektem, bude přes tuto díru viditelné.



TIP

Jestliže chcete vytvořit blok s obsahem, kterým se bude možné posunovat, je bezpečnější obsah do stránky vložit s vlastností `overflow`, zajišťující pevné zobrazení posuvníku po straně objektu, než se spoléhat na element `iframe` (jenž způsobuje řadu problémů s přístupností).

Nyní, když umíte definovat velikost objektu, je třeba zvážit, kolik prázdného prostoru ponechat kolem jeho vnějších okrajů (čímž objektu poskytnete prostor oddělený od dalších okolních objektů). Dvěma vlastnostmi, jež vám poskytnout požadovaný prázdný prostor, jsou `padding` a `margin`. Různý prázdný prostor můžete přiřadit každé straně objektu zvlášť (horní, spodní, levé a pravé). Vlastnost `padding` zajišťuje vnitřní okraj – tedy prostor mezi objektem a jeho rámečkem. Vlastnost `margin` zajišťuje vnější okraj – tedy prostor mezi rámečkem a ostatními objekty. Součet velikostí určených hodnotami vlastností `width`, `height`, `border`, `padding` a `margin` je roven celkové velikosti prostoru, který blok elementu na stránce zabere – tomu se říká *blokový model* CSS. Obrázek 6.10 ukazuje, jak celková podoba mezer ovlivňuje čitelnost.

Uzký
textový
obsah
se špatně
čte

Obsah s více prostorem
se čte lépe.

Obrázek 6.10: Rozestup mezi vaším obsahem (textem nebo jinými jeho částmi) může zajistit, aby se obsah na stránce snadněji četl

I když je prázdný prostor užitečný k definování relativních vztahů mezi objekty (efektivním zvýrazněním hran objektů), dokud kolem hran neuděláte čáru (ke

zvýraznění seskupení vašeho obsahu), uživatelé se mohou domnívat, že spolu nic nesouvisí. K dispozici máte několik voleb. Použít můžete vlastnost `border` (viz obrázek 6.11), určenou k přidávání rámečku (s možností určení barvy, stylu a síly čáry), vlastnost `outline` (která má podobný vizuální účinek, ale nemůžete definovat každou stranu zvlášť) nebo byste mohli třeba přidat nějaké zajímavé efekty CSS3, jako jsou vlastnosti `border-radius` (určující zaoblení rohů rámečku) nebo `box-shadow` (vytvářející stín za objektem).



Obrázek 6.11: Rámečky vám umožňují vizuálně seskupovat obsah a ukazují, jaký prostor objekt zabírá

Nyní se podívejte na to, jak můžete na stránce objekty umísťovat. Ve výrazném, osobitém designu je umísťování vším. Naštěstí pro designéry nabízí CSS vlastnost `position` (jež umožňuje určit pozici objektu podle okolí nebo pevné umístění na stránce, takže se objekt nehne ani při posunování stránkou dolů a nahoru.) Pokud vám některé z těchto možností nadchli, pak vlastnost `position` použijte s vlastnostmi `top`, `bottom`, `left` a `right`, které vám umožňují určit, kde se objekt na stránce zobrazí. Jinak ještě můžete použít vlastnost `float`, zajišťující plovoucí umístění objektů vedle sebe (toto plovoucí umístění pak ukončuje vlastnost `clear`).



DŮLEŽITÉ

Internet Explorer 6 nepodporuje pevné (fixní) umístění, což je škoda, protože tento typ umístění nabízí možnost nechat objekt zobrazený, i když se uživatel stránkou posune dolů. A i když existují jistá nouzová řešení, musíte s tímto problémem u Internet Exploreru 6 počítat.

Někdy můžete chtít svůj obsah překrýt nebo skrýt. Jestliže nabízáte postupné vysouvání, pak získáte řešení svého problému. Mimo skrývání textu pomocí záporného horního nebo levého vnějšího okraje (vlastnosti `margin`) máte k dispozici vlastnosti `visibility` (hodnota `hidden` skryje obsah objektu a ponechá jen prázdný prostor) a `display` (hodnota `none` objekt ze stránky zcela odstraní). K překrývání obsahu (a nastavení umístění) můžete použít vlastnost `z-index`, na-

stavující prioritu v pořadí zobrazení obsahu, když pomocí ní přidělíte číselnou hodnotu objektu v pořadí vrstev (funguje to opravdu skvěle).



Překrývání obsahu může být skvělé, ale pamatujte si, že pokud se váš obsah ve výsledku stane nečitelným, pak můžete zdržet návštěvníka v jeho postu stránkou (a omezit efektivitu osobitosti vašeho obsahu). Podobně jako v případě blízkosti musíte mezi sebou vzájemně vyvážit efekt a jeho použitelnost.

Dosud jste se naučili, jak CSS dokáže změnit rozměry, prioritu, umístění, viditelnost, a dokonce jak se vypořádat s překrýváním obsahu. Pokud na stránce máte seznamy (uspořádané nebo neuspořádané), pak jim můžete definovat vlastní typ odrážky (obrázek 6.12) nebo obrázek a ten pak umístit dle své chuti (v závislosti na typu seznamu, který použijete). Ačkoliv tato kniha nepokrývá všechny detaily, snadno můžete změnit také vzhled tabulek a jejich rámečků nebo přidat více mezer buňkám, což působivému vzhledu také pomůže.

