

## Obsah

Pre koho je tento návod .....	2
Ako začať .....	2
Tvorba webových stránok .....	2
FRONTEND .....	2
HTML .....	3
CSS .....	3
JavaScript .....	4
Responzivita .....	4
Polovica cesty .....	4
BACKEND .....	5
Webový server .....	5
PHP, C#, Java, Python .....	6
Databáza .....	7
JavaScript .....	7
DEVELOPER BOSS .....	8

## Pre koho je tento návod

Ak uvažujete o tom, že by ste sa chceli živiť ako tvorca webových stránok, programátor, či vývojár mobilných aplikácií, potom je tento návod pre vás.

Alebo ak sa chcete iba naučiť základy programovania a nájsť si nový job v niektorej počítačovej firme, tak vám tento návod poskytne prehľad o tom, kde a ako začať.

Volám sa Roman Dufek. Už dlhé roky sa žívím programovaním. Vytváral som programy pre DOS, WINDOWS a ako jeden z prvých pre internet. Od roku 2007 mám vlastnú firmu, v ktorej sa v súčasnosti venujeme najmä tvorbe webových stránok a tvorbe programov a aplikácií podľa konkrétnych požiadaviek našich zákazníkov.

Tento návod som vytvoril pre vás, ak sa chcete zorientovať v aktuálnych trendoch, technológiách, ale hlavne vám dávam návod ako krok za krokom vybudovať kariéru v lukratívnej oblasti výpočtových technológií.

Potenciál na trhu je obrovský. Firmy zápasia medzi sebou o programátorov, pretože ich je veľký nedostatok. Programátori sú platení veľmi dobre a hravo strčia do vrečka svojím príjmom aj právnikov, či lekárov. Na rozdiel od lekárov a právnikov, ak chcete byť programátorom, nemusíte dlhé roky stráviť štúdiom na vysokej škole a nemusíte skladať žiadne skúšky.

V pohode sa zamestnáte ako junior programátor aj s minimom znalostí.

## Ako začať

Ak sa pozriete na trh s požiadavkami na programátorov, tak najžiadanejší sú programátori pre webové stránky, C#, Javu, Python, C++, PHP.

Osobne odporúčam začať tvorbou webových stránok a postupne sa posúvať k ďalším programovacím jazykom. Príležitostí na trhu je dostatok, na naštudovanie a pochopenie to nie je zložité a praktické výsledky vidíte veľmi rýchlo.

## Tvorba webových stránok

Webové stránky to nie sú iba informačné stránky, kde firmy informujú o svojich službách a produktoch. Pokiaľ by ste chceli robiť iba tento jednoduchý typ webov, tak na to nepotrebujete programátorské kurzy. Vystačíte si s jednoduchými klikacími systémami typu Wordpress.

To však nie je to, čo vás bude dobre živiť. Pokiaľ sa chcete posunúť na vyšší level a zarábať mesačne aspoň 3 tisíc Euro, potrebujete získať znalosti programovania pre FRONTEND aj BACKEND.

## FRONTEND

Pojem FRONTEND sa používa pre tú časť webu, ktorá je na prvý pohľad viditeľná. Sú to texty, ich usporiadanie, obrázky, nastavenie farieb atď.

---

Je to definovanie vzhľadu, textov a obrázkov na webe. Sú to tie časti webu a techniky programovania, ktoré vznikli ako prvé a určili spôsob akým sa webové stránky v počiatkoch internetu vytvárali.

FRONTEND programátorské znalosti je to, čím by ste mali začať. Ak túto fázu preskočíte a vhupnete hneď do klikacích systémov typu Wordpress, tak vám budú chýbať potrebné základné znalosti a výsledkom nebudú prvotriedne webové stránky. Mnohé požiadavky zákazníkov nezvládnete a budete frustrovaný z toho aké slabé výsledky dosahujete. Mnohé veci budete robiť spôsobom pokus – omyl a bude vám trvať veľmi dlho zapracovanie v podstate aj triviálnych požiadaviek zákazníkov.

Ak to myslíte s tvorbou webových stránok a prácou programátora nazozaj vážne a ak chcete, aby vás klienti a zamestnávateľ chválili, tak určite nepreskočte FRONTEND. Je to podobné ako keby ste chceli jazdiť na motorke a pritom neviete jazdiť a udržať rovnováhu na bicykli.

A čo teda FRONTEND znalosti tvorí?

Je to HTML, CSS, JavaScript a knižnice, ktoré sú nad týmito tromi základmi webu postavené. Pojem knižnica je odborné pomenovanie pre voľne dostupné zdrojové kódy iných programátorov, ktorí výsledky svojej práce dali k dispozícii komukoľvek, kto má záujem. Tieto knižnice, alebo sa môže stretnúť aj s pojmom framework, stiahnete bezplatne z internetu.

