

**Vysoké učení technické v Brně**  
**Fakulta informačních technologií**



**Programová podpora pro Fair Trade**  
**Bakalářská práce**

# Programová podpora pro Fair Trade

Odevzdáno na Fakultě informačních technologií Vysokého učení technického v Brně.

*Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Petra Chmelaře. Další informace mi poskytli pracovníci a pracovníce Společnosti pro Fair Trade a rozvojové vzdělávání. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.*

.....  
Juraj Krivosudský

## **Abstrakt**

Práce vznikla jako programová podpora pro nevládní neziskovou společnost Společnost pro Fair Trade a rozvojové vzdělávání. Klade si za cíl položit základy informačního systému, který bude schopný integrovat a usnadňovat činnost této organizace. V aktuální verzi se jedná zejména o skladové hospodářství sjednocující management zboží na několika pobočkách a synchronizaci účetních položek pro potřeby účetnictví. Je zde uvedeno několik modulárních prvků, které umožní ulehčit další rozšíření a vývoj systému. Systém je postaven na technologiích PHP a MySQL s doplňkovou funkcionalitou JavaScriptu, která je však v zájmu co nejširší kompatibility minimalizována a pokud možno nahrazována alternativními řešeními.

## **Klíčová slova**

fairtrade, fair trade, férový obchod, informační systém, IS, skladové hospodářství, účetnictví, web, modularita, PHP, MySQL, WWW, HTML, CSS

## **Abstract**

This work has been created as a support for non-profit NGO Society for Fair Trade and Development Education. It's aimed to build a base of an information system, that will be able to integrate and simplify the organisation's running. In the actual version it's foremost about stock control system unifying the management of commodities on several affiliates and synchronization of book entries for accounting purposal. To make the further development easier, several modular components are introduced. The system is based on technologies of PHP and MySQL with additional functionality of JavaScript which is, for the sake of widest compatibility possible, minimalised and substituted with alternative solutions if possible.

## **Key Words**

fairtrade, fair trade, information system, IS, stock control, accounting, web, modularity, PHP, MySQL, WWW, HTML, CSS

# Obsah

<b>Úvod.....</b>	<b>6</b>
<b>1. Praktická východiska práce.....</b>	<b>7</b>
1.1. Co je Fair Trade.....	7
1.2. Proč podpořit Fair Trade.....	7
1.3. Zadání a jeho vývoj.....	8
<b>2. Technologická východiska.....</b>	<b>9</b>
2.1. Technologická situace před realizací.....	9
2.2. Volba aplikačního prostředí.....	9
2.3. Volba databázového prostředí.....	10
<b>3. Návrh, realizace.....</b>	<b>11</b>
3.1. Databázový model.....	11
3.1.1. Návrh.....	11
3.1.2. Realizace, optimalizace.....	12
3.2. Aplikační model.....	13
3.2.1. Struktura.....	13
3.2.2. Role, Uživatelská práva.....	14
3.3. Uživatelské rozhraní.....	15
3.4. Vybrané problémy.....	18
3.4.1. Modul pro prohlížení dat – tabulka.....	18
3.4.2. Modul pro editaci dat – formulář.....	19
3.4.3. Stromové menu.....	20
<b>4. Závěr.....</b>	<b>22</b>
<b>5. Literatura.....</b>	<b>23</b>
<b>6. Přílohy.....</b>	<b>24</b>
6.1. A : Uživatelský manuál.....	24
6.2. B : Programová dokumentace.....	24

# Úvod

Tato práce je dokumentací informačního systému, který byl vyvíjen v rámci programové podpory pro Fair Trade. Ta si klade za cíl pokrýt informační procesy, na kterých funguje neziskový nevládní projekt Společnost pro Fair Trade a rozvojové vzdělávání (dále Společnost), a integrovat je do kompaktního celku. Výhody tohoto přístupu jsou zřejmé, kromě lepší kontroly nad celým procesem je přínosem informačních systémů taky vyšší efektivita – odstraňují se redundantní informace, redukuje se možnost chyb při přenášení dat mezi nativně nespolupracujícími systémy, činnostem, které nelze vykonávat systematicky, je alespoň vytvářeno rozhraní umožňující lepší zpětnou kontrolu.

Mým úkolem v tomto procesu bylo navrhnout a implementovat multiuživatelské rozhraní umožňující vzdáleně přistupovat k datům a vykonávat nad nimi potřebné operace, dále postavit základ nejdůležitější části tohoto systému, skladového hospodářství. Jelikož se z technologického hlediska nedá spoléhat na vybavenost budoucích (i potencionálních) uživatelů konkrétním softwarovým vybavením, je zde důležitá přístupnost služeb z co nejširšího výběru běžných zařízení. S ohledem na další vývoj bylo vhodné už teď stavět na modulech, které budou moci být použity i při budování dalších součástí systému – jak ve snaze usnadnit kolegům či kolegyním práci na tomto procesu, tak ve snaze umožnit udržování systému v co nejpřehlednější a nejkompaktnější formě.

V první části tohoto dokumentu budou vysvětlena praktická východiska pro vznik systému : příčiny vzniku a fungování Společnosti, jejich aktuální situace, vyhlídky do budoucna - a z toho plynoucí moje motivace pomoci a zapojit se do tohoto projektu podpory. Dále bude shrnuto zadání této fáze projektu a jeho vývoj v průběhu implementace.

Druhá část představí technologická východiska : postavení informačních technologií ve fungování Společnosti před započítím projektu, zejména v oblastech přímo se týkajících této fáze. Budou shrnuty možnosti výběru použitých aplikačních a databázových prostředků s ohledem na dosud používané technologie, na cenu (a z toho plynoucí prioritu) případného pořízení kvalitnějších alternativ a taky na moje možnosti a schopnosti používat to-které řešení. Logickým důsledkem je pak shrnutí výhod, ale především omezení plynoucích z konečného výběru.

Ve třetí části je pak nastíněn návrh a realizace systému. Nejprve je v databázovém modelu popsána struktura navrhnuté databáze a s ní související datové procesy. V aplikačním modelu jsou spíše všeobecně popsány zvolené přístupy a postupy u implementace aplikace – vybrané zajímavější či originální části řešení jsou zařazeny až na konec této kapitoly, za popis uživatelského rozhraní. Je tak činěno proto, že tyto konkrétní problémy, ač jsou zajímavé hlavně z aplikačního hlediska, můžou souviset i s funkcionalitou uživatelského rozhraní či databáze.

Závěr obsahuje shrnutí dosažených výsledků a náhled na další fáze vývoje, na možnosti růstu systému. Následuje sekce Literatura, kde jsem uvedl bibliografické citace, a přehled příloh, který obsahuje krátké informace o jednotlivých přílohách a lokalitách kde je možno najít.

# 1. Praktická východiska práce

## 1.1. Co je Fair Trade

Na stránkách <http://www.fairtrade.cz/>, patřícím právě Společnosti, se o Fair Trade dočteme :

*„Spravedlivý obchod - Fair Trade je obchodní partnerství, jehož cílem je přímá a účinná podpora znevýhodněných výrobců z rozvojových zemí. Toho se snaží docílit především poskytováním „férových“ obchodních podmínek pro zapojené výrobce, důrazem na dodržování základních norem pracovního práva a ochrany životního prostředí a zvyšováním informovanosti spotřebitelů o situaci malých zemědělců a řemeslníků v rozvojových zemích.*

*Fair Trade tak poskytuje konkrétním lidem v rozvojových zemích šanci vymanit se z bludného kruhu bídy vlastními silami a žít důstojný život.*

*Fair Trade současně dává nám, spotřebitelům, jedinečnou možnost snadno a účinně podpořit jiný ekonomický model, dát hlas jinému způsobu obchodování a výroby a v neposlední řadě získat velmi kvalitní výrobky za odpovídající cenu.“[1]*



Obrázek 1: Logo FairTrade

Ve světě funguje nesčetně organizací a spolků věnujících se podpoře Fair Trade, v ČR se první aktivity v této oblasti objevily jenom před několika lety. Za tuto dobu – a nutno říct že právě díky organizacím typu Společnosti – se zboží Fair Trade zbavilo nálepky raritního zboží, které si člověk koupí párkrát za život.

## 1.2. Proč podpořit Fair Trade

Jednou z premis, na kterých fungování Fair Trade stojí, je dobrovolnost. Jenom díky nezištné podpoře dobrovolníků a dobrovolnic na celém světě je možné zboží, za které dostali výrobci na opačné straně zeměkoule férově zapláceno, prodávat u nás za ceny, které si můžeme dovolit aspoň svátečně. Jako sympatizant myšlenek férového obchodu jsem uvítal možnost tuto aktivitu podpořit svým kusem práce.

Pro mě - a taky pro většinu mých spolužáků či spolužaček – je před odchodem do důchodu vysokoškolské studium posledním úsekem v životě, kdy máme podmínky a motivaci pro to, abychom se mohli hlouběji věnovat práci na projektu, který nás zrovna neživí. Nebo jinak řečeno – po škole se pravděpodobně dostaneme jenom k realizaci projektů komerčních, nebo dostatečně malých na to, abychom se jim mohli věnovat po práci či o víkendu. Projekt takového rozsahu, jaký má tato Programová

podpora pro Fair Trade, by v komerční sféře stál peníze, které by nezisková organizace sháněla jenom těžko.

Toto byl rozhodující argument, který v mém případě zavázil u výběru zadání bakalářské práce. Jsem rád, že se našli další, kteří smýšleli podobně, takže projekt po odvedení mého podílu práce nevyhasne – a tímto bych jim chtěl vyjádřit respekt a popřát hodně úspěchů. Zároveň bych chtěl poděkovat Ing. Petru Chmelaři za rady a připomínky k mé práci, ale zejména za vedení podobných „dobrovolnických“ projektů na půdě fakulty, umožňující tak veřejně-prospěšným uskupením najít dobrovolnickou podporu v řadách studentů či studentek.

### 1.3. Zadání a jeho vývoj

Jak už jsem naznačil, tato práce je jenom začátkem širšího projektu programové podpory. Po konzultacích s Ing. Chmelařem a lidmi ze Společnosti jsme došli k závěru, že největší prioritou je vyřešení skladového hospodářství. Je tomu tak jednak z existenciálního hlediska (nepořádek na skladech navozuje nepořádek v účetnictví, tím pádem v daních...) a jednak kvůli tomu, že na skladovou evidenci navazuje, ať už přímo nebo nepřímo, i velká část dalších plánovaných součástí systému (e-shop, podpora individuálních komunitních prodejců).

Z konzultací tedy vzešel požadavek na systém, který umožňuje správu zboží u obchodní jednotky, která má :

- několik skladů, nebo-li poboček, s částečnou autonomií (určování prodejní ceny, nákup zboží...),
- větší počet uživatelů s různou mírou oprávnění vykonávat operace; tato práva udělují privilegovaní uživatelé,
- většinu uživatelů vázanou na svou „domovskou“ pobočku, tj. bez potřeby operovat na pobočkách ostatních.

K základním operacím má patřit, kromě definice atributů jednotlivých entit (zboží, dodavatelé), nákup a prodej zboží v několika fázích, přesuny zboží mezi pobočkami a generování statistik o stavu zboží a účetních operací. Přístup do systému nesmí být vázán na fyzickou přítomnost na některé z poboček a/nebo použití speciálního hardwarového nebo softwarového vybavení (např. práce „z domu“).

Zejména pro potřebu dalších „aplikací“ (např. e-shop) bude vhodné hned od začátku evidovat o zboží i informace ne přímo související se skladováním, včetně kategorizace do hierarchicky řazené struktury s neomezeným počtem úrovní.

V průběhu implementace se v zadání objevilo taky několik změn, většina na úrovni detailů či doplňkové funkcionality (např. přidat vlastnost k některé z entit), některé však znamenaly závažnější zásah do hotového kódu a/nebo databázového návrhu – takovým bylo například zjištění, že se Společnost bude muset, navzdory tomu, že se s tím nepočítalo, stát plátkyní DPH.





### 2.3. Volba databázového prostředí

U výběru databázového (dále DB) prostředí se výběr, podobně jako u prostředí aplikačního, ve velké míře řídil rozšířeností a z toho vyplývající finanční dostupností či množstvím a kvalitou podpůrných informací nacházejících se na internetu. Ve hře byly vesměs dva hráči – MySQL a PostgreSQL. Jelikož je MySQL přeci rozšířenější (zejména na hostingových službách) a v posledních verzích udělalo velký pokrok v „dohánění“ očekávané funkcionality (cizí klíče, transakce, zamykání tabulek, poddotazy...), padla volba na systém MySQL. Ten je orientovaný zejména na rychlost a vhodný na menší projekty, jakým je i tento informační systém.

Jak jsem už přestřel, nevyhly se tomuto výběru komplikace – ty největší byly absence podpory cizích klíčů a transakcí. Tyto MySQL už od verze 4 plně podporuje, ač jenom v tabulkách typu InnoDB, proto nikoho nenapadlo, že na hostingu s MySQL sady 5 by s tím mohly být problémy. Poskytovatel hostingových služeb totiž na svých databázích ponechal jenom možnost vytvářet tabulky typu MyISAM, ve kterých podpora pro zmiňované vlastnosti dostupná není.

Tento fakt byl v našem případě horší o to, že se poskytovatel o této skutečnosti příliš nezmiňuje a DB při pokusech vytvořit InnoDB tabulku nebo použít na MyISAM tabulce transakci nehlásí chybu, jenom tyto nepodporované vlastnosti u operace ignoruje a provede ji podle dostupných možností. Jelikož jsem systém vyvíjel na vlastním serveru a na hosting nahrával zkušební verze jenom průběžně (s cílem zavčas objevit případné chyby vzniknuvší z přenosu; ty však na první pohled viditelné nebyly), byly tyto nedostatky objeveny celkem pozdě, což znamenalo zejména zpětnou práci na ošetření chybějící funkcionality aplikační cestou.

## 3. Návrh, realizace

### 3.1. Databázový model

Viz. E-R diagram na následující stránce.

#### 3.1.1. Návrh

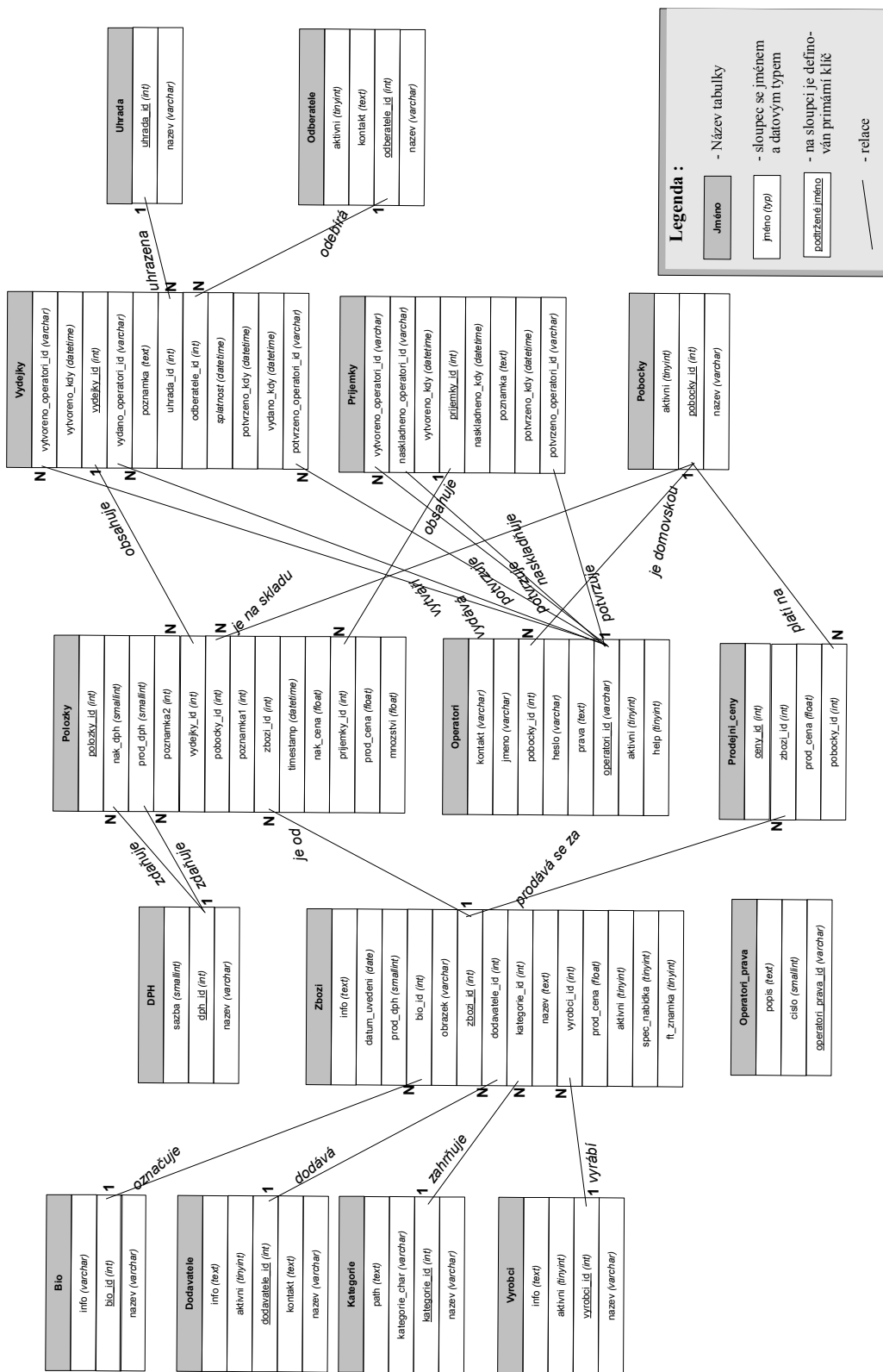
Podle zadání a předpokládaných potřeb aplikace vznikl model, který se po dalším upřesňování požadavků ustálil na situaci, kterou je možné ztvárnit následujícím entitně-relačním diagramem :

Vzhledem na omezení, vyplývající z omezení DB serveru (viz. předešlá kapitola), nutno podotknout, že v uváděných relacích není možno hledat cizí klíče – alespoň ne jako „pravé“ klíče definované na DB úrovni. Případná integritní omezení, která by jinak mohl být ošetřena na této úrovni, jsou řešena v aplikaci použitím zvláštních funkcí.

