

 **WORDPRESS.ORG**
vodní stránka Podpora Blog Šablony



WordPress

od základů
k profesionálnímu použití

Michal Bureš, Adam Laita, Marek Laurenčík

- Úvod do redakčního systému WordPress, jeho instalace a obsluha
- Tvorbav webových stránek a příspěvků
- Práce s textem, obrázky, mediálními soubory a soubory PDF
- Nastavení vzhledu webu, tvorba menu a práce se systémem uživatelů
- Práce se šablonami, pluginy a widgety
- Využití vlastního kódu, vlastního formátování a vlastních datových tabulek
- Vytvoření vlastní šablony





WordPress

od základů
k profesionálnímu použití

Michal Bureš, Adam Laita, Marek Laurenčík

Upozornění pro čtenáře a uživatele této knihy

Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě bez předchozího písemného souhlasu nakladatele. Neoprávněné užití této knihy bude **trestně stíháno**.

Michal Bureš, Adam Laita, Marek Laurenčík

WordPress

od základů k profesionálnímu použití

Vydala Grada Publishing, a.s.
U Průhonu 22, Praha 7
obchod@grada.cz, www.grada.cz
tel.: +420 234 264 401
jako svou 8194. publikaci

Odpovědný redaktor Petr Somogyi
Sazba Daniel Boček
Ikony Freepik (www.flaticon.com)
Počet stran 264
První vydání, Praha 2021
Vytiskla TISKÁRNA V RÁJI, s.r.o., Pardubice

© Grada Publishing, a.s., 2021
Cover Design © Grada Publishing, a. s., 2021
Cover Photo © Depositphotos/SergeyNivens

Názvy produktů, firem apod. použité v knize mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

ISBN 978-80-271-4462-4 (ePub)
ISBN 978-80-271-4461-7 (pdf)
ISBN 978-80-271-0178-8 (print)

Obsah

	Předmluva.....	9
1	Úvod do tvorby webových stránek.....	11
	1.1 Fungování webu.....	11
	1.2 Hosting a doména.....	13
	1.3 Programovací jazyky a databáze.....	14
	1.4 Protokoly a názvy souborů.....	15
	1.5 Zásady pro tvorbu webu.....	17
	1.6 Prostředky pro tvorbu webu.....	18
	1.7 Základní charakteristika WordPressu.....	19
2	Instalace.....	21
	2.1 Lokální server na vlastním zařízení.....	21
	2.2 Výběr a objednání hostingu a domény.....	26
	2.3 Automatická instalace WordPressu.....	26
	2.4 Ruční instalace WordPressu.....	27
3	Základní nastavení.....	33
	3.1 Přihlášení.....	33
	3.2 Přehled funkcí.....	35
	3.3 Základní konfigurace a nastavení.....	37
	3.4 Multisite.....	42
4	Práce s obsahem.....	47
	4.1 Stránky a příspěvky.....	47
	4.2 Práce se stránkami.....	48
	4.3 Práce s příspěvkem.....	52
	4.4 Rubriky a štítky.....	55
	4.5 Komentáře.....	57
	4.6 Knihovna médií, vkládání obrázků a jiných souborů.....	60
5	Obsahový editor Gutenberg.....	69
	5.1 Představení a historie editoru.....	69
	5.2 Práce v blocích.....	70

5.3	Práce s textem	73
5.3.1	Odstavce a formátování textu	73
5.3.2	Tvorba nadpisů	77
5.3.3	Tvorba seznamů	78
5.3.4	Hypertextové odkazy	80
5.4	Skupina TEXT	82
5.4.1	Tvorba tabulek	82
5.4.2	Další bloky ve skupině TEXT	85
5.5	Skupina MEDIA	87
5.6	Skupina VZHLED	92
5.7	Skupina WIDGETS	97
5.8	Skupina EXTERNÍ OBSAH	102
5.9	Tvorba a použití vlastních bloků	102
6	Šablony	107
6.1	Základní informace, výchozí šablony	107
6.2	Typy šablon	107
6.3	Vyhledání a instalace šablony	108
6.4	Přizpůsobení vzhledu	111
6.5	Odvozená šablona (child theme)	113
6.6	Vlastní šablona	114
7	Pluginy	115
7.1	Princip pluginu	115
7.2	Výběr a instalace pluginů	115
7.3	Plugin pro tvorbu e-shopu	118
7.4	Několik zajímavých pluginů	119
7.5	Speciální typy pluginů (Drop-in, Must-Use)	121
7.6	Údržba pluginů	122
8	Ostatní funkce	125
8.1	Widgety	125
8.2	Menu	126
8.3	Editor	129
8.4	Uživatelé	130
8.5	Nástroje	134
9	Provoz	137
9.1	Aktualizace	137
9.2	Bezpečnost	138