Medzi najznámešie FRONTEND knižnice, či frameworky, patria jQuery a Bootstrap.

## HTML

HTML je úplný základ. Je to spôsob zápisu, ktorým definujete obsah webovej stránky, t.j. texty, obrázky atď.

Prostredníctvom HTML stanovíte, čo je hlavný nadpis, čo je podnadpis, čo je odsek textu, ktorý text sa zdôrazňuje, vytvárate zoznamy, tabuľky atď.

Štúdiom HTML musíte určite začať. Každý iná cesta je krok vedľa.

## CSS

CSS je nadstavba nad HTML, ktorou definujete vzhľad webovej stránky. Prostredníctvom CSS určujete farby textu, farby pozadia, orámovanie prvkov na webe, tieň, vzájomné usporiadanie prvkov, čo má byť vľavo, vpravo, hore, dole, ako ďaleko sú prvky od seba, veľkosť písma, tvar písma a mnoho iných vecí.

Pomocou CSS tiež stanovíte rozdiely v zobrazení tej istej stránky na počítači, tablete, mobile.

Vďaka CSS môžete vytvárať rôzne animácie a efekty.

CSS je hneď druhá vec, ktorú sa musíte po HTML naučiť. Akýkoľvek iný postup je opäť krokom vedľa.

## JavaScript

HTML a CSS ešte nie je skutočné programovanie. HTML a CSS sa nezaruďujú medzi programovacie jazyky. Sú to iba akési značkovacie jazyky, ktoré používate na stanovovanie statického, teda nemenného, obsahu webu a jeho vzhľadu.

JavaScript už je programovací jazyk. Je to jazyk prostredníctvom, ktorého viete webové stránky rozšýbať. Ich obsah sa môže meniť na základe toho, kde a ako kliká návštevník webu.

Osobne odporúčam v začiatkovej fáze po naštudovaní HTML a CSS sa oboznámiť s jazykom JavaScript iba v základných rysoch. Neštudujte ho úplne podrobne, iba sa naučte ako sa začleňujú do kódu webovej stránky JavaScript knižnice, ktoré sú voľne dostupné na internete.

Naučte sa ako sa používa knižnica jQuery a knižnica Bootstrap. Tie sú dnes nutným štandardom a sú rokmi overené. Na webe nájdete množstvo návodov a doplnkov, ktoré sú nad nimi vybudované. Rôzne fotogalérie, slidery, prvky na výber dátumu, kalendáre atď.

Neštudujte však v tejto fáze úplne detailne všetky možnosti a syntax jazyka JavaScript. To je chyba mnohých začínajúcich vývojárov. Tu stratia veľa času, hoci JavaScript v tejto fáze neposunie vaše možnosti príliš dopredu. Naučiť sa detailne využívať možnosti JavaScriptu na to príde čas trochu neskôr.

## Responzivita

Responzivita je pojem a technika, ktorá sa používa na to, aby webové stránky boli pekné a dobre funkčné na každom zariadení, t.j. na počítači, tablete, mobile ...

Kým neprišiel prvý iPhone, tak ľudia pozerali webové stránky výlučne na počítači. Príchod iPhone to zmenil. Bol to prvý obchodne úspešný model telefónu, ktorý umožňoval plnohodnotné pozieranie webových stránok na mobile.

Dnes je tu iPhone a množstvo Android telefónov. Návštevnosť na webe telefóny valcujú. Dve tretiny návštevnosti dnes tvoria mobilné telefóny a iba tretina je prístup na webové stránky z počítačov.

Je preto bezpodmienečne nutné vedieť vytvárať responzívne webové stránky. Ak to budete ignorovať, tak sa pripravujete o 66% potenciálnych zákazníkov.

Vďaka knižnici Bootstrap dokážete vytvárať responzívne webové stránky pohodlne a jednoducho.

## Polovica cesty

Keď ste zvládli HTML, CSS, základy JavaScriptu a viete vytvárať responzívne webové stránky vďaka knižniciam jQuery a Bootstrap, ste v polovici cesty. Teraz prichádza chvíľa posunúť sa ďalej. Posunúť sa na BACKEND, ktorý je skutočným tajomstvom prvotriednych tvorcov webových stránok a ktorý vám otvorí bránu k tomu, aby ste sa mohli živiť programovaním.

---

## BACKEND

BACKEND to je skutočné programovanie, to sú skutočné prvotriedne webové stránky. Keď ste zvládli prvú fázu, FRONTEND, otvoria sa vám prekvapivé možnosti ako vďaka BACKEND technológiám vytvárať naozaj prekvapivé a kreatívne riešenia.