- První odrážka
- Druhá odrážka
- ✓ Třetí odrážka
- 1. Čtvrtá odrážka
- l. Pátá odrážka

Obrázek 6.12: Použijte vlastní odrážky (vytvořením obrázku), abyste určitému typu seznamu dodali ten správný vzhled

Nyní je čas na prozkoumání dvou částí CSS, jež mají největší vliv na to, jak na sebe vzájemně působí návštěvníci a obsah webu (nikoliv odstraněním čehokoliv ze své pozice, což hraje významnou roli při uspořádávání vašeho vizuálního pořádku a hierarchie). Následujícím dvěma vlastnostem, jež jsou obtěžkané již známými a v designu velmi důležitými koncepty barev a typografie, jsem věnoval celou část knihy.

Nejdříve se věnujme barvě. Vy už víte, že barvy mají vliv na lidské emoce a jsou silným psychologickým nástrojem. CSS umožňuje nejenom nastavit barvu popředí nebo pozadí objektu, ale umožňuje také nastavit vlastní obrázek na pozadí a umístit jej tak, aby objektu dodal správný efekt nebo texturu. CSS3 umožňuje také použití alfa průhlednosti a neprůhlednosti (zastínění). Jen musíte zajistit správný kontrast a dobrý výsledek finálního efektu (viz obrázek 6.13).

Barva v textu je skvělá!

Obrázek 6.13: Vyberte barvu a ujistěte se, že dobře kontrastuje se stránkou (použitím silných nebo jemných tónů)

Pokud chcete textu dodat zajímavý vzhled, musíte se dále pečlivě věnovat typografii. Pro začátek nastavte fonty (pomocí vlastnosti `font-family` k určení knihovny písem bezpečné pro web – více v kapitole 4) a použijte populární typ písma, jako je Georgia (viz obrázek 6.14), Trebuchet MS, Arial nebo Verdana. Pak se vrhněte na vizuální šarm pomocí vlastností `font-size` (nastavující velikost textu), `font-style` (nastavující kurzivu), `font-weight` (nastavující tučnost) a `font-variant` (umožňující vytvořit velká písmena stejné velikosti jako malá – kapitálky).

Líbí se vám Georgia?

Obrázek 6.14: Výběr správného typu písma pro váš design vyžaduje určité zvažování ohledně použití písma na webu

Dále máte k dispozici vlastnosti určující textové styly, jako jsou třeba vlastnosti `text-align` (která vám umožňuje měnit zarovnání textu doleva, doprava, na střed nebo do bloku, čímž bude text zarovnaný k okrajům elementu po obou stranách), `text-indent` (k odsazení nebo přesazení textu), `text-decoration` (umožňující podtržení, přeškrtnutí nebo umístění čáry nad text) a `text-transform` (díky níž můžete měnit malá písmena na velká nebo nechat první písmeno slova zobrazit velké). Některé příklady použití těchto vlastností ukazuje obrázek 6.15.

```
font-weight: bold; font-style: italic;  
text-align: left; text-align: center; text-align: right;  
text-decoration: underline;  
text-decoration: line-through;  
text-decoration: overline;
```

Obrázek 6.15: Text můžete stylovat a uspořádávat mnoha způsoby, aby byl zajímavý

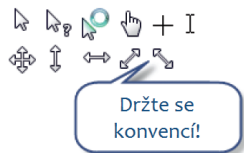
A pokud vám to nestačí, pak máte k dispozici také vlastnosti, jako jsou `letter-spacing` (určující mezery mezi písmeny), `word-spacing` (určující mezery mezi slovy), `line-height` (určující velikost mezery mezi řádky), `vertical-align` (podobně jako `text-align` zarovnáva text, ale po svislé ose, tedy nahoru, na účarí, na střed či dolů) a `white-space` (dokáže kontrolovat zalomení řádků).

A ještě několik dalších vlastností také ovlivňuje vizuální podobu textu (a mnohem více jich najdete v CSS3, pokud chcete využít nejmodernější technologie), takže neváhejte zkoušet, co funguje.



Prázdný prostor je důležitý, zvláště v případě textu. Ačkoliv neexistuje konkrétní postup, jak zajistit nejlepší čitelnost textu, vezměte v potaz zmíněné faktory a hrajte si s hodnotami vlastností, abyste našli stav, s nímž budete (vy i vaši návštěvníci) spojeni. Návštěvníci si přece chtějí čas strávený na webu užít!

Kvůli jejich jedinečnosti stojí za zmínku ještě několik dalších vlastností. První z nich je schopnost CSS vyladit uživatelské rozhraní prohlížeče (třeba aby byl obsah více sladěný s vizuálním stylem). Změna barev *posuvníku* (proprietární část CSS, jež funguje jen v některých prohlížečích) nebo podoby *kurzoru* (viz obrázek 6.16) se může tvářit jako dobrá myšlenka, ale její realizace může některé návštěvníky zmást.



Obrázek 6.16: Úprava podoby kurzoru může být zábavná, ale uživatelé budou složitěji sledovat svou pozici na stránce

Jedna z vlastností je ale výjimkou z tohoto pravidla: vlastní podoba *výběru* textu. Pokud ji provedete správně, může vylepšit uživatelskou zkušenost a čitelnost těm, kteří vyberou obsah tažením kurzorem (čímž svou pozici označí). Jakmile změníte barvu výběru, musíte zajistit, že vybraný obsah zůstane viditelný a že bude dostatečně kontrastní, aby uživatelé věděli, že výběr provedli správně.