9.3	Zálohování.....	139
9.4	Výkon.....	139
9.5	SEO.....	141
10	Úvod do tvorby kódu a práce s databázemi.....	143
10.1	Základní příkazy jazyka HTML.....	143
10.2	Úvod do tvorby kaskádových stylů CSS.....	147
10.3	Základní příkazy jazyka PHP.....	153
10.4	Úvod do databází.....	166
10.5	Základní příkazy jazyka SQL.....	175
10.5.1	Jednoduchý příkaz SELECT.....	176
10.5.2	Filtrace záznamů.....	177
10.5.3	Výpočty a standardní funkce.....	180
10.5.4	Dotazy z více tabulek.....	183
10.5.5	Tvorba souhrnů.....	185
10.5.6	Poddotazy.....	187
10.5.7	Tvorba pohledů.....	188
10.5.8	Akcční dotazy.....	188
11	Použití vlastního kódu.....	191
11.1	Přímé použití kódu HTML ve WordPressu.....	191
11.2	Použití vlastních stylů CSS.....	196
11.3	Použití vlastního kódu PHP v rámci WordPressu.....	202
11.4	Výběry z vlastních tabulek v databázi.....	209
11.5	Úprava vlastních tabulek v databázi.....	212
11.6	Práce s příspěvky a stránkami.....	215
12	Tvorba vlastní šablony.....	221
12.1	Princip tvorby vlastní šablony.....	221
12.1.1	Úvodní kroky při tvorbě vlastní šablony.....	221
12.1.2	Ukázka zobrazení obsahu stránek.....	227
12.2	Struktura šablony.....	229
12.3	Vysvětlení pojmů CPT, CF a shortkód.....	231
12.4	Příklad vlastní startovní šablony.....	236
12.4.1	Soubor functions.php.....	236
12.4.2	Složka assets.....	241
12.4.3	Složka css.....	242
12.4.4	Složka js.....	242
12.4.5	Soubor header.php.....	242
12.4.6	Soubor footer.php.....	243
12.4.7	Soubor index.php.....	244
12.4.8	Složka components.....	246
12.4.9	Soubor single.php.....	247
12.4.10	Soubory page.php, page-sidebar.php, 404.php a style.css.....	249

12.4.11	Soubor sidebar.php.....	250
12.4.12	Složka languages	250
12.4.13	Nové překlady.....	251
12.4.14	Přenos pot souboru.....	252
12.4.15	Screenshot a favicon.....	255
12.4.16	Šablona s CPT, taxonomií a shortkódy.....	255
12.5	WP-CLI aneb pokročilá správa pomocí příkazového řádku.....	259
Rejstřík.....		261
O autorech		264



Předmluva

Tato kniha je věnována tvorbě webových stránek pomocí redakčního systému WordPress. Tento systém je základem velkého počtu webů, které jsou v současné době na internetu provozovány. Důvody jsou především dva: Prvním je skutečnost, že WordPress představuje vyvážený kompromis mezi přístupem pro začátečníky a pokročilé – umožňuje jak tvorbu jednoduchého webu bez jakýchkoliv znalostí webového programování, tak i komplexního webového systému, v němž může vývojář uplatnit vlastní programovací postupy. Druhým důvodem je, že se pomocí WordPressu mohou vytvářet jak klasické weby, tak i internetové časopisy, různá fóra apod., jež umožňují aktivní účast většího množství uživatelů na tvorbě webu.

Příruček, které by se věnovaly tvorbě webu pomocí redakčního systému WordPress, je na českém trhu málo; většina informací je roztroušena po různých webových stránkách. A právě tuto mezeru se snaží zaplnit kniha, kterou držíte v rukou.

Při přípravě knihy jsme se snažili pojmut její obsah tak, aby byla užitečná jak pro čtenáře, kteří s WordPressem teprve začínají, tak i pro pokročilejší vývojáře, kteří by chtěli doplnit své stránky vlastním kódem a rozšířit tak výrazně jejich možnosti. Odpovídá tomu i rozdělení látky do kapitol.

První tři kapitoly představují úvod do tvorby webových stránek a základní informace o systému WordPress. Naleznete v nich také návod na instalaci WordPressu a jeho nastavení. Tvorbě webových stránek uživatelským způsobem jsou věnovány kapitoly 4 a 5. Tyto kapitoly důsledně vycházejí ze současného pojetí WordPressu, založeného na jednotné koncepci bloků a použití editoru Gutenberg.

Základem tvorby webu ve WordPressu je předem připravená šablona, jejíž možnosti jsou zpravidla doplňovány přidavnými moduly (pluginy), které jsou dostupné na internetu. Tomu se věnují kapitoly 6 a 7. Vývoj webu se ovšem neomezuje jen na vytváření vlastního obsahu, zahrnuje i tvorbu menu a další možnosti. Pozornost je také třeba věnovat jeho rychlému a bezpečnému provozu. Tato problematika je popsána v kapitolách 8 a 9.

Poslední dvě kapitoly knihy (11 a 12) jsou věnovány využití vlastního kódu ve WordPressu. Zahrnují přímou tvorbu obsahu pomocí kódování HTML, úpravu vzhledu s využitím CSS stylů a využití vlastního programovacího kódu v jazyku PHP. Najdete v nich informace o tvorbě vlastní šablony i o tvorbě vlastních programových modulů. Aby bylo toto téma co nejpřístupnější, v kapitole 10 je nejprve podán přehled základních elementů jazyka HTML, základní informace o tvorbě stylů CSS a úvod do jazyka PHP. Tato kapitola obsahuje i úvodní informace o databázi



MySQL, která se při tvorbě webových stránek používá nejčastěji, včetně základních prvků jazyka SQL, využívaného při komunikaci webových stránek s tabulkami v databázi.

Při psaní této knihy jsme vycházeli z poslední verze WordPressu (5.6) a ze standardní šablony *Twenty Twenty-One*, která se v současné době používá jako výchozí.

V knize jsou použity následující typografické prvky:

- Názvy příkazů v nabídkách, tlačítek, jednotlivých voleb v oknech, souborů a složek jsou zvýrazněny **tučným písmem**.
- Volby v dialogových oknech, kde se předpokládá rozhodování uživatele, nebo zadávané parametry jsou vyznačeny **tučnou kurzívou**.
- Názvy pluginů, šablon a webové adresy jsou zvýrazněny *kurzívou*.
- Příkazy a kódy programů jsou odlišeny písmem `courier`.
- Klávesové zkratky a klávesy jsou zvýrazněny KAPITÁLKAMI.