Teraz je tá chvíľa, kedy by ste mohli presedlať na Wordpress. Wordpress je dnes najpoužívanejšia platforma na tvorbu webových stránok.

**Osobne Wordpress neodporúčam a ani nepoužívam.**

Možno vás to prekvapí, prečo nie Wordpress, keď je najpoužívanejší. Hneď vám vysvetlím dôvody.

Wordpress je OK, ak chcete vytvárať pekné weby a bežné firemné weby rýchlo a jednoducho. Ak sa do toho pustíte, tak čoskoro zistíte, že neštandardné požiadavky zákazníkov sa až tak jednoducho realizovať cez Wordpress nedajú a zrazu začnete tápať a hľadať rôzne obchádzky ako obmedzenia Wordpressu obísť. Budete strácať čas, výsledkom budú nie dobre funkčné weby a mnohokrát budete nútený zákazníkom povedať „To sa nedá“.

Myslím, že povedať zákazníkovi „To sa nedá“ je situácia, do ktorej sa nechcete veľmi často dostať. Určite by ste radšej povedali „Viem, že v tomto systéme sa to nedá, ale mi to vieme urobiť. Dokážeme to, pretože to robíme inak ako väčšina.“

## Webový server

Aby ste pochopili ako funguje Wordpress a našli jeho lepšiu alternatívu, musíte spoznať pojem webový server.

Webový server je program, resp. skupina programov, ktoré bežia na samostatnom počítači. Tento samostatný počítač prijíma požiadavky na vygenerovanie HTML kódu pre konkrétnu webovú stránku, ktorú si chce užívateľ zobrazíť.

Predstavte si napríklad, že niekto si pozerá na webe predpoveď počasia. Zadá dátum a miesto, pre ktoré chce vedieť aké bude počasie. Táto požiadavka dorazí na webový server. Ten ju prijme a vygeneruje odpoveď vo forme HTML kódu s informáciami o počasí pre daný dátum a miesto.

Pomocou JavaScriptu by ste toto nedosiahli, potrebujete webový server, ktorý sa vie dostať k množstvu údajov. Potrebujete vedieť ako tieto údaje prečítať, spracovať a vytvoriť z nich HTML kód.

V súčasnosti sú najpoužívanejšími webovými servermi Apache a IIS.

Aby ste s nimi mohli pracovať a požiadavky, ktoré z webu prichádzajú vedeli spracovať, potrebujete sa naučiť programovací jazyk, ktorý daný webový server pozná.

V prípade servera Apache je to najčastejšie jazyk PHP a v prípade webového servera IIS je to najčastejšie jazyk C#.

## PHP, C#, Java, Python

Okrem jazykov PHP a C# existujú aj ďalšie programovacie jazyky ako napr. Java a Python, ktoré na strane webového servera viete využiť. Ako začiatočníkom však odporúčam buď PHP alebo C#.

Osobne preferujem pustiť sa smerom k webovému serveru IIS a k jazyku C#, hoci celosvetovo je viac webov na strane Apache + PHP. PHP je programovací jazyk, nad ktorým je postavený aj Wordpress.

### Prečo C# a nie PHP?

Komunita okolo PHP sa vytvorila ako alternatíva voči C# z prozaického dôvodu. Nástroje pre vývoj v PHP bolo roky zadarmo a nástroje pre C# boli platené.

V PHP vyvíjali weby individuálni vývojári, malé a začínajúce firmy, ktoré si nemohli dovoliť investovať. V C# sa vytvárali najmä veľké projekty, za ktorými stáli bohaté firmy, ktoré mali dostatok financií na vývoj. Pomocou PHP viete vytvárať webové stránky. Pomocou C# viete okrem webov vytvárať aj Windows programy aj mobilné aplikácie.

Vývojári v PHP sú platovo na tom horšie ako vývojári v C#. C# programátori sú viac cenení a vyhľadávaní.

V roku 2014 sa do vedenia spoločnosti Microsoft dostal Satya Nadella. Microsoft je autorom webového servera IIS a programovacieho jazyka C#. Microsoft určite poznáte ako autora Windows. Je to silná spoločnosť, ktorá bola dlhé roky najbohatšou spoločnosťou sveta a aj vďaka tomu dokázala priniesť na trh IIS a jazyk C#, ktoré majú oveľa viac možností ako PHP.

A práve v roku 2014 po zmene vedenia nastal zlom. Microsoft sa rozhodol, že dá vývojárske nástroje, ktoré dovtedy predával za tisícky Eur, bezplatne všetkým vývojárom.