Názvy tabulek a sloupců jsou, pro eliminaci případných problémů s jejich identifikací, definovány bez použití diakritiky, interpunkce nebo mezer. Význam (či použití) jednotlivých tabulek by se dal shrnout v následovném, abecedně řazeném seznamu :

- **Bio** – používané „Bio“ známky, včetně dalších informací.
- **Dodavatele** – dodavatelé zboží, včetně kontaktních údajů a dalších informací.
- **DPH** – aktuálně používané sazby DPH pro prodej a nákup.
- **Kategorie** – kategorizace pro zboží; více viz. Kapitola 3.4.3.
- **Odberatele** – odběratelé zboží, včetně kontaktních údajů.
- **Operatori** – uživatelé systému, včetně jejich hesel (resp. kontrolních součtů MD5), práv, nastavení a dalších informací.
- **Operatori\_prava** – definice uživatelských práv a jejich popisky; více viz. Kapitola 3.2.2.
- **Pobočky** – přehled používaných poboček.
- **Polozky** – skladové položky, fyzická reprezentace zboží, včetně souvisejících kontextových informací.
- **Prijemky** – příjmové doklady, včetně podílejících se osob a dalších informací.
- **Prodejni\_ceny** – prodejní ceny pro dvojice zboží-pobočka.
- **Uhrada** – používané typy úhrad.
- **Vydejky** – výdejové doklady, včetně podílejících se osob, odběratele a dalších informací.
- **Vyrobcí** – výrobci zboží, včetně dalších informací.
- **Zbozi** – definice zboží, veškeré potřebné statické či přednastavené informace.

## Entitně – relační diagram, I.S. Pro Fair Trade



**Schéma 1: Entitně - relační diagram**

### 3.1.2. Realizace, optimalizace

U formování tabulek jsem se řídil několika konvencemi, které usnadňují orientaci ve struktuře DB :

- Jména tabulek začínají velkými písmeny, jména sloupců malými,
- pokud sloupec slouží jako primární klíč, jeho jméno nese název tabulky a končí na „\_id“,
- sloupce v různých tabulkách, které mají být propojeny relací, nesou, pokud možno, stejné jméno nebo alespoň jeho část.

Zejména poslední konvence často umožňovala v SQL dotazech používat místo „*JOIN tabulka2 ON (tabulka1.sloupec1 = tabulka2.sloupec2)*“ přehlednější formulaci „*JOIN tabulka2 USING (sloupec)*“. Pro klíče jsem volil datový typ „int“, který by měl být celkem dlouhou dobu postačující a v případě potřeby taky není problém rozšířit ho.

Optimalizace databáze – či případně dotazů, které jsou nad ní vykonávány – by jistě měla být také součástí návrhu, zejména definováním vhodných datových typů a vytvářením indexů nad vhodnými sloupci. Vyjma předefinování datových typů, které by u „zeštíhlování“ mohlo vést k poškození dat, je však činnosti vedoucí k optimálnější fungující databázi možno provádět za plného provozu systému. Navíc je tento přístup vhodnější kvůli tomu, že už po krátkém používání databáze jsou nám známy některé důležité fakty, které mohly být u návrhu jenom tušeny. Jedná se zejména o poměry zápisů a čtení z té-které tabulky (skutečnost důležitá u stanovování indexů : index rapidně pomáhá při čtení, u častějších zápisů však může být spíše na škodu) a relativní počty dat v tabulkách (skutečnost důležitá u případného dělení SELECT dotazů používajících klauzuli JOIN : zda-li je „nákladnější“ použít dva DB dotazy na tabulky a jejich propojení řešit na aplikační úrovni, nebo použít jeden dotaz, kde optimalizátor díky výrazně rozdílným počtům řádků v propojovaných tabulkách vytváří mohutné mezitabulky).

Už u vytváření aplikace jsem se snažil redukovat výrazně „slabé“ dotazy, například omezováním vnořených dotazů na minimum, či používáním klauzule LIMIT u dotazů s očekávaným počtem výsledných řádků. Jak jsem ale napsal v předchozím odstavci, řada optimalizačních kroků čeká až na ostrý provoz a větší množství dat v tabulkách. V systému je proto několik dotazů, o kterých se dá s velkou pravděpodobností říct, že optimální nejsou. Nechal jsem je ale v původním stavu, kde je sice použito např. několikanásobného propojení typu JOIN, je ale jasně znatelný požadovaný výsledek dotaz a člověk provádějící optimalizaci bude mít možnost problematický dotaz jednoduše lokalizovat a předat příkazu EXPLAIN, který je pomůckou při určování (ne)efektivity dotazů, místo aby ho musel složitě skládat a dohledávat v kódu aplikace.

### 3.1.3. Transakce

Jak už bylo naznačeno, na aktuálním databázovém serveru momentálně není možnost provádět DB operace v transakčním módu. To je v případě systémů, kde jde o víc než jenom obrázky nebo vtipy v databázi, celkem závažná věc : u selhání jednoho ze série dotazů, které na sebe navazují, může dojít buď k inkonzistenci databáze, nebo nepozorovanému „nepořádku“ na skladech, který se ale může při účetní uzávěrce ukázat jako velká nepříjemnost.

Univerzální „workaround“ pravděpodobně neexistuje. U několikanásobných operací typu UPDATE je možno tyto spojit v jeden dotaz – jestli jeden z poddotazů selže, ostatní se taky neprovedou a nastane něco jako rollback. U kombinované série dotazů tyto odevzdám speciální funkci jako pole dotazů – kdyby jeden z nich selže, funkce uloží do souboru všechny run-time informace, ukončí běh a poprosí

uživatelé, aby o této skutečnosti informoval některého ze správců, který pak situaci vyhodnotí a případný rollback provede manuálně.

## 3.2. Aplikační model

### 3.2.1. Struktura

Z předcházejících kapitol už je známo, že na aplikační úrovni bude použito skriptovacího jazyka PHP bez použití toho, co je v PHP zvané objektově-orientovaný přístup – a na druhou stranu s využitím modulů, které je možno podle potřeb vkládat do existujících nebo nových skriptů, takže je možno užitečné části kódu recyklovat.

Stejně jako u databázového návrhu jsem se i zde držel několika konvencí, které, pokud jsou známy, pomáhají v orientaci ve struktuře aplikace :

- skripty, které vykonávají funkcionalitu spojenou s vykreslováním rozhraní pro uživatele (tj. je uživatel může identifikovat jako „novou stránku“), jsou umístěny v hlavním adresáři a většinou nesou jméno, které aspoň zhruba identifikuje jejich účel (např. *vydejka.php*, *pobocky.php*),
- skripty, které jsou určeny jenom na vykonání operací – neprodukují výstup a po skončení jsou přesměrovány na jinou stránku – jsou umístěny tamtéž, ale jejich jméno je uvedeno podtržítkem (např. *\_vydejka\_akce.php*, *\_login.php*),
- skripty, které jsou zamýšleny jako moduly poskytující pomocnou funkcionalitu pro některé z výše uvedených (vykreslování, ověření přihlášení, definice funkcí), jsou umístěny v podadresáři *data/*.

Pro uživatele je jednoznačně identifikovatelných (po načtení stránky je vidí v adresovém řádku prohlížeče) jenom několik skriptů. Tím hlavním je *main.php*, který má na starosti vygenerování hlavičky stránky, vykreslení hlavního menu a zavolání modulu – skriptu pro vykreslení obsahu stránky. Dalšími jsou například *index.php*, který má na starosti přihlášení uživatele, či tiskové sestavy (*doklad.php*, *faktura.php*), generovány v novém okně jako dokumenty pro tisk nebo archivaci.

### 3.2.2. Role, Uživatelská práva

Ze zadání vyplývá požadavek poměrně podrobného udělování uživatelských práv. Tato se vztahují na jednotlivé operace nebo skupiny operací v systému.

Pro lepší pochopení budiž nápomocen následující diagram použití :

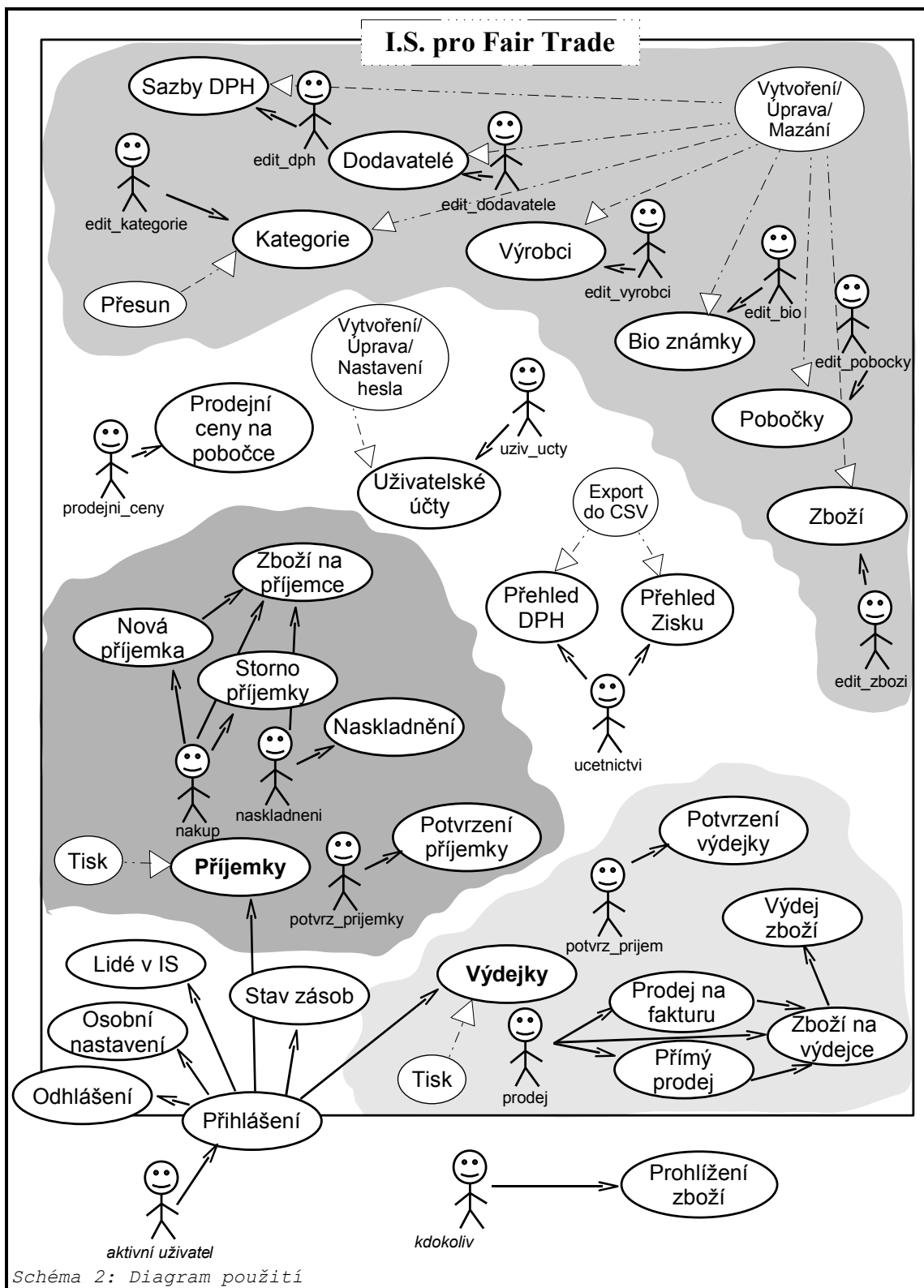


Schéma 2: Diagram použití

Můžeme zde vidět jednotlivé činnosti, tak jak jsou implementovány v systému. Názvy „rolí“ v grafu jsou identická s těmi, které má každý uživatel uložené v databázi. Tato jsou po přihlášení načtena a spolu s dalšími identifikačními údaji jsou uchovávány v globální proměnné `$_SESSION`.

Zvláštní práva, které nemohly být zaneseny do grafu, jsou :

- tzv. superuživatel, který má práva „fsechna“ a tímto je nadřazen nad všechna existující i budoucí práva; tato práva mu nebo jí můžou být přiděleny jenom dalším superuživatelem,
- práva zvaná „akce\_vsude“ samy o sobě uživateli nepřinášejí žádná další privilegia; avšak v kombinaci s právy, která jsou nějakým způsobem vztažena na pobočku (např. příjem zboží, přeskládňování atd.), umožňují uživatelům vykonávat operace i na jiné než domovské pobočce. U těchto operací je jim nabídnut výběr poboček, i když i tu je jako přednastavená hodnota domácí pobočka.

Dále je možno v diagramu vidět dvě role uvedené kurzívou :

- *aktivní uživatel* je vlastně každý uživatel jenž nemá (nebo jemuž od nich odhlédneme) přidělena žádná jiná práva,
- *kdokoliv* je libovolný návštěvník stránek fairtrade.cz. Je mu umožněno prohlížet si zboží a jeho detaily, vyhledávat podle kategorie a názvu a filtrovat podle dalších kategorií.

### 3.3. Uživatelské rozhraní

Pro výběr uživatelského rozhraní byl bez váhání zvolen značkovací jazyk HTML. Je tomu tak nejenom kvůli tomu, že je brán jako základní výstup PHP skriptů. Jeho přednostmi jsou vysoká standardizace (i když v každém prohlížeči nevypadají naprosto stejně, nejhorší co se může u případné nekompatibility stát, je kosmetický defekt na stránce) a nízké nároky na hardwarové a softwarové vybavení. Kvalitní HTML kód je navíc přinejhorším čitelný i v nepřeložené podobě.

Ačkoliv šla podpora standardů tohoto jazyka – a tím i možnost spolehnout se na korektní zobrazení stránky v neznámém prohlížeči – celkem slušně nahoru, ještě pořád není považováno za slušné ignorovat menšinové uživatele s prohlížeči nepodporujícími například zobrazování obrázků, zpracování CSS stylů či spouštění JavaScriptů. Takovými jsou například uživatelé se zrakovým postižením, starým hardwarovým vybavením, znakovým terminálem jako jedinou možností dostat se na internet – nebo paradoxně ti, co používají výdobytky moderní techniky v podobě kapesních počítačů či mobilních telefonů. Jejich prohlížeče často nemají podporu CSS stylů a JavaScriptů a/nebo si kvůli pomalému mobilnímu připojení vypínají zobrazování obrázků.

Takovým se často stává, že se je pro ně stránka neovladatelnou, nebo musí čekat nepřiměřeně dlouho na to, aby se jim načetla nepoužitelná stránka nepřipomínající nic z reálného světa. Tomuto jsem chtěl zabránit a tak jsem se pokoušel rozhraní stavět tak, aby příliš nediskriminovalo žádnou skupinu potencionálních uživatelů - minimálně na takové úrovni, aby bylo každému či každé umožněno, i když nepohodlně, plné využití možností systému.