Ať se Vám daří!

Autoři



Úvod do tvorby webových stránek

O významu webových stránek není jistě třeba ztrácet slov. Vlastní webové stránky jsou dnes samozřejmou prezentací každé firmy nebo instituce. Web dnes už dávno nefunguje jen jako jednoduché informační médium, ale umožňuje různé způsoby komunikace návštěvníka webu s jeho provozovatelem. Typickými příklady jsou různé objednávky, přihlášky, internetový obchod, knihovnictví, internetové bankovníctví, různá diskuzní fóra, sociální sítě atd. Klade to samozřejmě stále větší nároky nejen na tvůrce, ale i zadavatele webu. Web, který není pravidelně aktualizován, ztrácí postupně smysl.

Těžko je ovšem možné požadovat například od nebohého živnostníka, drceného státní byrokracií, aby se ještě začal učit programovat. Od devadesátých let minulého století je proto patrná snaha mít k dispozici prostředky, které by umožnily vytvářet (nebo aspoň upravovat a udržovat) web i pracovníkům, kteří programování webových stránek neovládají. Velice perspektivní a účinnou cestou se ukázal vývoj tzv. **redakčních systémů**. Patrně nejpoužívanějším z nich je v současné době systém WordPress, kterému je věnována tato kniha.

1.1 Fungování webu

Celosvětová počítačová síť, označovaná také zkratkou „www“ (z anglického World Wide Web) byla původně vyvinuta v USA pro výměnu informací mezi univerzitami a dalšími vědeckými pracovišti. Později se tohoto systému chopily (jak jinak!) státní instituce a armáda. Komerční využití internetu, které dnes zcela převažuje, přišlo až o něco později.

Internet byl od počátku vyvíjen jako prostředek zcela nezávislý na počítačovém vybavení na straně uživatele (tedy osoby, která s některým webem pracuje). Vedlo to k tomu, že internetové dokumenty, označované také jako stránky, mohly obsahovat pouze text. Prakticky každý web však obsahuje i grafické informace, které se textem vyjádřit nedají. Pionýři webu tento problém vyřešili tak, že všechny obrázky, fotografie a jiné grafické objekty byly uloženy v samostatných souborech a internetový dokument se na tyto soubory odvolával pomocí odkazů.

Princip klient – server

Je jasné, že fungování internetu musí být zajištěno na dvou stranách: na straně provozovatele webu, který informace poskytuje, a na straně návštěvníka webu, který tyto informace přijímá.

Tento princip se označuje jako systém „klient – server“.¹ Obdobně je možné rozdělit jednotlivé propojené počítače, jež tvoří celosvětovou internetovou síť, na dva základní druhy:

- **servery**, které uchovávají a nabízejí dokumenty s informacemi,
- **klienty**, které tyto dokumenty přejímají, zobrazují a nabízí uživatelům.

Na straně klienta musí být k dispozici program, který zajistí správné zobrazení internetového dokumentu. Tyto programy se označují jako **webové prohlížeče**. Jedním z nich je program **Microsoft Internet Explorer**, který je součástí operačního systému Windows. Dalšími často používanými prohlížeči jsou například **Firefox**, **Google Chrome**, **Edge**, **Opera** apod. Tyto programy lze bezplatně nainstalovat z řady webů.

Práce webového prohlížeče

Internetový dokument (stránka) však nemůže zahrnovat jen vlastní obsah, ale také řídicí příkazy pro práci prohlížeče. Ty se vytvářejí v jazyku **HTML** (Hypertext Markup Language). Takto vytvořený dokument má příponu **htm** nebo **html**.

Každý soubor HTML musí rovněž obsahovat informace, jak se má zobrazovaný text naformátovat. To je možné realizovat dvěma způsoby:

- Údaje o vzhledu písma, odstavce apod. zapsat přímo v souboru pomocí příkazů jazyka HTML.
- Oddělit od sebe vlastní obsah a informace o jeho formátování. Formátovací informace se zapíše pomocí nástroje zvaného „styly“ a v jazyku HTML se budeme na vytvořené styly odvolávat.

Druhý způsob je samozřejmě výhodnější, a proto je současným trendem při tvorbě webu přesunout co nejvíce formátovacích informací právě do stylů.²

Je nutné zdůraznit, že internetové prohlížeče neumí nic jiného než zpracovat obsah vytvořený pomocí jazyka HTML, popřípadě využít pro naformátování zobrazovaného obsahu pravidla uložená v zaznamenaných stylech. Pro přípravu webu je možné použít různé způsoby, ale ze serveru se do klientského počítače nakonec vždy odešle jen kód v jazyku HTML.



Většina prohlížečů umožňuje zpracovávaný kód HTML zobrazit. Do plochy prohlížeče klepnete pravým tlačítkem myši a z místní nabídky použijete položku **Zobrazit zdrojový kód**.

Soubory tvořící webové stránky

Pod pojmem „web“ se rozumí systém řady souborů, z nichž část může být ještě umístěna v podadresářích. Jedná se zejména o tyto soubory:

- Soubor s názvem **index**. Jedná se o úvodní stránku celého webu, která se při připojení zobrazí každému návštěvníkovi jako první. Soubor **index** musí být umístěn v kořenovém adresáři publikovaného webu.
- Další soubory tvořící webové stránky. Mohou sem patřit soubory typu HTML nebo soubory vytvořené v programovacím jazyku PHP (viz dále).
- Obrázky, fotografie a další grafické soubory.

