

PB - Vyšší odborná škola a Střední škola managementu, s.r.o.

# Absolventská práce

2006

Petr Šimčík

PB - Vyšší odborná škola a Střední škola managementu, s.r.o.

Nad Rokoskou 111/7, Praha 8

Obor: Aplikace výpočetní techniky

Název absolventské práce:

# **Linux jako platforma pro webdesign**

Školní rok: 2005/2006

Vypracoval: Petr Šimčík

Vedoucí absolventské práce: Milan Randák

### Prohlášení:

Prohlašuji, že jsem absolventskou práci na téma „Linux jako platforma pro webdesign“ vypracoval samostatně.

Použitou literaturu a podkladové materiály uvádím v příloženém seznamu literatury.

V Praze dne 14/4/2006

### Poděkování:

Touto cestou bych rád poděkoval Martinu Podholovi za jeho připomínky, opravy a kritiku k části mé práce věnující se editoru Vim a historii operačního systému Linux.

Dále bych rád poděkoval vedoucímu mé práce, panu profesorovi Milanovi Randákovi za jeho přístup a podnětné návrhy k mé práci.

Úvod.....	7
1    Základní pojmy .....	8
1.1    Co je to webdesign a co pro něj potřebujeme?.....	8
1.2    Vytváření a ladění vizuální stránky webu .....	8
1.3    Propagace, použitelnost a úspěšnost webu.....	9
1.4    Použití CSS a pravidel W3C. ....	10
1.5    Historie Internetu a webdesignu, W3C. ....	10
1.6    Co je to Linux, Historie a současnost Linuxu .....	11
2    Programy potřebné pro webdesign.....	13
2.1    Grafické programy .....	13
2.2    Wysiwyg editory .....	14
2.3    Editory kódu.....	14
3    Co nám nabízí Linuxové operační systémy? .....	14
3.1    Grafické programy .....	15
3.1.1    Sodipodi .....	15
3.1.1.1    Ovládací prvky .....	15
3.1.1.2    Nástroje .....	15
3.1.2    Inkscape.....	17
3.1.2.1    Ovládací prvky .....	17
3.1.2.2    Nástroje .....	17
3.1.3    Gimp .....	18
3.1.3.1    Ovládací prvky .....	18
3.1.4    Nástroje .....	19
3.1.5    F4L .....	20
3.2    Editory kódu.....	21
3.2.1    Quanta+ .....	21
3.2.2    Bluefish.....	23
3.2.3    CSSed .....	24
3.2.4    Vim a Emacs.....	25
3.2.4.1    Vi, Vim.....	25
3.2.4.1.1    Představení a historie.....	25
3.2.4.1.2    Funkce a ovládání .....	26
3.2.4.2    Editor Emacs .....	27
3.2.4.2.1    Představení a historie.....	27
3.2.4.2.2    Funkce a ovládání .....	28
3.2.4.2.3    Needitorové funkce editoru Emacs.....	29
3.2.5    MC Editor .....	30
3.3    WYSIWYG .....	30
3.3.1    NVU .....	30

4	Tvorba webových stránek.....	32
4.1	Moje osobní Internetová prezentace .....	32
4.1.1	Grafika - 1.část - Základní design .....	32
4.1.2	Kód - 1.část - Základ stránky.....	34
4.1.3	Ladění - 1.část - Zarovnání základních částí stránky .....	35
4.1.4	Grafika - 2.část - Navigace .....	35
4.1.5	Kód - 2.část - Navigace .....	36
4.1.6	Grafika - 3.část - Grafická náplň webu.....	36
4.1.7	Kód - 3. část - Zbylý kód a náplň webu.....	38
4.1.8	Ladění - 2. část - Doladění vzhledu ve většině prohlížečů.....	39
4.2	Můj Blog.....	40
4.2.1	Instalace publikačního systému phpRS .....	40
4.2.2	Grafika - Základní design .....	40
4.2.3	Kód - Základ stránky, návrh vzhledu, CSS.....	41
4.2.4	Ladění - Přizpůsobení mým představám.....	42
4.3	RSS - Informační kanály stránek .....	42
5	Závěr .....	43
	Resumé .....	44
	Seznam literatury .....	45
	Seznam příloh.....	46

# Úvod

Svoji práci jsem zaměřil na webdesign, jelikož mě tvorba webových stránek zajímá a chtěl bych se mu věnovat i po dokončení studia. Zároveň mě fascinuje svět Linuxu, ale pod Linuxem jsem ještě webdesign provozovat nezkoušel, a proto to chci v mé práci napravit. Linux mě zaujal ihned, jak jsem se s ním poprvé setkal, musím přiznat, že nejvíce tím, že je zdarma. Jak jsem ale pronikal do tajů tohoto operačního systému, oblíbil jsem si na něm i spoustu jiných vlastností, např. propracovanost, konfigurovatelnost, rychlost a dostupnost. Ve své práci se chci věnovat v Linuxu dostupným programům pro webdesign také proto, že jsem prozatím žádnou publikaci, která by je popisovala neviděl, a tak chci touto prací přispět k popularizaci Linuxu, pomoci začínajícím uživatelům Linuxu se orientovat v těchto programech a poskytnout jim základní informace k tomu, aby si mohli zvolit program, který bude vyhovovat právě jim.

Myslím si, že téma mé práce je velice aktuální, protože původně výlučně serverový systém Linux se jakožto operační systém pro desktop začal profilovat teprve v posledních několika letech a ukazuje se, že může v mnoha ohledech konkurovat ostatním operačním systémům. Linux směle může konkurovat ostatním operačním systémům např. v administrativě, jako platforma pro programátory, jako multimediální stanice (tzv. multimediální centrum domácnosti). Proč by tedy nemohl být dobrou platformou pro tvorbu internetových prezentací?!

Ve své práci bych chtěl seznámit čtenáře s tím, co je to webdesign, na co všechno musí dobrý webdesigner myslet při tvorbě internetové prezentace, jaká byla historie webdesignu a internetu vůbec. Dále nastínit historii a současnost Linuxu, možnosti jeho využití s příklady nasazení u nás i ve světě a vysvětlit, co to Linux je. V další části jsem se rozhodl popsat, jaké programy potřebujeme pro tvorbu webových stránek, jaká je nabídka nejpoužívanějších programů pro tvorbu webu a grafiky pro web používaných v Linuxových distribucích, a seznámit čtenáře mé práce se vzhledem, funkcemi a ovládáním těchto programů. Na konci své práce bych chtěl předvést uplatnění některých programů v praxi prostřednictvím vytvoření svých vlastních webových stránek a poté shrnout své poznatky o tom, jestli je Linux připraven stát se platformou pro tvorbu internetových stránek a zjistit, zda mu v tomto oboru některé druhy programů chybí či v něčem vyniká.

# 1 Základní pojmy

## 1.1 Co je to webdesign a co pro něj potřebujeme?

Webdesign je zjednodušeně řečeno tvorba internetových stránek, tedy alespoň té jejich části, která je vidět. Pro webdesign by nám teoreticky mohl stačit program Poznámkový blok (Notepad) a znalost HTML - jazyku, ve kterém je většina Internetových stránek napsána. Tak jednoduché to ale zdaleka není. Tyto prostředky by nám přestaly stačit hned, jakmile bychom chtěli do stránek začlenit nějaký složitější tvar, obrázek, zvláštní členění odkazů či atypické rozmístění textu. Pro složitější tvorbu stránek, a tedy „opravdový“ webdesign bychom toho potřebovali mnohem víc, počínaje grafickými editory pro tvorbu všemožné grafiky, přes wysiwyg editory umožňující vizuální tvorbu internetových stránek bez znalosti kódu a textové editory se zvýrazněním syntaxe programovacích jazyků pro webdesinery, kteří upřednostňují tvorbu stránek pomocí psaní kódu. Jak vidno webdesign se skládá z mnoha podkategorií. Je to estetická stránka webu, navigace, problematika použití barev, zrychlení a optimalizace grafiky, použití animací a zvuku, použitelnost stránek, volba architektury odkazů, přizpůsobení cílovému uživateli, registrace stránek pro vyhledávače a do katalogů, použití CSS, pravidel W3C a v neposlední řadě užití skriptů na straně uživatele.

## 1.2 Vytváření a ladění vizuální stránky webu

Kategorie estetická stránka webu, navigace, problematika použití barev, zrychlení a optimalizace grafiky a použití animací a zvuku můžeme dohromady označit jako vytváření a ladění vizuální stránky webu. Vizuální stránka webu je první věc, podle níž si návštěvník webu udělá svůj první dojem a většinou se rozhoduje, jestli náš web opustí nebo ho bude prohlížet dále. Proto by si měl webdesigner nechat záležet na tom, aby stránky vypadaly navenek co nejlépe. Navigace je pak druhá věc, kterou bude návštěvník webu hledat. Webdesigner by se měl tedy postarat, aby hledal co nejkratší dobu nebo nejlépe vůbec. Použití správných barev je opět klíčové, sice můžeme mít skvělou navigaci a krásné obrázky, ale pokud bude muset návštěvník přivírat oči kvůli příliš ostrým nebo je naopak napínat kvůli nevýrazným či splývajícím barvám, rychle naši stránku opustí. Správné barvy by měl webdesigner vybírat podle toho, jaký bude web plnit účel, kterou část stránek chce zvýraznit, např. navigaci nebo reklamu, podle jejich psychologického významu a také podle toho, jak spolu kontrastují, případně splývají. Zrychlení a optimalizace grafiky jsou velmi důležité pro uživatele s pomalejším připojením, ale ani návštěvník s rychlým pevným připojením nebude rád, když bude muset čekat na načtení důležitých obrázkových prvků, například navigace stránky. Optimalizace se řeší snížením bitové hloubky či kvality obrázku tak, aby změna nebyla viditelná pouhým okem, případně také rozřezáním větších obrázků na několik jednotlivých kusů, které se budou načítat zvlášť, a tedy rychleji. Ruku v ruce s optimalizací grafiky jde použití animací a zvuku. Pokud chceme na web nějaké animace a zvuk umístit, čeká nás rozhodování, v jaké kvalitě a jaký

formát? Použít všeobecně rozšířený nebo menšinový, ale lépe vypadající? Ozvučit jen něco nebo celou stránku?

## 1.3 Propagace, použitelnost a úspěšnost webu

Zajištění použitelnosti stránek, zvolení architektury odkazů, přizpůsobení cílovému uživateli, registrace stránek pro vyhledávače a do katalogů a užití skriptů na straně uživatele - všechny tyto věci patří do kategorie propagace, použitelnost a úspěšnost webu. Zajistit použitelnost stránek by měl každý dobrý webdesigner a to nejen pro zdravé lidi, ale i pro ty uživatele internetu, kteří mají nějaké postižení, buď jsou barvoslepi a mohou vidět jen některé barvy, a nebo jsou slepi úplně a využívají hlasového výstupu počítače ke čtení stránek. Sice tato skupina lidí není tak početná, ale pokud se jedná o potenciální návštěvníky webu, měl by s nimi autor stránek dopředu počítat. Zvolení výhodné architektury webu je také velmi důležité. Při jejím navrhování bychom měli vzít v úvahu možnost nejen se ponořit hluboko do sítě stránek, ale také se moci kamkoli vrátit, přeskočit na jiné téma apod. Web by měl být co nejvíce přizpůsoben cílovému uživateli. Jinak bude vypadat strohý web pro programátory zpřístupňující např. nové verze knihoven, jinak web teenagerského časopisu, od kterého se očekává, že bude zářit barvami a pestrostí, a jinak web prestižní firmy, kde by měla být jednoduchá, ale poutavá grafika a velké množství snadno dosažitelných informací. Na tento aspekt tvorby webu by měl webdesigner myslet co nejdříve, nejlépe by ho měl napadnout již ve chvíli, kdy se jde o práci ucházet, a podle toho nabídnout svoji vizi stránek. Registrace stránek do vyhledávačů i katalogů je zásadní pro zvýšení návštěvnosti stránek. Této registraci se zkráceně říká SEO. SEO probíhá na několika frontách. Jsou to registrace do vyhledávačů, registrace do katalogů, registrace kamkoliv jinam a vytvoření seznamu vyhledávaných slov v META tazích. Vyhledávače a katalogy jsou servery, které shromažďují odkazy na stránky a pak je nabízejí zpravidla s dalšími službami koncovému uživateli, u nás mezi ně patří např. [www.seznam.cz](http://www.seznam.cz), [www.centrum.cz](http://www.centrum.cz), [www.atlas.cz](http://www.atlas.cz) a [www.jyxo.cz](http://www.jyxo.cz), mezi světové potom všemocný [www.google.com](http://www.google.com) a [www.yahoo.com](http://www.yahoo.com). Registrace na všech ostatních stránkách nám přinese sice méně, ale návštěvnost určitě zvedne také. Mezi tyto další stránky patří např. diskusní fóra, stránky s hodnocením webových stránek a profesní sdružení. Nakonec je důležité vytvořit seznam slov, podle kterých se dají naše stránky najít, to se dělá s pomocí takzvaných tagů META, které jsou umístěny v hlavičce každé stránky. Tento seznam vytváříme pro případ, že by se potenciálnímu návštěvníku nechtělo naše stránky hledat v kategoriích katalogů a hledal by je za pomoci vyhledávání slov ve vyhledávači. Užití skriptů na straně uživatele sice patří mezi vedlejší činnosti a klidně je můžeme vynechat, ale s rozvojem internetu se tato funkce stránek prosazuje i vyžaduje čím dál tím více. Přitom nemusí jít o vyloženě složité skripty, slouží například ke změně tlačítek, kontrole vyplněného formuláře, vytvoření složitější webové galerie, pohybu textu či obrázku a podobným efektům. Stránky vypadají profesionálněji, když obsahují třeba jen již zmíněnou změnu odkazů po jejich označení. Vždy si ale webdesigner musí uvědomovat tenkou čáru mezi množstvím a kvalitou, praktičností a přelácaností svých stránek.

## 1.4 Použití CSS a pravidel W3C.

Použití CSS a pravidel W3C není sice povinné, ale dobré internetové stránky bez jednoho ani druhého nejdou vytvořit. CSS jsou jazykem, který umožňuje formátování jakékoli části stránek bez závislosti na jejich obsahu. CSS nám umožňuje pozicovat text, obrázky, tabulky, vytvářet rámečky, relativně umístěné plochy a spoustu dalších prvků stránek. To všechno šlo dříve i pomocí tabulek a HTML, ale stránky měly nepřehledný kód, byly mnohem náročnější na internetové připojení a nebyly editovatelné. Pro přidání nabídky, tlačítka apod. bylo nutné celé stránky vytvořit téměř znovu. Pravidla W3C jsou jakousi normou pro tvorbu a zpřístupnění internetových stránek. Tato norma je doporučena pro všechny internetové stránky bez rozdílu zaměření, určuje pravidla pro dostupnost obsahu a pro programovací jazyk, kterým jsou psány, takzvanou validitu kódu.

## 1.5 Historie Internetu a webdesignu, W3C.

V roce 1962 firma DARPA zahájila výzkum počítačů a jejich možnou spolupráci. V roce 1969 se podařilo zapojit čtyři uzly do sítě nazvané ARPANET, koncovka je z anglického NET, což znamená síť nebo pavučina. O tři roky později byl ARPANET rozšířen o 17 směrovačů a 38 počítačů a v téže roce Ray Tomlison píše první e-mailový program. O rok později, v roce 1973 byla zveřejněna specifikace TCP a v roce 1980 byl nasazen experimentální provoz TCP/IP ve verzi Ipv4 a vznikl DNS protokol.

Původně byl Internet vojenský projekt určený pro výměnu dat. Po jeho uvolnění se postupně prosadil mezi studenty a profesory na amerických univerzitách jako médium pro výměnu dokumentů. Stále ovšem neměl tu podobu, kterou známe dnes, a existovaly-li tehdy nějaké internetové prohlížeče, byly zcela jistě textové. První zlom přišel v roce 1993, kdy nastala revoluce s příchodem prvního grafického prohlížeče s názvem Mosaic. Jeho autory byl Marc Andreessen a několik dalších studentů z University of Illinois. V roce 1994 vypukly války Internetových prohlížečů a nastaly staré špatné časy webdesignu. V žádné jiné době nebylo možné vidět na internetu více stránek s prohlášením, že tato stránka se zobrazuje jen v jednom z prohlížečů. Hlavními soupeři na tomto elektronickém bojišti byly firmy Microsoft a Netscape, jež se snažily vytlačit svého soupeře vydáváním stále nových a nových funkcí, které konkurenční prohlížeč nepodporoval. Přestože již v roce 1994 byla založena organizace W3C, jež se snažila vytvářet standardy pro web a internetové prohlížeče, válka se táhla ještě rok a skončila vítězstvím Microsoft Internet Exploreru a drtivou porážkou prohlížeče Netscape, který sice existuje také dodnes, ale je to spíše jiné rozhraní pro zobrazovací jádra prohlížečů Mozilla a Internet Explorer než samostatný prohlížeč. Druhá válka prohlížečů začala v roce 1998, kdy byla založena Mozilla Foundation a na trh vypuštěn její prohlížeč Mozilla, který byl součástí Mozilla Suite - balíku, který obsahuje i emailového klienta a wysiwyg editor internetových stránek. Tato válka prohlížečů stále trvá a nový prohlížeč Mozilla Firefox si od svého vypuštění v roce 2003 získal svou jednoduchostí, bezpečností a propracovaností 15% uživatelů z celkového koláče uživatelů Internetu. Tato válka prohlížečů je ale již o mnoho klidnější a vede se schopností prohlížeče dodržovat normy W3C.