Pomocí vlastnosti `content` můžete do stránky dodatečně vložit vlastní obsah (ke stylistickým účelům). Ale radím vám, abyste tuto vlastnost nepoužívali pro obsah, který je důležitý pro celkový kontext stránky (v případě, že bude CSS vypnuto, nebo při použití čtečky obrazovky by se takový obsah nezobrazil). V základě platí, že použití vlastnosti `content` vám poskytuje efektivní způsob, jak popsat objekty na stránce, které jsou oddělené od ostatního obsahu.

A nakonec vyžaduje trochu pozornosti také vlastnost `font-face` v případě CSS3, neboť umožňuje použití vlastní typografie (pokud máte k dispozici licencovaný

typ písma). Pokud dokážete najít legální typ písma, jenž odráží náladu vašeho designu lépe než běžné webové fonty, pak tuto šanci využijte. Podobně jako řada dalších aspektů CSS3 je i vlastnost `font-face` v prohlížečích podporovaná různě a jen těžko se brzy objeví nějaký standard fungující ve všech prohlížečích.



ODKAZ

Pokud se chcete dozvědět více o tom, jak tato funkcionalita této vlastnosti funguje, pak navštivte adresu <http://meiert.com/en/indices/css-properties/> nebo si kupte knihu *HTML5 a CSS3 Výukový kurz webového vývoje* (vydal Computer Press).

Mobilní, tiskový nebo výchozí design

Množství voleb v rámci CSS 1, 2 a 3 může být pro začátečníka ohromující. Nicméně možnost volby je dobrá věc a stojí za to uvážit všechny volby dostupné pro každý objekt na stránce, abyste určili jeho optimální úroveň osobitosti (nebo neutrality). Pokud už s kódem máte zkušenosti, ale příliš se nevěnujete designu, zvažte otestování několika myšlenek (třeba formou zpětné vazby) a poté svoji myšlenku, testy a náčrty umístěte do něčeho reálného.

Výběr médií vám umožňuje cílit na specifické typy zařízení (jako jsou tiskárny nebo projekční zařízení) se styly orientovanými na jejich potřeby. V minulosti se argumentovalo, že pro web je nejlepší mít jedinou šablonu stylů nebo jediný vizuální layout pro všechna zařízení. Ale tento argument neobstojí. Designéři si dnes ovědomují, že jedna velikost nemůže být vhodná pro všechna zařízení. Zobrazení webu na obrazovce, pro tisk, na mobilních zařízeních, na televizi a dalších zařízeních by se mělo lišit podle potřeb toho kterého zařízení, uživatele a očekávání, jak by měl styl fungovat. (Obrázek 6.17 jde o krok dále ukázáním dotazu na médium, jenž stojí nad selektorem, a stylu založeného na funkci, jako je velikost okna.)

```
@media screen and (min-width: 321px) and (max-width: 750px) {}
```

zobrazení

minimální šířka

minimální výška

Obrázek 6.17: Dotaz na médium (media queries) v CSS3 vám umožňuje zacílit určité proměnné pracovního prostředí

CSS vám umožňuje vzít všechny vaše koncepty a myšlenky ohledně layoutu a promítnout je do reality. Dobře promyšlený design může vypadat skvěle na obrazovce stolního nebo přenosného počítače, ale pokud stejný design zobrazíte na chytrém telefonu s malou dotykovou obrazovkou, výsledek nemusí pracovat tak, jak má. Přenosná zařízení jsou oblíbená, takže s nimi musíte počítat.



TIP

Někteří designéři, kteří tvrdohlavě trvají na používání pevné šířky designu a používají šířky podle zařízení, přenesli stejnou techniku také na mobilní platformy. Nastavení pevné šířky může fungovat například na iPhone, ale už ne na iPadu. Jak to tak vypadá, není již pevná šířka designu dobrým řešením. Plovoucí layouty jsou lepší.

Pokus zaměřit se na všechna dostupná zařízení je neskutečný úkol. V případě CSS je dotaz na přenosná (handheld) zařízení ještě stále bezvýznamný kvůli jeho podpoře ze strany výrobců zařízení. Co se týká vašich voleb, pak můžete mobilní uživatele nechat používat existující verzi pro stolní počítače (což není dobrá volba), využít dotaz na média v rámci CSS3 a jako hodnotu zvolit typ handheld, abyste pomohli chytrějším zařízením upravit uživatelské rozhraní stránky pro mobilní zařízení, nebo můžete pro uživatele mobilních zařízení vytvořit samostatný web (provedením některých optimalizací kvůli jejich spotřebě). Pokud máte čas, pak projděte všechny tři typy – počínajíc poslední volbou až se propracujete k té první.



DŮLEŽITÉ

Většina designérů se domnívá, že rozlišení je jediným faktorem ovlivňujícím prostor, s nímž musí pracovat, ale faktem je, že většina uživatelů nepoužívá maximalizované okno se stejným DPI a v celoobrazovkovém režimu, a navíc zobrazené panely nástrojů a postranní panely omezují celkovou velikost zobrazovacího prostoru.

Pamatujte si, že můžete nabídnout alternativní šablony stylů pro každou situaci. Jestliže jste dobrodruzi, nabídněte různá témata, aby si návštěvníci mohli vybrat to, které nejlépe odpovídá jejich osobnímu vkusu. Přidání dalších voleb je časově náročné, ale můžete tak skutečně udeřit hřebíček na hlavičku v ohledu personalizace a povzbuzení uživatelů k interakci. S alternativními šablonami stylů můžete nabídnout také verzi s vysokým kontrastem kvůli zvýšení přístupnosti.