¹ Tento princip se samozřejmě uplatňoval v praktickém životě již dávno před příchodem počítačů. Návštěvník hospody funguje již od středověku jako klient, hostinský jako server.

² Princip použití stylů při přípravě webu je podobný způsobu, jak se využívají styly v textových editorech, například ve Wordu.

- Soubory ve formátu PDF, které lze zobrazit pomocí programu **Acrobat Reader**. Soubory PDF se mohou na webu prohlížet, tisknout nebo stáhnout do návštěvníkova počítače.
- Soubory typu CSS obsahují definice použitých stylů. Tyto soubory nemají vlastní obsah, ale zajišťují jednotný vzhled celého webu.
- Podle potřeby mohou být na webu umístěny i dokumenty Wordu, sešity Excelu nebo jiné soubory, které jsou zpravidla určeny ke stažení ze serveru do počítače na straně klienta.

1.2 Hosting a doména

Pro provoz vlastního webu si musí budoucí provozovatel zajistit dvě věci:

- Zaregistrovat si vlastní **doménu**, což je identifikace vytvářeného webu. Označení domény musí být v rámci celosvětové počítačové sítě jedinečné (obdobně jako adresy elektronické pošty). Registrace nové domény se provádí přes internet, kde se ověří, zda navrhovaný název domény je nebo není již někým používán.
- Zajistit si přístup na některý z počítačových serverů, kde se webové stránky provozují. Tento přístup se nazývá **hosting** nebo **webhosting**. V podstatě se jedná o pronájem diskové plochy na některém ze serverů zapojených do internetové sítě.

Celosvětový systém domén tvoří strom, v němž se rozlišují tři úrovně. Odpovídá tomu i název domény, který se skládá ze tří částí oddělených tečkami. Jednotlivé části názvu (směrem odzadu) odpovídají jednotlivým úrovním v systému domén:

- Poslední část názvu (doména prvního řádu) je často označení státu, kde je web provozován: „cz“ znamená Českou republiku, „sk“ Slovensko, „de“ Německo atd. Kromě toho ještě existují mezinárodní domény prvního řádu označující typ provozovatele: „com“ jsou komerční organizace, „org“ neziskové organizace a „gov“ instituce státní správy.
- Předposlední část názvu se označuje jako doména druhého řádu a je vlastní identifikací provozovatele webu.
- První část názvu (doména třetího řádu) může mít dvojitý charakter. Často se používají písmena „www“, nebo je první částí názvu individuální rozlišení domény třetího řádu v rámci domény řádu druhého.

Při registraci domény je možné vytvořit novou doménu druhého nebo třetího řádu, odvozenou od domény druhého řádu. Příkladem domény druhého řádu je *www.seznam.cz*, příkladem domény třetího řádu je *mojedomena.kvalitne.cz*.

Hosting je zpravidla současně spojen s nabídkou některých služeb potřebných pro provozování webu. Jedná se zejména o podporu programovacího jazyka PHP, přístup k databázi a u řady hostingů také o možnost instalace některého redakčního systému pro tvorbu webu, například WordPressu.

Registrace domény a objednávka hostingu jsou na sobě nezávislé, ale řada provozovatelů webového serveru (tento provozovatel se často označuje jako **provider**) nabízí možnost provést obě akce najednou. Provozování domény a hostingu může být bezplatné, nebo provozované za mírný poplatek (řádově stovky korun za rok). Bezplatný hosting obvykle nabízí menší plochu na disku a na vytvořeném webu se zobrazují reklamy, což u placeného hostingu není. Z toho plyne, že pro zkušební účely postačí bezplatný hosting, zatímco pro provoz „ostrého“ webu se vyplatí použít hosting placený.

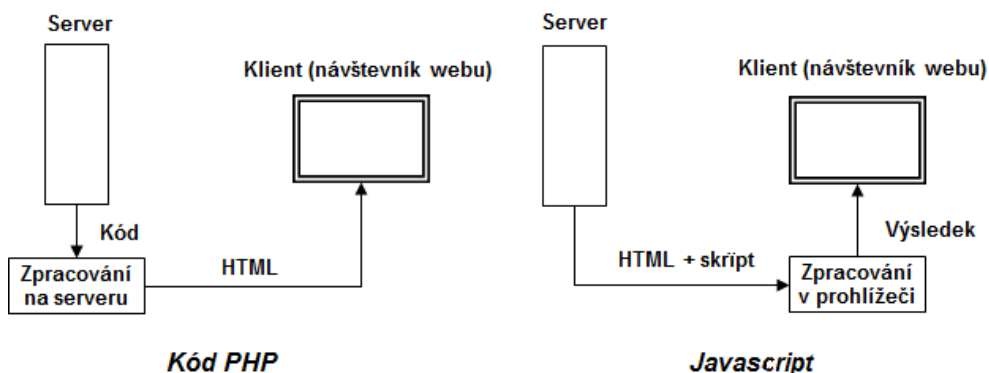
Objedávka hostingu se provádí přes internetové stránky providera. Zájemce má zpravidla možnost výběru z několika možností lišících se rozsahem služeb a cenou, popřípadě si může zvolit bezplatný hosting. Po vyplnění potřebných údajů a potvrzení objednávky zájemce ob-

drží e-mail s potřebnými informacemi, především přihlašovacím jménem a heslem (heslo si je většinou možné zvolit při objednávce). Přihlašovací údaje jsou nezbytné pro publikování vytvořeného webu, tedy pro přepokopírování potřebných souborů na server.