Jako celý internet, i webdesign vznikl v Americe, a to nejspíše s rozšířením prvních grafických prohlížečů. Protože první platformou podporující grafiku byl Apple Macintosh a jeho systém Mac OS, většina designérů té doby používala právě počítače Apple, a i když to dnes již zdaleka neplatí, většina amerických webdesignerů mu zůstala věrná. V Evropě to tak ale není. Internet a webdesign jako profese se do ní dostaly až ve druhé vlně, kdy již byly dostupné Windows a grafický internetový prohlížeč pro ně. Dnes se dostáváme do fáze, kdy by bylo možné počítat s další platformou pro webdesign, a to je právě Linux, BSD a ostatní Unixové systémy, i když se stále protloukají řadou nevýhod, například tím, že na nich neběží většina komerčních aplikací.

Jak jsem již napsal, W3C byla založena pro podporu webových standardů v roce 1994 a od té doby postupně vydala standardy HTML 2.0, HTML 3.2, HTML 4.0, XML 1.0, XHTML 1.0, HTML 4.01 a XHTML 1.1. W3C se stará také o standardy CSS a třeba i grafiky, například formátu SVG. Kromě toho vydala doporučení pro úpravu stránek pro všeobecnou přístupnost.

## 1.6 Co je to Linux, historie a současnost Linuxu

Linux samotný je jen jádro systému, které je samo o sobě pro běžného uživatele téměř nepoužitelné. Linux je proto dodáván v tzv. linuxových distribucích. Linuxová distribuce je Linuxové jádro dodávané s velkým množstvím programů, jejichž složení i množství se liší od distribuce k distribuci. Tyto distribuce se dělí na komerční a nekomerční. Mezi komerční patří např. Red Hat Linux a Mandriva, mezi nekomerční potom Slackware, Debian, Arch Linux, Ubuntu a Aurox. Kromě těchto distribucí existují ještě další, úžeji specializované, např. na vývoj softwaru, pro bezpečnost sítí, na multimédia, záchranu dat, speciální serverové distribuce apod. Linux podporuje mnoho softwaru, u každé činnosti si můžeme vybrat, kterou alternativu zvolíme, na výběr máme z několika kancelářských balíků, internetových prohlížečů, emailových klientů, grafických programů, multimediálních přehrávačů, textových editorů, prohlížečů a organizérů digitálních fotografií a celých grafických prostředí. Výhodou Linuxu je mimo jiné i jeho šíření pod licencí GPL. Ta umožňuje každému používat program zcela svobodně, upravovat ho i dále šířit za podmínky, že bude opět pod licencí GPL. Šíření Linuxu pod touto licencí je výhodné i pro uživatele, kteří ho chtějí jen používat. Díky tomu, že na programu spolupracují lidé z celého světa, jsou si navzájem testery a objeví více chyb než při testování komerčním způsobem. Díky tomuto přístupu jsou chyby v systému velice rychle opraveny a systém je pak mnohem bezpečnější než většina komerčních řešení. Pro vyzkoušení Linuxu nám může posloužit tzv. Live Distribuce, která existuje téměř od každé běžné distribuce. Její výhodou je, že je spustitelná z vyjimatelného média a obsahuje veškeré vybavení, které má standardní nainstalovaná distribuce. Pouze startuje a načítá se o něco déle.

Linux je implementací operačního systému UNIX, která je zcela volně šiřitelná a upravitelná. Tvůrcem Linuxu je Linus Torvalds, který práce na Linuxu započal jako student univerzity ve finských Helsinkách v roce 1991 v rámci předmětu „Operační systémy“ v jazyce assembler jako svůj první pokus o vlastní operační systém. Toto rozhodnutí nabylo v inspiraci ze systému Minix, kde se mu řada věcí nelíbila a pro jejich změnu se inspiroval systémem Unix. V témže roce se mu podařilo napsat

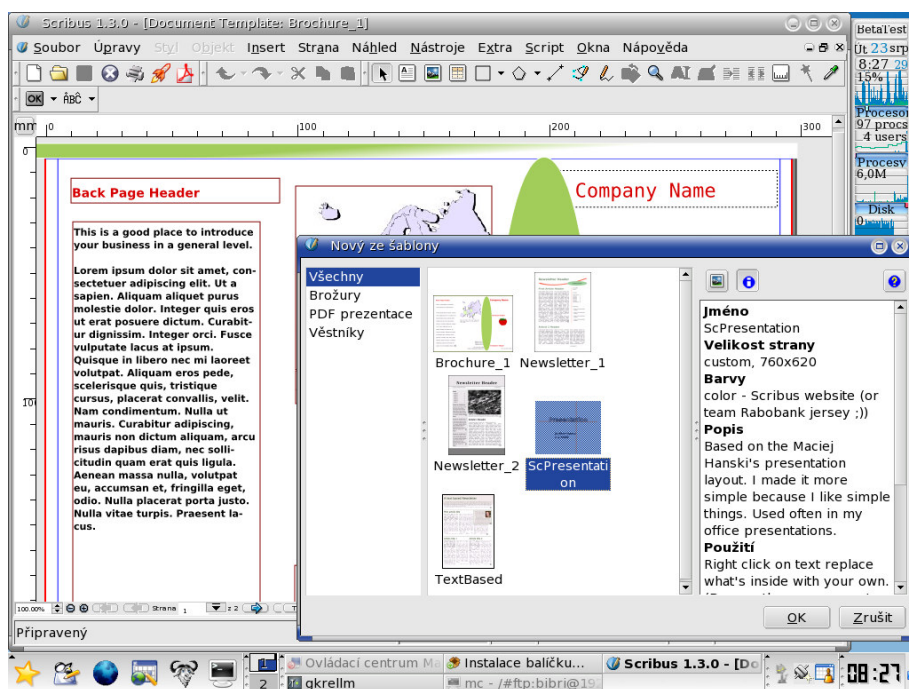
funkční verze jádra, která uměla pracovat se signály normy POSIX a ošetřovat rutiny správy paměti. Toto jádro bylo psáno na 360Kb paměti. Spouštělo se jako program v již běžícím systému. V srpnu roku 1991 byla uvolněna třetí verze jádra Linux, která se v té době jmenovala Phreax , ta byla schopna startovat nezávisle, zaváděla se z diskety. Toto jádro administrátor, Linusův kamarád, umístil i pro ostatní na školní ftp server a pojmenoval ho Linux (Linusův Unix). Další verze byla přepsána do jazyka C a řada studentů si jej oblíbila a začala pro něj rozšiřovat ovladače hardwaru. Jádro získávalo stále větší oblibu ve světě a přispěvatelé přibývali. Linus vše koordinoval do fungujícího celku. Značení jádra je takové, že první číslo udává hlavní řadu, v současnosti je 2. Linus prohlašuje, že 3 nikdy nebude. Druhá číslovka znamená hlavní revizi. Pokud je sudá, jedná se o stabilní verzi pro normální použití, lichá je testovací pro příští stabilní. Poslední číslo značí „drobnou revizi“. Současně se vyvíjí jak 2.2. větev, tak 2.4. a 2.6., od řady 2 byla zavedena modularita ovladačů, proto je nyní snadnější programovat ovladače zařízení. Ty mohou být přidávány do jádra za běhu bez restartu a opět odstraňovány. Ve verzích 2.6 oproti 2.4 proběhla řada významných změn, byl přepsán systém POSIX vláken a inheritance protected módu uživatelské paměti pro virtuální procesory a řada dalších majoritních změn. Jádro je důkladně testováno a prověřováno jak na stabilitu, tak na bezpečnost. První zdrojové kódy jádra byly malé, nezabíraly ani pětinu diskety, dnešní kód má ve zkomprimované podobě skoro 40MB. Aktuální verze řady 2.6 je 2.6.16rc4 a řada 2.4 má 2.4.33pre2. Systém, který nyní používám k psaní této práce, má verzi jádra 2.6.16

K rozšíření Linuxu pomohla velkou měrou Open Source Software Foundation (nadace pro otevřený software), nadace pro podporu tvorby svobodného softwaru, pod licencí GPL . Linusovi se líbil projekt pana Richarda Matthew Stallmana GNU. Pro GNU byla založena právě OSS Foundation, která se stará o GNU/GPL licenci pro software. V rámci projektu GNU byla vytvořena řada uživatelského a systémového software, který je dodáván spolu s jádrem Linux jako plný operační systém GNU/Linux. Tato licence se vztahuje jak na samotný Linux, tak na ostatní programy, které jsou součástí linuxových systémů. Základem filozofie Open Source Software Foundation je přesvědčení, že by měl být veškerý software volně dostupný jak v normální použitelné formě, tak jako zdrojový kód programu a upravitelný každým, kdo by o to projevil zájem. To je mimo jiné také hlavní výhodou Linuxu.

Další, co ovlivnilo Linux byl Unixový operační systém BSD, ze kterého vychází i komerční Mac OS X. Od BSD jsou odvozeny některé síťové utility a démoni používaní v Linuxu.

Dnes je už Linux plnohodnotným systémem s obrovským množstvím aplikací pro téměř jakoukoli oblast činnosti s počítačem. Podporuje nejnovější hardware, systémy pro více procesorů, je ideálním systémem jak pro servery, tak pro desktopy, dá se využít i v mobilních telefonech či PDA, dokonce existují i Linuxové distribuce určené pro tzv. Multimediální centra domácností. Jádro je plně přizpůsobitelné a nasazení Linuxu nic nebrání. Linux se využívá ve filmových společnostech k tvorbě triků (Byl použit například při natáčení filmů Matrix nebo Titanic ) , v Pacifiku je to systém používaný na lodích k monitorování oceánografických dat, používají ho i některá velká evropská města jako hlavní systém ve státní správě a najdeme ho třeba i na polárních stanicích, kde zpracovává vědecká data. U nás Linux například řídí veškerý letový provoz na letišti v Ruzyni a soupravy pražského Metra, Pražská Plynárenská společnost před dvěma lety nasadila linuxové

řešení do zařízení na regulaci tlaku v potrubí a směšování plynů. Příkladů ze života je mnoho, pokud vlastníte nějaké zařízení na bázi inteligentního řízení spotřeby, topení nebo i dvd rekordér, je téměř 50ti procentní šance, že toto zařízení řídí právě Linux.



Obrázek 1 : Mandriva Linux - distribuce, kterou jsem použil jak při testování všech programů, tak při psaní této práce.

## 2 Programy potřebné pro webdesign

### 2.1 Grafické programy

V prostředí Microsoft Windows k nim patří velice oblíbený, komerční program Adobe Photoshop, se kterým se setkal snad každý, kdo si chtěl alespoň vyzkoušet tvorbu počítačové grafiky. Tento program má mnoho konkurentů a některé z nich najdeme i v operačním systému Linux. Jsou to hlavně programy The Gimp, komerční Neopaint for Linux a Pixel. Tyto programy, jak už název kapitoly napovídá, slouží k tvorbě grafiky, tj. různých menu, zaoblených prvků na stránkách, lišt, firemních znaků, reklamních bannerů a složitějších log a nápisů na stránkách, které nezvládá klasické formátování textu. Proto obsahují standardně nástroje pro tvorbu čar, křivek, obrazců, jejich vyplnění barvou či texturou, grafické filtry např. pro rozmazání, zostření, rozložení do dlaždic apod. Výše popsané prvky jistě stačí pro tvorbu statické webové prezentace, ale co když chceme své stránky udělat živější? Použijeme technologii Flash, jejímž tvůrcem je společnost Macromedia. Téměř jediným použitelným programem pro tvorbu animací Flash je Macromedia Flash MX. Technologie Flash je používána pro tvorbu internetové grafiky, jednoduchých her a pro tvorbu celých dynamických internetových stránek. Její jedinou nevýhodou je hardwarová náročnost, která je ale již smazávána modernizací počítačového vybavení uživatelů internetu.

## 2.2 Wysiwyg editory

Takzvaně „What You See Is What You Get“ editory dávají možnost tvorby stránek opravdu každému, kdo je schopen používat alespoň základní funkce tohoto programu. Mezi wysiwyg editory patří například komerční Macromedia Dreamweaver či Microsoft Frontpage, které jsou oba určeny pro platformu Microsoft Windows. Jejich konkurencí a zároveň zástupci wysiwyg na platformě Linux jsou Mozilla Composer a z něj vycházející NVU (N-view). Práce s nimi se podobá práci v běžných kancelářských programech typu Microsoft Word či Open Office - Writer. Uživatel napíše požadovaný text, importuje obrázky, pomocí „editoru tabulek“ napozicuje vše tak, jak potřebuje, nastaví velikost a barvu písma, barvu pozadí a stránky jsou hotovy. Ve výsledku vypadají naprosto stejně jako vypadaly při editaci kódu. Wysiwyg editor toho ale umí ještě mnohem víc, kromě vizuální editace nabízí samozřejmě i možnost nahlédnutí do kódu a jeho úpravu a také možnost zvýraznění jak kódu tak určitých částí stránky přímo při editaci. Když je to tak jednoduché, tak proč se nepoužívají jen wysiwyg editory? Odpověď je poměrně jednoduchá - některé věci prostě wysiwyg editor nezvládne, například produkovat čistý kód a používat takzvané externí kaskádové styly (dále jen CSS) bez zásahu do kódu. Editory wysiwyg totiž používají čistý jazyk HTML, a tak se i u dosti jednoduchých stránek stává kód nečitelným a nepřehledným. Nepřehlednost kódu pak způsobuje jeho nevaliditu a ta pak špatnou interpretaci v různých prohlížečích na různých platformách. Sice se některé vizuální editory snaží CSS začít používat, ale nikdy není kód tak validní a čistý, jako když ho tvůrce stránek napíše ručně v textovém editoru.

## 2.3 Editory kódu

Editory kódu jsou preferovány spíše menšími webdesignerskými firmami či jednotlivci, kteří chtějí mít kód svých stránek neustále pod kontrolou a nespokojí se s často složitým a nepřehledným výstupem vizuálních editorů. Editory kódu nejsou jen standardními programy umožňujícími psát text, ale umožňují také zvýraznění syntaxe, doplňování některých znaků zvoleného programovacího jazyka, mají v sobě zabudovány debugery a některé obsahují například knihovnu hotových scriptů a šablon dynamických stránek. Obsahují funkce vizuálních editorů, k nim přidávají spousty svých, ale nedávají možnost pracovat jinak než s kódem. Nejoblíbenějším freewarovým editorem pro platformu Microsoft Windows je nejspíše program PSPad a pod Linuxem jsou to programy Vi a GNU Emacs.

# 3 Co nám nabízí Linuxové operační systémy?

Hned po instalaci systému máme možnost začít pracovat, protože Linuxové operační systémy obsahují hned v základu několik programů vhodných pro tvorbu webových stránek.

Především jsou to grafické programy, z nichž asi nejpokročilejší a nejvíce použitelný je program GIMP pro tvorbu bitmapové grafiky, dále pak program Sodipodi umožňující vytvářet

vektorovou grafiku a jeho nástupce Inkscape, který se již také dostává do většiny nejnovějších distribucí Linuxu. Pro úpravu kódu je v Linuxu opravdu požehnaně editorů, pro fandy klávesových zkratk a účelnosti je zde dvojice editorů Vi a GNU Emacs, které jsou asi nejoblíbenější programy pro tvorbu jakýchkoli dokumentů, programů či webových stránek skalních uživatelů Linuxu. Dále jsou zde editory dostupné podle instalovaného grafického prostředí, v Gnome je to Bluefish a v KDE je to Quanta+, která je použitelná jako editor kódu i jako vizuální editor. Pro vizuální tvorbu je pod Linuxem k dispozici také editor NVU konkurující komerčním projektům hlavně cenou, je totiž zdarma. Dalším editorem, který Linux nabízí, je editor CSSed zaměřený na editaci CSS.

## 3.1 Grafické programy

### 3.1.1 Sodipodi

Sodipodi je grafický editor pracující s vektory. Vektory dávají namalovanému grafickému prvku schopnost zvětšovat se a zmenšovat se bez ztráty kvality a větší možnost úpravy obrázku co se týče velikosti a tvaru než editory bitmapové grafiky, což je dáno použitím jiné metody vykreslování výstupu. Program pracuje s formátem SVG vytvořeným W3C, konsorciem starajícím se o standardy internetu. Tento formát využívá XML a je tak měnitelný i v obyčejném textovém editoru, díky formátu XML je také přenositelný a je možné ho teoreticky zobrazit jak v obyčejném webovém prohlížeči, tak i v prohlížeči mobilního telefonu. Protože formát ještě není moc rozšířen, umožňuje Sodipodi také export do formátu PNG.