O JavaScriptu platí předešlé dvojnásob. Ačkoliv je pomocí něj prohlížení stránek neuvěřitelně příjemné, je problém napsat takový kód, aby spolehlivě běžel aspoň ve třech nejrozšířenějších prohlížečích. Dost často je to řešeno (viz. například i skript pro zobrazování kontextové nápovědy, který jsem použil v systému) počátečním testováním druhu prohlížeče a následným větvením kódu doufajíce, že se ostatní prohlížeče zařadí do té správné větve - nehledě na to, že se tím zvětšuje zátěž linky kvůli stahování „nepoužitelného“ kódu. Další nepříjemností je softwarová náročnost JavaScriptu ve větším



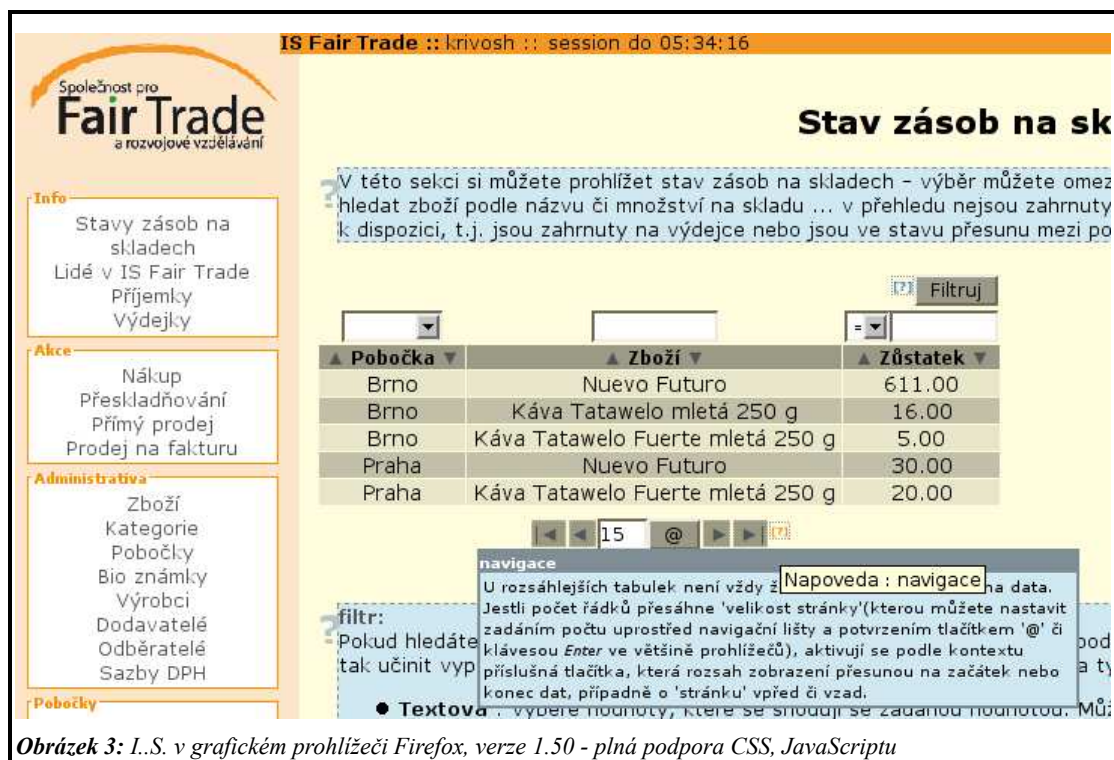
množství, způsobující pomalejší odezvu stránek či prohlížeče a sekundárním omezením výkonu i pro ostatní programy na klientském počítači.

Proto jsem u implementace rozhraní využíval JavaScriptu jenom tam, kde to bylo opravdu nevyhnutné a žádoucí – konkrétně například u kontextové nápovědy, u které stačí najet kurzorem nad ikonu otazníku a objeví se nápovědná zpráva v místech, kterých se přímo týká. Ani uživatelé bez fungujícího JavaScriptu ale nejsou úplně mimo hru – na stejnou zprávu bude v rámci stránky (většinou na její konec) přesunut uživatel, který na ikonku klikl.

Pro srovnání uvádím tzv. screenshoty systému zobrazeném v prohlížečích s různou mírou podpory „problémových“ webových technologií :



V grafickém prostředí jsem zvolil výřez obrazovky v sekci „Stav zásob na skladech“ se zapnutou kontextovou nápovědou. V době snímku byl kurzor právě nad ikonou nápovědy pro navigační lištu :



### 3.4. Vybrané problémy

Pro podrobnější přehled funkcí systému doporučuji přílohu A – Uživatelský manuál – ve kterém je popsána veškerá funkcionální systém, včetně obrazové dokumentace.

Pro detailní informace o kódu a jednotlivých modulech je dostupná příloha B – Programová dokumentace – v provedení statického HTML webu a PDF. Kromě souhrnných přehledů se tam nacházejí syntakticky zvýrazněné a komentované zdrojové kódy celého systému.

Pro detailnější popisy jsem vybral následovné části :

#### 3.4.1. Modul pro prohlížení dat – tabulka

Jednou z nejčastějších požadavků nejenom v tomto systému je přehledný pohled na vybraná data – a jelikož je zde použita relační databáze, je nejlogičtější pohledem právě tabulka.

I když je celkem pravděpodobné, že bych s trochou snahy mohl použít už existující funkční řešení – ať už v jiné open-source aplikaci (např. phpMyAdmin), nebo přímo dedikovanému projektu – rozhodl jsem se jít vlastní cestou a vytvořit vlastní řešení.

V souladu se zadáním a tím, co jsem napsal o mém postoji k webovým rozhraním v kapitole 3.3. by toto řešení mělo být, v případě zájmu mých následovníků či následovnic, co nejsnadněji adaptovatelné i v dalších pokračováních programové podpory pro Fair Trade a taky použitelné i v prohlížečích s omezenou podporou webových technologií.

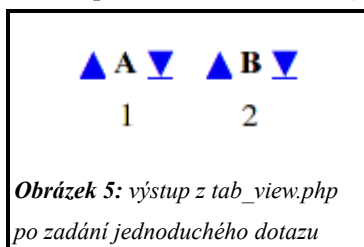
Vznikl proto modul *tab\_view.php*, který slouží na zobrazení (My)SQL dotazu v tabulce, s volitelnou možností vyhledávání v datech a řazení podle zvoleného kritéria. Zvolil jsem metodu přímého vložení kódu, který předpokládá nastavení jediného povinného parametru – SQL příkaz SELECT dotazu, který má vykreslit – a zároveň umožňuje různá „vylepšení“ parametry volitelnými. Metodu volání funkcí jsem zavrhl hned z několika důvodů :

- relativně složitější volání – kromě direktivy na vložení definice funkce, *include 'tab\_view.php';* která je v současné podobě postačující, by se musela volat samotná funkce,
- nelze zde využít žádné výrazné výhody, které obvykle použití funkcí přináší – vrácení hodnot, vícenásobné použití funkce, rekurze, vlastní adresný prostor a podobně,
- parametry se nemusí zadávat v řadě mezi závorkami volání funkce, což může být u většího množství složitějších proměnných (např. asociativní pole polí) celkem příjemné z pohledu přehlednosti kódu,
- menší paměťová náročnost, i když ta nezvykle hrát u serverů s PHP příliš důležitou roli

Jedním z nejjednodušších volání modulu tedy může být například :

```
$tab_sql = "SELECT 1 AS A, 2 AS B";  
include ('data/tab_view.php');
```

Ten by měl vrátit tabulku o dvou sloupcích a dvou řádcích (včetně halvičky).



Tento příklad samozřejmě postrádá – kromě smysluplných dat – také definici stylu, který se očekává v externím CSS stylesheetu. Volitelné parametry jsou popsány ve zdrojovém kódu modulu. S jejich použitím může, ve souznění se šablonou I.S. Fair Trade, výstup modulu vypadat například i takhle :

▲ Zboží ▼	▲ Množství ▼	▲ -i- ▼	▲ Na přesun ▼
Káva Tatawelo Fuerte mletá 250 g	20.00	<a href="#">info</a>	5
Nuevo Futuro	5.00	<a href="#">info</a>	
Nuevo Futuro	12.00	<a href="#">info</a>	3
Nuevo Futuro	13.00	<a href="#">info</a>	

**Obrázek 6:** výstup z `tab_view.php` po složitějším dotazu a CSS stylování

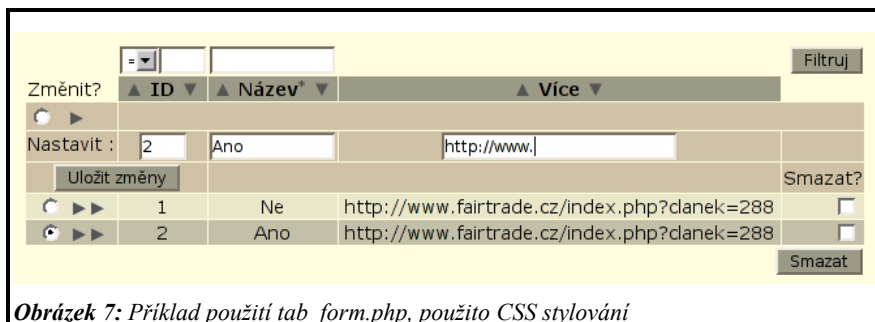
### 3.4.2. Modul pro editaci dat – formulář

Další podobně frekventovanou činností je jednoduchá aktualizace dat v tabulce, případně přidávání a mazání hodnot. Vycházejíc z předchozího modulu, nazval jsem tento modul `tab_form.php`. Ač pracují na podobném principu a vycházejí ze stejného základu, je tady jeden velký rozdíl : Jenom hodně těžko bychom mohli aktualizovat libovolnou tabulku – ta mohla vzniknout kombinací různých jiných tabulek, případně vložením úplně fiktivních sloupců.

Modul má opět jediný povinný parametr – název tabulky s případným omezením na zobrazená data. Ta mohou být zadána klauzulemi WHERE a GROUP BY. Funkcionalitu JOINů (např. záměna „cizích klíčů“ za příslušné hodnoty) zde částečně nahrazuje parametr \$trans, který je možno naplnit asociativním polem hodnota->jméno - a narazí-li modul u výpisu dat na hodnotu z pole, nahradí j=i jménem z dvojice.

Výhodou oproti `tab_view.php` zde je, že tabulku (resp. její hlavičku a formulář) může vypsát i když neobsahuje žádná data – identifikátory sloupců totiž bere nikoliv z průchodu prvního řádku vrácených dat, ale používá SQL příkazu DESCRIBE, který zároveň vrátí i informace o sloupci s primárním klíčem, s povinnou ne-NULL hodnotou atd.

Podrobný seznam dalších nepovinných parametrů je k nalezení ve zdrojovém kódu modulu (viz. například přílohu B), bližší informace o používání jsou k nalezení i v příloze A, Uživatelském manuálu.

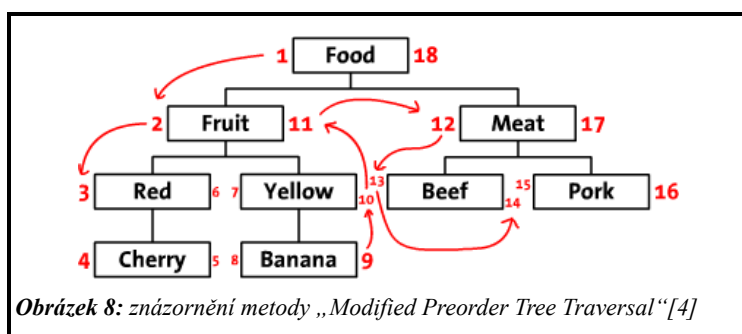


### 3.4.3. Stromové menu

Jednou z výzev zadání byla kategorizace zboží do hierarchické struktury – stromu. V případě, že se, stejně jako tady, předpokládá neomezená šířka a hloubka stromu a ukládání stromu do relační databáze, jedná se o zajímavý problém, který zatím nemá „univerzálně dobré“ řešení.

Na první pohled nejsnadnějším a nejpřirozenějším řešením bývá ke každému uzlu stromu evidovat uzel rodičovský – jejich teoretickým propojením je možno vybudovat celý strom. Ohromnou nevýhodou této metody je, že spoléhá na rekurzivní funkci, která potřebuje pro zjištění každé větve stromu právě jeden dotaz na databázi, co jí činí prakticky nepoužitelnou už u středně velkých stromů. Naopak, zápis nového uzlu kamkoliv do databáze je velice jednoduchý, avšak v našem případě je plánovaný poměr čtení a aktualizací stromu výrazně ve prospěch čtení.

Dalším řešením bývá budování více nebo méně složitých indexovacích informací. Ty většinou dávají celkem slušné výsledky při čtení, avšak i u malých změn se často musí celý index vybudovat odznovu, navíc většinou opět s použitím rekurze. Jedním z takových je tzv. „Modified Preorder Tree Traversal“ [4], který si u každého uzlu eviduje číselné ID levého a pravého souseda a uzel, resp. jeho potomky, pak lokalizuje přes rozdíly těchto indexů :



Obrázek 8: znázornění metody „Modified Preorder Tree Traversal“ [4]

Pak jsem narazil na zmínky o tzv. „genealogickém přístupu“, který si sice taky u nod vytváří doplňkové informace, avšak v celkem přirozeném a hlavně snadno zpracovatelném formátu. Ve zkratce jde u tohoto přístupu o přiřazení jedinečného identifikátoru ke každému uzlu a evidenci „cesty“ přes tyhle identifikátory u každého uzlu. Různé přístupy se liší v názoru, jak má identifikátor vypadat : jestli to je jeden znak[5] (resp. hodnota o typu *byte*), dává to možnost mít ve stromu 256 uzlů, ať už v jedné úrovni vedle sebe, nebo v 256 úrovních za sebou – nebo něco mezi tím.

Strom kategorií	Identifikátor	Cesta
•	0	–
└─ Potraviny	A	0
└─┬─ Jídlo	B	0A
└─┬─ Nápoje	C	0A
└─┬─┬─ Káva	D	0AC
└─┬─┬─ Ostatní	E	0A
└─ Tiskoviny	F	0
└─ Oblečení	G	0
└─ Ostatní	H	0

**Tabulka 1 :** Příklad DB reprezentace kategorií genealogickým přístupem

Problém – a jeden z hlavních terčů kritiky tohoto přístupu – je hlavně datový typ, do kterého se ukládá cesta : v tomto, nejjednodušším případě se stejně musí pro „cestu“ rezervovat 255 znaků. A ač to není příliš pravděpodobné, pro praxi to znamená, že jestli má být toto řešení opravdu univerzální, je nutno pro „cestu“ vyhradit datový typ *text*, nad kterým se, kromě jiného, operace vykonávají relativně delší dobu než nad datovými typy s pevnou délkou. Otázkou – podle mých zjištění zatím nezodpovězenou – zůstává, jestli tahle „ztráta“ (SELECTy s použitím textových funkcí plus datový typ *text*) převáží zisk, který oproti ostatním metodám tato získává bezkonkurenčně nejjednoduššími operacemi pro čtení, aktualizace i vkládání nových uzlů do stromu.

Jelikož ale tuto metodu využívají i diskusní servery s desítkami uzlů (vím alespoň o jednom : [kyberia.sk](http://kyberia.sk)), rozhodl jsem se taky k hledání odpovědi připojit. Z datového typu *text* pak lze udělat znouzectnost : při relativně volné délce identifikátoru, jeho definicí jako řadícího klíče a použití vyhrazeného znaku oddělovače v zápisu cesty je možno při dotazu na hierarchii stromu tento hned i řadit podle onoho klíče. Předchozí příklad pak v databázi, v tabulce Kategorie, vypadá následovně :

kategorie_id	kategorie_char	path	nazev
1	0		
2	+potraviny	0	Potraviny
3	jídlo	0-+potraviny	Jídlo
4	napoje	0-+potraviny	Nápoje
5	_ostat_jdl	0-+potraviny	Ostatní
6	_obleceni	0	Oblečení
7	tiskoviny	0	Tiskoviny
8	_ostatni	0	Ostatní
9	kafe	0-+potraviny-napoj	Káva

**Tabulka 2:** DB reprezentace kategorií v systému

„Identifikátor“ zde nahradil název kategorie\_char a ačkoliv musí být jedinečný, je žádoucí možnost ho měnit, proto do tabulky přibyl primární klíč kategorie\_id, který kategorii jednoznačně charakterizuje. Jsou v něm dovoleny jenom znaky a-z a '+', '\_', díky tomu je pomocí něj snadno a jednoznačně řadit uzly v rámci jedné úrovně; pomlčka v cestě (path) zas slouží jako oddělovač. Dotaz :

```
SELECT K.path, CONCAT(K.path, '-', K.kategorie_char) fullpath, kategorie_id,
kategorie_char, nazev, kids FROM Kategorie K LEFT JOIN (SELECT path, COUNT(*) kids FROM
Kategorie GROUP BY path) P ON (CONCAT(K.path, '-', K.kategorie_char) = P.path) ORDER BY
fullpath DESC
```

pak vygeneruje strom, který byl k vidění v prvním příkladu – odzadu. Kdybychom si vystačili s jednoduchým stromem, kde jsou jednotlivé úrovně odsazeny jenom počtem odsazovacích znaků (např. mezer), nebylo by jednak potřeba v dotazu zjišťovat „kids“, tj. počet childů každého uzlu (tahle informace je ale užitečná i z jiných důvodů, např. možnost mazání) a hlavně by bylo možné strom seřadit

„odpředu“ a přímo vypsát. Není však od věci, ani příliš náročné, zjistit strukturu tohoto odsazení už při prvním průchodu (všechny algoritmy na vykreslení stromu používali několik průchodů, tenhle jsem si vytvořil protože mi to přišlo zbytečné) – je k tomu ale logicky potřeba strom projít „odspodu“, do výstupního bufferu dáta vpisovat taky „odzadu“ a tento po průchodu najednou vypsát.