1.3 Programovací jazyky a databáze

Programovací jazyk HTML prochází neustálým vývojem, nicméně má své limity. Všechny požadavky, které se dnes na weby kladou, dnes již není tento jazyk schopen zabezpečit. Z tohoto důvodu se k jazyku HTML postupně přidaly další programovací prostředky. Jsou to především dva jazyky:

- Jazyk **Javascript** provádí kód na straně klienta. Jedná se o tzv. klientský skript, což znamená, že prováděné příkazy tohoto jazyka jsou přímo součástí kódu HTML. Příkazy se provedou v klientském počítači při zobrazování stránky. Umožňuje to oživit webové stránky dynamickými efekty.
- Jazyk **PHP** je skriptovací jazyk, který funguje na straně serveru. Kód se tedy provede dříve, než se stránka odešle k zobrazení na klientském počítači. Tento programovací jazyk má zásadní důležitost mimo jiné právě při práci s redakčními systémy typu WordPress, protože celý redakční systém je vytvořen právě v tomto programovacím jazyku.



Obrázek 1.1 Schéma práce Javascriptu a jazyka PHP

Využití databáze

Pod pojmem **relační databáze** se míní programový systém určený pro efektivní uložení a zpracování velkého množství dat. Údaje uložené v databázi se mohou snadno filtrovat, třídít a sloužit jako základ pro provádění různých výpočtů. Mezi nejčastěji používané databázové systémy patří například **MS Access**, **MS SQL Server** nebo **Oracle**. Některé databázové možnosti nabízejí i poslední verze Excelu.

K tvorbě webových stránek se využívá databázový systém MySQL, v poslední době pak i systém MariaDB. Práce s oběma systémy je prakticky stejná. Jedna ze služeb poskytovaných při objednání hostingu je automatické vytvoření databáze, ve které je možné uchovávat potřebná data ve formě tabulek. Tabulky se při tvorbě webových stránek nejčastěji využívají dvěma způsoby:

- V databázi jsou uloženy údaje, které se na webu prezentují v přehledné podobě jako seznam nebo tabulka. Jedná se zejména o nabídky, ceníky, seznamy termínů atd.

- V databázových tabulkách je uložen vlastní obsah jednotlivých webových stránek a také struktura menu celého webu. Na webovém serveru pak se pak nemusí umístit řada souborů typu HTML, ale po zvolení některé nabídky se její obsah zobrazí v klientském počítači návštěvníkovi webu.

Tato spolupráce webu s databází přináší dvě podstatné výhody:

- Web je výrazně úspornější a vyžaduje méně diskové plochy na serveru.
- Při úpravě prezentovaných údajů (například ceníku nebo nabídky) není třeba měnit kód odpovídajícího webového dokumentu, ale postačí upravit potřebnou databázovou tabulku.

Přesun valné většiny textových informací do databázových tabulek představuje v současné době dominantní způsob tvorby webových stránek. Tato koncepce je také základem redakčních systémů (včetně WordPressu).

Jazyk PHP

Jak již bylo uvedeno, kód v jazyku PHP se provede dříve, než se obsah webové stránky odešle směrem ke klientovi. V praxi to znamená, že se na webovém serveru pomocí kódu PHP vytvoří část nebo celý obsah webové stránky, tento výsledek se stáhne do klientského počítače a poté se zobrazí návštěvníkovi webu v prohlížeči. Protože se kód provede na straně serveru a na webové stránce se zobrazí až jeho výsledek, provádění kódu není nijak závislé na používaném prohlížeči.

Soubory, které obsahují kód v tomto programovacím jazyku, mají příponu *php* a je třeba je na webovém serveru také uložit. Využití jazyka PHP poskytuje při tvorbě webu řadu možností:

- *Komunikace s databází My SQL:* Pomocí prostředků jazyka se vytvoří spojení mezi webem a databází. Přes příkazy jazyka PHP je pak možné údaje uložené v databázových tabulkách získat, převést do kódu HTML a odeslat do klientského počítače.
- *Odesílání e-mailových zpráv:* Pomocí kódu PHP je možné automaticky odeslat návštěvníkovi webu zprávy s různým obsahem.
- *Práce se soubory:* Jazyk PHP obsahuje příkazy pro stažení určeného souboru z klientského počítače na server. Umožňuje také tvorbu nových souborů. To je možné využít například pro export dat z webu do souboru typu XML, určeného pro další zpracování.
- *Vyhodnocení údajů zadaných návštěvníkem webu:* Většina webů obsahuje jeden nebo více klientských formulářů, které jsou určeny pro zadávání údajů (objednávky, platební příkazy, dotazníky apod.). Pomocí příkazů v jazyku PHP je možné tyto údaje vyhodnotit, zapsat do tabulky v databázi, nebo je odeslat provozovateli webu pomocí e-mailu.

1.4 Protokoly a názvy souborů

Aby se obsah webu zobrazil v prohlížeči správně, je třeba před název domény přidat ještě informaci o tzv. **protokolu**, což je způsob, jak prohlížeč zobrazí obsah webového serveru. Kompletní adresa webového serveru tedy zahrnuje protokol, následovaný dvojtečkou a dvěma lomítky, a název domény, například: *https://www.gradac.cz*. Komunikace s webovým serverem používá dva protokoly:

- Protokol *http* nebo *https*: v prohlížeči se zobrazí obsah webu, se soubory na webovém serveru není možné manipulovat.
- Protokol *ftp*: tento protokol neumožňuje soubory na webovém serveru prohlížet, umožňuje však s nimi pracovat obdobným způsobem jako programy **Průzkumník** nebo **Tento počítač**, tedy lze soubory kopírovat, mazat, vytvářet nové složky apod.