#### 3.1.1.1 Ovládací prvky

Prostředí programu se mi na první pohled nezdálo nejšťastněji zvolené - dělí se mezi nezávislý panel a kreslicí plochu, která neobsahuje žádné ovládací prvky. To má za následek dlouhé zvykání si na umístění ovládacích prvků, které se nacházejí pouze na nezávislém panelu. To se ovšem změnilo ve chvíli, kdy v novém dokumentu klikneme pravým tlačítkem kamkoli na plochu a ve vyskočivším menu vybereme volbu Nový ukotvený panel a ten starý zavřeme. Toto rozložení už mi vyhovovalo o mnoho více, ale bylo by lepší jako výchozí. Panel se skládá z několika menu, v nichž jsou nástroje rozděleny podle funkcí. Každé menu lze od panelu oddělit a vytvořit si tak několik menších panelů.

#### 3.1.1.2 Nástroje

V menu Soubor se nachází klasická sestava nástrojů - Nový, Otevřít, Uložit, Tisk, Importovat obrázek do dokumentu a Exportovat dokument do PNG.

V kategorii Upravit je možnost dát krok Zpět, Vpřed, Kopírovat a Duplikovat vybrané objekty.

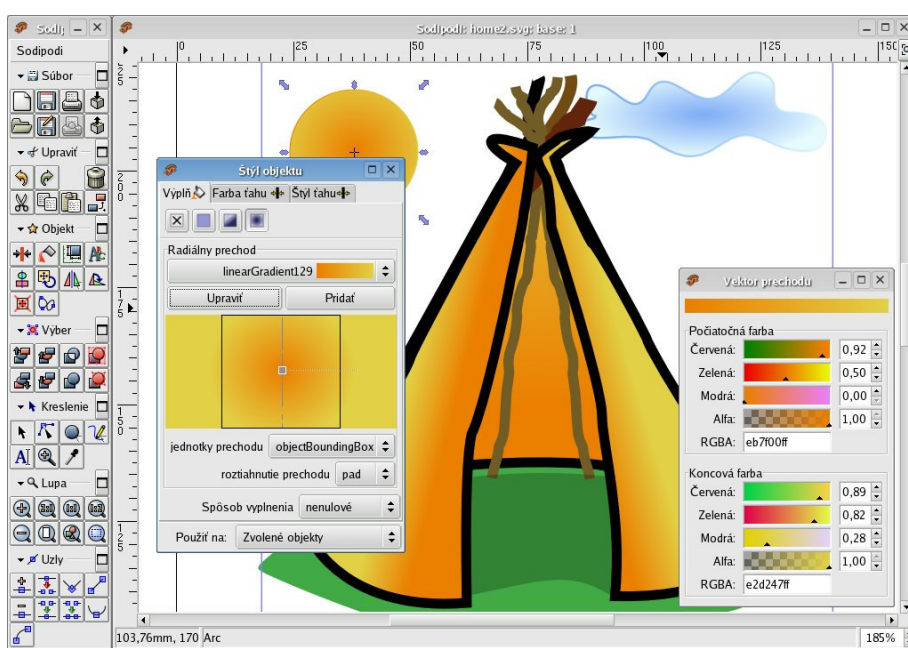
Ve skupině Objekt najdeme nástroje pro úpravu již nakreslených objektů - nastavení velikosti tahu, jeho barvy, výplně objektů, použitého písma, otáčení objektů a jejich zarovnání k ostatním.

Menu Výběr obsahuje nástroje pro práci s výběry, jejich sjednocování, rozdělování a všeobecně manipulaci s nimi.

Nejvíce používané nástroje najdeme v sekci Kreslit. Jsou zde nástroje pro úpravu objektů pomocí uzlů, kreslení geometrických objektů, čar nebo textu. Pod tlačítka Geometrické útvary a Kreslení čar jsou ukryty bližší volby. Pod prvním z nich je možnost kreslit čtyřúhelníky, zaoblit jejich hrany, vytvářet kruhy, elipsy, mnohoúhelníky hvězdy a spirály. Pod Kreslením čar najdeme Kreslení od ruky, Rovné čáry, Křivky a Kaligrafické čáry.

Menu Zvětšit obsahuje nástroje pro zvětšování a zmenšování, respektive přibližování a vzdalování obrázku.

Poslední sekci jsou Uzly, všechny nástroje v tomto menu jsou určeny pro práci s uzly, jejich spojování, vytváření a jejich úpravu.



Obrázek 2: Vektorový editor Sodipodi v akci.

## 3.1.2 Inkscape

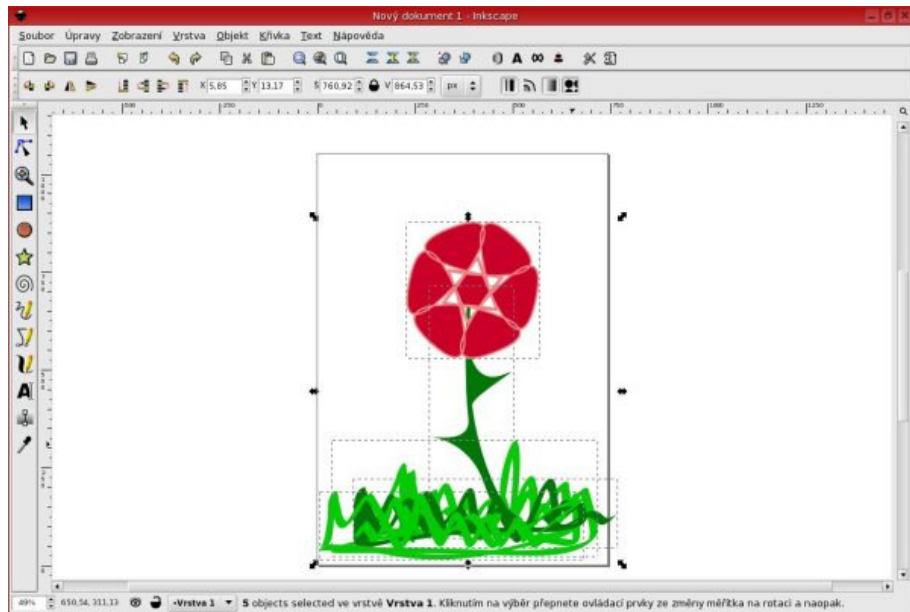
Inkscape je vektorový grafický editor vycházející ze Sodipodi. Oproti Sodipodi je ale natně vylepšený, má několik nových praktických funkcí a přepracované, uživatelsky přívětivé ovládání. V době psaní této práce je ve verzi 0.43, což není definitivní, ale přesto zcela funkční verze programu. Program je dostupný pro všechny platformy, tedy nejen pro Linux. Stejně jako jeho předchůdce používá formáty SVG a pro export PNG.

### 3.1.2.1 Ovládací prvky

Ovládací prvky Inkscape jsou na rozdíl od Sodipodi umístěny po levé straně a na horním panelu hlavního okna programu, což je pohodlné a intuitivní. Uživatel je na toto rozložení zvyklý z ostatních programů, ať už grafických či kancelářských. Ikony umístěné nalevo zastupují nástroje velice podobné programu Sodipodi, tj. nástroje pro základní kreslení, různé tvary (čtverec, hvězda, elipsa), Beziérovky křivky, ovládání uzlů a barevných přechodů a mimo jiné kaligrafické pero, které se mi zdá, ač vychází ze stejného principu jako u Sodipodi, lépe zpracované. V horní liště jsou tradičně umístěny volby Kopírování, Ukládání, Vpřed, Zpět, několik nástrojů pro přizpůsobení obrázku, zarovnání jeho částí a správu barev a editor XML, ve kterém se dají veškeré grafické prvky upravit v jazyce XML. Pod těmito volbami se vždy podle zvoleného nástroje objeví jeho nastavení - síla čáry, sklon pera, zaoblení apod. Kromě těchto ikonami reprezentovaných nastavení se zde nacházejí tradiční rozbalovací menu obsahující pokročilejší volby.

### 3.1.2.2 Nástroje

Nástroje jsou klasické, až na několik výjimek jsou stejné jako u Sodipodi. Rovné čáry a křivky jsou tvořeny buď od ruky, nebo pomocí rovných čar a jsou upravitelné pomocí uzlů. Kružnice lze tvořit jak celé, tak jenom jejich části. Čtverec či Obdélník umožňuje tvořit i útvary se zaoblenými hranami. Hvězda, jak název napovídá, nám umožňuje vytvářet rozličné hvězdy pravidelného tvaru a pomocí nastavení zaoblení i roztodivné, ale efektní tvary připomínající výšivky, sněhové vločky, květiny nebo keltské vzory. Spirála je nástroj velice zvláštní, ale co se týče možností využití, řekl bych, zbytečný, protože je velice málo nastavitelný a dovoluje tvořit právě jenom spirály. U Kaligrafického pera bych se rád zastavil trochu blíže, protože je zde nejlépe zpracované, alespoň co se týče mně známých programů. Vektorové programy řeší kresbu kaligrafickým perem buď kresbou dvou linek současně, kdy prostor mezi nimi je vyplněn barvou (Inkscape, Sodipodi), nebo pomocí různých štětců (Expression od Microsoftu, který sice není pro Linux, ale srovnání jsem si vybral proto, že je stejně jako Inkscape ke stažení zdarma.) . Řešení, které nabízí Inkscape, se mi zdá velice použitelné a v několika ohledech lepší než použití štětců, například je možné do již vytvořené kaligrafie dále zasahovat a upravovat její tvar, aniž by bylo nutné použít další tahy perem, můžeme vytvářet například trny na stonku růže, různé boule a výdutě, vše jen posouváním stávajících bodů.



Obrázek 3: Vektorový program Inkscape - ukázka využití kaligrafického pera a nástroje Hvězda.

### 3.1.3 Gimp

Gimp je grafický program funkcemi snad nejvíce podobný Adobe Photoshopu, zvláště pokud použijeme jeho verzi zvanou GIMPshop. Je velice vhodný právě pro tvorbu webové grafiky, tlačítek, složitějších vzorů, pro úpravu fotografií a zvládá i tvorbu animací. Jako jeho hlavní formát slouží XCF, ale dokáže pracovat téměř z jakýmkoli formátem včetně pracovních formátů konkurenčních programů.

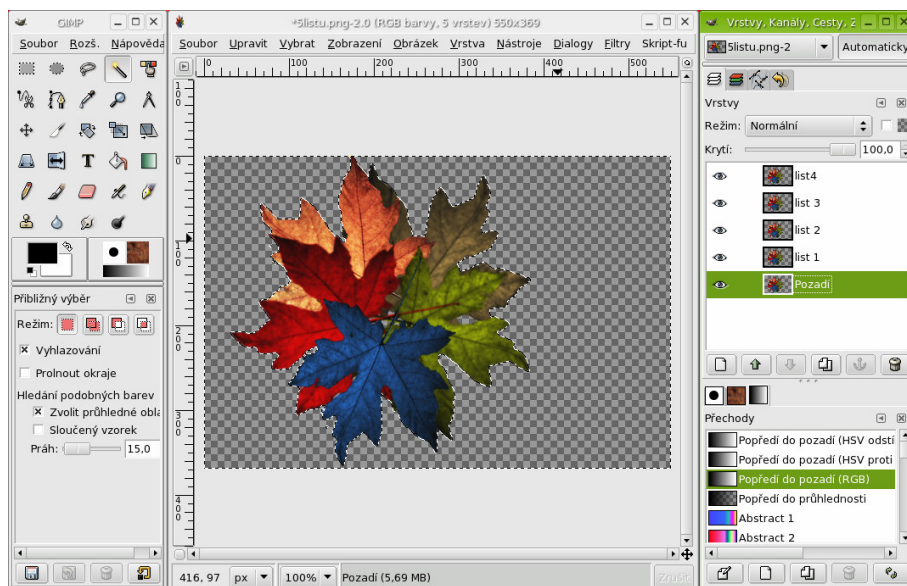
#### 3.1.3.1 Ovládací prvky

Gimp je rozdělen do několika nezávislých oken, na což si uživatel musí chvíli zvykat, ale poté se stává rozložení oken velice intuitivním a praktickým.

Jedno celé okno obsahuje pouze nástroje reprezentované ikonami a jejich nastavení a doladění. Jsou zde základní nástroje pro manipulaci s dokumentem - Otevření, Vytvoření nového dokumentu, Ořezávání, Tisk, Uložení. K dispozici jsou různé druhy výběrů oblastí - Obdélníkový, Eliptický, Ruční (laso), Spojitých oblastí (magická hůlka), podle barvy, inteligentní nůžky a Beziérovovy křivky. Dále nástroje pro přemístění, přiblížení, rozřezání, rotaci, zvětšení, změnu perspektivy či překlopení obrázku a nakonec nástroje kreslení - Tužka, Štětec, Pero, Razítko, Rozprašovač, Výplň barvou či vzorem, Guma, Rozmazávač, Gradient, Výběr barvy, Pipeta, Text. Druhé, hlavní okno je téměř celé tvořeno plochou ke kreslení či úpravě obrázku. Kromě této plochy obsahuje i měřicí lišty a hlavně několik rozbalovacích menu - Soubor, Úpravy, Výběr, Zobrazení, Obrázek, Vrstva, Nástroje, Dialogy, Filtry a Script-Fu. Třetí okno zobrazuje vrstvy, palety, zmenšený obrázek a jiné detaily nástrojů. Toto okno na rozdíl od ostatních dvou při práci nemusíme vůbec použít, případně jej lze rozdělit, konfigurovat a sestavit tak, aby odpovídalo představě uživatele.

### 3.1.4 Nástroje

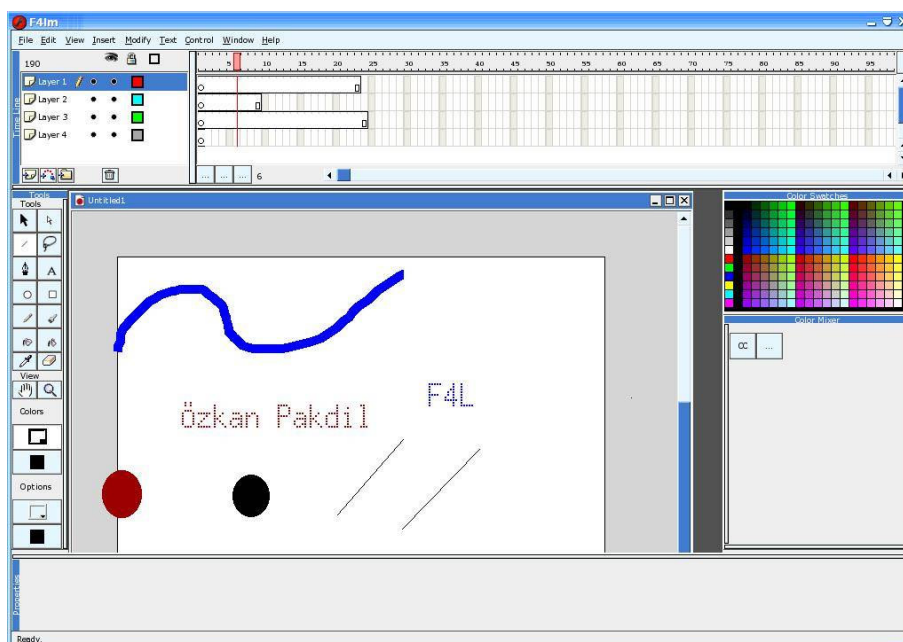
Gimp obsahuje obrovské množství různých nástrojů, jednoduché pro základní kreslení jako jsou Štětec a Tužka s velkým množstvím různých stop od normálních přes kaligrafické, až k různým předvoleným objektům. Stopy lze samozřejmě doplňovat, a to buď stahováním s Internetu, nebo jejich vlastním tvořením. Pokud budeme v programu Gimp kreslit, budou tyto dva nástroje asi nejčastěji používány. Uživatele jiných programů by mohla zmást absence nástrojů pro kreslení tvarů - kruhu, čtverce apod. Ty jsou zde nahrazeny výběrem, který lze bez problémů vyplnit či obtáhnout zvolenou stopou. Vzhledem k bohaté možnosti výběrů v programu Gimp je kreslení výběrem velice použitelné a praktické. Pokud bychom chtěli kreslit složitější obrázky, nevystačíme si pouze se základními čarami. Pro tyto případy jsou zde nástroje Rozmazávání, Beziérový křivky a Razítko. Rozmazání zprůměruje barvy kolem rozmazávaného místa a provede efekt podobný tahu prstem v čerstvé malbě. Beziérový křivky jsou známé z vektorových editorů, jsou to křivky upravitelné za pomoci uzlů. Nástroj Razítko umožní odebrat vzorek nějaké oblasti v obrázku a pak s ní zacházet jako s reálným razítkem. Kromě běžných nástrojů jsou zde také tzv. Filtry. Filtry umí obrázek přizpůsobit tak, aby vypadal například jako malba olejovými barvami, jako by byl namalován na plátně, umí upravit barvy, rozmazat či zostřit obraz. Použitím filtrů lze upravit obrázek doslova k nepoznání. K vytváření pokročilé grafiky je výborná znalost filtrů nutností. Gimp obsahuje filtrů opravdu nespočet a stejně jako u stop štětců je možné je stahovat z Internetových stránek. Dalším nástrojem je Script-Fu. Scripty-Fu jsou napsané v programovacím jazyku Scheme, založeném na jazyku Lisp. Tyto Scripty fungují jako další Filtry, ale dokáží i obrázky přímo vytvářet pomocí zadaných parametrů.