Řešení v podobě skriptů

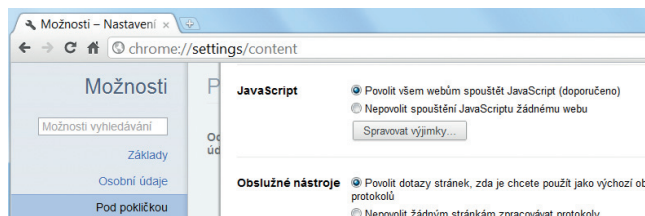
CSS je skvělým jazykem využitelným k vizuálnímu odlišení stránek, ale není to jazyk, který dodá interaktivitu. Neposkytuje prožitek, který uživatelé očekávají (za hranicí tradičního statického webu). Internet ve své geniálnosti designérům nabízí jazyk, který zdaleka převáží CSS v ohledu celkové složitosti, ale za to nabízí ideální platformu poskytující návštěvníkům požadovanou interaktivitu. Tímto jazykem je JavaScript, a jde skutečně o velmi mocný nástroj.

Zaprve a především platí, že ačkoliv JavaScript není jedinou metodou, jak uživatelům poskytnout interakci a chování, je jedním z několika, kteří pracují na straně klienta (v prohlížeči uživatele místo zpracovávání skriptů na serveru). A co víc, je to ten typ jazyka, jenž lidem umožňuje reagovat stisknutím kláves, kliknutím myši a klepnutím na dotykovou obrazovku. JavaScript je proto pro designéry ideální.

**TIP**

Skriptovací jazyky pracující na straně klienta, jako je JavaScript, vám umožňují interagovat se vstupním zařízením návštěvníka, ale nezapomínejte na to, že jazyky pracující na straně serveru vám dovolují pamatovat a zpracovat tyto akce do něčeho produktivního a přizpůsobeného. Jde o obousměrný způsob.

Pokud jste už s JavaScriptem pracovali, pak víte, že jeho hlavní nevýhodou jsou situace, kdy lidé podporu JavaScriptu vypnou (viz obrázek 6.18), aby se vyhnuli rušivému chování některých špatně implementovaných skriptů (například vysoce intruzivním skriptům brojícím proti kliknutí pravým tlačítkem). Úkolem designéra není obvykle pouze ujistění se, že web funguje i při vypnutí podpory skriptů (takže skripty používejte, ale mějte nouzové řešení), ale také vyhnutí se újmě na prožitku uživatelů nebo schopnosti přistupovat na web a používat jej.



Obrázek 6.18: Podporu skriptů lze vypnout různými způsoby, takže nepředpokládejte, že váš web je v bezpečí

I když styly plodí osobitost pomocí vizuální stimulace, řešení chování (pomocí skriptů) nabízí poutavější prožitek tak, že uživatelům umožní na webu provádět více činností než jen čtení. Tato úroveň interakce lidem pomáhá zapamatovat si jejich zkušenost s webem, protože akce spouští vizuální paměť mozku (tedy nejen ty části užívané ke čtení). Tyto interakce jim také poskytují způsob, jak se vyhnout nudnosti, tím, že jim představí zajímavé aktivity a nástroje.

JavaScript umí víc než jen poskytnout interaktivní prvky reagující na vstup uživatele nebo jeho aktivitu. Reakce skriptů mohou také vést ke změně kódu CSS nebo HTML či k výměně obsahu za jiný. Toto dynamické chování vám umožňuje skrze volby uživatelů proměnit celkovou zkušenost s webem a připravit tak přizpůsobitelné, flexibilní, interaktivní a na uživatele orientované rozhraní.

Čím skripty přispívají k osobitosti webu

Síla JavaScriptu je nepopíratelná. A i když měl také své porodní bolesti (jako v dobách špatně kopírovaných skriptů), jazyk jako takový vám dovoluje vytvořit téměř jakoukoli aplikaci nebo funkcionalitu. Jestliže vaši uživatelé požadují určitou pomoc nebo funkci (třeba průvodce výrobky), můžete je implementovat nevtíravým způsobem (stačí k tomu trocha vědomostí a zkušeností a případně trocha pomoci od frameworku, jakým je třeba knihovna jQuery), a to protože jedinou překážkou je vaše představivost.

Pro osobité designéry může být volba mezi dynamickým a statickým webem těžká. Na jedné straně chcete, aby se uživatelé cítili jako součást prožitku. Na druhé straně přidaná složitost vyvinuté funkcionality může bránit v rozvoji webu, pokud se nezvládne správně. I když tato kniha se nevěnuje celé řadě věcí, které lze vytvořit skriptováním (jde koneckonců o designovou knihu), platí, že JavaScript nabízí několik triků poskytujících designu trochu interaktivity (jež lidi zaujme).



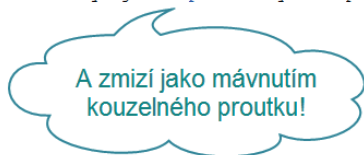
Interaktivita má osobitější hodnotu než něco animovaného, jako je třeba video. Když uživatel vykoná nějakou akci spíše, než aby se jen pohyboval po stránce, mozek rozezná reakci na událost.

První nástrojem JavaScriptu vylepšujícím práci na designu je DOM (*Document Object Model* – česky *objektový model dokumentu*). Ten vám umožňuje zacílit určité části struktury stránky (a následně je třeba stylovat) vyfiltrováním hierar-

chie dokumentu a odkázáním na objekt. Na každý objekt na stránce lze pomocí tohoto modelu odkazovat (odkazem může být název elementu, třída nebo identifikátor), takže na zacílený objekt můžete použít chování.