V adresním řádku (najdete ho u horního okraje okna prohlížeče) je první protokol považován za výchozí. Pokud do tohoto řádku zapíšete jen samotný název domény, automaticky se zobrazí obsah požadovaného webu. V některých případech (například při tvorbě hypertextových odkazů v kódu HTML) je však nutné použít kompletní adresu včetně protokolu.

Názvy souborů

Webové servery využívají operační systém Unix, který ve srovnání s operačním systémem Windows má odlišná pravidla pro názvy souborů:

- nepřipouští se názvy s českou diakritikou, mezerami, čárkami a podobnými znaky,
- v názvech se **rozlišují** malá a velká písmena,
- při zápisu cest k souborům se používá obyčejné (nikoliv zpětné) lomítko.

Názvy všech souborů, které hodláte kopírovat na webový server, je proto zapotřebí upravit tak, aby respektovaly tyto zásady. Názvy by proto měly obsahovat pouze:

- malá písmena abecedy bez písmen s háčky a čárkami,
- číslice,
- pomlčku a podtržítko.

Redakční systém WordPress je schopen sám upravit názvy souborů tak, aby odpovídaly požadavkům webového serveru. Je však rozumné uvedené zásady důsledně dodržovat i při práci s WordPressem. Jejich nedodržení může vést k tomu, že se například špatně pojmenovaný obrázek nebude na webu zobrazovat.

Kopírování souborů na webový server

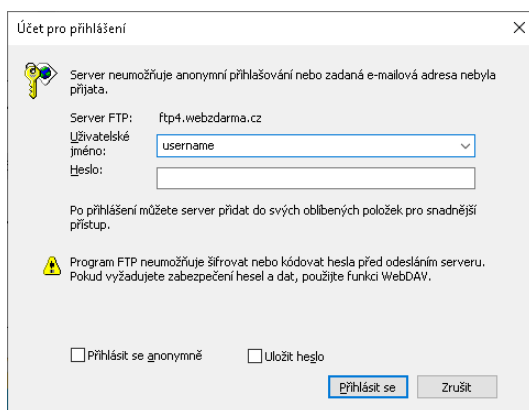
Kopírování potřebných souborů na webový server provedete pomocí internetu, spuštěného s využitím protokolu FTP. Pro přístup na webový server, kde je umístěn vytvářený web, je zapotřebí znát internetovou adresu webového serveru, přihlašovací jméno a heslo. Tyto informace sděluje provozovatel webového serveru po objednání webhostingu, zpravidla formou e-mailu.

Při práci se soubory na webovém serveru postupujete takto:

1. Spustíte program **Tento počítač**.
2. Do adresního řádku, umístěného v okně pod hlavní nabídkou, zapíšete adresu webového serveru včetně protokolu (například *ftp://www.grada.cz*).
3. V dalším zobrazeném okně zadáte přihlašovací jméno a heslo.
4. Klepnete na tlačítko **Přihlásit se**.

V okně programu **Tento počítač** se prostor přidělený na webovém serveru zobrazí jako samostatný disk. Pro práci se soubory a složkami použijete stejné postupy jako při práci s běžnými lokálními nebo síťovými disky.

Pohodlnější práci s protokolem FTP nabízí některé programy určené přímo pro práci se soubory, například program **Total Commander**. Z internetu si můžeme také bezplatně stáhnout programy určené přímo pro práci s protokolem FTP, například program **FileZilla**.



Obrázek 1.2 Přihlášení se protokolem FTP

1.5 Zásady pro tvorbu webu

Web každé instituce by měl být vytvářen s vědomím, že je vizitkou provozovatele a pro většinu lidí pravděpodobně představuje první kontakt s institucí, která web provozuje. Dobře vytvořený web by proto měl splňovat některé zásady:

- *Logičnost*: Struktura menu by měla být logická a srozumitelná.
- *Přehlednost*: Návštěvník nebude ochoten strávit půl hodiny tím, aby hledal, kde vlastně firma sídlí. Web by měl splňovat zásadu: „Čím častěji hledaná informace, tím snazší cesta.“ K přehlednosti webu také nepřispívá, pokud jsou klíčové informace utopeny v záplavě drobných detailů.
- *Střízlivost*: Důležitý je obsah webu, vizuální forma je jen prostředek. Při použití grafických efektů nepochybně platí stará dobrá zásada, že méně často znamená více.
- *Čitelnost*: Tenké černé písmo na duhovém nebo tmavomodrém pozadí je sice efektní, ale návštěvníkovi příliš nepomůže.
- *Stručnost*: Řada návštěvníků webu nebude ochotna číst na obrazovce příliš dlouhé texty. Pokud je potřebujete návštěvníkovi předat, je často lepší umístit na web jen stručný souhrn spolu s odkazem na soubor PDF, který bude obsahovat plný text.
- *Nezávislost na počítačovém vybavení návštěvníka*: Web musí fungovat v různých prohlížečích, tabletech a mobilech.
- *Odolnost vůči chybám*: Tento požadavek se týká především formulářů, do kterých bude návštěvník zadávat své údaje. Počítejte s tím, že řada návštěvníků bude při jeho vyplňování dělat chyby a není možné, aby se web i při malé chybičce zhroutil nebo zamrzl.
- *Dobrá vyhledatelnost*: Pro vyhledávací portály jako Google, Seznam apod. je důležitou informací nadpis jednotlivých stránek, proto je nutné věnovat jeho formulaci patřičnou pozornost.
- *Vstřícnost vůči uživateli*: Pokud web obsahuje ceník zboží či služeb, návštěvník jistě ocení možnosti si tento ceník stáhnout jako soubor. U adresy firmy je dobré přidat mapku (ne každý přijede autem s navigací), u kontaktů nezapomeňte na telefonní číslo (řada lidí nechce komunikovat pouze elektronicky a dává přednost osobnímu styku).