Obrázek 4: Gimp - Použití vrstev.

### 3.1.5 F4L

F4L znamená Flash For Linux a je to zatím jediný program pro Linux, který by měl pracovat ve větší míře s technologií Flash. Ostatní programy sice mají jisté možnosti, např. tvorba tlačítek v Open Office org či Gimpu, ale kvalit editoru zaměřeného přímo na Flash nemohou nikdy dosáhnout. Program F4L je možností jak tvořit soubory swf, flashové bannery, menu a animace i pod Linuxem. Tedy byl by, kdyby byl dokončen. Program začal být vyvíjen v roce 2003, bohužel zůstal ve velice rané verzi a vývoj se zpomalil, ne-li úplně zastavil. I ve své verzi 0.2.1, která je zatím poslední, umí využívat programovací jazyk Flashe - Action script, přesto je pro tvorbu flashových animací prozatím nepoužitelný. Jeho menu a nabídky sice obsahují všechny funkce, které by webdesigner potřeboval, ale většina tlačítek je mrtvých a nepoužitelných. Prozatím je proto F4L jen horším vektorovým editorem a bohužel jím asi i zůstane.



Obrázek 5: Skromné možnosti editoru F4L se bohužel nejspíše už nerozšíří.

## 3.2 Editory kódu

### 3.2.1 Quanta+

HTML editor určený pro KDE (což pochopitelně nevyklučuje jeho použití s jiným správcem oken, pokud jsou v systému přítomny patřičné knihovny) je součástí téměř každé distribuce Linuxu. (výjimkou jsou samozřejmě distribuce neobsahující KDE)

Uživatelské rozhraní Quanta je složené ze dvou částí. Tím nejdůležitějším je samozřejmě editor - okno, kam píšeme text. Okno v levé části zpřístupňuje adresáře a soubory, projekty, strukturu dokumentu a nápovědu. Poměr rozdělení pracovní plochy má samozřejmě uživatel plně pod kontrolou.

Vlastní editor je totožný s editorem kwrite, má několik základních nastavení, která ovlivňují automatické odsazování, chování bloků, zalamování řádků apod. Další funkcí je zvýrazňování syntaxe. Editor podporuje několik různých jazyků (od SQL přes C či PHP, až po standardní CSS a HTML). Editor dále podporuje současnou práci s více soubory, které je možné přepínat pomocí záložek. Kromě těchto základních funkcí, jež má téměř každý linuxový textový editor, jsou zde lišty s nástroji pro tvorbu tabulek, seznamů, rámců, formulářových prvků, odkazů, prvků textu a všeobecně html znaků.

V Quantě lze Internetové stránky tvořit jak psaním kódu, tak pomocí vizuální tvorby, ale ta zde není až tak uživatelsky přívětivá. K tvorbě lze použít režim vizuální editor nebo kombinovaný editor, kde lze při vizuální tvorbě kontrolovat kód výstupu či obráceně - psát v kódu a používat vizuální editor pro kontrolu výstupu. Kromě těchto dvou režimů je zde také režim textový a režim náhled, ve kterém nelze stránku upravovat, ale je zobrazena přesně jako v prohlížeči webových stránek. Chování editoru je nastavitelné pomocí rozbalitelných menu v horní liště programu, těchto menu je třináct - Soubor, Úpravy, Pohled, Záložky, Projekt, Nástrojové lišty, DTD, Značky, Moduly, Nástroje, Okno, Nastavení, Nápověda.

Soubor obsahuje volby pro načítání dokumentů a projektů, pro jejich ukládání, tisknutí a vytváření nových stránek.

V Úpravách jsou volby pro kopírování, vkládání, vpřed a zpět, funkce hledání v textu a nahrazování slov.

Pohled umožňuje nejen přepínat mezi jednotlivými druhy editace, ale obsahuje i volby lomu písma a zobrazení či skrytí některých panelů, které zrovna nejsou k práci potřeba.

Záložky umožní uživateli si na stránce označit jakoukoli oblast a k té se pak pohodlně vrátit kdykoli bude chtít. Oblastí, které takto označíme, může být samozřejmě víc.

Možnost Projekt umožňuje vytvářet celé skupiny stránek, obrázků, dynamického obsahu tak, aby byly přehledně roztrženy do složek, k nimž by byl snadný přístup. Umožňuje celé takto vytvořené projekty udržovat pohromadě v jedné složce, případně na FTP serveru.

Nástrojové lišty - tato volba je velice užitečná pro programátory, kteří si mohou vytvořit svoji vlastní lištu nejpoužívanějších nástrojů přímo pro určitý programovací jazyk, tu archivovat či ji poslat kolegovi emailem. Je zde také možnost stažení nástrojových lišt z Internetu.

DTD je menu umožňující změnit či nastavit typ dokumentu, který budeme vytvářet.

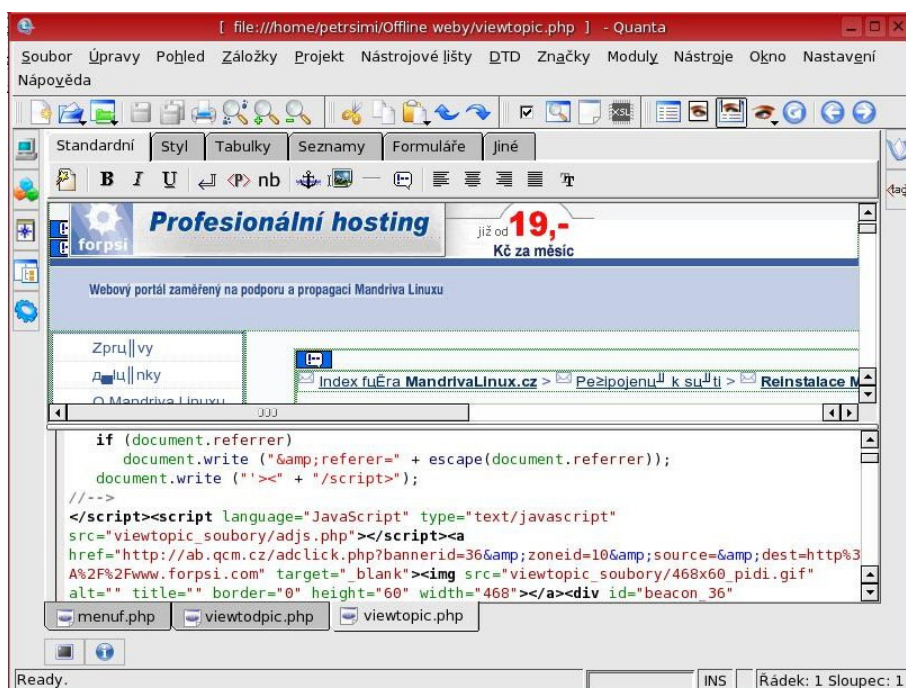
V menu Značky jsou rozříděny podle použití HTML tagy a to na Standardní, Styl, Tabulky, Seznamy, Formuláře a Jiné. Mezi standardní značky patří například vložení obrázku, zarovnání písma či zalomení textu. Styl obsahuje značky, které lze ovlivnit pomocí CSS, tj. div, span, nadpisy a také barevnou paletu, s jejíž pomocí můžeme ostylované prvky obarvit. Tabulky, Seznamy a Formuláře obsahují základní tagy, s jejichž pomocí lze výše zmiňované prvky stránky vytvářet. V podmenu Jiné jsou značky, které nespádají do žádné z kategorií, je zde odkaz na e-mail, speciální tagy, vložení aktuálního času, průvodce rámci, vložení CSS stylu nástroj pro tvoření odkazů.

V Modulech jsou nainstalovány různé moduly, například pro kontrolu funkčnosti odkazů, kontrolu validity kódu či hledače chyb v programovacím jazyku.

Menu Nástroje obsahuje veliké množství nástrojů pro zvýraznění syntaxe, kontrolu pravopisu a nastavení textu. Pro zvýraznění syntaxe máme na výběr přes sto deset možností, a to v oblasti skriptů, programovacích jazyků a databází, ale jsou zde i některé konfigurační jazyky linuxových programů. Co se týče nastavení textu je zde nastavení kódování, jichž je na výběr kolem čtyřiceti, styl odsazování, ukončení řádku, změna řádku na komentář a zpět, převod z malých písmen na velká a obráceně, nastavení zalamování slova a nástroj pro nastavení celé kostry HTML dokumentu, vložení stylu CSS a META tagů.

Další rozbalitelné menu Okno zahrnuje nastavení uživatelského rozhraní a lze si zde vybrat podle pohodlí ze čtyř vzhledů.

Poslední z menu, nepočítáme-li nápovědu, která obsahuje informace o Quantě, uživatelskou příručku a odkaz na webové stránky Quanty+, je menu Nastavení. To zahrnuje veškerá nastavení editoru pohromadě. Nastavení klávesových zkratk, zvýraznění, kde můžeme u každého jazyka vybrat jaká z jeho částí bude zvýrazněna jakou barvou, výchozí nastavení pohledu, nastavení kontroly pravopisu, pravidel ukládání. Je zde několik podmenu, kde lze nastavit nástrojové lišty a klávesové zkratky.

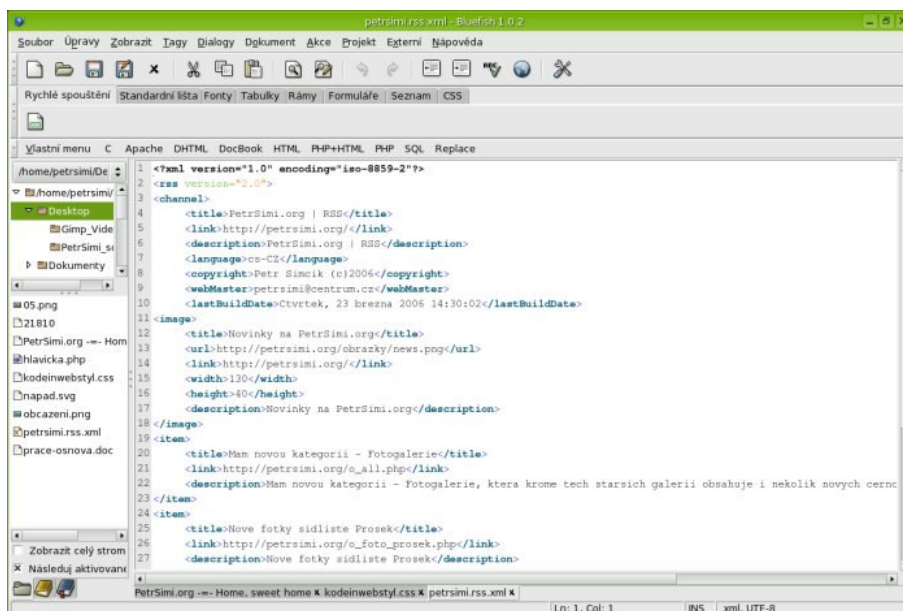


Obrázek 6: Quanta+ v režimu kombinovaného editoru.

## 3.2.2 Bluefish

Bluefish je textový editor, který lze nasadit v jakémkoli grafickém prostředí. Jeho výhodou je, že není závislý na knihovnách některého z grafických prostředí, a tak je jeho instalace velice snadná a téměř nehrozí, že by bylo nutné instalovat ještě nějaké podpůrné balíčky. Editor Bluefish není co se týče tvorby webových stránek tak všestranný jako Quanta, a tak většinou nepatří k základnímu vybavení většiny distribucí. Je to editor textu, a tak se zde uživatel na rozdíl od Quanty neseťká s vizuální tvorbou webových stránek.

Uživatelské rozhraní editoru Bluefish je děleno do tří částí, což lze ale samozřejmě změnit. Výchozí nastavení obsahuje jednu postranní lištu, ve které lze procházet adresáře a vyhledávat editovatelné soubory, nástrojovou lištu s hlavním menu a pruhem, který obsahuje hlavně tlačítka pro rychlé vkládání tagů HTML a hlavní okno, kde budeme při používání programu trávit většinu času. Nástrojová lišta je plně přizpůsobená potřebám webdesignera. Obsahuje tlačítka pro vložení všech myslitelných částí webové stránky, formátování textu a rozbalovací menu s některými nejpoužívanějšími funkcemi PHP, DHTML a dalších webových programovacích jazyků. Ač je editor Bluefish určen pro tvorbu webových stránek, velice dobře zvládá i jiné jazyky než jsou základní HTML, CSS a PHP. Velkou předností je pro mě u editoru Bluefish možnost změnit kódování textu, což mi velice pomohlo při přechodu na Linux. Kromě změny kódování je zde ještě několik velice užitečných funkcí, např. kontrola pravopisu a možnost zobrazení webu v některém z externích prohlížečů přímo při editaci. Stejně jako u Quanty máme možnost vytvářet celé projekty skládající se z více stran nebo scriptů a společně je ukládat a editovat. Editor Bluefish nabízí ty samé funkce co ostatní editory a přitom vyniká přehledností a použitelností.

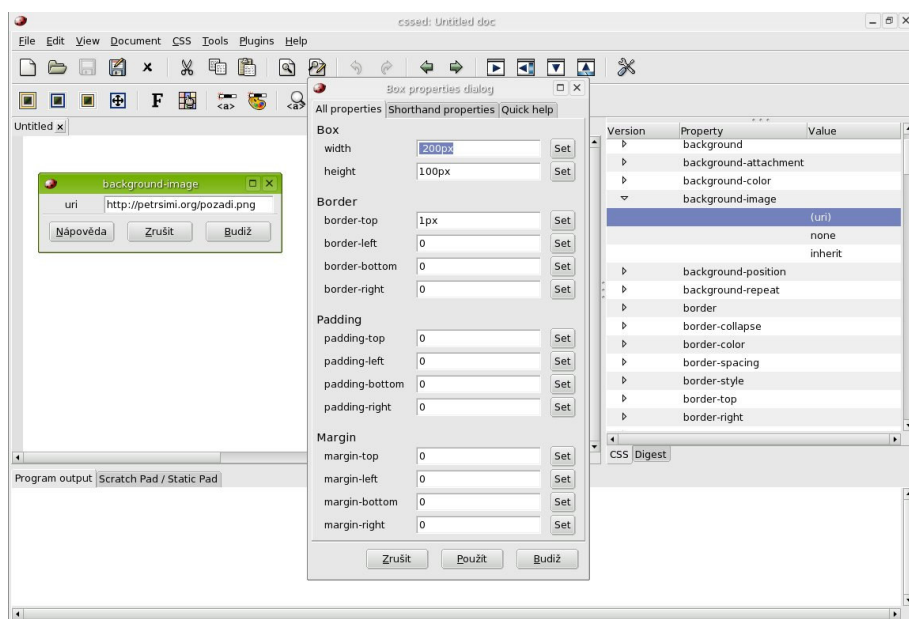


Obrázek 7: Bluefish při editaci XML.

### 3.2.3 CSSed

CSSed je velice povedený editor CSS, který je vynikající pomůckou pro začínající tvůrce stylovaných stránek. Prozatím je ve verzi 0.4.0, která zdaleka není úplná, a proto obsahuje spoustu chybek, většinou v ovládní, ale přesto je zcela dostačující pro tvorbu jakkoli náročných kaskádových stylů.

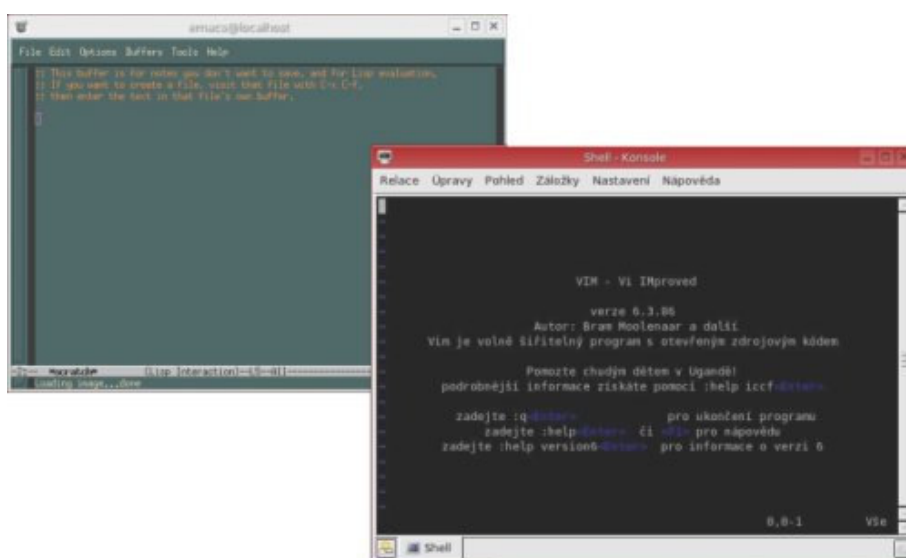
Uživatelské rozhraní je pro program tohoto typu standardní. Celé okno se dělí na 3 části, jedna obsahuje kód, druhá jednotlivé položky stylu a poslední, spodní, slouží k zobrazení výstupu programu. Pomocí jednotlivých položek můžeme postupně sestavovat styl a definovat ke každé položce hodnoty. V horní liště programu je umístěno několik ikonek, které provedou uživatele atributy border, margin a padding, případně hned celým stylem blokového prvku. Je zde možnost vložit ID či CLASS, jednoduše nastavit písmo, atributy i barvu každého prvku stránky. CSSed je i ve své rané verzi velice užitečným pomocníkem při editaci CSS. Obsahuje dokonce i validátor CSS, který hned zkontroluje funkčnost vytvořeného stylu. Kromě CSS může CSSed zvýrazňovat i ostatní jazyky, např. HTML, PHP, Python, ale nemá pro jejich editaci žádné nástroje, takže se hodí spíše k malým opravám než k programování v některém z nich. Samozřejmostí je možnost otevřít více souborů pomocí záložek, což je u Linuxových programů dobrým zvykem.