Použitím směsice funkcí a příkazů deklarujících „když se něco stane, pak proveďte tyto konkrétní úkoly“ může interagovat s uživatelem – počkáte na určitou událost a pak na ni budete reagovat. Je skvělé mít možnost zacílit konkrétní elementy a události jako `onClick` (kliknutí myši), ale mít možnost těmto objektům změnit styly vám nabízí mnohem zajímavější volby. Představte si, že budete schopni například na vyžádání změnit fonty. Velmi syrový příklad předvádí obrázek 6.19, ale stejně tak byste mohli přidat elementu odkaz na třídu, abyste zkrátili dlouhé skripty měnící styly.

```
document.getElementsByTagName("p")[1].style.display:none;
```



Obrázek 6.19: JavaScript dokáže změnit CSS na požádání a umí produkovat skvělou interaktivní estetiku

Schopnost změnit vzhled objektů téměř nekonečným množstvím způsobů nabízí samozřejmě širokou řadu možností, a to nejenom v přidávání nebo odebrání zdůraznění objektům, jak je potřeba. Poskytnutí informací uživatelům (nebo jejich vyžádání), jakmile je potřebují (postupné odkrytím), je něco, čeho můžete s JavaScriptem snadno dosáhnout. Mimo přidávání nebo odebrání obsahu na stránce na vyžádání dokáže JavaScript použít sofistikovanější mechanismus, jako je Ajax. Ten umožňuje nastavovat nebo sbírat data bez nutnosti opakovaného načítání stránky nebo jakýchkoli jiných změn.

JavaScript umožňuje vylepšovat uživatelská rozhraní se skvělými výsledky. Je jasné, že interaktivita a web spolu fungují bezvadně. A pokud budete chtít uživatelům nabídnout něco víc než jen základní prožitek ve formě čtení, pak po přemýšlejte o potenciálních výhodách JavaScriptu ve vašem osobitém designu.

Frameworky ve vizuálním designu

JavaScript si zaslouží obdiv za poskytnutí bohaté funkcionality vylepšující dynamiku vašeho webu. Nicméně je nutné také zmínit *frameworky* postavené na Ja-

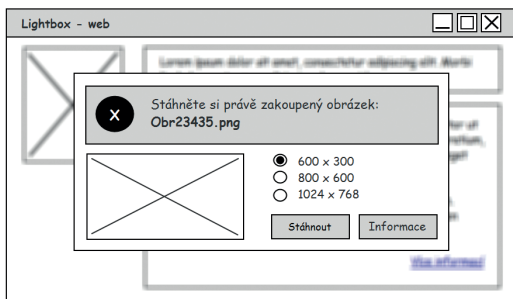
vaScriptu. Ty zjednodušují proces vylepšování interaktivity a funkčních animací (někdy nahrazující Flash) a inspirují designéry k nabídnutí skvělých a unikátních věcí na jejich webech.

Z mnoha dostupných frameworků se žádný neuchytil tak dobře jako knihovna jQuery. Řada argumentů tvrdí, že používání frameworků není vůbec nutné. Ale i když je pravda, že k jejich používání budete potřebovat určité základní znalosti JavaScriptu, jejich přímočarost a optimalizovaná přirozenost z nich dělá cenné kandidáty pro jejich potenciální využití na webu, zvláště pokud spěcháte.



Framework je knihovna předpřipravených kódů vykonávajících určité funkce podle požadavků vývojáře. Zbaví vás tvrdé práce na vývoji takových nástrojů a umožní vám jejich zprovoznění pouhým vložením kódu a zavoláním funkce, díky čemuž jsou pro nováčky zdaleka nejrychlejším řešením začlenění i sofistikovaných efektů JavaScriptu do jejich designů.

Schopnost jednoduše funkcionalitu připojit k vašemu webu může usnadnit vylepšení uživatelského rozhraní. V posledních letech lidé začali interaktivitu na webu očekávat, a protože takto dynamické chování dokáže vylepšit rozhraní, interaktivita nabízí nové úrovně osobitosti ke zkoumání. Jen uživatele nezahľte přetíženým obsahem nebo vtíravými animovanými efekty jen kvůli zaujetí pozornosti.



Obrázek 6.20: Lightboxy (vysvícení) strhávají pozornost na konkrétní objekty na stránce

U webu se neočekává totéž co u tisku. Lidé s webem interagují kontrolovaným způsobem (používají myš nebo klávesnici). Z tohoto chování můžete těžit a ovlivnit své uživatele. Dříve jsem se věnoval psychologii podnětu a odezvy, a vy jste se naučili, že podmiňování uživatelů určitými událostmi (jako je třeba kliknutí

na obrázek spouštějící lightbox nebo dialog vyžadující pozornost – viz obrázek 6.20) vám dovoluje posílit očekávání toho, co se může stát v budoucnosti.

Při tom všem je JavaScript cenným nástrojem k prozkoumání. Dovoluje vám sesbírat data o vašich uživateli pomocí analytiky (abyste viděli, jak web procházejí a jak jejich prožitek zlepšit), interakce s uživateli (a jejich psychologického ovlivnění) a tvorby konkrétních úchvatných efektů, které postupně zobrazují informace nebo stránky (a jež doprovází animované efekty, jako je třeba změna stylu určená k zachycení pozornosti uživatele). I když JavaScript není tak vizuálně zaměřený jako CSS, jeho skripty nabízejí množství věcí, jimiž můžete sami prezentovat, kontrolovat a zaujímat uživatele. To je skvělá zpráva pro osobité designéry. Takže tyto možnosti prozkoumejte!