Část těchto zásad je platná i pro tvorbu dobré prezentace v PowerPointu. Pokud máte zkušenosti s PowerPointem nebo jiným prezentačním programem, dobře je uplatníte i při tvorbě webu.

Tvorba webu má několik fází:

- *Počáteční rozvaha*: Sem patří stanovení hlavní cílové skupiny, k níž se bude web obracet, a stanovení jeho rozsahu. Uvažte, zda bude pro požadované účely stačit web s několika stránkami, nebo potřebujete vytvořit rozsáhlý systém využívající databázi a programování.
- *Příprava podkladů*: Zpravidla je rozumné vytvořit si delší texty nejprve ve Wordu nebo jiném textovém procesoru, a teprve po důkladné revizi je převést do webové podoby.
- *Grafická stránka webu*: Mnohé prostředky pro tvorbu webu (například WordPress) nabízejí řadu grafických šablon, vytvořených profesionálním designéry. Pokud chcete vytvořit web „na zelené louce“, je často nejjednodušší svěřit právě grafické ztvárnění webu odborníkovi.
- *Vlastní tvorba webu*: Prostředek pro tvorbu webu zvolíte podle rozsahu a náročnosti webu.
- *Testování webu*: Tento bod je nutný zejména u složitějších webů. Testování by mělo provést více osob, z nichž aspoň některé nebudou mít s tvorbou webu nic společného. Zde nepochybně platí totéž, co u testování programů: nejlepším testerem je běžný uživatel.

- *Zajištění provozu webu:* Tento bod se bohužel mnohdy podceňuje. Web musí být pravidelně aktualizovaný a zálohovaný. Může to spolehlivě zajistit jen k tomu určená zodpovědná osoba.

Nakonec ještě zmíníme několik věcí, kterým se je radno při tvorbě webu vyhnout:

- *Pravopisné chyby a překlepy:* Je třeba upozornit na to, že programy pro tvorbu webu nemají zpravidla žádné funkce pro kontrolu pravopisu. U delších textů je proto vždy vhodné napsat je nejprve v textovém editoru, důkladně je zkontrolovat a teprve potom převést na web.
- *Nelogické a zmatené věty:* Tady platí v podstatě totéž co u předchozího bodu. Textový obsah webu by měl být vytvářen z pečlivě připravených podkladů.
- *Chybné odkazy:* Všechny obrázky a hypertextové odkazy je třeba pečlivě zkontrolovat.
- *Různá „překvapení“:* Pro návštěvníka webu není příliš příjemné, jestliže se náhodným klepnutím ocitne na úplně jiném webu, ze kterého se musí pracně vrátet. Dalším příkladem je nabídka bezplatných služeb, ovšem jakmile ji návštěvník webu použije, zobrazí se požadavek na zadání čísla účtu. Jedná se o dosti rozšířené nešvary.
- *Přemíra grafických efektů:* Přeplnění webu blikajícími a běžícími texty, popřípadě střídajícími se obrázky je často kontraproduktivní a řadu návštěvníků spíše odradí. Přeplněnost webu grafikou navíc zpomaluje jeho načítání. Nelze počítat s tím, že každý návštěvník bude mít k dispozici nejnovější model počítače a vysokorychlostní připojení k internetu.
- *Vlastní obsah webu je klíčový:* To je podstata webu, pro kterou se návštěvník rozhodl web navštívit. Neměl by být utopen v záplavě reklam a bannerů s druhotným obsahem.

1.6 Prostředky pro tvorbu webu

Pro tvorbu webu je možné použít dvě cesty:

- přímou tvorbu kódu,
- využití některého ze systémů, které jsou schopné kód webu vytvořit.

První cesta představuje tvorbu webu úplně od začátku a její nespornou výhodou je, že tvůrce webu má vše zcela pod kontrolou. Nároky na moderní weby jsou dnes však takové, že si tvůrce webu nevystačí se znalostí jazyka HTML a pravidel pro tvorbu stylů. Pro tvorbu webu je zapotřebí zvládnout základy programování aspoň v jednom jazyku (zpravidla PHP) a práci s databází. Rovněž aktualizace webu není jednoduchá záležitost a mnohdy je nad síly provozatele webu.

Z tohoto důvodu začaly již velmi brzy vznikat prostředky, jež měly tvorbu webu usnadnit. Prvními produkty tohoto typu byly například program NVU, nebo svého času poměrně rozšířený produkt Microsoft FrontPage. Tyto programy byly v podstatě generátory kódu HTML, který byl navíc vytvořen velice nepřehledně. Dnes již tento přístup patří minulosti.³ Revoluci v tvorbě webu přinesly tzv. redakční systémy, které celý obsah webu včetně struktury nabídky zapisují do tabulek v připojené databázi, zpravidla v systému MySQL. To umožňuje, aby vlastní web tvořilo relativně malé jádro. Vlastní tvorbu webu pak zajišťuje uživatelské rozhraní, které tvůrce webu umožňuje pracovat s databázovými tabulkami. Tato koncepce se ukázala jako velice perspektivní a označuje se jako „systém pro správu obsahu“ neboli **Content Management System** (CMS).