Dnes už teda peniaze nie sú prekážkou a aj individuálni programátori môžu používať profesionálne Visual Studio.

Ak chcete väčší potenciál a lepšiu perspektívu zamestnania a odmeňovania, vyberte si webový server IIS a programovací jazyk C#. Určite neoľutujete. Budete si môcť vyberať z lepšie platených pracovných ponúk a nemusíte ostať iba pri webových stránkach.

Osobne poznám viacerých programátorov, ktorí prešli z PHP na C# a už sa nechcú vrátiť späť. Nepoznám však žiadneho, ktorý by prešiel opačným smerom.

**Neurobte rovnakú chybu ako mnohí iní začiatočníci, ktorí išli cestou Wordpress alebo v tom menej horšom prípade cestou PHP a až následne Wordpress.**

Začnite ihneď na perspektívnejšie strane. Začnite s BACKEND technológiami IIS, .NET, C# a vývojovým prostredím Visual Studio.

Niektorí menej znalí PHP nadšenci vám povedia, že potrebujete Wordpress, aby si mohli majitelia webov meniť obsah webových stránok sami. Ak sa pustíte smerom k jazyku C#, tak tam Wordpress k dispozícii nie je a preto by ste mali ísť cestou PHP.

Pre C# ale existujú Wordpress alternatívy. Najpoužívanejšou je Umbraco. Vďaka Umbracu dokážete so znalosťami C# bez znalostí PHP robiť rovnaké veci ako umožňuje Wordpress. Alebo, keď na to príde, urobíte si vlastnú náhradu za Wordpress. Osobne však nevidím na to dôvod.

## Databáza

Vráťme sa k nášmu príkladu s počasím. Keď chcete zobrazovať počasie pre rôzne miesta a rôzne dni, musíte mať informácie pre predpovede počasia niekde uložené. A hlavne musíte ich mať uložené tak, aby ste ich mohli veľmi rýchlo prečítať a odoslať na webové stránky.

Na to slúži databáza a vy ako programátor musíte ovládať prácu s databázou. Musíte sa naučiť ako sa do nej dáta ukladajú a ako sa z nej dáta čítajú.

Bez databázy prvotriedny web nevytvoríte. Bez databázy budete odkázaný iba na tvorbu základných jednoduchých webov, ktorých obsah si majiteľ sám meniť nedokáže. Vďaka databáze dokáže Wordpress aj Umbraco poskytnúť rozhranie, cez ktoré si majiteľ webu môže meniť texty na webe, obrázky, pridávať nové podstránky atď.

Najpoužívanejšie databázy na strane PHP sú MySQL a MariaDB. Na strane C# je to MS SQL.

Ak sa pustíte na tú perspektívnejšiu vývojársku dráhu, na stranu C#, odporúčam používať databázu MS SQL. Môžete tiež využiť MySQL alebo MariaDB aj v C#. Avšak pre Umbraco, ako náhradu za Wordpress, budete potrebovať databázu MS SQL.

Či už používate MySQL, MariaDB alebo MS SQL, tak na prácu s nimi potrebujete poznať jazyk SQL a ten je vo svojich základných črtách totožný pre všetky uvedené databázy.

## JavaScript

Trochu znalí vývojári sa možno budú čudovať, prečo som zaradil JavaScript do časti pre BACKEND. Je to preto, pretože až tu využijete JavaScript naplno.

Až vo chvíli, keď ovládnete HTML, CSS, C# a SQL, je ten správny čas na to, aby ste za ponorili detailne do JavaScriptu.

Teraz je ten správny čas na to, aby ste využili JavaScript, JSON, AJAX a začali vytvárať webové stránky, ktoré budú naozaj perfektné. Stránky, ktoré budú nielen vizuálne príťažlivé, ale budú bezchybne funkčné rýchle a dokážete na ne doplniť akékoľvek funkcie, ktoré si zákazník vymyslí.

Neurobte chybu ako takmer každý začínajúci web programátor, že budete detailne študovať JavaScript skôr ako ho dokážete naozaj naplno využiť.

## **DEVELOPER BOSS**

Ovládnite HTML, CSS, C#, SQL, JavaScript a buďte DEVELOPER BOSS. Buďte programátor, ktorý je lepšie platený ako bežný tvorcovia webových stránok. Začnite tvorbou webových stránok, pretože je to najrýchlejšie na štart.

Keď sa rozhodnete alebo sa vám naskytne príležitosť, môžete sa od webov potom posunúť k vývoju hier, mobilných, či podnikových aplikácií. Programovací jazyk C# je na to ideálny.

Všetky potrebné technológie, programovacie jazyky a techniky sa naučíte aj v našom [kurze programovani a tvorby webových stránok](#).