Obrázek 8: Vkládání obrázku do stylu CSS pomocí CSSed.

## 3.2.4 Vim a Emacs

Tyto dva programy jsou pro někoho, kdo Linux nikdy nepoužíval, jako z jiného světa. Jejich základní verze jsou totiž dostupné jen v textovém režimu, případně v konzoli (Okno z textovým režimem dostupné v grafickém prostředí.), a neobsahují žádné viditelné ovládací prvky. Tím uvádí většinu neznalých lidí v přesvědčení, že jsou to programy zastaralé a nic neumí, opak je ale pravdou. Většina uživatelů Linuxu a jiných Unixových systémů si během svého užívání operačního systému vybere právě jeden z těchto editorů, a to nejen ke tvorbě internetových stránek, ale například i ke psaní mailů, dokumentů, programování a spoustě činností, která se s těmito programy po jejich zvládnutí stává mnohem rychlejší a efektivnější než při používání jednoho programu s grafickou nadstavbou pro každou jednotlivou činnost.



Obrázek 9: Textové editory Emacs (vlevo) a Vim (vpravo).

### 3.2.4.1 Vi, Vim

#### 3.2.4.1.1 Představení a historie

V roce 1988 napsal Bram Moolenaar obdobu editoru Vi pro Amigu. Amiga v té době nedisponovala žádným použitelným editorem. Tento program byl založen na kódu editoru Vi SteVle, proto byl pojmenován Vi Imitation.

Roku 1991 byla uveřejněna první veřejná verze tohoto editoru - Vim 1.14. Tato verze byla stále dostupná pouze pro Amigu, na UNIX byla přenesena až verze 1.22, kdy došlo k celé řadě vylepšení a editor byl přejmenován na Vi Improved. Další přelomovou verzí byla verze 3 (1994), kdy se objevila podpora více oken, a verze 4 z roku 1996, kdy byla do editoru zapracována možnost grafického rozhraní. Další verzí byla verze 5 (1998), kde největší změnou bylo zvýrazňování syntaxe. Nyní 2005 je poslední verzí verze 6 (z roku 2001). Roku 2002 byla změněna licence směrem k licencím typu GPL, což umožnilo využít Vim jako základní editor v některých linuxových distribucích. Tato změna licence byla umožněna faktem, že z původního kódu, který Bram Moolenaar využil, již ve Vimě nezůstalo nic. Program vi je jednoduchý, malý a umožňuje přehlednou

práci s textem. Jelikož je to program jednoduchý a nezvládá většinu pokročilejších funkcí jako je např. kontrola pravopisu, zarovnávání textu apod., je zde možnost spolupracovat s ostatními programy, které lze pomocí editoru lehce vyvolat, a tak poskytné uživateli veškeré funkce, na které si vzpomene. Vim má možnosti pluginů, rozšíření. Těch jsou již dnes stovky a přidávají mu obrovské schopnosti pro editace textů a nejen to. Obzvláště programátoři ocení automatické vytvoření programově zdrojové hlavičky dokumentu při vytvoření nového zdrojového souboru v různých jazycích. K jejich určení používá Vim autodetekci. Pro psaní těchto rozšíření Vim používá svůj vlastní skriptovací vnitřní jazyk. Pluginy je možné psát také jako přenositelné binární soubory v jazyku C nebo C++.

### 3.2.4.1.2 Funkce a ovládání

Jelikož editor vi a vim (což je jeho pohodlnější a novější klon.) pracuje pouze v textovém režimu a nemá viditelné ovládací prvky, neobejdeme se při jeho ovládání bez klávesových příkazů (obvykle zadaných jediným písmenem) a několika pracovních režimů. Tyto režimy jsou tři : příkazový, editační a režim ex.

Po spuštění programu se uživatel nachází v příkazovém režimu, který umožňuje měnit text pomocí příkazů. Do editačního režimu se můžeme přesunout příkazem „i“ a zpět do režimu příkazů pomocí klávesy Esc.

Editační režim slouží k hlavní činnosti textového editoru, tzn. vkládání a úprava textu. K pohybu po stránce můžeme používat klávesy pohybu nebo klávesy h,j,k a l. Kromě přímého použití editačního režimu lze vkládat text pouze za momentální pozici kurzoru příkazem „a“ nebo na nový řádek použitím příkazu „o“. Mazat text můžeme také i jinak než v editačním režimu - buď opět za kurzorem pomocí příkazu „x“, nebo můžeme smazat celý řádek pomocí „dd“, případně použít příkaz „dw“ pro mazání slova, které se právě nachází pod kurzorem. Pokud bychom náhodou smazali to, co jsme neměli, můžeme pomocí příkazů „p“ nebo „P“ text zpět vyvolat, případně můžeme vrátit poslední akci příkazem „u“. Chceme-li text nahradit, můžeme to udělat pomocí příkazu „R“, pokud místo toho použijeme „cw“, tak se po ukončení vkládání odstraní zbytek původního slova. Příkazem „-“ donutíme text změnit malá písmena na velká.

Editor, který se musí obejít bez podpory myši, nutně potřebuje nějaké prostředky pro pohyb po stránce. Kromě posunu o jedno pole, který není u větších textů příliš vyhovující, je zde také posun na začátek slova pomocí „b“, na začátek následujícího slova pomocí „w“ a také na začátek řádku díky „0“ a na konec řádku díky „\$“. Při práci s dlouhými soubory je zde možnost posunu o stranu dopředu příkazem „Ctrl-F“ a o stranu dozadu pomocí „Ctrl-B“. Pokud bychom se chtěli přesunout na určitý řádek souboru, stačí k tomu číslo řádku a příkaz „G“. Velice výhodné jsou v tomto směru také příkazy „/“ a „?“, které přesunou kurzor na slovo, jež napíšeme za příkazem. Příkazy lze také spojovat, takže například „d\$“ smaže vše od kurzoru do konce řádku.

Zatím jsem se nezmínil o režimu ex. Režim ex slouží k práci se soubory v editoru vi a vyvolá se příkazem „:“ z příkazového režimu. Např. příkazem „:w“ v příkazovém režimu nejdříve zavedeme režim ex a poté uložíme soubor. Pokud bychom přidali k příkazu ještě „q“, tak celý

editor vim ukončíme. Pokud bychom si přáli neukládat soubor a pouze editor ukončit, potřebujeme příkaz „:q!“

Samotné „:q“ by nám k tomu stačilo pouze v případě, že byl soubor uložen. Kromě těchto možností je tu i možnost načíst jiný soubor příkazem „:e jméno\_souboru“, případně použít „!“, pokud jsme stávající soubor neuložili a nechceme tak učinit. Jestliže by bylo nutné vložit obsah jednoho souboru do druhého, lze tak učinit příkazem „:r jméno\_souboru“.

Pokud se chceme podívat, kde náš soubor leží, můžeme tak učinit buď přímo z editoru příkazem „:!“, který nám umožní zadávat příkazy shellu (příkazové řádky v Linuxu.), nebo ho pozastavit klávesovou zkratkou „Ctrl-z“.

Editor vim je samozřejmě schopný provádět daleko více funkcí než jsou ty, které jsem vyjmenoval. Tyto funkce jsou většinou přístupné pomocí kombinace již uvedených příkazů. Např. kopírování či přesouvání textu lze provést pomocí příkazu „dd“, který smaže řádek, a příkazem „P“ smazaný řádek umístíme někam jinam. To samé lze udělat příkazem „yy“, který ale řádek jen zkopíruje a nesmaže. Ke kopírování také můžeme používat systém značek, který původně jen umožňoval lepší orientaci v dokumentu. Značku umístíme příkazem „ma“ pro značku „a“, jinak samozřejmě jdou použít všechna písmena abecedy, jak malá tak velká. Pokud ji budeme chtít vyhledat použijeme příkaz „a“. Pokud tedy budeme chtít text kopírovat, tak se po umístění značky přesuneme na konec kopírovaného textu a použijeme „y’a“. Kdybychom místo kopírování chtěli přesouvat, použijeme „d’a“. Nejvhodnějším způsobem jak ve vimu kopírovat a přenášet texty, jsou tzv. Registry. Registr je prostor pro ukládání textu, který si uživatel přeje kopírovat či přenášet mezi částmi dokumentu. Registr vyvoláme např. „a (pro registr a)“, takže např. „aay“ zkopíruje řádek do registru „a“ a příkaz „ap“ ho zkopíruje na novou pozici kurzoru. Pomocí registru lze přenášet i text mezi soubory. Pokud bychom chtěli nějaké části textu nahrazovat, můžeme použít takzvané indikátory. Indikátor „g“ nahradí každé slovo, které se shoduje se vzorem, a indikátor „c“ se na každé slovo před jeho nahrazením zeptá.

Tímto končí výčet schopností samotného editoru, ale kdyby nám něco chybělo, není problém použít příkaz „:x,y! náš\_příkaz“, který spustí externí program a provede jeho efekt mezi řádkem x a řádkem y. Například můžeme napsat program pro jazyk Perl, který pak pomocí takto spuštěného překladače spustíme. Další nastavení editoru vi je možné pomocí příkazu „:set“ nebo nastavením souboru .vimrc v našem domovském adresáři.

## 3.2.4.2 Editor Emacs

### 3.2.4.2.1 Představení a historie

Emacs je dalším editorem dostupným v textovém režimu, ale na rozdíl od programu vim je to velice rozsáhlá aplikace, téměř by se dalo říci, že integrované prostředí. Emacs je textový editor, jehož počátky sahají do 80. let. Do dnešního dne se stále vyvíjí a vydávají se nové verze. Emacs je populární pro svou přizpůsobitelnost a rozšiřitelnost; obsahuje interpret programovacího jazyka LISP, pomocí kterého je možné vytvářet moduly s novými funkcemi. Existují rozšíření, která umožňují používat Emacs jako internetový prohlížeč, e-mailový klient apod. Emacs je součástí

projektu GNU, je šířen pod licencí GNU GPL. Existují verze pro mnoho platforem včetně GNU/Linuxu, Microsoft Windows, MS-DOS, Mac OS X atd. Název Emacs vznikl jako zkratka pro Editor MACroS („editorová makra“), neboť se původně jednalo pouze o sbírku maker pro editor TECO. V rámci humoru se tato zkratka vykládá i různými jinými způsoby (např. „eight megabytes and constantly swapping“, „osm megabytů a pořád swapuje“, což naráží na poměrně vysoké systémové nároky Emacsu, přinejmenším na dobu jeho vzniku). Emacs v sobě integruje vnitřní možnost modulů a k jejich psaní používá interpret skriptovacího jazyka LISP. Díky němu je možné napsat vlastní rozšíření, a tím podpořit všestrannost editoru. Většina nových distribucí obsahuje dvě verze editoru, jednu pro textový režim, která je rychlejší, ale pomaleji vyvíjená, a verzi pro grafický režim, která je o něco rozsáhlejší, uživatelsky přívětivá, ale také pomalejší.

### 3.2.4.2.2 Funkce a ovládání

I když je Emacs rozsáhlejší a modernější než vim, neobejdeme se ani zde bez klávesových zkratk. Vytvoření nového souboru je v Emacsu možné pomocí zkratk „Ctrl-x“ „a Ctrl-f“, při stisku kláves se dole na obrazovce objeví nápověda s aktuálním adresářem, následně můžeme vytvořit buffer, kam se vloží obsah nového souboru. Editor Emacs obsahuje mnoho editačních režimů, např: pro ASCII, sázecí systém TEX, programovací jazyk C apod. Každý z režimů má potom specifické vazby mezi řídicími klávesami a jim odpovídajícími příkazy. Emacs si režim určuje většinou sám podle rozšíření u jména souboru. Pokud je Emacs spuštěn v grafickém prostředí, otevře se navíc ještě jedno okno obsahující nabídky, posuvníky a další prvky.

Práce v Emacsu je při základní editaci textu mnohem intuitivnější než ve vimu, kurzor se pohybuje pomocí šipek či kombinace „Ctrl“ a kláves p, n, f a b. V editoru Emacs je ale samozřejmost, že si tyto funkce může uživatel přenastavit, což platí téměř u veškerého ovládání editoru. Kromě těchto klávesových zkratk jsou tu ty, které se pohybují po řádcích „Ctrl-a“ a „Ctrl-e“, a ty, které umožňují skákat po stránkách „Ctrl-v“ a „Alt-v“.

Pro ukončení práce v editoru je třeba použít kombinaci „Ctrl-x Ctrl-c“, v tomto případě jde o takzvaný předřazený příkaz, kdy „Ctrl-x“ může sloužit i jako předpříkaz jiné zkratky, například uložení pomocí „Ctrl-s“ nebo „Ctrl-f“ pro výměnu editovaného souboru za jiný.

Po stisknutí „Ctrl-f“ vyskočí okno, ve kterém se můžeme probírat soubory a najít ten, který chceme editovat, po jeho vybrání okno zase zmizí. Editor takto umí otevřít i více editovaných souborů najednou, mezi nimiž lze potom přepínat pomocí zkratky „Ctrl-x b“. Pokud si uživatel nepamatuje názvy všech přepínaných oken, je zde zkratka „Ctrl-x Ctrl-b“, která všechna okna zobrazí v přehledném seznamu, mezi seznamem a ostatními okny lze pak přepínat pomocí „Ctrl-x o“. Seznam lze také definitivně zavřít a to pomocí „Ctrl-x k“.

Jelikož se ze začátku ovládání programu zdá více než složité a autor s tím počítal, je pro Emacs dostupná obsáhlá nápověda i s výukovým programem. To vše je skryto pod zkratkou „Ctrl-h“ a „Ctrl-h t“. Kromě samotného výukového programu je zde také informační systém, kde lze nalézt další podrobnosti k Emacsu. Ten se vyvolává zkratkou „Vtrl-h i“ a zobrazí seznam témat, mezi nimiž si lze vybrat klávesou „m“ a vypsáním názvu tématu, pokud bychom chtěli prohlížet všechna témata stačí zmáčknout mezerník. Mezi tématy se dá pohybovat pomocí kláves „u“, „p“ a „l“. „u“

nás přemístí do nadřazeného, „p“ do předcházejícího a „l“ do naposledy prohlíženého tématu. Po stisknutí klávesy „h“ je dostupná nápověda k informačnímu systému. Jelikož jsou nápověda a informační systém zobrazovány přímo v editoru, lze se v ní pohybovat běžnými příkazy pro pohyb v textu.

Stejně jako vim i Emacs používá systém značek a registrů. Umístit značku můžeme pomocí „Ctrl-@“, případně pomocí „Ctrl-Space“, text mezi kurzorem a značkou se poté nazývá oblast. Pokud bychom chtěli použít registr, musíme ještě zmáčknout „Ctrl-x“ a jméno registru. S oblastí potom může libovolně operovat, například ji vymazat a umístit do tzv. Kill ring, což je místo, kam se umísťuje všechen smazaný text pomocí „Ctrl-w“. Pokud bychom nechtěli smazat text, ale pouze zkopírovat do Kill ringu, musíme místo „Ctrl-w“ použít „Alt-w“. Smazanou oblast pak můžeme pomocí „Ctrl-y“ přemístit na jiné místo v textu. Výhodou používání Kill ringu je to, že se takto můžeme při kopírování vrátit i několik bloků nazpátek.

Pokud bychom dali při kopírování přednost registrům, musíme nejdříve nastavit značku pomocí „Ctrl-space“ nebo „Ctrl-@“, a poté „Ctrl-x x“ a jméno registru. Chceme-li potom text umístit, stiskneme „Ctrl-x g“ a následně jméno našeho registru.

Práci v Emacsu nám může v mnohém ulehčit vyhledávání. To se provádí klávesovou zkratkou „Ctrl-s“ a psaním textu, který chceme vyhledat. Po každém zadaném písmenu Emacs vyhledá první výskyt zatím zadaného řetězce. Pokud vás zajímá jiný výskyt hledaného slova, stačí zmáčknout znovu „Ctrl-s“. Kdybychom chtěli hledat zpětně musíme zadat „Ctrl-r“ a kdybychom chtěli vyhledaný výraz nahradit, použijeme „Alt-%“ a změnu potvrdíme mezerníkem, pokud chceme nahradit všechny stejné výrazy, použijeme místo toho „Alt-x replace-string“.