Minulost, přítomnost a budoucnost

Schopnost tvořit ještě krásnější a osobitější webové stránky byla pozitivně ovlivněna (z části) vývojem jazyků, jako jsou HTML, CSS a JavaScript. Ale jak asi víte, věci nejsou v takové harmonii, jak by se mohlo zdát. S množstvím prohlížečů a řadou možností stylování nebo chování spočívající v rukou uživatelů je třeba se zaměřit na stránky bezchybné, kompatibilní a použitelné pro návštěvníky.

Dříve žili weboví uživatelé uprostřed války prohlížečů mezi společnostmi Microsoft a Netscape, které změnilo prostředí svými proprietárními doplňky. Některé výsledky tohoto vývoje (například AJAX) dnes stále existují a jsou široce přijímané. Jiné neuspěly a v některých případech po sobě zanechaly problémy s nekonzistentním vykreslováním, s nimiž se musí weboví designéři vypořádat. Vaše designy závisejí na schopnosti prohlížečů zobrazit váš obsah, takže pro ně musíte svoji práci ladit.

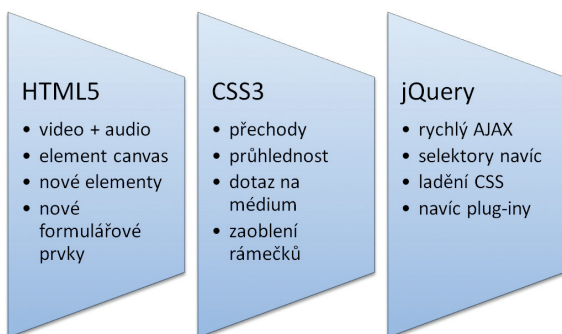
**DŮLEŽITÉ**

Nekonzistentní vykreslování by mělo být menším problémem pro vývoje, jakmile výrobci prohlížečů zajistí, že jejich nástroje budou odpovídat specifikacím. Jakmile bude vykreslování stejné, pak se budete muset postarat pouze o přijetí nově vytvářených standardů.

Současný stav webu je samozřejmě v případě desktopových prohlížečů mnohem lepší než v předchozích letech. V případě starých verzí Internet Exploreru byly

časté problémy s divným vykreslováním, a tak se zavedla technika podmíněných komentářů k jejich napravení. Nyní je třeba řešit problém s vykreslením na kapesních zařízeních, kde zatím neplatí žádné široce přijímané standardy. Další stadium designu se točí kolem konceptů flexibility a použitelnosti, takže se ujistěte, že jsou vaše designy osobité a odpovídají těmto potřebám.

I přes současné problémy a staré války vypadá budoucnost webdesignu růžově. Problémem zůstává kompatibilita, protože zařízení a nástroje používané lidmi se neustále mění a vyvíjejí. Navíc si řekněme na rovinu, že přemluvit uživatele, aby se vzdali starých verzí Internet Exploreru, není snadné. Posunout se vpřed a vytvořit osobitý design již nebude založeno na tom, co můžete vytvořit s omezenými nástroji, ale na vaší důmyslnosti a kreativitě. Zvažte, jak v současnosti mění HTML5, CSS3 a jQuery vizuální stránku věci, a pochopíte, jak mocným se Internet stává (viz obrázek 6.21).



Obrázek 6.21: Zde je několik věcí, které přinesly jazyky HTML5, CSS3 a jQuery

Se všemi nekonečnými možnostmi rolování a dokumenty, jež mohou být stejně tak interaktivní jako počítačové hry, není zapůsobení na uživatele nijak jednoduché. Z této nekonečné řady možností si můžete odnést jednu radu, a to že složitost může poskytnout nové úrovně osobitosti, ale nemusí znamenat ten největší přínos pro koncového uživatele. Jakmile skončíte se zkoumáním všech dostupných nástrojů pro kódování, pamatujte si, že je třeba také pečlivě zvážit a zdrženlivě použít vše, co poskytují v designu. Čára mezi inovací a zneužitím je velmi tenká.

Docela postupné vylepšení

Tři techniky zajistí, že váš obsah bude fungovat, i když nebudou určité technologie na straně návštěvníka dostupné nebo budou vypnuté. Některé fungují lépe než

jiné, takže je důležité prozkoumat metody ovlivňující kompatibilitu (negativně a pozitivně) a pochopit, které metody jsou preferované.

Jednou z metod vedoucích k dosažení osobitého designu je princip *zpětné kompatibility* (základní postup znázorňuje obrázek 6.22). Web vytváříte pro uživatele, kteří budou těžit z plných možností uživatelského rozhraní (díky moderním standardům), a poté stránky doladíte pro starší zařízení a požadované prohlížeče. To je nejproblematictější metoda, protože starší nebo méně známé nástroje je možné snadno opomenout, což má vliv na potenciální (pravděpodobně nepoznané) uživatele, které chcete ohromit dech beroucím rozhraním.



Obrázek 6.22: Zpětná kompatibility používá složité postupy

Běžným problémem kompatibility zůstává oprava chyb prohlížečů a její provedení je stále silnou zbraní v arzenálu designéra. Flexibilní design, jenž není závislý na nutnosti prohlížet jej určitým způsobem nebo po provedení určitého úkolu (aby vše fungovalo na základní úrovni), ponechává vaše návštěvníky v následující situaci: pokud nesplňují vaše standardy, buď z webu odejdou, nebo provedou změnu vybavení. Jak ukazuje zkušenost, opuštění webu je snadnější volbou.