Pro vykreslování větví/odsazení stromu jsou většinou používány obrázky – to ale koliduje s mým cílem mít systém široce přístupný, proto na jejich vykreslení využívám 4 znaky Unicode ze sady WGL4, které jsou považovány za všeobecně dostupné : |, ⌋, ⊥, ⊞. Tímto zároveň umožňuji uživatelům přizpůsobovat si velikost písma ve stromu, kde se, na rozdíl od populárních obrázků, tyto znaky zvětší zároveň s nimi. Po menším zkoumání struktur stromů jsem přišel na to, že vhodnou substitucí těchto znaků za čísla 4-kové soustavy, se kterými se při přechodu na další řádek děje jenom jedna ze třech matematických operací. Pro bližší informace doporučuji zdrojový kód (např. *kategorie.php*), kde je celý proces podrobně popsán.

Mazání nódů či jejich přesuny i s potomky se taky vejdu do jednoho nebo dvou dotazů, ohledně detailů rovněž doporučuji nahlédnout do zdrojového souboru (*\_kategorie.php*).

## 4. Závěr

Cílem této práce nebylo napsat uzavřenou práci, která by po obhájení skončila v regálu knihovny a pravděpodobně už neužřela světlo světa. Naopak, byla psána pro existující organizaci, která je v současnosti ve stadiu rozšiřování, co do počtu poboček i objemu prodeje. Dosavadné řešení skladového hospodářství (a správy zdrojů vůbec), kde bylo nutné využívat několik vzájemně nespolupracujících aplikací a nejednou i přenosu informací „na papíře“, už proto přestalo být postačující. Informační systém, kterého základy jsem postavil, ještě zdaleka není ve své finální podobě - mé kolegy či kolegyně, kteří budou pokračovat ve vytváření dalších funkčních částí systému, čeká ještě kus práce.

Naštěstí však budou moci, díky modulární stavbě systému, využívat návrh rozhraní a vhodných univerzálních prvků, které systém používá. Taky u vytváření databázového modelu jsem se snažil o univerzálnost, aby každé další řešení mohlo na to současné přímo navazovat aneb byly potřebné přílišné zásahy do struktury dat.

I když se už v budoucnu pravděpodobně k tomuto projektu nevrátím s takovým nasazením, jako tomu bylo teď, neznamená to úplné přerušení spolupráce se Společností – slíbil jsem jim podporu v případě, že zrovna nebudou mít nikoho, kdo by byl schopen či ochoten pomoci s problémem či změnami na systému, který jsem jim právě odevzdal.

Věřím však, že se o českém Fair Trade v budoucnu doslechnu i jinak než z e-mailu s přemetem obsahujícím „chyba“ či „problém“ - tento projekt totiž má význam a co víc, má šance na úspěch. Jestli si pak budu moci říct, že jsem u toho taky kdysi stál, bude to příjemný pocit. I když to byl „jenom“ informační systém.



## 5. Literatura

- [1] Společnost pro Fair Trade a rozvojové vzdělávání : *Fair Trade*, URL : <http://www.fairtrade.cz/index.php?clanek=10> (leden 2007)
- [2] Achour, Mehdi aj. : *Manuál PHP*, kapitola 1, URL : <http://cz.php.net/manual/cs/introduction.php> (leden 2007)
- [3] Dudek, Jan : *Vývoj PHP - cesta správným směrem?* , magazín Interval.cz, 24. 1. 2005. URL : <http://interval.cz/clanky/vyvoj-php-cesta-spravnym-smerem/> (leden 2007)
- [4] Tulder, Gijs Van : *Storing Hierarchical Data in a Database*, sitepoint.com, 30.4.2003. URL : <http://www.sitepoint.com/article/hierarchical-data-database> (leden 2007)
- [5] Švec, Petr : *Diskuse vs. SQL*, . Builder - Informační server o programování, 8.8.2002. URL : [http://www.builder.cz/art/homepage/discuss\\_sql.html](http://www.builder.cz/art/homepage/discuss_sql.html)

## 6. Přílohy

### 6.1. A : Uživatelský manuál

Jako první příloha je uveden uživatelský manuál. Sice je vytvořen a zamýšlen jako návod k použití systému pro budoucí uživatele, může ale taky posloužit jako menší prohlídka jeho možností a fungování.

Jestli čtete tištěnou verzi, měl by být manuál přiložen hned za touto prací. V elektronické verzi ho najdete v souboru *Manual\_FairTrade.pdf*.

### 6.2. B : Programová dokumentace

Druhou přílohou je programová dokumentace k aplikaci, generována automaticky ze zdrojových kódů za použití nástroje [phpDocumentor](#). Obsahuje komentovaný přehled funkcí, modulů a zdrojových souborů s odkazy na zdrojové kódy, které jsou taky k dispozici v barevně zvýrazněné formě. Zdrojové kódy jsou pro velký rozsah (cca 140 stran A4) přítomny jenom v elektronické verzi správy, a to tak ve formátu PDF (soubor *Programova\_dokumentace.pdf*) vhodném pro tisk, tak v HTML (soubor *Programova\_dokumentace/index.html*) vhodném pro pohodlné prohlížení ve webovém prohlížeči.

INFORMAČNÍ SYSTÉM

FAIRTRADE.CZ

# Uživatelský manuál



*Aktuální verze : 22.1.2007*



# 1. Úvod

Vítejte v průvodci Informačním systémem pro FairTrade. Jeho úkolem je především seznámit Vás s jeho možnostmi a funkcemi, dále pak upozornit na některá úskalí a naopak výhody, které nemusí být zřejmé na první pohled.

Systém je pořád otevřen dalšímu vývoji, proto v době, kdy čtete tyto řádky, nemusí jeho stav a funkcionality zodpovídat tomuto manuálu. A naopak, pokud máte návrhy na jeho zlepšení či v systému objevíte chybu či nedostatky, neváhejte a kontaktujte autorský team.

Tento manuál z velké části kopíruje kontextovou nápovědu, která je obsažena v samotném systému (jak ji vypnout/zapnout se dočtete v kapitole 25) – navíc však obsahuje :

- informace, které se v kontextové nápovědě nemusí zobrazit "na první pohled", tj. může sloužit jako vodítko k některým funkcím systému
- informace o celém systému, tj. i o součástech, na které Vaše uživatelské práva (máte-li už v systému účet) nejsou dostačující
- doplňkové informace, které se nevešly nebo nehodily do kontextu nápovědy

Manuál je rozdělen do kapitol, které korespondují s částmi systému tak, jak je můžete navštívit v hlavním menu (viz. taky horní vnější roh každé stránky manuálu) – a podle menu jsou kapitoly taky řazeny.

Základ každé stránky manuálu tvoří tzv. screenshot (obrázek části nebo celé původní obrazovky) z části systému, která je zrovna popisována. Jak už bylo napsáno, komentáře k ní mohou být buď

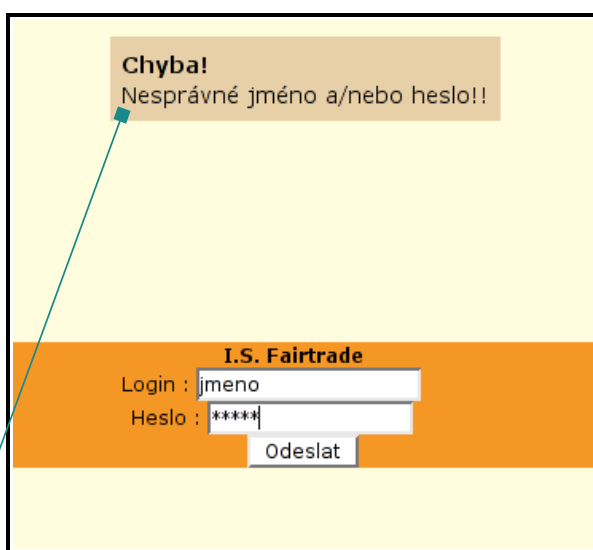
kopíí kontextové nápovědy – ty budou ovedeny takovýmto písmem a rámečkem,

nebo doplňující, nacházející se jenom v tomto manuálu – ty budou uvedeny standardním textem.

Kontextová nápověda (nebo komentář, bude-li to možné) bude taky do obrázku uvedena čarou na místo nebo místa, ke kterým se vztahuje.

## 2. Přihlášení

Prvním krokem do systému je, jak to bývá zvykem, přihlášení se pod svým přihlašovacím jménem a heslem. Tyto Vám vytvořil některý z oprávněných uživatelů; pokud jste si heslo po prvním přihlášení změnil/a, neexistuje reálná možnost ho ze systému zjistit. Proto v případě, že heslo zapomenete, kontaktujte opět někoho z uživatelů s oprávněním vytvářet a nastavovat cizí účty (kapitola 6) - vaše zapomenuté heslo sice nezjistí, ale může Vám nastavit nové, které si pak opět můžete změnit podle svých představ (viz. kapitola 25).



The screenshot shows a login interface for 'I.S. Fairtrade'. At the top, there is a yellow box with the text 'Chyba! Nesprávné jméno a/nebo heslo!!' (Error! Incorrect name and/or password!!). Below this, the login form has two input fields: 'Login : jmeno' and 'Heslo : \*\*\*\*\*'. A blue arrow points from the error message box to the 'Login' field. At the bottom of the form is a button labeled 'Odeslat' (Send).

Po zadání nesprávného jména a/nebo hesla budete na tuto skutečnost upozorněni; jestli zadáte jméno a heslo správně, budete přesměrováni na úvodní stránku.



The main menu of the Fair Trade system is displayed on the right side. It features the logo 'Společnost pro Fair Trade a rozvojové vzdělávání' at the top. Below the logo, there are several categories of links, each in a box with an orange header:

- Info**
  - Stavy zásob na skladech
  - Lidé v IS FairTrade
  - Příjemky
  - Výdejky
- Akce**
  - Nákup
  - Přeskladňování
  - Přímý prodej
  - Prodej na fakturu
- Administrativa**
  - Zboží
  - Kategorie
  - Pobočky
  - Bio známky
  - Výrobci
  - Dodavatelé
  - Odběratelé
  - Sazby DPH
- Pobočky**
  - Prodejní ceny
- Účetnictví**
  - Přehled DPH
  - Přehled Zisku
- Systém**
  - Uživatelské účty
- Osobní**
  - Změna údajů, nastavení
  - = ODHLÁŠENÍ - =



### 3.Tabulkové formuláře a pohledy na data

Během práce s informačním se budete často setkávat se dvěma druhy "standardních" prvků, které jsou si v mnohém podobné a umožňují pohodlnou a funkční práci s daty.

Prvním z nich je tzv. tabulkový pohled na data, který poskytuje přehled údajů v tabulce; položky jsou řazeny do řádků, ve kterých sloupce představují jednotlivé údaje. Podle nich je možno měnit pořadí a často i filtrovat podmnožiny zobrazených dat, viz. příklad :

▲ Pobočka ▼	▲ Zboží ▼	▲ Zůstatek ▼
Brno	Nuevo Futuro	611.00
Brno	Káva Tatawelo mletá 250 g	16.00
Brno	Káva Tatawelo Fuerte mletá 250 g	6.00
Praha	Nuevo Futuro	30.00
Praha	Káva Tatawelo Fuerte mletá 250 g	24.00

**Filtr :** Pokud hledáte konkrétní data v tabulce, či chcete nechat zobrazit jenom podmnožinu údajů z tabulky, můžete tak učinit vyplněním políček nad sloupce, kterých se budou filtrovací kritéria týkat. Ty mohou být třech druhů :

- **Textová :** vybere hodnoty, které se shodují se zadanou hodnotou. Můžete použít zástupné znaky %(procento) pro libovolný počet libovolných znaků(např. 'Ivan%' najde shodu na 'Ivan', 'Ivana', 'Ivanovice...') a \_(podtržítko) pro právě jeden libovolný znak(např. 'Ivan\_' najde shodu na 'Ivana' či 'Ivane', ne však na 'Ivan' či 'Ivanka...'). Hledání většinou nerozlišuje velká a malá písmena

- **Matematická :** nabízí základní aritmetické operace(výběr vlevo vedle políčka) pro práci s čísly. Pozor, desetinná místa jsou oddělena tečkou, ne čárkou!

- **Výběrová :** umožňuje vybírat jenom jednu z uzavřeného výběru hodnot.

Filtrování zrušíte smazáním filtrovacích kritérií. Potvrdit výběr kritérií můžete ve většině prohlížečů kromě tlačítka 'Filtruj' také zmáčknutím klávesy *Enter* z některého z filtrovacích polí.

**Navigace :** U rozsáhlejších tabulek není vždy žádoucí zobrazovat všechna data. Jestli počet řádků přesáhne 'velikost stránky'(kterou můžete nastavit zadáním počtu uprostřed navigační lišty a potvrzením tlačítkem '@' či klávesou *Enter* ve většině prohlížečů), aktivují se podle kontextu příslušná tlačítka, která rozsah zobrazení přesunou na začátek nebo konec dat, případně o 'stránku' vpřed či vzad.

Rozšířením tabulkového pohledu je tabulkový formulář, který navíc umožňuje zobrazená data upravovat, mazat či doplňovat. Používá se všude tam, kde jde o jednoduchý zápis dat a samozřejmě na tuto činnost máte příslušná práva.

**Změna :** V levém sloupci můžete vidět dva druhy ovládacích prvků v každém řádku : tzv.'checkbox', čili zaškrtnávací políčko a šipky směřující na příslušný řádek.

Šipky slouží na doplnění hodnot z daného řádku do vstupních polí formuláře (ta můžou být, podle zvládnutí tvůrce stránky, pod a nebo nad datovými řádky). Ačkoliv zároveň nastaví onen řádek jako vybraný pro zápis, nemusí to být pravidlem, protože

checkbox slouží právě na označení zapisovaného řádku (ať už půjde o vytvoření nového nebo přepsání existujícího). Můžete tak například snadno vytvářet obsahově příbuzné řádky bez potřeby opisování všech dat, když nejprve šipkami do vstupních polí vložíte existující hodnoty a pak checkboxem označíte 'nový' řádek.

I když by ve většině případů mělo fungovat odeslání změn zmáčknutím klávesy Enter z některého ze vstupních polí, u některých prohlížečů nezbyvá než používat tlačítka 'Uložit změny'.

**Smazat :** Položky je možno smazat označením příslušných řádků a potvrzením tlačítkem 'Smazat'. Zejména v případech, kdy by mohly být narušeny odkazy na položky z jiných tabulek, je možné, že některé položky smazat nebude možno. V tom případě budou odmítnuty všechny řádky označené na mazání; problémové řádky budou oznámeny.

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

---

**Info**

Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

---

**Akce**

Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

---

**Administrativa**

Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

---

**Pobočky**

Prodejní ceny

---

**Účetnictví**

Přehled DPH  
Přehled Zisku

---

**Systém**

Uživatelské účty

---

**Osobní**

Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 4. Úvodní obrazovka

V úvodní obrazovce, přihlásíte se-li poprvé nebo nemáte vypnuté zobrazování nápovědy, se dovíte o tom, jak vypadá a používá se kontextová nápověda. Kromě jiného se dočtete, že ač by mělo být prohlížení nápovědy funkční ve všech prohlížečích, se zapnutou podporou *JavaScript*-u získáte výhodu zobrazení nápovědy přímo u kurzoru při najetí nad ikonku nápovědy u komentovaného místa. Další důležitou informací je, že si zobrazování nápovědy můžete vypnout či zapnout v sekci *Osobní nastavení* (kapitola 25).

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 5. Stav zásob na skladech

V této sekci si můžete prohlížet stav zásob na skladech - výběr můžete omezit jenom na vybranou pobočku, hledat zboží podle názvu či množství na skladu ... v přehledu nejsou zahrnuty položky, které momentálně nejsou k dispozici, t.j. jsou zahrnuty na výdejce nebo jsou ve stavu přesunu mezi pobočkami.

▲ Pobočka ▼	▲ Zboží ▼	▲ Zůstatek ▼
Brno	Nuevo Futuro	611.00
Brno	Káva Tatawelo mletá 250 g	16.00
Brno	Káva Tatawelo Fuerte mletá 250 g	6.00
Praha	Nuevo Futuro	30.00
Praha	Káva Tatawelo Fuerte mletá 250 g	24.00

## 6. Lidé v IS FairTrade

V této sekci si můžete prohlížet další aktivní uživatele či uživatelky tohoto informačního systému - výběr můžete omezit jenom na vybranou pobočku, nebo hledat člověka podle jména. Toto je taky správné místo pokud sháníte někoho, kdo může upravovat účet váš (změna práv, domovské pobočky) či lidí, kteří mají problém dostat se do systému (zapomenuté heslo, zablokovaný účet).

▲ Login ▼	▲ Jméno ▼	▲ Pobočka ▼	▲ Účty? ▼
krivosh	Admin Krivosh	Brno	Ano
nig	Niguel Mauricio	Praha	Ne
pavelch	Pavel Chmelar	Brno	Ano
petrch	Petr Chmelar	Brno	Ano

**Niguel Mauricio**

Login : nig  
Pobočka : Praha  
Kontakt : nig.m@nuevo.cz

*Uživatel/ka může nastavovat a vytvářet uživatelské účty*

> [Na seznam lidí](#)

> [Změnit údaje](#)

**Účty :** Sloupec dává informaci o tom, jestli dotyčný nebo dotyčná má práva nastavovat cizí uživatelské účty.

**Login :** Po kliknutí na některý z loginů v tomto sloupci se zobrazí další informace o vybrané osobě.

Jestli máte práva nastavovat cizí uživatelské účty, objeví se Vám v detailech osoby taky odkaz "Změnit údaje". Tento vás přesune do sekce *Uživatelské účty* a nastaví na změnu příslušnou osobu.