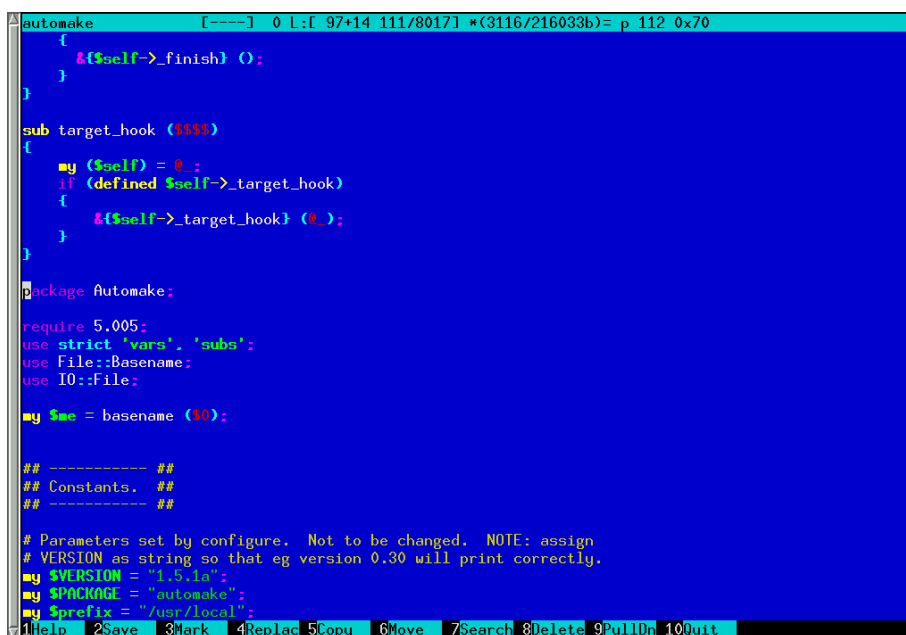
Editor Emacs může velice dobře používat makra, vždyť i jeho samotný název je od nich odvozen. Makra jsou velice efektivní nástroj pro práci s editorem a umožňují nám ulehčit si práci, kterou opakujeme. Nejdříve zmáčkneme „Ctrl-x (“, zadáme námi požadovanou operaci a pak zmáčkneme „Ctrl-x )“. Vše co jsme zadali mezi tyto zkratky, si editor zapamatoval a můžeme to znovu použít pomocí „Ctrl-x e“. Pokud bychom použili „Ctrl-x u“, bude Emacs tuto instrukci provádět až do konce dokumentu.

### **3.2.4.2.3 Needitorové funkce editoru Emacs**

Editor Emacs má mnoho funkcí, které z něj dělají více než pouhý editor textových dokumentů. Pomocí Emacsového mailového klienta lze psát zprávy, kopírovat text z ostatních dokumentů, odesílat, přijímat a číst emailové zprávy, opět z nich kopírovat a ukládat je jako dokumenty. To vše je možné ve zvláštním režimu, který se jmenuje RMAIL. Přestože je tento režim velice užitečný, používá většina uživatelů Linuxu jiný program pro přijímání mailů a v Emacsu emaily jen editují. Dalšími funkcemi Emacsu jsou např. režim GNUS, který slouží ke čtení zpráv ze sítě Usenet, Režim C pro práci s kódy jazyka C, režim TEX pro práci se soubory TEX. Každý z těchto režimů má speciální vlastnosti a funkce, které práci velmi usnadňují.

## 3.2.5 MC Editor

Midnight commander je předchůdce norton commanderu pro operační systém DOS. Pochází ze starých Unixových systémů a dodnes je vyvíjen a hojně používán. Editor je součástí Midnight Commanderu. Umožňuje přímo v souborovém manageru otevřít a upravit jakýkoli textový soubor. Nejvíce se snad funkcemi podobá Notepadu z Windows. Nepodporuje ani HTML zvýraznění textu (podporuje pouze některé programovací jazyky) a je určen jen k malým opravám textových souborů. Zcela určitě to není editor, ve kterém by šly naprogramovat celé stránky.



```
automake  [---] 0 L:[ 97+14 111/8017] *(3116/216033b)= p 112 0x70
{
    &{$self->_finish} ();
}
}

sub target_hook ( $$$ )
{
    my ($self) = @_;
    if (defined $self->_target_hook)
    {
        &{$self->_target_hook} (@_);
    }
}

package Automake;

require 5.005;
use strict 'vars', 'subs';
use File::Basename;
use IO::File;

my $me = basename ($0);

## ----- ##
## Constants. ##
## ----- ##

# Parameters set by configure. Not to be changed. NOTE: assign
# VERSION as string so that eg version 0.30 will print correctly.
my $VERSION = "1.0.1a";
my $PACKAGE = "automake";
my $prefix = "/usr/local";

1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit
```

Obrázek 10: MC editor při editaci jazyka perl.

## 3.3 WYSIWYG

### 3.3.1 NVU

Editor NVU je nejvíce podobný editoru Frontpage. Je to takzvaný editor wysiwyg, který umožňuje vizuální tvorbu webových stránek s minimální znalostí HTML.

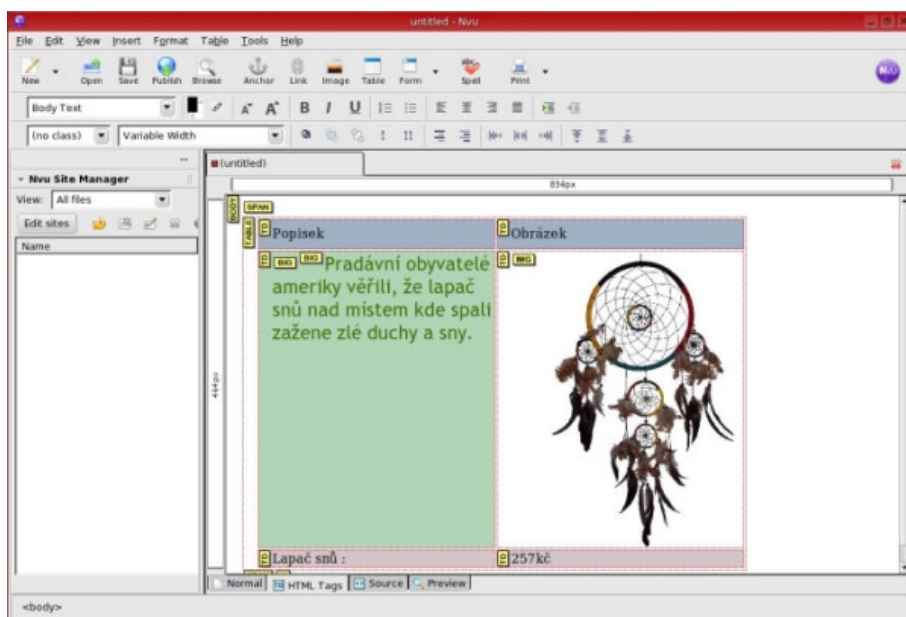
Pomocí Nvu můžeme velice snadno vytvořit jednoduché prezentace a ty poté umístit na Internet pomocí vestavěného nástroje pro uveřejňování. NVU podporuje kromě čistého HTML i CSS. Jelikož je NVU malý a je to jediný slušný wysiwig editor pro Linux, nachází se ve většině Linuxových distribucích.

Ovládání programu je velmi intuitivní a kdokoli pracoval někdy s nějakým kancelářským programem, by neměl mít žádné potíže s prací v NVU. Ovládání je rozloženo do standardních vysouvacích menu a ikon pro nejpoužívanější funkce, jakými jsou: Formátování textu, tabulky, formuláře, formátování písma a kontrola pravopisu. Okno programu je rozděleno na 3 části - menu, správce stránek a hlavní okno. Ve správci stránek můžeme hledat a spravovat vytvořené soubory a celé weby, správcem stránek lze přistupovat i na FTP server a soubory upravovat přímo na něm.

Hlavní okno pak slouží k vizuální tvorbě stránek. Použití NVU je opravdu jednoduché, pozicování obrázku a jeho zmenšování provádíme myší, stejně tak jeho umístění na stránku provedeme pouze přetažením ze složky. Stejně jednoduché je i tvoření textu, odstavců, seznamů a jejich upravování. Provádí se totiž stejně jako v kancelářských programech. Pokud bychom chtěli vytvořit tabulku, můžeme použít k tomu vytvořený nástroj, kde jenom vyplníme do kolonek rozměry, a tabulka je hotova. Stejným stylem je v NVU vyřešeno vkládání téměř jakéhokoli prvku stránek. Ve vysouvacích menu je pak kromě nástrojů na vytváření prvků stránky také několik programků pro kontrolu validity kódu, změny zobrazení a v neposlední řadě také praktický editor externích CSS.

Editor NVU má několik způsobů zobrazení stránky, jsou to : normální, HTML značky, zdrojový kód a náhled.

Normální mód bude asi nejvíce využíván, v něm totiž můžeme stránky vizuálně editovat. Mód HTML značky je užitečný v tom, že nám, aniž bychom se museli podívat do kódu stránky, zvýrazní všechny html tagy na stránce, takže víme, kde a co máme upravit pomocí CSS. V módu zdrojový kód vidíme a můžeme upravovat stránku v kódu. A náhled nám dá možnost si stránky prohlédnout tak, jak budou vypadat v prohlížeči webových stránek. Jediný mód, který mi zde chybí, je zobrazení kódu a vizuální editace zároveň, což ostatní programy umožňují.



Obrázek 11: Jednoduché použití wysiwyg editoru NVU.

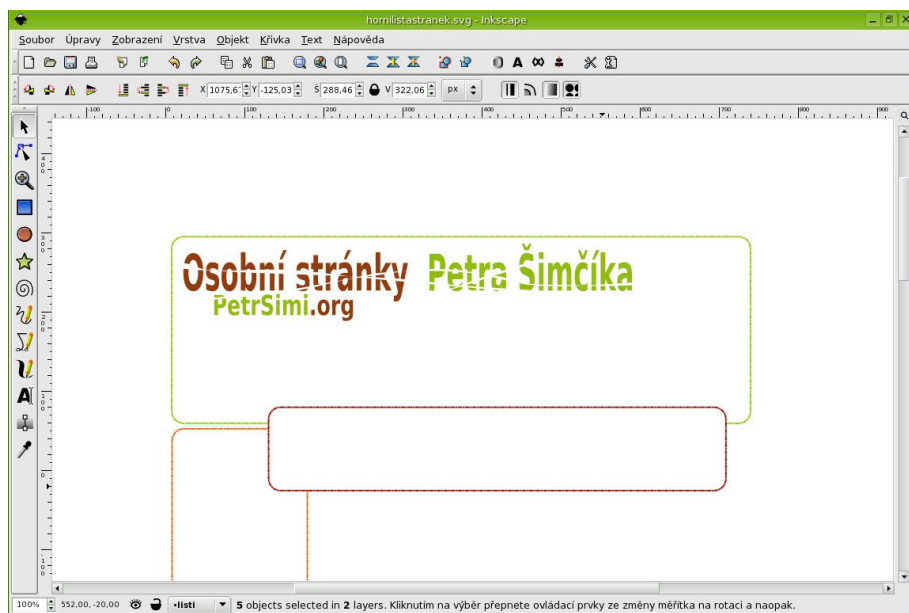
# 4 Tvorba webových stránek

## 4.1 Moje osobní Internetová prezentace

Pro podporu svého tvrzení, že webdesign je možné provozovat velice dobře i na Linuxové platformě, jsem se rozhodl zhotovit své osobní stránky právě pod systémem Linux a pouze za použití Linuxových programů. Svě vlastní stránky jsem sice již měl, ale byly již zastaralé a mně samotnému se už nelíbily. Svě stránky jsem se proto rozhodl naprosto předělat, dalo by se říci, že jsem je začal vytvářet úplně od začátku. Protože mám rád podzim, zvolil jsem design, kde převládá kombinace podzimních barev, bílé a motiv listů.

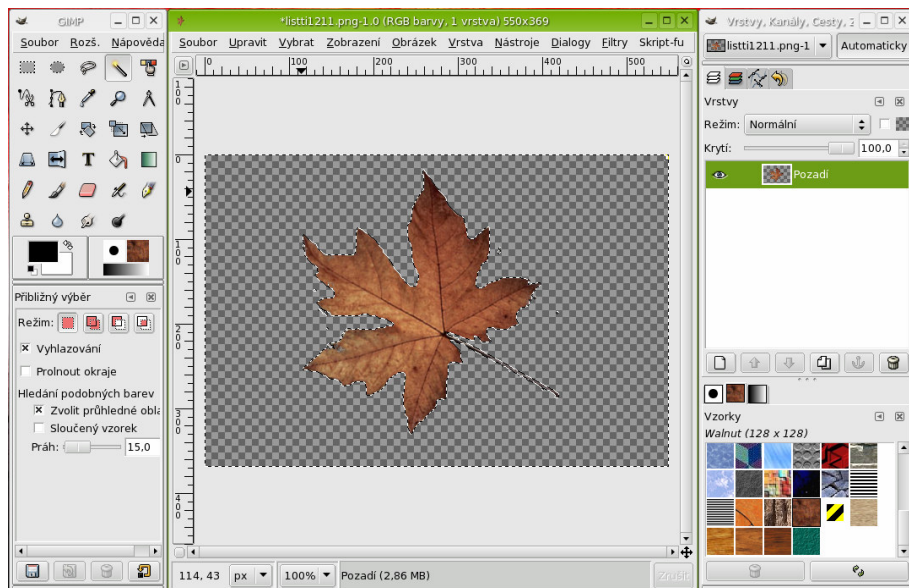
### 4.1.1 Grafika - 1.část - Základní design

Pro grafickou část stránek jsem použil dva programy, a sice Gimp a Inkscape. Protože se mi lépe pracuje v Inkscape, většinu grafiky jsem udělal pomocí něj a nejprve ve formátu SVG. Nejdříve jsem si pomocí nástroje pro tvorbu čtyřúhelníků vytvořil základní kostru stránky a do ní vložil text nadpisu. Přes něj jsem pro popraskaný efekt použil několik tahů kaligrafickým perem, které jsem přebarvil na barvu pozadí.



Obrázek 12: Základní kostra hlavičky mých stránek.

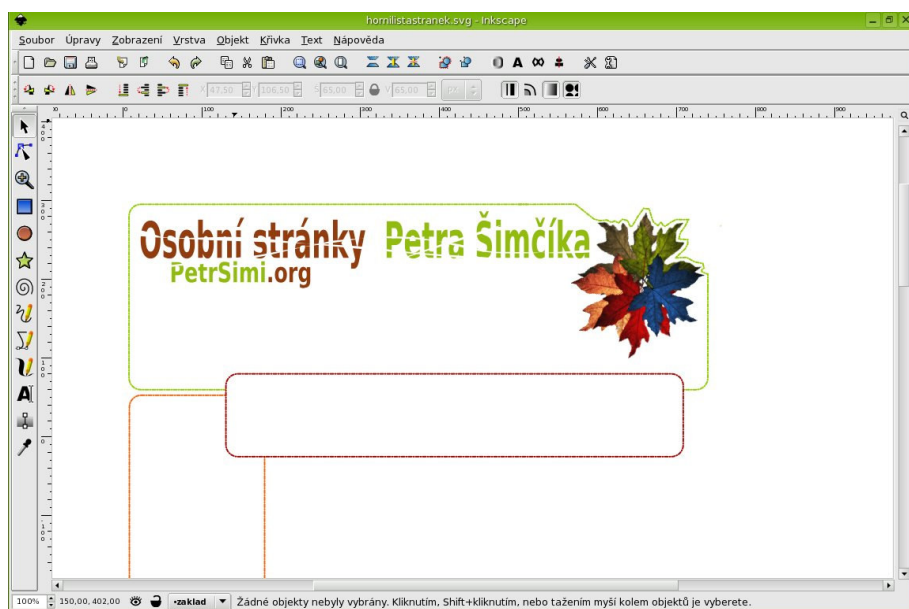
Dalším mým úkolem bylo sehnat fotografii listu. Tu jsem sehnal na internetu pomocí vyhledávače Google a otevřel ji v programu Gimp. Kromě samotného listu jsem musel celý zbytek obrázku vymazat. Nejprve jsem pro to použil nástroj ručního výběru, kterým jsem list obtáhl, poté jsem výběr invertoval, vyjmul celé okolí listu a jemné nedostatky domázl nástrojem guma. Protože jsem potřeboval, aby okolí listu bylo stále průhledné a list vypadal co nejlépe, uložil jsem ho ve formátu PNG bez jakékoli komprese.



Obrázek 13: Gimp - úprava listu.

Po tomto úkonu jsem se vrátil zpět do Inkscape, připravil jsem si roh čtverce s nápisem tak, aby nebylo jeho obtažení vidět, čehož jsem docílil vložením jiného čtverce s bílou výplní a žádným obtažením. Importoval jsem do kresby listy vytvořené v Gimpu a zmenšil je na požadovanou velikost.