TIP

Návštěvníci jen málokdy opustí váš web, pokud jsou problémy s vykreslením malé (někteří návštěvníci budou tvrdohlaví, což pomáhá), ale slabé uživatelské rozhraní by je mohlo odradit od návratu nebo delší zastávky. Při navrhování to mějte na mysli.

Směřujete-li při práci k ideálu a opravy v případech, kdy se něco pokazí, nejsou považovány za dobré řešení, pak co? Jako osobití designéři musíte vědět, že to, co nabííte, funguje v případě co nejširšího publika. Zničený design může nesprávně vykreslit obsah nebo se v něm nebude možné správně orientovat a to by narušilo jeho hodnotu. Právě proto je koncept *drobné degradace* zároveň s *postupným vylepšením* tak oblíbený.

Drobná degradace znamená, že pokud použijete pravděpodobně nedostupnou technologii, je třeba za ni poskytnout nějaké použitelné náhradní řešení (i když nebude tak pěkné). Postupné vylepšení na druhou stranu znamená, že při tvorbě webu každé vylepšení vrstvíte na jinak fungující webové stránky. Koncepty fun-

gují společně, protože vrstvení funkcí znamená, že když vylepšení není možné použít, degraduje se na něco s méně, ale stabilními funkcemi. Metodologie testování při práci s vrstvením znázorňuje obrázek 6.23.



Obrázek 6.23: Postupné vylepšení s drobnou degradací – jde o osobitou stabilitu v akci

Základním příkladem drobné degradace s postupným vylepšením v akci je vrstvení vaší HTML struktury nad obsah (takže jej odlišíte) následované umístěním mikroformátů na HTML (čímž obsahu přidáte kontext). Dále nad kód HTML přidáte CSS (čímž poskytnete osobitý vizuální styl) a následně umístíte vrstvu CSS3, abyste poskytli nějaké ty vizuální efekty navíc, u nichž dojde k degradaci, jestliže je nebude možné provést. A nakonec přidáte nějaký ten JavaScript (čímž dodáte osobitou funkcionalitu).

Jakmile máte v designu všechny vrstvy na svém místě, máte design bezpečnější pro zobrazení ve starších prohlížečích, protože se v případě nouze propadnou o vrstvu níže. Vrstvení také snižuje množství nutného ladění. Provést musíte pouze zvláštní ladění, na rozdíl od záplatování pro celé prohlížeče. Osobitý design vyžaduje, abyste věděli, jak takové věci vyřešit.

Nyní zvažte, jak byste mohli začít ladění nebo testování vašeho webu k vyladění vizuálních triků. Nechcete, aby si lidé mysleli, že webový prohlížeč zpackal váš obsah. A protože já předpokládám, že kódu rozumíte dostatečně, abyste vytvořili osobitý web, neposkytnu vám výklad do hloubky probírající proces ladění. Nicméně jedné metodě, která funguje zaručeně, se zde věnovat budu. Tuto metodu ladění v jejím jádře zobrazuje obrázek 6.24.

```
@media screen {
  body {
    background: #e0e0e0;
    color: #767676;
    font-family: Candara, Tahoma, Geneva, sans-serif;
  }
  h2 {
    font-style: italic;
    font-weight: normal;
    margin-top: 0.5em;
    padding-bottom: 0.5em;
  }
}
```

← problém nalezen

Obrázek 6.24: Rozděl a panuj: odstraňujte kód, dokud nenajdete závadný materiál

Ladění je zdlouhavý úkol, zvláště když musíte procházet stovku řádků kódu. Jakmile najdete něco, co nevypadá správně v určitém prohlížeči, nejrychlejším způsobem, jak problém vyřešit, je vrátit se k základům. Vyjměte veškerý kód v šabloně stylů nebo skript a začněte jej postupně přidávat po kouscích zpátky, dokud nedosáhnete bodu, kdy nastane chyba. V takové chvíli budete hned vědět, co chybu způsobuje. A pak můžete najít řešení, otestovat jiný kód nebo opravit hloupou chybu, až problémy vyřešíte.

Dostat se do stavu, kdy je veškerý kód v pořádku, může být náročné, ale ten čas stojí za to, pokud máte návštěvníkům zajistit bezchybný prožitek. Velmi se doporučuje, abyste web nejdříve otestovali v oblíbených prohlížečích (Internet Explorer, Firefox, Chrome, Safari a Opera) a do testu zahrnuli také všechny verze prohlížečů, které návštěvníci vašeho webu používají. V případě Internet Exploreru (kde toto pravidlo platí nejlépe) zahrnete verze 6, 7, 8 a 9. Pokud budete mít štěstí, podaří se vám je všechny na váš počítač nainstalovat.

**DŮLEŽITÉ**

Abyste zajistili spolehlivost, měli byste web testovat na současných zařízeních (ve výchozí konfiguraci podle výrobce). Pokud to ale není možné, dalším nejlepším postupem je použít virtualizovaná zařízení (identická) a nakonec využít emulátory (a dosáhnout replikace).