## 7. Příjemky

Každý příjem zboží patří pod nějakou příjemku - zde vidíte jejich přehled, s mnoha možnostmi řazení a filtrování - více viz. nápověda nad jednotlivými sloupci.

**Stav :** Příjemka se může nacházet ve stavu :

- **Otevřena** do doby, než je zboží přijata na sklad
- **Naskladněna** po přijmutí zboží(naskladnění); v tomto stavu je u příjemky možno měnit jenom poznámku, máte-li na to dostatečná práva
- **Potvrzena** je označení příjemky která je definitivně v pořádku a není třeba s ní provádět další operace
- **Storno** je vlastně zrušená příjemka; pro pořádek účetnictví by označování dokladů mělo být souvislé, proto chtěl-li někdo smazat např. příjemku č.21 když už existovala 22, musela být místo opravdového smazání 'stornována', aby nevznikla 'mezera'

Po kliknutí na stav budou zobrazeny detaily příjemky.

Více o zobrazení detailů příjemky najdete v kapitole 9.

**Akce :** Sloupec dává informaci o tom, kdo a kdy na příjemce vykonal onu akci. Podle pravidel filtrování(viz. nápověda u tlačítka 'Filtruj') můžete například vybrat akce na příjmkách z ledna 2007(hledat 2007-01%) či od některého z rodiny Novákových (hledat %Novák%)

řádky 3-10 z 10 ::

-i-	ID	Vytvořeno	Naskladněno	Potvrzeno	Suma	Poznámka, popis...
Otevřena	P-2007-18	2007-01-15 (Pavel Chmelař)	-	-	-	-
Otevřena	P-2007-15	2007-01-12 (Pavel Chmelař)	-	-	396.00	-
Potvrzena	P-2007-12	2007-01-12 (Pavel Chmelař)	2007-01-12 (Pavel Chmelař)	2007-01-12 (Pavel Chmelař)	3888.00	-
Potvrzena	P-2007-9	2007-01-10 (Admin Krivosh)	2007-01-10 (Admin Krivosh)	2007-01-10 (Admin Krivosh)	50.00	jenom test
Naskladněna	P-2007-4	2007-01-10 (Admin Krivosh)	2007-01-10 (Admin Krivosh)	-	1405.00	-
Naskladněna	P-2007-2	2007-01-10 (Admin Krivosh)	2007-01-10 (Niguel Mauricio)	-	3648.00	poznámka2
Potvrzena	P-2007-1	2007-01-08 (Petr Chmelař)	2007-01-08 (Petr Chmelař)	2007-01-10 (Admin Krivosh)	77760.00	poznámka 123

< < 7 @ > >

> Nový příjem

**Suma :** Informace o sumě která byla/má být zaplacená za položky na příjemce, v Kč včetně DPH.

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 8. Výdejky

Každý výdej zboží patří pod nějakou výdejku - zde vidíte jejich přehled, s mnoha možnostmi řazení a filtrování ... viz. např. sloupce :

**Stav** : Výdejka či faktura se může nacházet ve stavu :

● **Otevřena** do doby, než je zboží vydána ze skladu

- **Vydána** po vyjmutí zboží ze skladu; v tomto stavu je u výdejky či faktury možno měnit jenom druhotné informace, máte-li na to dostatečná práva
- **Potvrzena** je označení výdejky či faktury, která je definitivně v pořádku a není třeba s ní provádět další operace
- **Storno** je vlastně zrušená výdejka; pro pořádek účetnictví by označování dokladů mělo být souvislé, proto chtěl-li někdo smazat např. výdejku č.21 když už existovala 22, musela být místo opravdového smazání 'stornována', aby nevznikla 'mezera'

Po kliknutí na stav budou zobrazeny detaily výdejky či faktury.

**Akce** : Sloupec dává informaci o tom, kdo a kdy na výdejce nebo faktuře vykonal onu akci. Podle pravidel filtrování (viz. nápověda u tlačítka 'Filtruj') můžete například vybrat akce z ledna 2007 (hledat 2007-01%) či od některého z rodiny Novákových (hledat %Novák%)

**Suma** : Informace o sumě která byla/má být zaplacená za položky na výdejce či faktuře, v Kč včetně DPH.

(pobočky\_id LIKE 'Brno') : řádky 5-9 z 13 ::

-i-	ID	Pobočka	Vytvořeno	Vydáno	Potvrzeno	Suma	Odběratel	Poznámka, popis...
Otevřena	V-2007-21	Brno	2007-01-15 (Admin Krivosh)	-	-	-	-	-
Otevřena	V-2007-20	Brno	2007-01-15 (Admin Krivosh)	-	-	-	Tesco	Přímý prodej
Vydána	V-2007-16	Brno	2007-01-12 (Pavel Chmelar)	2007-01-12 (Pavel Chmelar)	-	1005.73	-	Přímý prodej
Vydána	V-2007-15	Brno	2007-01-12 (Pavel Chmelar)	2007-01-14 (Admin Krivosh)	-	23322.60	Tesco	Prodej na fakturu

1 4 @

> Nový Přímý prodej  
> Nový Prodej na fakturu

**Nová** : Kliknutím na doleuvedené odkazy vytvoříte novou výdejku (t.j. přímý prodej s okamžitou splatností) nebo fakturu (odběratele, splatnost atd. nastavíte dále).

## 9. Příjemka

Pod sekci "Příjemka" se odehrává celý proces příjmu zboží, tj. vytvoření příjemky kliknutím na odkaz "Nákup" v menu či "Nový příjem" pod seznamem příjemek – nebo následné naskladnění hotové příjemky a její potvrzení, jako detaily příjemky.

- Až do naskladnění zboží je možno zboží na příjemku libovolně přidávat, či zboží přidané omylem z příjemky vyjmout.
- Na druhou stranu, po naskladnění zboží je možné tisknout příjemku ve výstupní sestavě.
- Poznámku k příjemce je možno pozměňovat až do potvrzení příjemky povolanou osobou

Změnit? ▲ Množství ▼ ▲ Zboží ▼ ▲ Nák.cena ▼ ▲ DPH ▼

Vytvořit :    5% ▼

<input type="radio"/>	12.00	Káva Tatawelo Fuerte mletá 250 g	33.00	5%	<input type="checkbox"/>
<input type="radio"/>	10.50	Nuevo Futuro	80.00	5%	<input type="checkbox"/>

Suma : 1236.00 Kč (1177.14 bez DPH)

Naskladnit na pobočku :

Poznámka/popis :

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

**Info**

Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**

Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**

Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**

Prodejní ceny

**Účetnictví**

Přehled DPH  
Přehled Zisku

**Systém**

Uživatelské účty

**Osobní**

Změna údajů, nastavení  
== ODHLÁŠENÍ ==

**Výběr zboží :** Vybírat můžete z veškeré nabídky aktivního zboží. Pro rychlejší orientaci v (abecedně řazeném) seznamu můžete po najetí na lištu(at' už kliknutím nebo klávesou tabulátoru z předešlého políčka) ve většině lepších prohlížečů zadávat první písmena názvu zboží až do doby kdy bude nalezeno požadované zboží. Podobně by ve většině prohlížečů mělo fungovat potvrzení výběru zboží kromě kliknutí myši taky tabulátorem(a tím pádem přechodem na následující pole). Klávesa Enter může být prohlížečem chápána jako potvrzení výběru, ale taky jako pokus o odeslání všech údajů.

**Smazat :** Pokud ještě nedošlo k naskladnění zboží, je možno příjem zrušit. Zboží bude zapomenuto a příjemka bude smazána(případně stornována, existuje-li už příjemka s vyšším identifikačním číslem).

> tisk

poznámka 123

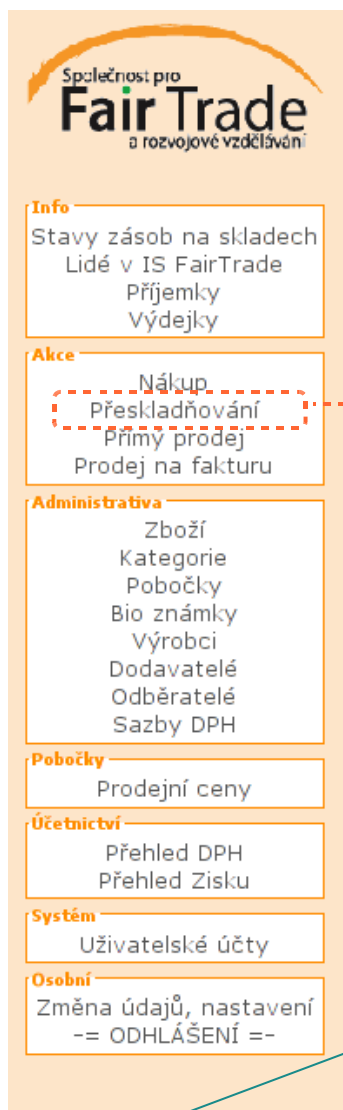
▲ Zboží ▼	▲ Množství ▼	▲ Jedn.Cena ▼	▲ DPH ▼	▲ Cena s DPH ▼	▲ Celkem ▼
Nuevo Futuro	600.00	128.00	0	128.00	76800.00
Nuevo Futuro	20.00	48.00	0	48.00	960.00

Suma : 77760.00 Kč (77760.00 bez DPH)

— —

Vytvořeno : 8.1.2007, 08:18:06 (Petr Chmelář)  
Naskladněno : 8.1.2007, 08:26:31 (Petr Chmelář)  
Potvrzeno : 10.1.2007, 05:56:50 (Admin Krivosih)

**Tisk :** Tisknout doklad je možné ke každé naskladněné příjemce; jestli je na příjemce zboží jenom od jednoho dodavatele, bude v dokladu jako 'Adresa dodavatele' právě jejich adresa, získaná z tabulky *Dodavatelé*. V opačném případě tam budou jenom vypsány jména podílejících se dodavatelů.



## 10. Přeskladňování

Tato sekce zahrnuje veškeré operace spojené s přesuny zboží mezi pobočkami - můžete tady iniciovat nový přesun, přijmout na 'svůj' sklad zboží které sem bylo posláno z jiné pobočky a taky vrátit zpátky zboží, které bylo z pobočky posláno, ale z nějakého důvodu nebylo doručeno. Zboží, které je 'na cestě', není evidováno na žádném ze skladů, tudíž nemůže být například prodáno.

**Odesílatel :** Změna odesílatele je vlastně možnost měnit 'domovskou pobočku' v kontextu přesunů zboží; je dostupná jen se zvláštními právy. Její změnou se patřičně přizpůsobí pohled na všechny pohyby (např. zboží na cestě 'od nás' může po změně být najednou adresováno 'k nám').

Zboží 'na cestě k nám' je zboží, které bylo posláno z jiné pobočky a 'zde', na domovské pobočce, ještě nebylo přijato.

Pobočka :  
Praha

> Zboží "na cestě k nám" (1)

> Zboží "na cestě od nás" (1)

Přesunout :

Pobočka	Položka	Zboží	Množství	-i-	Na přesun
Praha	20	Káva Tatawelo Fuerte mletá 250 g	20.00	info	5
Praha	45	Nuevo Futuro	5.00	info	
Praha	14	Nuevo Futuro	12.00	info	3
Praha	15	Nuevo Futuro	13.00	info	

Do pobočky :  
Brno

Zboží 'na cestě od nás' je zboží, které bylo posláno z domovské pobočky a ještě nebylo doručeno na pobočku cílovou.

**Položka :** Přesun se zde, na rozdíl od prodeje, neřídí zásadou *First in, first out* (tj. že se k akci vybere 'nejstarší' dostupné zboží) - proto, pokud na tom záleží, je výběr položek na přesun ponechán na posílajícím.

**Na přesun :** Do textových polí zadávejte množství kusů jednotlivých položek, které mají být přesunuty. Pokud je zadána nepřijatelná hodnota (například větší množství než je k dispozici), je odmítnuta celá sada zadaných hodnot. Fakticky ale není problém zadávat položky po jedné, na přesunu se to, kromě času odeslání, nijak neodrazí.

**Info :** Možnost zjistit více informací o dané položce, t.j. zobrazí příjemku, kterou bylo zboží nakoupeno (odtud například stáří zboží, dodavatel atd.)

**Přesunout :** Do vybrané pobočky přesune zboží o množstvích zadaných v polích 'na přesun' z předchozí tabulky.

**Vrátit** : U zaškrtnutých položek je zrušen přesun, t.j. jsou vráceny na sklad pobočky, ze které byly poslány (v tomto kontextu domovská pobočka).

> Zboží "na cestě k nám" (1)

Zboží "na cestě od nás" :

▲ Odesláno ▼	▲ Do pobočky ▼	▲ Zboží ▼	▲ Množství ▼	▲ Vrátit? ▼
2007-01-22 15:38:36	Brno	Káva Tatawelo Fuerte mletá 250 g	4.00	<input type="checkbox"/>

Vrátit vybrané

> Nový přesun

**Přijmout** : U zaškrtnutých položek je dokončen přesun, t.j. jsou doručeny na sklad cílové pobočky (v tomto kontextu domovská pobočka).

Zboží "na cestě k nám" :

▲ Odesláno ▼	▲ Z pobočky ▼	▲ Zboží ▼	▲ Množství ▼	▲ Přijmout? ▼
2007-01-22 15:38:36	Praha	Káva Tatawelo Fuerte mletá 250 g	4.00	<input checked="" type="checkbox"/>

Přijmout vybrané

> Zboží "na cestě od nás" (1)

> Nový přesun

## 11. Prodej

Tato kapitola zahrnuje dva úkony – prodej na fakturu i přímý prodej; už ze samotné podstaty se totiž jedná v podstatě o stejnou věc, přímý prodej je zjednodušením fakturovaného prodeje na okamžitou splatnost a platbu v hotovosti...

**Prodej na fakturu** : Před vytvořením faktury je potřeba potvrdit datum vytvoření, případně pobočku ze které chcete fakturu vést, máte-li možnost prodávat na více pobočkách. Odběratel, typ úhrady a splatnost můžou být nastavovány či měněny i dodatečně.

Z pobočky :  
Brno ▼

Datum vytvoření :  
2007-01-22 (rrrr-mm-dd)

Odběratel :  
--- ▼

Typ úhrady :  
Hotově ▼

Splatnost :  
2007-02-05 (rrrr-mm-dd) nebo za  dnů

Potvrd'

Z pobočky :  
Brno ▼

Datum vytvoření :  
2007-01-23 (rrrr-mm-dd)

Odběratel :  
--- ▼

Vytvořit

**Přímý prodej** : Před vytvořením výdejky je potřeba potvrdit datum vytvoření, případně pobočku ze které chcete výdejku vést, máte-li možnost prodávat na více pobočkách. Odběratel může být nastavován či měněn i dodatečně.

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

**Info**

- Stavy zásob na skladech
- Lidé v IS FairTrade
- Příjemky
- Výdejky

**Akce**

- Nákup
- Přeskladňování
- Přímý prodej
- Prodej na fakturu

**Administrativa**

- Zboží
- Kategorie
- Pobočky
- Bio známky
- Výrobci
- Dodavatelé
- Odběratelé
- Sazby DPH

**Pobočky**

- Prodejní ceny

**Účetnictví**

- Přehled DPH
- Přehled Zisku

**Systém**

- Uživatelské účty

**Osobní**

- Změna údajů, nastavení
- ODHLÁŠENÍ --

**Společnost pro Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 12. Výdejka

Zde si můžete prohlížet detaily příjemky a v závislosti od Vašich práv a stavu výdejky na ní taky provádět operace :

- Až do vydání zboží je možno zboží na fakturu libovolně( resp. v rámci skladových možností) přidávat, či zboží přidané omylem vrátit zpátky na sklad.
- Na druhou stranu, po vydání zboží je možné tisknout doklad ve výstupní sestavě.
- Poznámku k dokladu, odběratele, typ úhrady a splatnost je možno pozměňovat až do potvrzení faktury povolanou osobou

-z pobočky : **Brno**

Nuevo Futuro (611.00ks, á 140.00Kč vč. DPH 5%) (nebo ID : )

množství :  (nebo  bez DPH)

prodejní cena :  Kč včetně DPH 0% (nevyplněno -> běžná prod.cena a/nebo DPH)

Poznámka/popis :

Uzavřít výdejku :

Smazat výdejku :

**Zboží na výdejce :**

▲ Pobočka ▼	▲ Zboží ▼	▲ Množství ▼	▲ Jedn.cena ▼	▲ DPH ▼	▲ Celkem ▼	▲ Smazat? ▼
Brno	Káva Tatawelo Fuerte mletá 250 g	1.00	85.71	5%	90.00	<input type="button" value="x"/>

Suma : 90.00 Kč (85.71 Kč bez DPH)

Odběratel :

**Přidání zboží :** Zde můžete vybírat ze zboží dostupného na vybrané pobočce; vedle zboží můžete vidět taky dostupné množství, (jednotkovou) prodejní cenu a DPH. Toto jsou hodnoty, které jsou 'doplněny' k prodejní položce v případě, že nejsou vyplněny.