Poté jsem ještě nástrojem pro vytváření čar obtáhl list a napojil čáru tak, aby ladila barvou i velikostí k původnímu obdélníku.



Obrázek 14: Konečná verze hlavičky stránek.

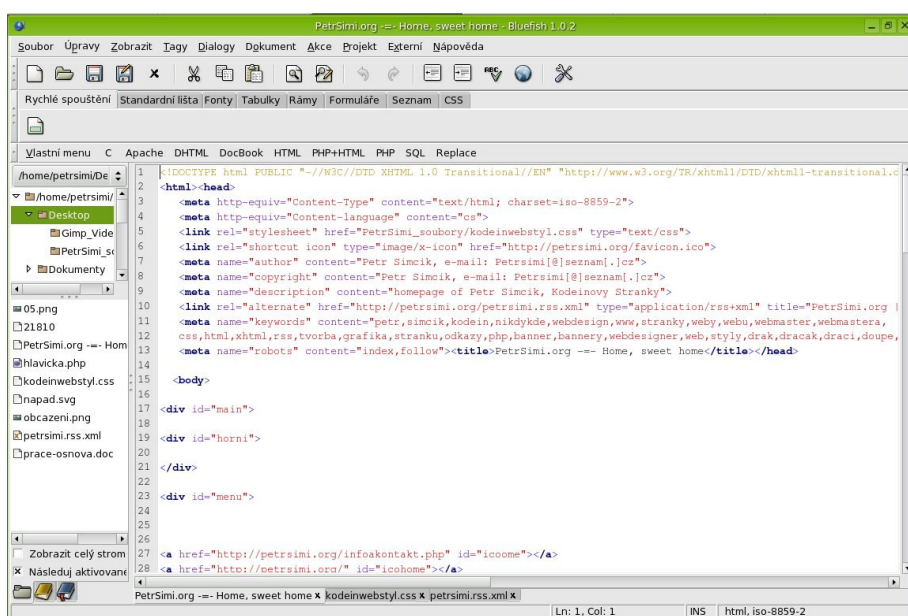
Stejným způsobem jsem pak vytvořil i dolní a boční lištu stránek s použitím jednotlivých barevných listů z jednotlivých vrstev Gimpu.

Protože jsem stále ještě neměl představu, jak budou vypadat tlačítka webu a ostatní grafické prvky, uložil jsem tyto prvky stránky v editovatelném formátu SVG a přešel k práci na kódu.

## 4.1.2 Kód - 1.část - Základ stránky

Pro editaci kódu jsem si vybral programy CSSed pro kaskádové styly a Bluefish pro vše ostatní. Bluefish by sice zvládl oboje, ale CSSed i přes své nedostatky poskytuje větší pohodlí a navíc dokáže kontrolovat validitu.

Do editoru Bluefish jsem zkopíroval z původních stránek kód v HTML a smazal všechny HTML tagy, které odporovaly XHTML. Poté jsem vše upravil tak, aby to odpovídalo standardům a zároveň bylo použitelné pro nový design, tj. smazal jsem všechny úpravy vzhledu pomocí HTML, dal jsem do hlavičky definici XHTML a rozdělil jsem stránku na několik DIVů.



```
1 |!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.c
2 |<html><head>
3 |<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
4 |<meta http-equiv="Content-language" content="cs">
5 |<link rel="stylesheet" href="PetrSimi_soubory/kodeinwebstyl1.css" type="text/css">
6 |<link rel="shortcut icon" type="image/x-icon" href="http://petsimi.org/favicon.ico">
7 |<meta name="author" content="Petr Simcik, e-mail: Petsimi[@]seznam[.]cz">
8 |<meta name="copyright" content="Petr Simcik, e-mail: Petsimi[@]seznam[.]cz">
9 |<meta name="description" content="homepage of Petr Simcik, Kodeinovy Stranky">
10 |<link rel="alternate" href="http://petsimi.org/petsimi.rss.xml" type="application/rss+xml" title="PetrSimi.org |
11 |<meta name="keywords" content="petr,simcik,kodein,nikdykde,webdesign,www,stranky,weby,webu,webmaster,webmasters,
12 |css,html,xhtml,rss,tvorba,grafika,stranku,odkazy,php,banner,bannery,webdesigner,web,styly,drak,dracak,draci,doupe,
13 |<meta name="robots" content="index,follow"><title>PetrSimi.org -> Home, sweet home</title></head>
14 |
15 |<body>
16 |
17 |<div id="main">
18 |
19 |<div id="horni">
20 |
21 |</div>
22 |
23 |<div id="menu">
24 |
25 |
26 |
27 |<a href="http://petsimi.org/infookontakt.php" id="icoome"></a>
28 |<a href="http://petsimi.org/" id="lcohome"></a>
```

Obrázek 15: Úprava kódu v editoru Bluefish.

Poté jsem spustil CSSed a zde udělal styly pro základní kostru stránky a navěsil na ní základní grafiku, horní boční a spodní lištu pomocí pozadí DIVu.

### 4.1.3 Ladění - 1.část - Zarovnání základních částí stránky

Kostra stránky je sice teoreticky hotová, ale po nahrání na FTP server a zveřejnění tomu stále něco chybí. Každý div si dělá co chce, tu a tam o několik pixelů přečnává a proto nastává fáze ladění. Otevřel jsem tedy v Linuxu dostupné prohlížeče Firefox a Opera a upravil CSS tak, aby na sebe části stránky plynule navazovaly a aby nebyl problém je naplnit textem. Testování pro Internet Explorer, který dělá největší problémy, je pod Linuxem téměř nemožné, a proto jsem ho řešil až po dokončení stránek.

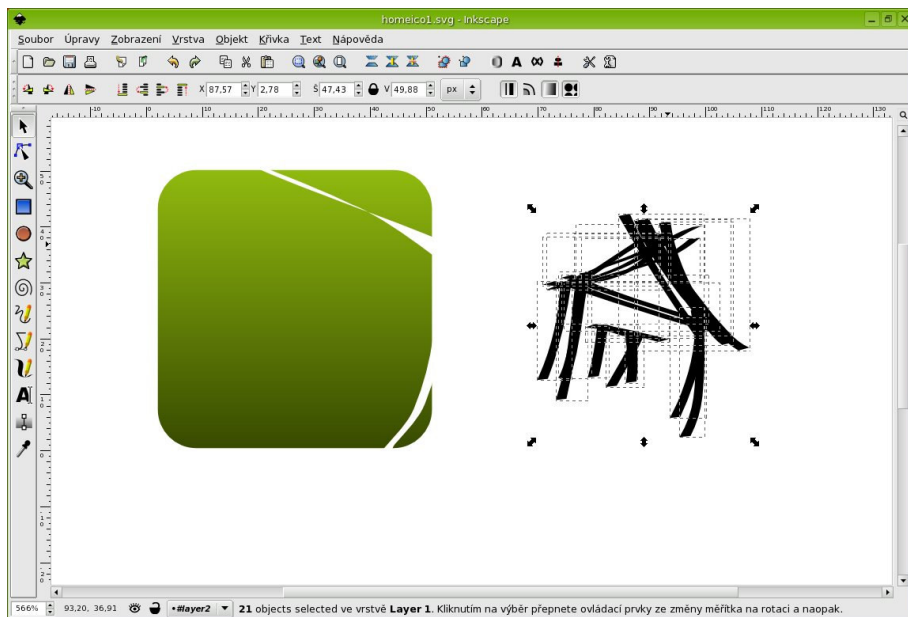


Obrázek 16: Kostra stránek po doladění tak, jak je zobrazena v prohlížeči.

### 4.1.4 Grafika - 2.část - Navigace

Protože kostra stránek byla hotová, mohl jsem přikročit k další fázi vytváření grafiky a k tomu mi opět posloužil Inkscape. Jednou z důležitých částí stránky je navigace, rozhodl jsem se, že bude dostupná jak textově, tak v grafické formě a to pomocí čtvercových tlačítek se zaoblenými hranami a barevným přechodem. Pomocí výběru čtyřúhelníku jsem udělal čtverec, ten jsem vyplnil přechodem a odstranil jeho ohraničení. Vedle plátna jsem pak od ruky nakreslil motivy jednotlivých

tlačítek, jakmile jsem byl spokojen, motiv jsem přebarvil na barvu pozadí, zkopíroval ho na povrch tlačítka a upravil jeho velikost. Pro efekt prasklin jsem opět použil kaligrafické pero.



Obrázek 17: Tvorba navigace pomocí Inkscape.



Obrázek 18: Výsledná tlačítka.

#### 4.1.5 Kód - 2.část - Navigace

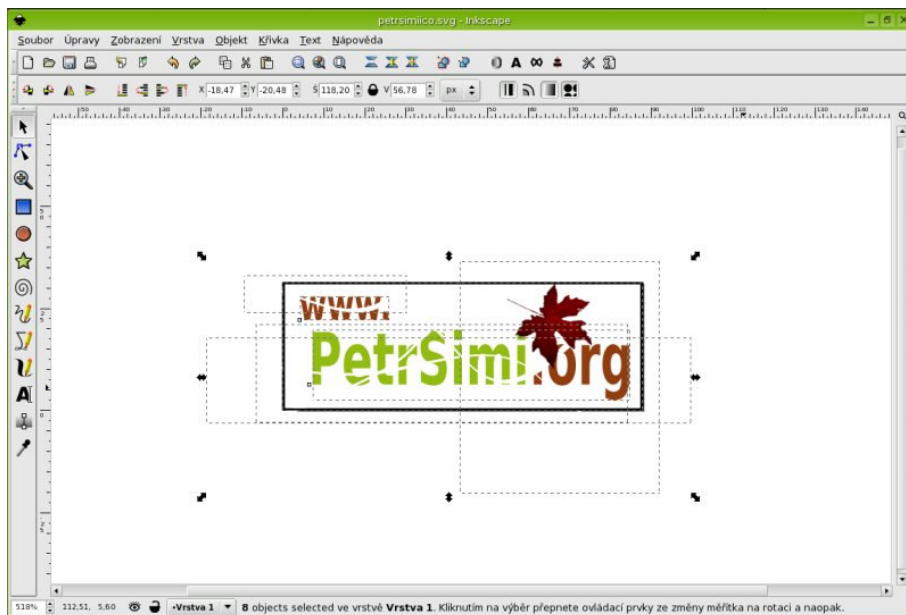
Grafiku pro navigaci jsem již měl, stačilo tedy znovu otevřít soubor se zkušební stránkou a napsat kód pro navigaci. Kód jsem pak umístil do nového souboru „menu.php“ a ten pomocí PHP funkce include, vložil do stránky. Tím, že jsem použil funkci include, jsem si ale znemožnil testování na lokálním disku a použití souborů s příponou html. Proto jsem hlavní soubor rozdělil dále na „hlavicka.php“ , „paticka.php“ a „index.php“ , přičemž jsem je všechny opět slepil pomocí funkce include. Tímto jsem si zjednodušil editaci navigace při velkém počtu stránek a vkládání obsahu.

Poté jsem si v CSSed opět otevřel kód kaskádových stylů a nadefinoval styl všem odkazům.

#### 4.1.6 Grafika - 3.část - Grafická náplň webu

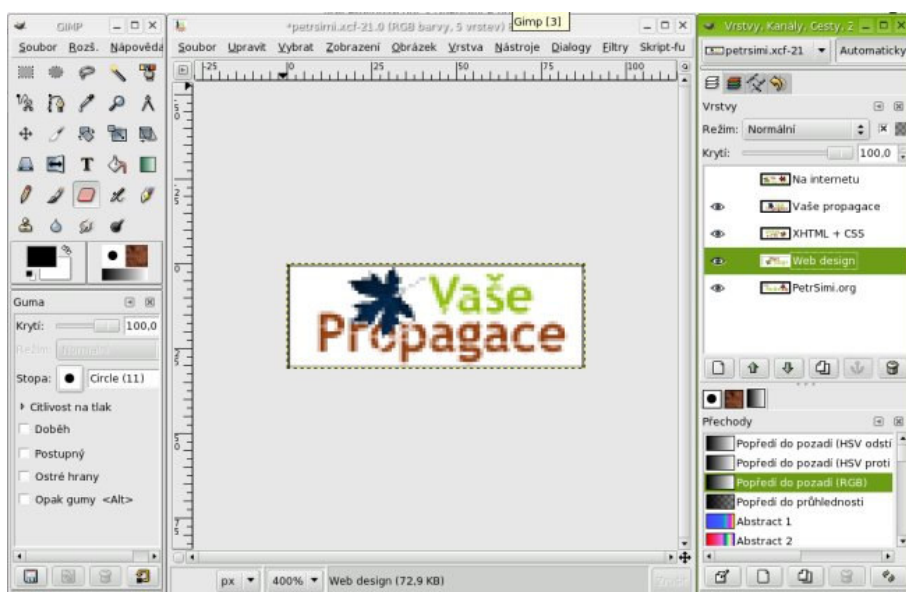
Web je v této fázi již skoro hotový, stačí ho již jen naplnit informacemi , kromě textu jsou to hlavně obrázky, které tvoří ne zrovna zanedbatelnou náplň webu. Zbylou grafiku na web tvoří fotografie, výměnné ikonky a jiný grafický obsah, u mě například snímky mnou udělaných stránek v sekci webdesign. Nejdříve jsem se zaměřil na výměnné ikonky, protože ty bylo nutné udělat zcela znovu, aby se hodily k nové vizuální stránce webu.

Použil jsem Inkscape. Nejdříve jsem vyrobil ikonku pro PNG formát, tudíž statickou. Postup byl velice podobný výrobě horní lišty, použil jsem to samé písmo, jeden z listů vyrobených v Gimpu a opět praskliny. To celé jsem orámoval černým obdélníkem a vše přizpůsobil standardní velikosti webové ikony tj. 88px na 31px.



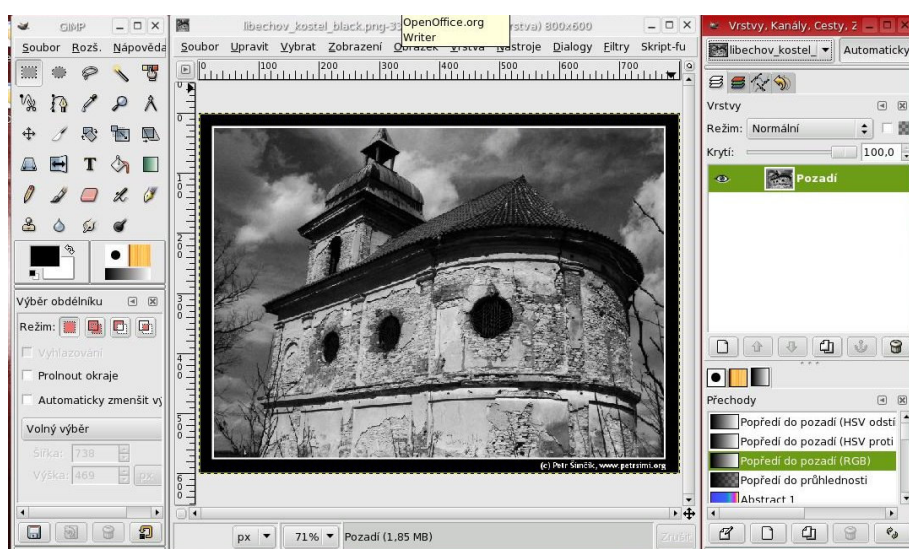
Obrázek 19: Výměnná ikonka webu vytvořená v programu Inkscape.

Animovaná ikonka byla vyrobena podobně, ale skládá se z několika po sobě jdoucích obrázků, a proto jsem opět musel použít Gimp. Nejdříve jsem tedy udělal stejným postupem jako v případě statické ikony dalších pět vrstev a pak je otevřel v Gimpu. Všechny otevřené obrázky jsem potom seřadil do jednoho souboru jako vrstvy a uložil jako animovaný gif.



Obrázek 20: Vrstvy animované ikony v Gimpu.

Dalším problémem byly fotografie. Jejich velikost tak jak jsou neupravené z digitálního fotoaparátu, je příliš velká na prezentování na internetových stránkách. Digitální fotografie bez jakékoli úpravy zabírá něco mezi jedním až dvěma megabajty prostoru a to i u malé prezentace s několika fotkami nabobtná stránku do gigantických rozměrů co se týče zabraného místa, nemluvě o tom, že návštěvník stránek nebude mít nejspíše náladu čekat, než se mu obrovská fotografie načte. Ač je Gimp pro úpravu fotografií trošičku neohrabaný, má pro jejich úpravu nejvíce funkcí a mně vyhovuje. Proto jsem všechny fotografie pro mé stránky upravoval právě v Gimpu. Většinou stačilo je jen pomocí funkce Resize zmenšit na velikost, kterou jsem pokládal za dostatečnou a tou je 800px na 600px, u některých jsem použil některé funkce z menu Barvy např. Jas-kontrast, Posterizovat, Odbarvit, nebo Odstín-sytost. Všechny fotografie jsem pak uložil do formátu PNG, protože používá bezztrátovou kompresi. Samozřejmě jsem vytvořil i malé náhledy.