³ Poslední verze programu MS FrontPage je z roku 2003.

Díky uvedené koncepci se jedná o software, který umožňuje poměrně jednoduchou a hlavně rychlou tvorbu a aktualizaci obsahu internetových stránek. Proto dnes přestává platit, že se o internetové stránky musí starat osoba s rozsáhlými znalostmi programování. Při používání CMS vytvářejí obsah stránek editoři, kteří nemusí mít žádné znalosti toho, jak se webové stránky vytvářejí. To samozřejmě značně snižuje náklady na tvorbu a provoz webových stránek, ať už se jedná o osobní stránky, komerční weby, informační weby úřadů a veřejných institucí nebo rozsáhlé komunitní portály. Systémy CMS se uplatní všude tam, kde se obsah webu často mění, přidává ho více lidí nebo je žádoucí jeho pohodlná správa.

Protože se veškerá komunikace s databází provádí přes internet, pro tvorbu a správu webu postačí jen některý internetový prohlížeč. Díky tomu lze provádět správu webu z jakéhokoliv počítače, který má přístup k internetu. Web tak může být neustále udržovaný a aktualizovaný. Pokud má s webem pracovat více uživatelů, zpravidla mohou být rozděleni do více skupin s různou úrovní práv. Pak je snadné určit, co určitý uživatel smí provádět.

Rozsáhlejší systémy CMS nabízejí tvůrcům webu řadu předem připravených šablon pro zajištění designu webu a programových modulů provádějících různé činnosti. Z pohledu vývoje je proto použití CMS výhodné v tom, že vývoj webu je s použitím šablon a připravených programových modulů rychlejší a jednodušší, než kdyby se měl web tvořit úplně od začátku.

Systémy CMS jsou poměrně snadno instalovatelné přímo z internetu. Nabídka je značně široká, od velmi jednoduchých systémů s malými možnostmi až po důmyslné systémy umožňující uplatnit i vlastní tvorbu stylů a vlastní programování. Mezi nejznámější CMS systémy patří WordPress, Joomla, Drupal a Webnode. Zejména první dva jsou dnes rozšířené po celém světě. Jedná se o tzv. **open source software**, tedy software s otevřeným kódem. To znamená, že je zdarma dostupný každému, kdo o něj má zájem, a tvůrci těchto CMS systémů drží krok s vývojem nových technologií.

Mezi systémy WordPress a Joomla existuje určitý rozdíl v koncepci. WordPress je tvořen relativně malým stálým jádrem, další funkčnosti se mohou podle potřeby přidat použitím vhodné šablony a instalací **pluginů** (což jsou dodatečné moduly rozšiřující možnosti webu). U systému Joomla je řada funkcí zahrnuta již v jádru, výsledkem je složitější instalace a obsluha. Z toho plyne, že se WordPress hodí spíše pro menší weby, Joomla pro větší. Je ovšem třeba zdůraznit, že s postupujícím vývojem se rozdíl mezi oběma systémy stírají.

1.7 Základní charakteristika WordPressu

WordPress je redakční systém, který je vyvíjen jako open source projekt. První verze vznikla v roce 2003 a od té doby se WordPress stal nejpoužívanějším publikačním nástrojem pro blogy na celém světě. Wordpress ale prošel rozsáhlým vývojem a pomocí jeho redakčního systému bylo vytvořeno i velké množství „neblogových“ webů. S využitím nabízených pluginů a vlastního programování je možné pomocí WordPressu vytvořit sociální síť, diskusní fórum nebo elektronický obchod.

Mezi hlavní výhody WordPressu patří:

- *Snadná instalace:* Pro menší weby je dokonce možné provést registraci domény, objednání hostingu a instalaci WordPressu najednou.
- *Možnost tvorby webu bez psaní kódu:* Vytvoření jednoduchého webu zvládne v podstatě každý, postačí základní znalosti práce s počítačem a některého z textových editorů.
- *Hotový vzhled:* WordPress nabízí velké množství šablon vytvořených profesionálními designéry.

- *Snadná tvorba menu:* WordPress umožňuje přehlednou tvorbu menu i s poměrně složitou strukturou.
- *Podpora databáze:* V databázi, která obsahuje informace o vytvářeném webu, je možné poměrně snadno vytvářet vlastní tabulky a seznamy, jež pak lze využívat ve vytvářeném webu.
- *Velká nabídka pluginů:* WordPress nabízí velké množství doplňkových modulů, například kontaktní formuláře pro vkládání dat, které výrazně rozšiřují možnosti webu.
- *Malé náklady:* Základní instalace WordPressu je bezplatná, šablony a pluginy se nabízejí bezplatně nebo za nevelký poplatek.
- *Začlenění dalších uživatelů:* Na webu je možné vytvořit celý systém uživatelů s různými stupni práv.

Použití WordPressu má i některé nevýhody:

- Pokud se na vytvářený web kladou vysoké nároky, jeho tvorba se zpravidla neobejde bez tvorby vlastních stylů a vlastního kódu.
- Nabízených šablon je mnoho, ale nalezení optimální šablony není vždy úplně snadné.
- Určitou slabinou WordPressu je tvorba tabulek, i když poslední verze systému představují zřetelný krok vpřed.
- Protože se jak vlastní jádro WordPressu, tak i různé pluginy a šablony poměrně často upravují, je třeba dbát na pravidelné aktualizace.