Nejrychlejší prodej jedno kusu zboží(jakožto nejčastějšího prodejního množství) vypadá ve většině prohlížečů tak, že kliknete(nebo se klávesou tabulátoru přesunete) na tuto lištu, zadáním prvních pár znaků z názvu žádaného zboží jej v (abecedně řazeném) seznamu vyberete a jedním nebo dvěma zmáčknutími klávesy *Enter* potvrdíte.

**Nebo ID :** Pokud se vám z jakéhokoliv důvodu výběr zboží přes rolovací lištu nehodí, máte možnost zadávat zboží přes jeho ID, pokud ho samozřejmě znáte. Takto zadané zboží má přednost před zbožím zadaným z rolovací lišty, jsou-li zadány obě.

**Cena bez DPH :** Jestli z nějakého důvodu preferujete zadání jednotkové ceny bez DPH(cena uvedená v liště u zboží je včetně DPH), můžete ji zadat do pole 'bez DPH' - tato cena bude upřednostněna před cenou z pole 'ceny včetně DPH'. Interně je však evidována cena s DPH(t.j. dopočítána ze zde zadané ceny), může tedy dojít k nežádoucímu zkreslení ceny kvůli zaokrouhlení na halíře.

**Smazat** : Pokud ještě nedošlo k vydání zboží, je možno výdej zrušit. Zboží bude vráceno na sklad a výdejka bude smazána (případně stornována, existuje-li už výdejka s vyšším identifikačním číslem).

Po vydání je transakce definitivně ukončena, to znamená, že zboží není možno jenom tak vrátit. Další akce jako změna poznámky či odběratele slouží už jenom pro dokumentační účely.

-z pobočky : Brno

[tisk](#)

▲ Zboží ▼	▲ Množství ▼	▲ Jedn.cena ▼	▲ DPH ▼	▲ Cena s DPH ▼	▲ Celkem ▼
Káva Tatawelo Fuerte mletá 250 g	3.00	74.79	19%	89.00	267.00
Nuevo Futuro	1.00	141.90	5%	149.00	149.00
Suma : 416.00 Kč (366.27 Kč bez DPH)					

Odběratel : Tesco

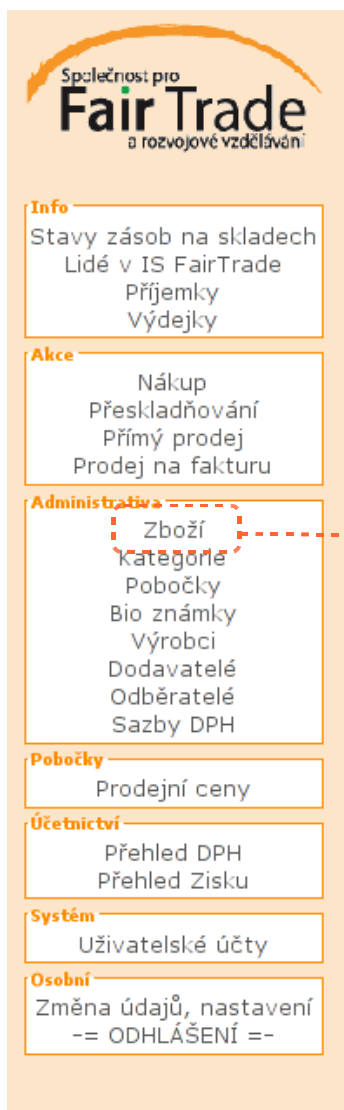
Vytvořeno : 16.1.2007, 02:58:39 ([Admin Krivosh](#))  
Vydáno : 16.1.2007, 02:59:09 ([Admin Krivosh](#))  
Potvrzeno : 16.1.2007, 03:01:47 ([Admin Krivosh](#))

**Tisk** : Tisknout doklad je možné ke každé vyskladněné faktuře; 'Adresa odběratele' a další potřebné údaje jsou závislé od parametrů nastavených zde, resp. v tabulce *Odběratelé*.



- Info**
  - Stavy zásob na skladech
  - Lidé v IS FairTrade
  - Příjemky
  - Výdejky
- Akce**
  - Nákup
  - Přeskladňování
  - Přímý prodej
  - Prodej na fakturu
- Administrativa**
  - Zboží
  - Kategorie
  - Pobočky
  - Bio známky
  - Výrobci
  - Dodavatelé
  - Odběratelé
  - Sazby DPH
- Pobočky**
  - Prodejní ceny
- Účetnictví**
  - Přehled DPH
  - Přehled Zisku
- Systém**
  - Uživatelské účty
- Osobní**
  - Změna údajů, nastavení
  - = ODHLÁŠENÍ = -





### 13. Zboží – správa

V této sekci můžete nastavovat zboží, veškeré údaje, zejména pro externí potřebu (informace pro zákazníky), ale taky pro fungování prodeje (běžná prodejní cena, DPH)

Jak si jistě hned všimnete, tabulka má nezvykle moc sloupců; proto se v zobrazení systému jenom málokdy na šířku vejde na obrazovku a proto je i obrázek tady vložen ve dvou částech. Jelikož se však jedná o stěžejní objekt celého systému, je tato obsáhlost zcela namístě.

**Aktivní :** V případě, že je z nějakého důvodu žádoucí 'vyřadit zboží z provozu' (neukazovat ho v nabídkách, na stránkách), ale už není možno jej smazat (je v oběhu), je možno nastavit zboží jako neaktivní.

**Speciální nabídka :** Jestli je zboží ve speciální nabídce; tato vlastnost slouží pro web, v rámci I.S. nehraje žádnou roli.

**Bio :** Informace o tom, jestli je zboží 'bio' a/nebo jakou 'bio' známku nese; tyto jsou, včetně dalších informací, definovány v sekci *Bio známky* (kapitola 16).



**Kategorie** : Každé zboží je zařazeno do nějaké kategorie; tyto jsou definovány(a řazeny do hierarchické struktury) v sekci *Kategorie*(kapitola 14). (Právě jednu) kategorii vyberete zaškrtnutím políčkem, v případě, že chcete rozbalit větev stromu, klikněte na její 'kořen'. Ten by měl být označen zvláštní šipkou('►') a taky označen výraznější barvou.

**Prodejní cena** : Cena včetně DPH, která bude nabízená jako běžná prodejní cena(t.j. cena, za kterou bude zboží prodáno v případě nezadání jiné ceny). Nad toto nastavení ale ještě může být nadřazena vlastní prodejní cena některé pobočky, je-li zadána v sekci *Prodejní ceny*(kapitola 21).

**Obrázek** : Zde se očekává adresa ilustračního obrázku ke zboží nebo prázdné pole v případě, že zboží obrázek nemá. Doporučuje se zadávat kompletní URL(<http://server/jmeno>), v rámci serveru by teoreticky měla fungovat i absolutní adresa(např. </img/zbozi/obrazek.jpg> by mělo fungovat stejně jako <http://www.fairtrade.cz/img/zbozi/obrazek.jpg> , pozor ale na úvodní lomítko). Systém nepozná jestli je v adrese obrázku chyba nebo obrázek neexistuje, pokusí se ho zobrazit.

**Více** : Sem patří veškeré 'další' informace o zboží, které mají být zobrazeny v přehledech zboží. Jestli je místo textu zadán jenom hypertextový odkaz ( <http://adresa> ), stane se tento v kontextu aktivním - doporučuje se zejména v případě, že o zboží už existuje rozsáhlejší článek.

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

Info

Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

Akce

Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

Administrativa

Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

Pobočky

Prodejní ceny

Účetnictví

Přehled DPH  
Přehled Zisku

Systém

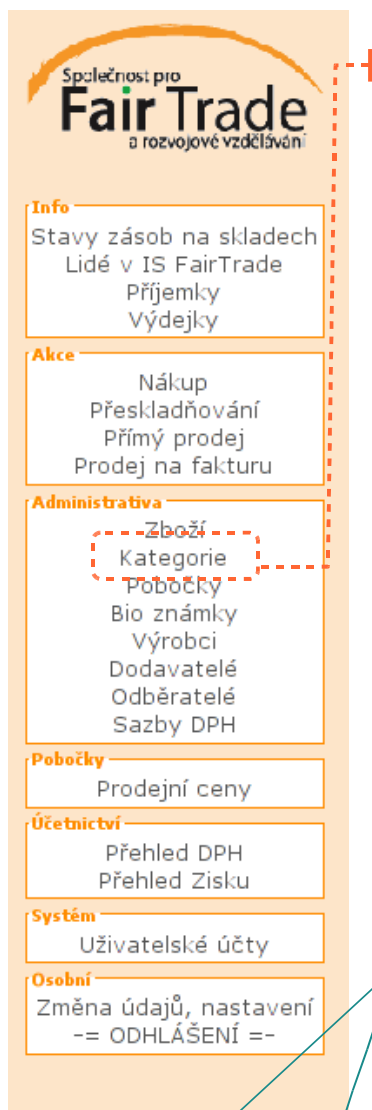
Uživatelské účty

Osobní

Změna údajů, nastavení  
== ODHLÁŠENÍ ==

Jak už bylo napsáno, tato data jsou definována zejména pro externí potřebu.

Některé atributy ještě nejsou systémem plně využívány(viz. například "Speciální nabídka"); chcete-li se však na zboží podívat na zboží z pohledu "vnějšího" návštěvníka, navštivte adresu <http://www.fairtrade.cz/zbozi.php>, kam bude tak další funkcionality postupně dopracována.



## 14. Kategorie – správa

V této sekci můžete nastavovat kategorie, ve kterých je zařazeno veškeré zboží. Ty jsou řazeny hierarchicky do stromu, ve kterém je můžete téměř libovolně přesunovat a řadit. Začít můžete kliknutím na název kategorie (jestli chcete provádět operace na už existující) nebo ve formuláři vytvoření kategorie...

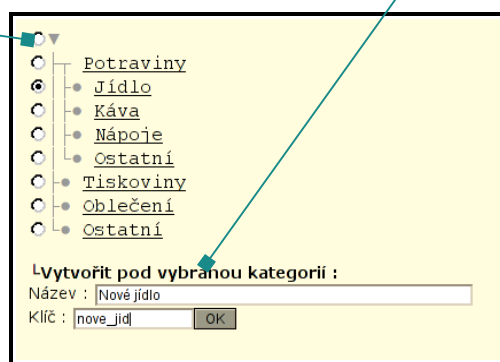
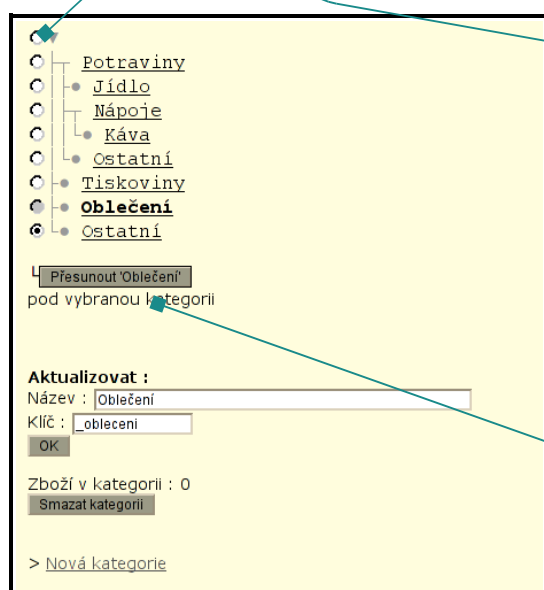
**Klíč** : 'Řadící klíč' umožňuje řadit kategorie v rámci stromu:

- nemusí(i) když to kvůli přehlednosti není špatné) být podobný názvu - pro uživatele je (téměř) neviditelný
- maximální délka je 10 znaků, musí být jedinečný
- může obsahovat jen znaky **a-z** a **0-9**, dále **+(plus)** a **\_(podtržítka)**
- podtržítka a tečka pomáhají řadit v rámci jedné podkategorie - např. "+novinky" je nad "karminove", ta nad "modre" a ta nad "\_a\_t\_d" ...

**Kořen** : Toto je kořenová kategorie, pod kterou jsou stromově umístěny další kategorie. Na rozdíl od běžných kategorií do ní není možné umísťovat zboží (jenom kategorie).

Po výběru kategorie, se kterou chcete provádět operace (kliknutím na její název), můžete označit zaškrťovací políčko vlevo - tato kategorie se pak stane cílovou pro přesun.

**Nová** : Jestli chcete vytvořit novou kategorii, zvolte v seznamu zaškrťovacích políček tu, pod kterou Vaše kategorie vznikne. Dále vyplňte jméno a řadící klíč a pak už můžete akci potvrdit zmáčknutím OK



**Přesun** : Přesune kategorii, kterou jste zvolili kliknutím na její název, do kategorie označené zaškrťovacím políčkem. Kategorie se přesunuje včetně zboží v ní a podkategorií - to je logický důvod proč není možno kategorii přesunout do vlastní podkategorie.

## 15. Pobočky – správa

Pod pobočkami jsou v tomto kontextu myšleny jak sklady, tak i samostatně fungující 'jednotky', ze kterých je možno prodávat atd. Každý uživatel má přiřazenou právě jednu 'domovskou pobočku', ze které mu plynou příslušná privilegia/omezení.

**Aktivní :** V případě, že je z nějakého důvodu žádoucí 'vyřadit pobočku z provozu'(neukazovat ji v nabídkách...), ale už není možno ji smazat(na pobočku se váže zboží či výdejka, některý uživatel ji má nastavenou jako 'domovskou'), je možno nastavit pobočku jako neaktivní. Tento status je plně vratný.

## 16. Bio známky – správa

V této sekci můžete nastavovat 'Bio známky', které je pak možné přiřazovat k jednotlivému zboží. Tento parametr je pro zboží povinný, je proto žádoucí nechat v seznamu známek i možnost typu 'žádná', zároveň s odkazem na článek / vysvětlení, proč žádnou nemá. Bio známky je možné libovolně přidávat a upravovat, avšak mazat jenom v případě, že nejsou přiřazeni k žádnému zboží.

**Více :** Tyto 'další' informace o Bio známce jsou zobrazeny především u detailů zboží, je-li nalezeno z vnější(veřejné) části webu FairTrade. Jestli je místo textu zadán jenom hypertextový odkaz ( <http://adresa> ), stane se tento v kontextu aktivním.

**Info**  
 Stavby zásob na skladech  
 Lidé v IS FairTrade  
 Příjemky  
 Výdejky

**Akce**  
 Nákup  
 Přeskladňování  
 Přímý prodej  
 Prodej na fakturu

**Administrativa**  
 Zboží  
 Kategorie  
 Pobočky  
 Bio známky  
 Výrobci  
 Dodavatelé  
 Odběratelé  
 Sazby DPH

**Pobočky**  
 Prodejní ceny

**Účetnictví**  
 Přehled DPH  
 Přehled Zisku

**Systém**  
 Uživatelské účty

**Osobní**  
 Změna údajů, nastavení  
 -= ODHLÁŠENÍ -=

**Společnost pro Fair Trade a rozvojové vzdělávání**

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
**Dodavatelé**  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 17. Výrobci – správa

V této sekci můžete nastavovat výrobce, které je pak možné přiřazovat k jednotlivému zboží. Výrobce je možné libovolně přidávat a upravovat, avšak mazat jenom v případě, že nejsou přiřazeni k žádnému zboží.

Změnit?	ID	Aktivní?	Název	Více	Smazat?
<input type="radio"/>	3	Ano	Asprovega	http://www.fairtrade.cz/index.php?clanek=388	<input type="checkbox"/>
<input type="radio"/>	4	Ano	Idulgashena	http://www.fairtrade.cz/index.php?clanek=292	<input type="checkbox"/>
<input type="radio"/>	5	Ano	Conacado	http://www.fairtrade.cz/index.php?clanek=205	<input type="checkbox"/>
<input type="radio"/>	6	Ano	El Ceibo	http://www.fairtrade.cz/index.php?clanek=204	<input type="checkbox"/>
<input type="radio"/>	7	Ano	Talon Sports	http://www.fairtrade.cz/index.php?clanek=300	<input type="checkbox"/>
<input type="radio"/>	8	Ano	Design Inc.	http://www.fairtrade.cz/index.php?clanek=298	<input type="checkbox"/>
<input type="radio"/>	14	Ano	Commercio Alternativo	Itálie	<input type="checkbox"/>

## 18. Dodavatelé – správa

V této sekci můžete nastavovat dodavatele, které je pak možné přiřazovat k jednotlivému zboží. Dodavatele je možné libovolně přidávat a upravovat, avšak mazat jenom v případě, že nejsou přiřazeni k žádnému zboží.