Obrázek 21: Gimp má k úpravě fotek bezkonkurenční množství funkcí.

Snímky mnou vytvořených stránek jsem pořídil pomocí programu Ksnapshot, což je program, který v Linuxovém grafickém prostředí KDE zodpovídá za funkci klávesy Print Screen. Takto vytvořené snímky jsem poté znovu otevřel v Gimpu a pomocí výběru ořízl nepotřebné části jako jsou např. lišty prohlížeče a nakonec je zmenšil na požadovanou velikost.

#### 4.1.7 Kód - 3. část - Zbýlý kód a náplň webu

Téměř poslední věcí, kterou bylo nutno udělat, bylo doplnit starý obsah a přidat nový. Starý obsah jsem nejdříve otevřel v Bluefish editoru a rozdělil do předem připravených záložek podle druhu obsahu. Poté jsem všem prvkům stránky přiřadil třídy a identifikátory. Následně stačilo v CSSed pro všechny tyto prvky nadefinovat styly barvy, velikostí, fontů apod. V Bluefish editoru jsem pak otestoval i zvýraznění PHP na funkci „send“, která mi umožní mít na svých stránkách vzkazový formulář.

Jelikož byly původní stránky v kódování windows-1250, musel jsem je ještě před umístěním na web převést do Linuxem používaného kódování. Mohl jsem si vybrat z UTF-8 a ISO-8859-2, z čehož

jsem zvolil to druhé, protože je více podporováno. Tuto změnu jsem provedl opět v programu Bluefish, když jsem byl s kódováním spokojen, přesunul jsem stránky na web.

#### 4.1.8 Ladění - 2. část - Doladění vzhledu ve většině prohlížečů

Výhodou využívání XHTML a CSS pro tvorbu prezentace je to, že je opravdu velká šance, že stránky se budou zobrazovat tak, jak jsme si představovali při jejich torbě. Ne ve všech prohlížečích to ale platí stoprocentně. Mezi prohlížečem Firefox od Mozilla Foundation a prohlížečem Opera není zas tak velký rozdíl a stránky bylo nutné upravit jen velmi málo, stačilo v CSSed upravit soubor stylů a stránky se zobrazovaly zcela korektně. Problém nastal v případě prohlížeče Internet Explorer, protože ten pod Linuxem za běžných okolností není dostupný a bohužel je většinový. Po poradě s ostatními lidmi na internetových fórech zabývajících se Linuxem jsem zjistil, že Internet Explorer na Linuxu jde spustit pomocí programu wine a speciálních skriptů. Tuto variantu řešení, tj. Instalaci Internet Exploreru jsem bohudík při testování svých stránek nemusel použít, protože mě několik uživatelů Windows ujistilo, že se zobrazují korektně.



Obrázek 22: Ukázka vzhledu stránek v prohlížeči Firefox.

## 4.2 Můj Blog

Druhou částí mých stránek je můj blog, internetový deníček, kam zapisuji své myšlenky a zážitky, které by mohly být přínosné nebo by mohly pobavit uživatele internetu.

### 4.2.1 Instalace publikačního systému phpRS

Jelikož nejsem příliš zblhlý v php a MySQL, rozhodl jsem se postavit svůj blog na již vytvořeném a volně dostupném redakčním systému. Tímto redakčním systémem bylo phpRS. Jeho instalace je vskutku jednoduchá, stačí pomocí jeho části SQL vytvořit novou databázi, tu pak zohlednit v souboru config.php, jehož úpravy jsem samozřejmě dělal opět v programu Bluefish, a celý systém zkopírovat na FTP. Poslední částí instalace je nastavení v administračním režimu stránek, s jehož pomocí jsem nainstaloval potřebný layout.

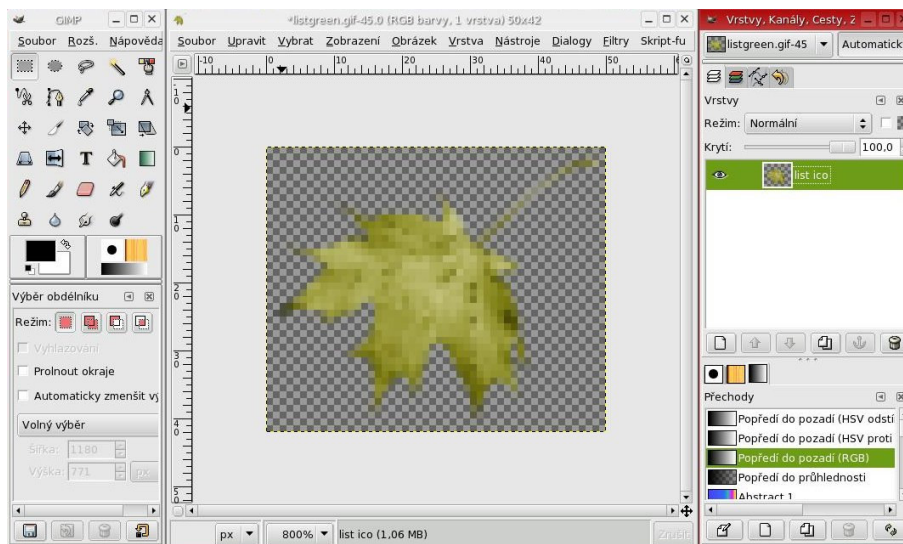
### 4.2.2 Grafika - Základní design

Protože je základní layout phpRS poněkud nevzhledný, musel jsem ze začátku pro svůj nový blog vytvořit alespoň novou hlavičku stránky. Tu jsem tvořil opět v kombinaci programů Inkscape a Gimp. Gimp opět posloužil k úpravě listů, které jsem tentokrát získal pomocí digitálního fotoaparátu. Použil jsem stejný postup jako u základu mých stránek a tímto způsobem vytvořil několik různě barevných listů. Ty jsem opět importoval do programu Inkscape a propletl je s nadpisem mého blogu. Celá hlavička blogu je složená vlastně jen z písma, importovaných listů a „prasklin“ vyrobených s pomocí kaligrafického pera.



Obrázek 23: Hlavička blogu v programu Inkscape.

Na začátku každého příspěvku se v phpRS implicitně zobrazuje obrázek dot.gif, což je poměrně nevzhledná modrá tečka, proto jsem se s Gimpem pustil ještě do jednoho listu, tentokrát jsem ho však musel po očištění od okolí ještě převést na menší počet barev a uložit ho jako průhledný gif. Byla by tu sice možnost nahradit gif také obrázkem PNG, ale při takto malém rozměru obrázku mi gif připadal výhodnější a navíc jsem nemusel zasahovat do kódu stránky a stačilo obrázek jen vyměnit.

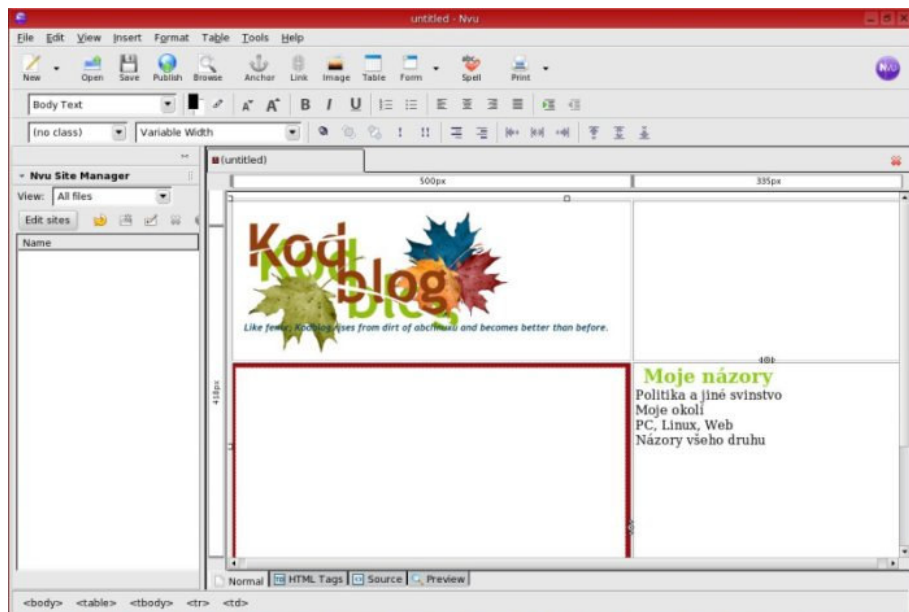


Obrázek 24: Malý list zvětšený a upravený v programu Gumo.

### 4.2.3 Kód - Základ stránky, návrh vzhledu, CSS

PhpRS sice vytvoří hned po instalaci použitelný základ, ale já jsem potřeboval, aby byl systém co nejjednodušší a vyhovoval mi po vzhledové stránce. Proto jsem v základních souborech vzhledu pomocí editoru Bluefish odstranil reklamu, horní odkazovou lištu a ostatní přebytečné prvky. Abych se přesvědčil o použitelnosti wysiwyg editoru NVU a vyzkoušel si jeho funkce, použil jsem ho k jednoduchému návrhu stránky blogu pomocí tabulek, které phpRS používá. Použil jsem přitom funkce pro vytváření tabulek, jejich úpravu, vkládání obrázků a formátování textu.

Poslední věci, která mi zbývala dokončit v kódu, byly styly CSS. Pomocí CSSed jsem vybral potřebná písma, změnil barvy a odstranil ohraničení příspěvků.



Obrázek 25: Na návrhu svých stránek se mi podařilo otestovat program NVU.

## 4.2.4 Ladění - Přizpůsobení mým představám

Ladění stránek v systému phpRS probíhá na webu. Přesto jsem využil editor Bluefish k vytvoření menu a poté jsem ho teprve překopíroval do příslušné kolonky administračního rozhraní.

Poslední věcí, kterou bylo nutno doladit, bylo zobrazování v různých prohlížečích. I když jsou stránky blogu naprosto jednoduché, stejně měly problém při zobrazování v Internet Exploreru. V ostatních prohlížečích se zobrazovaly bez problémů. Tento problém jsem opravil jemným doladěním stylů, speciálně pro Internet Explorer.

## 4.3 RSS - Informační kanály stránek

RSS jsou kanály, kterými lze zveřejnit novinky na vašich stránkách, aniž by se na ně uživatel internetu musel neustále dívat. To, že neobtěžujete uživatele zbytečnými návštěvami stránek, vám zachová jeho delší přízeň a on vaše stránky bude navštěvovat pokaždé, když se v RSS kanálu objeví nějaká zajímavá novinka.

Protože stránky bez návštěvníků jsou k ničemu, rozhodl jsem se udělat RSS kanál jak pro svůj blog tak pro své osobní stránky. Jelikož jsou mé stránky rozsahem malé, připadlo mi zbytečné dělat RSS automaticky generované, a proto jsem se vrhl na jejich tvorbu přímo v XML. K tomu jsem použil opět Bluefish, protože zvýrazňuje kód a nejlépe se mi ovládá. Otestoval jsem je pomocí prohlížeče Firefox, který čtení RSS podporuje. Pomocí Bluefishu jsem také zeditoval hlavičky stránek a menu tak, aby v nich byla možnost si RSS kanál zapnout.

## Závěr

Během mé práce jsem přišel na to, že i když je pod Linuxem spousta programů, které jsou volně dostupné a jsou na téměř profesionální úrovni, pořád zde něco chybí. Dle mého názoru zde chybí komerční programy společností Adobe a Macromedia. To, že jsou na platformách Mac OS X i Microsoft Windows dostupné programy těchto firem, z nich dělá ve webdesignu přednostní platformy. Macromedia Flash MX, Macromedia Fireworks a Adobe Photoshop jsou totiž ve světě webdesignu přece jen pojmy a ten, kdo tyto programy neovládá, je na druhé až třetí koleji. V Linuxu jsou sice pokusy tento stav nějak napravit, ale zatím se to daří jen částečně. Macromedia Flash měl být nahrazen F4L, ale tento program uvázl ve vývoji na mrtvém bodě a technologie Flash proto pod Linuxem není dostupná. Fireworks co se týče funkcí nemají v Linuxu konkurenci, lze je tedy zastoupit, ale uživatel zaplatí přechod na open source programy ztrátou některých funkcí. Pokud jde o mě, ztráta Macromedia Fireworks mě mrzela ze všech ztrát nejméně, což by ale určitě neplatilo v případě větších designerských firem nebo profesionálů zvyklých na svůj komfort. Adobe Photoshop má dva konkurenty, komerční Pixel a nekomerční Gimp. Pixel je ve vývoji zatím dosti pozadu, ale po svém dokončení by měl být plně konkurenceschopný Photoshopu. Gimp je zatím to nejlepší, co může Linux a open source nabídnout, ale nepodporuje barevný režim CMYK, který je k tvorbě grafiky hojně používán, nehledě na to, že používá zcela jiné rozložení ovládacích prvků, které nemusí vyhovovat každému. I po dokončení Pixelu nebo vylepšení Gimpu však bude chybět jedna věc, kterou open source zatím nabídnout nemůže, a tím jsou balíčky programů, které jsou uzpůsobeny tak, aby mezi sebou spolupracovaly, například tak jako to dělají právě programy společnosti Macromedia v novém Macromedia Studio 8. Dalším programem, který pod Linuxem znatelně chybí, je Macromedia Dreamweaver, který je opět součástí výše zmíněného balíku. Tento program je totiž nezbytný pro větší firmy zabývající se tvorbou stránek ve velkém, pro které je důležitá rychlost zpracování stránek a je pro ně nemyslitelné, aby zaměstnanec ztrácel čas nějakou nadbytečnou prací s kódem. Tento nedostatek bychom mohli nahradit programem NVU, který, přestože poskytuje zadarmo hodně funkcí a je velice použitelný, nemůže Dreamweaveru prozatím konkurovat. V jedné množině programů však Linux neztrácí. Jsou to editory kódu. Editor vim je dokonce tak populární, že dosáhl zpracování pro Windows. Ostatní programy určené k editaci kódu v ničem nezaostávají za svou konkurencí z Windows a v lecčems ji i předstihují. Poslední slabinou Linuxu je, že na něm chybí prohlížeč se zobrazovacím jádrem Internet Exploreru od společnosti Microsoft, ale i zde se blýská na lepší časy a bylo vytvořeno několik skriptů, které umožňují pohodlnější instalaci tohoto prohlížeče pod Linuxem.

Po vytvoření mých stránek jsem přes všechny nedostatky, které jsem výše uvedl, přesvědčen, že tak jak je Linux připraven vystoupit ze stínu Microsoft Windows a Mac OS X a stát se plnohodnotným a uživatelsky přívětivým operačním systémem, tak je také připraven pomalým, ale jistým krokem dojít na své místo jakožto použitelná webdesignerská platforma, byť prozatím jen pro samostatné webdesignery či menší firmy.

# Resumé

Svou práci jsem začal tím, že jsem kompletně přeinstaloval svůj počítač a migroval jsem na operační systém Linux. To mi umožnilo zakusit slasti a strasti tohoto operačního systému a vyzkoušet všechny programy, včetně těch pro webdesign, v praxi. Zkoušení programů na vlastní kůži mi umožnilo je velice dobře poznat a díky tomu jsem se stal téměř nezávislý na jiných zdrojích jako jsou knihy nebo webové stránky.

Ve své práci jsem seznámil čtenáře s tím, co je to webdesign, vyjmenoval a vysvětlil některé pojmy ve webdesignu a nastínil historii tohoto oboru. Dále jsem se zabýval Linuxem jako takovým, jeho historií, současností a možnostmi jeho využití, včetně příkladů nasazení u nás i ve světě.

V další části jsem představil několik nejpoužívanějších programů určených pro webdesign a provozovatelných na operačním systému Linux. Popsal jsem jejich uživatelské rozhraní a ovládání.

Nakonec své práce jsem využil některé z těchto programů při tvorbě mých webových stránek a v závěru jsem shrnul, co Linuxu chybí, aby se mohl stát plnohodnotnou platformou pro webdesign.

I started my work in this document by reinstalling the operating system on my computer and I began using only Linux. It enabled to experience advantages and disadvantages of this operating system by testing all programs including those for webdesign in practice. The program testing enabled me to get acquainted with them very well thanks to which I became independent of other sources such as books and websites.

In my work, I made readers familiar with that what means webdesign, named and explained some terms in webdesign and summed up the history of this branch. Further on I was concerned with Linux as such, its history, its present and the possibility of its use including examples of application in our country and throughout the world.

In further part I presented several most used programs for webdesign and for Linux operating system. I described their user interface and control.

At the end of my work I used some of these programs in course of creating my website and I ended up resuming what Linux lacks to reach a full value for webdesign platform.

# Seznam literatury

1. Linda Weinmanová - *<velká kniha webdesignu.4>* - Zoner press, ZONER software, s.r.o. 2004.  
ISBN 80-86815-10-2
2. Matt Welsh, Mathias Kalle Dalheimer, Terry Dawson, Lar Kaufman - *Používáme Linux* - Computer Press, a.s. 2003.  
ISBN 80-7226-698-5

# Seznam příloh

Příloha č. 1: CD s webovými stránkami a grafikou v editovatelných i webových formátech.