Mimo výjimky v případě více verzí Internet Exploreru můžete instalovat poslední dostupné verze prohlížečů. V případě Internet Exploreru můžete stáhnout a použít volně dostupné nástroje IETester a Microsoft Virtual PC, jež vám umožní prohlédnout si váš web v požadované verzi prohlížeče. Jakmile váš web vypadá ve všech těchto prohlížečích dobře, zvažte testování webu na mobilní platformě, pokud to je možné. Reálná zařízení jsou pro testování nejlepší, ale pomoci vám mohou také různé emulátory telefonů iPhone, Android, Nokia a vlastních prohlížečů.

Klíč k tomu, abyste si mohli svůj web prohlédnout v různých prohlížečích, je v použití vykreslovacího jádra (které vykresluje obsah na obrazovku) spíše než v náhodném výběru prohlížečů. Počítat se všemi prostředími, v nichž se bude web zobrazovat, je složité, ale jak už zmínily předchozí kapitoly, způsob konzumace vámi poskytnutého obsahu určují uživatelé. Vaší prací jakožto designéra je zajistit, aby z něj uživatelé měli co nejvíce.



TIP

Typy mobilních značek lze emulovat, včetně platform Android, Apple, Blackberry, dotMobi, LG, Mobile Firefox, Microsoft (Windows Mobile), Motorola, NetFront, Nokia, OpenWave, Opera (mini nebo Mobile), Palm (Pre a další), Samsung a SkyFire.

Osobitost v budoucích standardech

Web napomáhá ve vývoji vašich komunikačních technik mnoha zajímavými způsoby, ale problémy s kompatibilitou zde zůstanou i v následujících letech. Protože nikdo neví, jaké nové nástroje se objeví v dalších 50 letech, jste nuceni pracovat konzervativně a myslet na budoucnost, abyste se vyhnuli vkládání odpadu z porušeného webu. To nejlepší, v co můžete doufat, je hladký a bezproblémový přechod.

S vývojem počítačů a sociální oblasti s ní spojené je práce s Internetem stále více frustrující. S tím, jak se vzájemně kříží kulturní očekávání, se uživatelé musejí vypořádávat s různorodými významy v zobrazení webových stránek. Přespříliš složité weby a služby uživatele nutí přizpůsobit své chování požadavkům uživatelského rozhraní. Protože lidé s počítači interagují stále více způsoby, je třeba, aby komunikace člověka se strojem byla co nejvíce plynulá a přirozená, aby se vyhnula kognitivním nebo behaviorálním bariérám. Web lidstvu poskytuje základní služby a vy jste zavázáni být profesionály v každém přístupu a implementaci.



DŮLEŽITÉ

Osobitý design je o dobré komunikaci a rozvinutí dobrého vztahu s návštěvníkem. Dokud návštěvníka nepovzbudíte k interakci nebo komunikaci s vámi poskytnutým obsahem, nemáte k dispozici základy k vybudování vzájemné důvěry, empatie a smysluplného, hodnotného prožitku. Ujistěte se proto, že jsou vaše interakce smysluplné.

Zvažte, jak se po čase vizuály znehodnocují, a to nejenom z důvodu evoluce v oblasti prohlížečů, ale také kvůli procesu stárnutí. Tento proces zadržují aktualizace webu, ale designéři také musí neustále přizpůsobovat rozhraní osobitému chování uživatelů a zároveň jej udržet autonomní. I přes vědomí o proměnlivosti a specifických potřebách uživatelů musí designéři rozhraní navrhovat tak, aby fungovala v nejlepším zájmu uživatelů, čímž uživatelé zůstávají u moci. Obrázek 6.25 ilustruje, jak čas hraje důležitou roli ve vnímání (a posuzování) moderního webu.

Poslední aktualizace: 20. 10. 2011

Obrázek 6.25: Lidé jsou fascinováni a ovlivněni časem. Zajistěte, aby web nebyl zastaralý.

Jazyky, jako jsou HTML5 a CSS3, jsou dnes na vzestupu, a další jazyky se neustále vyvíjejí, aby změnily způsob, jímž lidé používají Internet. Vezměte si například přechody a animace CSS3, kterých jste dříve dosáhli pouze pomocí technologie Adobe Flash – přidáním vrstev animace a vylepšení můžete návštěvníka zaujmout. Grafiku vykreslující jazyky, jako jsou SVG nebo ML, vám umožňují využít na webu vektorovou grafiku (jakmile budou jejich možnosti podporovat prohlížeče).

Shrnutí

Internet je ohniskem nových aktivit prováděných lidmi, jakými jsou třeba vývojáři prohlížečů, kteří chtějí Internet návštěvníkům obohatit (což je pro váš web skvělá zpráva). Příklady těchto novinek můžete vidět například v předponách u nových vlastností CSS3 (ještě nedokončené), jež vám umožňují testovat je ve vašem designu. Tím máte možnost tyto nové nástroje už dnes využít na svém webu. Tyto nové možnosti až neuvěřitelně překonávají dosavadní možnosti prohlížečů.

Osobitý design a nástroje k implementaci krásných, nových a fascinujících funkcí ve vašem uživatelském rozhraní již existují. Ačkoliv musíte kódovat opatrně, pokud u kódování přemýšlíte a provádíte validaci, testování a vylepšování své práce, pak můžete vašim návštěvníkům z webu udělat lepší místo. Také tím poskytnete částem vašeho layoutu skvělou příležitost, jak vyniknout. Kód diktuje vizuální atraktivitu a její implementaci. Budete jej muset ovládnout, pokud chcete dosáhnout požadovaných výsledků; a vždy si pamatujte, že váš zdrojový kód – každá jeho část – je důležitý.