Změnit?	ID	Aktivní?	Název	Kontakt	Více	Smazat?
<input type="radio"/>	24	Ano	Ashley Exports Sri Lanka	-	http://www.fairtrade.cz/index.php?clanek=214	<input type="checkbox"/>
<input type="radio"/>	25	Ano	Ekumenická Akademie	Ekumenická akademie Praha Na Mláčcích 1 101 00 Praha 10 - Vršovice	http://www.ekumakad.cz/	<input type="checkbox"/>
<input type="radio"/>	26	Ano	Commercio Alternativo	Blalala 16 Ferrara	Itálie	<input type="checkbox"/>

**Kontakt :** Kontaktní údaje jsou využívány zejména u tisku příjmového dokladu, je-li naskladňováno zboží od toho-kterého dodavatele. Na formátování je možné použít běžné HTML značky, jmenovitě např. <br> na označení konce řádku..

## 19. Odběratelé – správa

V této sekci můžete nastavovat dodavatele, které je pak možné přiřazovat k jednotlivému zboží. Dodavatele je možné libovolně přidávat a upravovat, avšak mazat jenom v případě, že nejsou přiřazeni k žádnému zboží.

Změnit?	ID	Aktivní?	Název	Kontakt	Smazat?
<input type="radio"/>	1	Ano	Tesco	Tesco Stores ČR, a.s. Veselská 663 199 00 Praha-Letňany	<input type="checkbox"/>

## 20. Sazby DPH – správa

V této sekci můžete nastavovat aktuální sazby DPH. Tyto jsou nabízeny jednak jako výběr např. u prodeje či nákupu, jednak se podle nich stanovují daňová pásma pro přehledy např. v účetních statistikách či faktuře.

**ID :** Podle ID jsou řazeny sazby DPH v nabídkách (t.j. ne v přehledech, tam jsou řazeny podle velikosti sazby) - sazba s nejnižším ID je nabízena jako první, tím pádem by se dalo říct, že je 'preferována'.

**Sazba :** Sazba je daná v procentách, t.j. by měla nabývat hodnot 0-100%. Hodnoty sazeb zde jsou v ohledu na zboží jenom informativní, pro výběr (nejdou vázány)

Změnit?	ID	Sazba*	Název*
<input type="radio"/>	1	5	Snížená
<input type="radio"/>	2	19	Základní
<input type="radio"/>	3	0	Žádná

## 21. Prodejní ceny na pobočce

Ačkoliv je pro každé zboží definována běžná prodejní cena, která je globálně nabízena jako 'předvolená' u prodeje, v případě potřeby je možné nastavit pro některou z poboček 'výjimku', t.j. vlastní předvolenou prodejní cenu.

**Pobočka :** Možnost měnit 'domovskou pobočku' v kontextu nastavení prodejních cen; je dostupná jen se zvláštními právy. Její změnou se patřičně přizpůsobí pohled na pobočku a ceny/zboží pro ni nastaveny.

Změnit?	Zboží(+orig.cena)*	Prod.Cena*
<input type="radio"/>	Káva Tatawelo Fuerte mletá 250 g (91.43,-)	90.00
<input type="radio"/>	Nuevo Futuro (149.00,-)	140.00

**Zboží :** Názvy aktivního zboží (rozuměj : toho, které je možno prodávat) a v závorce, pro informaci, cena včetně DPH, která je přednastavena z nastavení v sekci Zboží (kapitola 13).

**Prodejní cena :** Cena včetně DPH, která bude v případě prodeje z vybrané pobočky nabízena jako běžná prodejní cena (t.j. cena, za kterou bude zboží prodáno v případě nezadání jiné ceny).

Společnost pro  
**Fair Trade**  
a rozvojové vzdělávání

Info

Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

Akce

Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

Administrativa

Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

Pobočky

Prodejní ceny

Účetnictví

Přehled DPH  
Přehled Zisku

Systém

Uživatelské účty

Osobní

Změna údajů, nastavení  
== ODHLÁŠENÍ ==

**Společnost pro Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
Uživatelské účty

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

## 22. Přehled DPH

Zde můžete generovat přehled příjmů v daňových pásmech za zvolené časové období. Každý řádek představuje výdejku nebo fakturu, sloupce se generují podle aktuálně nastavených sazeb daní (viz. kapitolu 20, *Sazby DPH*) - pro každou sazbu je spočítána suma základu a daně pro položky na výdejce/faktuře. Poslední sloupec představuje celkovou sumu na dokladu, včetně DPH.

**Období :** Jako vstup pro období se očekávají dva datумы (od a do) ve formátu rrrr-mm-dd. Přednastavené je období posledních 100 dní.

**Export :** Tento odkaz vygeneruje a nabídne na stažení CSV soubor, ve kterém je uložena tabulka, kterou vidíte na stránce - včetně veškerých řazení a filtrací. Formát CSV je používán jako textová reprezentace tabulky; sloupce jsou v něm odděleny znakem středníku (;). Tento formát je schopna otevřít a zpracovat velká většina tabulkových procesorů typu MS Excel, OO Spreadsheet atd.

Za období od-do :  
2006-10-14 - 2007-01-22  
Nastavit období

Filtruj

ID	Vystaveno	Základ(0%)	Dan(0%)	Základ(5%)	Dan(5%)	Základ(19%)	Dan(19%)	Celková částka
V-2007-1	2007-01-08	0.00	0.00	141.90	7.10	0.00	0.00	149.00
V-2007-9	2007-01-10	0.00	0.00	670.47	33.53	0.00	0.00	704.00
V-2007-11	2007-01-15	0.00	0.00	80.95	4.05	0.00	0.00	85.00
V-2007-13	2007-01-14	0.00	0.00	80.95	4.05	0.00	0.00	85.00
V-2007-15	2007-01-14	0.00	0.00	210.19	10.51	1941.76	3688.24	23322.60
V-2007-16	2007-01-12	0.00	0.00	957.84	47.89	0.00	0.00	1005.73
V-2007-38	2007-01-16	85.00	0.00	84.76	4.24	0.00	0.00	174.00
V-2007-39	2007-01-16	0.00	0.00	141.90	7.10	224.37	42.63	416.00

> export do formátu CSV ("Excel")

## 23. Přehled zisků

Zde můžete generovat přehled zisků za zvolené časové období. Každý řádek představuje prodané zboží na jedné výdejce nebo příjemce. Ve sloupcích je, kromě zmiňovaných položek, součet nákupní a prodejní ceny bez DPH za zboží v uvedeném množství a v posledním sloupci rozdíl těchto hodnot, t.j. zisk.

Za období od-do :  
2007-01-14 - 2007-01-16  
Nastavit období

Filtruj

ID	Vystaveno	Pobočka	Zboží	Množství	Prodej	Nákup	Zisk
V-2007-11	2007-01-15	Brno	Káva Tatawelo mletá 250 g	1.00	80.95	54.00	26.95
V-2007-13	2007-01-14	Brno	Káva Tatawelo mletá 250 g	1.00	80.95	54.00	26.95
V-2007-15	2007-01-14	Brno	Káva Tatawelo mletá 250 g	1.00	80.95	54.00	26.95
V-2007-15	2007-01-14	Brno	Káva Tatawelo Fuerte mletá 250 g	1.00	1.73	54.00	-52.27
V-2007-15	2007-01-14	Brno	Káva Tatawelo Fuerte mletá 250 g	1.00	1.81	54.00	-52.19
V-2007-15	2007-01-14	Brno	Nuevo Futuro	1.00	9.07	128.00	-118.93
V-2007-15	2007-01-14	Brno	Káva Tatawelo mletá 250 g	1.00	9.52	54.00	-44.48
V-2007-15	2007-01-14	Brno	Nuevo Futuro	1.00	9.07	128.00	-118.93
V-2007-15	2007-01-14	Brno	Káva Tatawelo Fuerte mletá 250 g	1.00	9.07	54.00	-44.93
V-2007-15	2007-01-14	Brno	Nuevo Futuro	1.00	90.70	128.00	-37.30
V-2007-15	2007-01-14	Brno	Nuevo Futuro	22.00	19411.76	2816.00	16595.76

> export do formátu CSV ("Excel")

## 24. Uživatelské účty

V této sekci můžete vytvářet účty novým uživatelům či uživatelkám, případně měnit údaje o těch stávajících.

**Login :** Login slouží jako přihlašovací jméno - a zároveň jedinečná identifikace člověka v I.S., proto musí být jedinečný. Doporučuje se nepoužívat v něm diakritiku, mezery či interpunkci.

**Uživatelské účty :**

Uživatel/ka :  
nig - Niguel Mauricio

Login: \*nig (max 16 zn.)  
Heslo: (max 30)  
Kontrola hesla: (Pokud nechcete měnit, nevyplňujte)  
Jméno: \*Niguel Mauricio (max 50)  
Kontakt: nig.m@nuevo.cz (max 100)  
Domovská pobočka: Praha  
Aktivní? Ano

**Práva:**

<input type="checkbox"/> Všechna práva nebo : <input checked="" type="checkbox"/> UZIV_UCTY <sup>[i]</sup> <input type="checkbox"/> AKCE_VSUDE <sup>[i]</sup>	<input checked="" type="checkbox"/> NAKUP <sup>[i]</sup> <input checked="" type="checkbox"/> NASKLADNENI <sup>[i]</sup> <input type="checkbox"/> POTVRZ_PRIJEMKY <sup>[i]</sup>	<input checked="" type="checkbox"/> PRESKLADNOVANI <sup>[i]</sup> <input type="checkbox"/> UCETNICTVI <sup>[i]</sup> <input type="checkbox"/> EDIT_ZBOZI <sup>[i]</sup> <input type="checkbox"/> EDIT_KATEGORIE <sup>[i]</sup> <input type="checkbox"/> EDIT_POBOCKY <sup>[i]</sup> <input type="checkbox"/> EDIT_BIO <sup>[i]</sup> <input type="checkbox"/> EDIT_VYROBCE <sup>[i]</sup> <input type="checkbox"/> EDIT_DODAVATELE <sup>[i]</sup> <input type="checkbox"/> EDIT_ODBERATELE <sup>[i]</sup> <input type="checkbox"/> EDIT_DPH <sup>[i]</sup>	<input checked="" type="checkbox"/> PRODEJNI_CENY <sup>[i]</sup> <input checked="" type="checkbox"/> PRODEJ <sup>[i]</sup> <input type="checkbox"/> POTVRZ_VYDEJKY <sup>[i]</sup>
--	---	---	---

**Společnost pro  
Fair Trade**  
a rozvojové vzdělávání

**Info**  
Stavy zásob na skladech  
Lidé v IS FairTrade  
Příjemky  
Výdejky

**Akce**  
Nákup  
Přeskladňování  
Přímý prodej  
Prodej na fakturu

**Administrativa**  
Zboží  
Kategorie  
Pobočky  
Bio známky  
Výrobci  
Dodavatelé  
Odběratelé  
Sazby DPH

**Pobočky**  
Prodejní ceny

**Účetnictví**  
Přehled DPH  
Přehled Zisku

**Systém**  
**Uživatelské účty**

**Osobní**  
Změna údajů, nastavení  
== ODHLÁŠENÍ ==

**Pobočka :** Domovskou je pobočka, ve které má operátor/ka své 'hlavní působiště' - v praxi I.S. to například znamená, že tam může vykonávat operace se zbožím. Jestli má nastavená práva 'akce všude', bude mu u výběru operace závislé na pobočce nabízená domovská jako první.

**Aktivní :** Uživatelský účet není možno smazat - můžete ho však nastavit jako neaktivní; a takový účet není možné se přihlásit.

**Práva :** V této sekci můžete uživateli nastavit práva k přístupu do různých částí I.S., či případně omezit operace které může činit. Více informací o té - které položce je uchováno v malém [i] u každého z nich, resp. v popisku tohoto znaku. Ten zobrazíte v závislosti od použitého prohlížeče např. po najetí kurzorem jako 'bublinu' či text ve stavovém řádku prohlížeče...





## 25. Změna vlastních údajů

Zde si můžete nastavit osobní údaje - ty aktuální jsou vždy předvyplněny, kromě hesla, které je kvůli bezpečnosti nutno zadat a potvrdit u každého pokusu o změnu údajů.

Login:	<input type="text" value="krivosh"/>
Heslo:	<input type="password" value="*****"/> (max.30 znaků)
Kontrola hesla:	<input type="password" value="****"/>
Jméno:	<input type="text" value="Admin Krivosh"/> (max.50 znaků)
Kontakt:	<input type="text" value="krivosh@gmail.com"/> (max.100 znaků)
<input type="button" value="Odeslat"/>	
Zobrazovat nápovědu :	
<input type="button" value="Ne"/> <input type="button" value="Potvrd"/>	

**Nápověda :** Zvláštní položkou v nastavení je zobrazování nápovědy; pokud si myslíte, že už systém dostatečně ovládáte, případně vás nápověda obtěžuje, můžete si její zobrazování vypnout zvolením 'Ne' a potvrzením.

Nápovědu si samozřejmě můžete analogickým postupem kdykoliv aktivovat zpátky.

## 26. Odhlášení

Z bezpečnostních důvodů je po skončení práce se systémem doporučeno **vždy** provést odhlášení. Po kliknutí na příslušný odkaz bude vaše 'sezení' ukončeno a budete přesměrováni zpátky na přihlašovací stránku.

Sezení by mělo být ukončeno taky zavřením prohlížeče, ve kterém jste se přihlásili. Je zde ale riziko, že prohlížeči zůstalo otevřeno okno, které uniklo Vaší pozornosti – bez odhlášení může z tohoto okna případný narušitel pod Vaším loginem vstoupit do systému aniž by znal přihlašovací údaje.

Session byla automaticky ukončena po vypršení časového limitu. Chcete-li pokračovat v práci, přihlašte se znovu, prosím	
<b>I.S. Fairtrade</b>	
Login :	<input type="text"/>
Heslo :	<input type="password"/>
<input type="button" value="Odeslat"/>	

Toto riziko částečně řeší časový limit nečinnosti(viz.údaj "session do" v horní liště hlavního okna), po kterém bude sezení dočasně ukončeno a budete přesměrováni na přihlašovací stránku. Po úspěšné identifikaci budete přesměrováni zpátky.



# Obsah

1. Úvod.....	2
2.Přihlášení.....	3
3.Tabulkové formuláře a pohledy na data.....	4
4.Úvodní obrazovka.....	5
5.Stav zásob na skladech.....	6
6.Lidé v IS FairTrade.....	6
7.Příjemky.....	7
8.Výdejky.....	8
9.Příjemka.....	9
10. Přeskladňování.....	10
11. Prodej.....	11
12. Výdejka.....	12
13. Zboží – správa.....	14
14. Kategorie – správa.....	16
15. Pobočky – správa.....	17
16. Bio známky – správa.....	17
17. Výrobci – správa.....	18
18. Dodavatelé – správa.....	18
19. Odběratelé – správa.....	18
20. Sazby DPH – správa.....	19
21. Prodejní ceny na pobočce.....	19
22. Přehled DPH.....	20
23. Přehled zisků.....	20
24. Uživatelské účty.....	21
25. Změna vlastních údajů.....	22
26. Odhlášení.....	22



**FAIRTRADE**

# Programova dokumentace pro :: IS pro FairTrade



_doruceni.php	2
_kategorie.php	2
_login.php	3
_logoff.php	3
_nasklad.php	3
_nova_faktura.php	4
_nova_prijemka.php	4
_odberatel_vydej.php	5
_potvrd_prij.php	5
_potvrd_vydej.php	5
_poznamka_prij.php	6
_poznamka_vydej.php	6
_presklad.php	6
_smaz_prij.php	7
_tab_form.php	7
_ucty.php	8
Function dump_prava	8
_vydejka_akce.php	9
Function vrat_polozky	9
_zmena_sebe.php	10
bio.php	10
ceny.php	11
funkce_filtr.php	11
Function filtr_sloupec	11
Function filtr_sql	11
Function math_selectbox	11
funkce_help.php	12
Function help_icon	13
Function help_text	13
funkce_put_get.php	13
Function generate_gets	13
Function put_hidden_inputs	14
funkce_transakce.php	14
Function create_log	15
Function go_transakce	15

html_body.php	15
html_end.php	15
html_head.php	15
dodavatele.php	16
kategorie.php	16
odberatele.php	16
pobocky.php	17
vyrobci.php	17
zbozi.php	17
konekt.php	17
session.php	18
Define AutoLogOff	18
Function ma_prava	18
store_posts.php	19
tab_form.php	49
tab_view.php	20
dodavatele.php	21
doklad.php	21
dph.php	22
export_csv.php	22
faktura.php	23
index.php	23
kategorie.php	24
lide.php	25
main.php	25
odberatele.php	26
pobocky.php	26
prehled_dph.php	27
prehled_zisku.php	27
presuny.php	28
prijemka.php	29
prijemky.php	29
prodej.php	30
prodej_faktura.php	30
prvni.php	30
ucty.php	31
vydejka.php	31
vydejky.php	32
vyrobci.php	32
zasoby.php	33
zbozi.php	33
zmena.php	34
../zbozi.php	35

## /is/\_doruceni.php

### Popis

#### dokonceni "cesty" zbozi mezi pobockama

... bud doruceni do cile, nebo storno (podle POST(akce))

volan z modulu presuny.php, po vykonani se vraci tamtez s prislusnym kodem v &msg=

- zdroj. kod: [ke stazeni](#)

### Includes

*include* ('data/session.php') (radek 16)

overi jestli je uzivatel prihlasen, aktualizuje session

*include\_once* ('data/funkce\_transakce.php') (radek 21)

da nam funkci check\_rollback()

## /is/\_kategorie.php

### Popis

#### manipulace s kategoriemi

dostava v \_POSTu :

- presun - potom je "od\_ID" IDcko ciloveho parenta
- zmena : novy kategorie\_char, nazev; IDcko ktere se ma menit
- vymaz : kategorie\_id : ta ktera se ma smazat
- vytvor : kategorie\_id - pozor, to je parent;  
a nazev, kategorie\_char

volan z modulu kategorie.php, po vykonani se vraci tamtez s prislusnym kodem v &msg=

- zdroj. kod: [ke stazeni](#)

### Includes

*include* ('data/session.php') (radek 21)

overi jestli je uzivatel prihlasen, aktualizuje session

*include* ('data/check\_delete/kategorie.php') (radek 93)

vlozi funkci na overeni

## /is/\_login.php

### Popis

#### nalogovani usera do systemu

overi prihlaseni a bud vrati s prislusnou hlaskou nebo presmeruje dovnitr systemu

volan z index.php, v POST dostava login, heslo a nepovinne i "kam" ma jit po uspesnem prihlaseni(default: main.php)

- zdroj. kod: [ke stazeni](#)

### Includes

*include* ('data/konekt.php') (radek 17)

pripoji se do DB (nebo skonci, v pripade neuspechu)

## /is/\_logoff.php

### Popis

#### provede likvidaci sessiony nebo-li odhlaseni

volan z menu main.php, pokracuje na index.php

- zdroj. kod: [ke stazeni](#)

## /is/\_nasklad.php

### Popis

#### naskladnovani novych polozek

premistení zboží na sklad, aktualizace příjemky

volano z modulu prijemka.php, po vykonani se vola tamtez s prislusnym &msg= , pripadne na ?co=prijemky jestli neidentifikoval "volajici" příjemku

- zdroj. kod: [ke stazeni](#)

## Includes

*include* ('data/session.php') (radek 16)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_nova\_faktura.php**

## Popis

**vytvoreni nove faktury nebo vydejky**

jaky doklad, to se rozhodne uz u volani z formulare, kde se nastavi prislusne parametry(NULL splatnost -> vydejka)

volano z prodej.php nebo prodej\_faktura.php, jestli uspesne vytvori, jde na novou vydejku(?co=vydejka), jinak zpatky

■ **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('data/session.php') (radek 16)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_nova\_prijemka.php**

## Popis

**vytvoreni nove prijemky**

volano odkudkoliv bez parametru; teoreticky by mohla dostavat predem dany popis a datum, bylo-li by pro to využiti

pokracuje na novou prijemku (prijemka.php) v pripade uspechu, na seznam prijemek (prijemky.php) v opacnem pripade

■ **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('data/session.php') (radek 16)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_odberatel\_vydej.php**

## Popis

**zmena veci ohledne odberatele (u vydejky ci faktury)**

... typ uhrady, splatnost a samozrejme odberatel ...

volan z modulu vydejka.php, po vykonani se vraci tamtez s prislusnym kodem v &msg= (pripadne na ?co=vydejky jestli neidentifikuje vydejku)

■ **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('data/session.php') (radek 17)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_potvrd\_prij.php**

## Popis

**potvrzovani prijemky**

volan z modulu prijemka.php, po vykonani se vraci tamtez s prislusnym kodem v &msg= (pripadne na ?co=prijemky jestli neidentifikuje prijemku)

■ **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('data/session.php') (radek 15)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_potvrd\_vydej.php**

## Popis

**potvrzovani vydejky**

volan z modulu vydejka.php, po vykonani se vraci tamtez s prislusnym kodem v &msg= (pripadne na ?co=vydejky jestli neidentifikuje vydejku)

- zdroj. kod: [ke stazeni](#)

**Includes**

*include* ('[data/session.php](#)') (radek 15)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_poznamka\_vydej.php****Popis****zmena poznamky u vydejky**

volan z modulu vydejka.php, po vykonani se vraci tamtez s prislusnym kodem v &msg= (pripadne na ?co=vydejky jestli neidentifikuje vydejku)

- zdroj. kod: [ke stazeni](#)

**Includes**

*include* ('[data/session.php](#)') (radek 15)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_presklad.php****Popis****inicializuje presun tovaru mezi pobočkami, volan z presuny.php**

POST-em dostava parametry :

- z\_pobocky a do\_pobocky - ID pobocek
- sadu typu kolik\_XY=Z, co znamena ze od polozky XY chceme presunout Z kusu

vraci se vzdy zpatky na ?co=presuny, s prislusnym &msg=...

- zdroj. kod: [ke stazeni](#)

**Includes**

*include* ('[data/session.php](#)') (radek 18)

**overi jestli je uzivatel prihlasen, aktualizuje session**

*include\_once* ('[data/funkce\\_transakce.php](#)') (radek 23)

**da nam funkci go\_transakce**

**/is/\_smaz\_prij.php****Popis****mazani prijemky**

zbozi na ni jeste nemelo byt naskladneno, tudiz se muze vymazat - za predpokladu, ze by nevznikla dira, to ji pak oznaci jako storno

volano z modulu prijemka.php, pokracuje na ?co=prijemka nebo ?co=prijemky podle (ne)uspesnosti

- zdroj. kod: [ke stazeni](#)

**Includes**

*include* ('[data/session.php](#)') (radek 17)

**overi jestli je uzivatel prihlasen, aktualizuje session**

**/is/\_tab\_form.php****Popis****provadi operace na univerzalni tabulce, je volan z tab\_form.php**

vstup dostava pres \$\_POST :

- change - id(prim.klic) meneneho nebo new pro novou polozku
- sadu val\_neco pro hodnoty sloupce .. sloupec primarniho klice se ignoruje, ten se bere z change
- sadu delete\_neco pro id sloupce ktere se maji mazat - POZOR, ZADNE TRANSAKCE
- zmena nebo vymaz nebo aktualizuj je neprazdna podle toho kterym se odesilalo buttonem
- tabulka - jmeno tabulky ve ktere se ma strachat
- prim\_sloupec - jmeno ID-ovskyho sloupce(ten ale stejne musi mit auto\_increment, jinak se neprida)
- kam\_zpatky - moznost zadat jinou navratovou adresu
- check\_delete - jestli je dan,, pokusi se jej pred mazanim includnout, zpustit z nej funkci

check\_delete(id), která vrati 1 pokud muze byt radek s ID smazan

**navratovy kody pres &msg=:**

- 0 - chyba u zapisu do DB
- 1 - aktualizovano
- 2 - vytvoreno
- 3 - chybi udaje
- 4 - smazano
- 5 - nemozno smazat(&ne= teckami oddeleny seznam id ktere se nemuzou smazat)

pripojeni do databaze se odehrava v data/session.php

u chyby ulozi do SESSION udaje, ktere mely byt zapsany - viz funkce data/store\_posts.php

- **zdroj. kod:** [ke stazeni](#)
- **see:** [session.php](#)
- **see:** [store\\_posts.php](#)

## Includes

*include* ('[data/session.php](#)') (radek [39](#))

**overi jestli je uzivatel prihlasen, aktualizuje session**

*include* ('Location: '.\$kam\_zpatky.'&msg=0') (radek [57](#))

**vlozi pozadovanou funkci na konktrolu mazani**

*include* ('[data/store\\_posts.php](#)') (radek [100](#))

**jestli se vyskytla chyba, ulozi do SESSION zadane udaje(budou vraceny uzivateli)**

*include* ('[data/store\\_posts.php](#)') (radek [135](#))

**jestli se vyskytla chyba, ulozi do SESSION zadane udaje(budou vraceny uzivateli)**

*include* ('[data/store\\_posts.php](#)') (radek [169](#))

**jestli nebyla zadana bezna akce,ulozi do SESSION zadane udaje(budou vraceny uzivateli)**

## /is/\_ucty.php

### Popis

### editace useru - vytvoreni, zmena udaju

volano z modulu ucty.php, pokracuje tamtez s patricnym &msg=

- **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('[data/session.php](#)') (radek [14](#))

**overi jestli je uzivatel prihlasen, aktualizuje session**

## Functions

**dump\_prava** (radek [21](#))

**funkce na upravu parametru pro prava do vyslednyho tvaru**

prava zadana pres \_POST upravi do DB tvaru

*void* **dump\_prava** ()

## /is/\_vydejka\_akce.php

### Popis

**provadi operace na vydejce : uzavreni, pridani/smazani polozky,...**

jako vstup dostava :

- "akce" - pres POST podle submit buttonu, pripadne "vyndat" pres GET
- podle toho pak potrebne udaje z tabulky Polozky

volano z modulu vydejka.php, vraci se tamtez s prislusnym &msg= nebo jestli neni mozno, taky na ?co=vydejky(vydejka smazana, nenalezena...)

- **zdroj. kod:** [ke stazeni](#)

## Includes

*include* ('[data/session.php](#)') (radek [19](#))

**overi jestli je uzivatel prihlasen, aktualizuje session**

`include_once ('data/funkce_transakce.php')` (radek 24)

**da nam funkci go\_transakce**

## Functions

**vrat\_polozky** (radek 32)

**vrati polozky z vydejky zpatky na sklad**

"podminkou" oznacene zbozi bud pripocte zpatky k polozkam ze kterych tyto nove "vznikli", nebo jsou-li tyto uz passe, necha je "zit" a jenom je da zpatky na pobočku

*void* **vrat\_polozky** ( *\$prvni\_podminka* )

▪ **\$prvni\_podminka**

## /is/\_zmena\_sebe.php

## Popis

**prepise registracni udaje aktualniho usera**

je volan z modulu zmena.php, tamtez se vraci s prislusnym &msg=

▪ **zdroj. kod:** [ke stazeni](#)

## Includes

`include ('data/session.php')` (radek 14)

**overi jestli je uzivatel prihlasen, aktualizuje session**

## /is/bio.php

## Popis

**manipulace s bio znamkami - vytvoreni, zmeny**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

▪ **zdroj. kod:** [ke stazeni](#)

## Includes

`include ("data/tab_form.php")` (radek 69)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

## /is/ceny.php

## Popis

**prodejni ceny zbozi na pobočkach**

za beznych okolnosti by se jedinecnost dvojice pobočka+zbozi dala resit na DB urovni (klik pres dva radky), to ale tab\_form.php neumí, tudiz se to resi na aplikacni urovni

vyber pobočky je samozrejmosti, nabídka zboží se resi odfiltrováním hodnot, které už na pobočce jsou(+, byl-li kliknut radek, musí tam zahrnout i ten)

je nutné aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

▪ **zdroj. kod:** [ke stazeni](#)

## Includes

`include ("data/tab_form.php")` (radek 123)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

`include_once ('data/funkce_put_get.php')` (radek 31)

**vlozi funkce na reprodukci PUT/GET parametru**

## /is/data/funkce\_filtr.php

## Popis

**pomucky pri filtrovani ... pouzitelne zejména (jenom?) v modulech tab\_view a tab\_form**

vsechny si produkují nějaký \$filtr\_vystup - do toho se dělá vystup, který pak vrátí

▪ **zdroj. kod:** [ke stazeni](#)

## Functions



**filtr\_sloupec** (radek 95)

**udela input nad sloupec ve filtrovacim radku nad tabulkou**

- **see:** [tab\\_view.php](#)

*void* **filtr\_sloupec** (*array* \$typy, *array* \$sirky, *string* \$sloupec)

- **string \$sloupec:** jmeno sloupce, nad kterym to vzniká
- **array \$typy**
- **array \$sirky:** pole které definuje jestli a jaké mají být ty inputy ... viz. popis napr. v [tab\\_view.php](#)

**filtr\_sql** (radek 48)

**funkce na poreseni filtrovani**

parametry nejsou ... vystaci si s \$\_GET

- **return:** vrati dodelek k dotazu nad tabulku

*string* **filtr\_sql** ()

**math\_selectbox** (radek 18)

**funkce pro vykreslovani selectboxu pro matematicke filtrovani**

(pouzitelna asi jenom ve funkci `filtr_sql` tady)

- **return:** vraci onen selectbox s matematickymi vyrazy, serazen podle situace

*string* **math\_selectbox** (*string* \$sloupec)

- **string \$sloupec:** jmeno filtrovaného sloupce (selectbox se pak bude jmenovat `op_filtr_$sloupec`)

## /is/data/funkce\_help.php

### Popis

**Funkce pro generovani zacatecnickych help-zpravicek v textu stranky...**

deklarovany jsou uz v `main.php`, pouzivany vsude, kde se objevuji napovedy(=vsude)

- **zdroj. kod:** [ke stazeni](#)

### Functions

**help\_icon** (radek 37)

**ikonka v kontextove napovede**

vrati hezky otaznik, který jednak na `onMouseOver` zobrazí popup s textem helpu a jednak slouží jako odkaz na `help_text` se stejným `$jmeno` (napr. jestli člověk nemá možnost javascriptu)

*string* **help\_icon** (*string* \$jmeno, *string* \$text)

- **string \$jmeno:** "Jmeno" textu .. bude se odkazovat přes `href="#jmeno"`, takže divoke znaky nejsou na místě
- **string \$text:** Text helpu. Cokoliv co html unese

**help\_text** (radek 17)

**vrati hezky text helpu. Ten bude taky cílem klikání na ikonku se stejným \$jmeno**

*string* **help\_text** (*string* \$jmeno, *string* \$text)

- **string \$jmeno:** "Jmeno" textu .. bude se odkazovat přes `href="#jmeno"`, takže divoke znaky nejsou na místě
- **string \$text:** Text helpu. Cokoliv co html unese

## /is/data/funkce\_put\_get.php

### Popis

**funkce na reprodukce \_PUT a \_GET parametru, používáno zejména v `tab_view.php` a `tab_form.php`**

- **zdroj. kod:** [ke stazeni](#)

### Functions

**generate\_gets** (radek 42)

**reprodukuje parametry dane pres \_GET**

Taky se vzdy ignoruje parametr 'msg', 'doplň' a 'ne'

- **return:** vraci neco jako &parametr1=hodnota1&parametr2...

*string* **generate\_gets** (**string** \$krome\_ceho)

- **string** \$krome\_ceho: nazev toho parametru, který se má vynechat

**put\_hidden\_inputs** (radek 15)

**Do formularu vklada hidden inputy z \_GET promennych, které by byly jinak zapomenuty**

Taky se vzdy ignoruje parametr 'msg', 'doplň' a 'ne'

*void* **put\_hidden\_inputs** (**string** \$krome\_ceho)

- **string** \$krome\_ceho: nazev toho parametru, který se má vynechat

Documentation generated on Sun, 28 Jan 2007 00:24:29 +0100 by [phpDocumentor 1.3.1](#)

## /is/data/funkce\_transakce.php

Popis

**funkce pro workaround nefungujících transakcí**

- **zdroj. kod:** [ke stazení](#)

Functions

**create\_log** (radek 17)

**vytvorí log soubor s jmenem=dnesní datum,**

do něj dá veškeré informace které by mohly pomoci k nalezení a odstranění chyby

*void* **create\_log** (**string** \$msg)

- **string** \$msg: Text který vypíše do logu - nebo na výstup jestli se nepovede vytvořit

**go\_transakce** (radek 37)

**udělá parodii na transakci**

udělá rychlý sled dotazu, po každém overi jestli ovlivnila aspoň nějaké řádky v případě chyby volá create\_log - vypíše tam stav ve kterém i chyba došlo

*boolean* **go\_transakce** (**array** \$trans)

- **array** \$trans: pole dotazu(stringu) které se mají vykonat

## /is/data/html\_body.php

Popis

**kus kódu který ukončí hlavičku a začne body**

- **zdroj. kod:** [ke stazení](#)

## /is/data/html\_end.php

Popis

**kus kódu, který se vkládá na konec stránky**

- **zdroj. kod:** [ke stazení](#)

## /is/data/html\_head.php

Popis

**kus kódu, který se vkládá na začátek stránky - začne hlavičku**

</head> a <body ..> pak v souboru html\_body.php - někdy je třeba vkládat pár věcí navíc na konec hlavičky

- **zdroj. kod:** [ke stazení](#)

## /is/data/check\_delete/dodavatele.php

Popis

**vloží funkci pro zjištění, jestli může tab\_form.php smazat položky \$ids z tabulky Dodavatele**

dodavatele nemůžeme mazat v případě, že jsou evidováni u nějakého zboží; Vrací seznam tech, co nemůžou být smazány, odděleny tečkami

- **zdroj. kod:** [ke stazení](#)
- **see:** [tab\\_form.php](#)

## Functions

**check\_delete** (radek 20)

**dodavatele nemozno mazat v pripade, ze jsou evidovani u nejakeho zbozi**

string **check\_delete** (string \$ids)

- **string \$ids**: funkce predpoklada na vstupu seznam IDcek oddeleny carkama

### /is/data/check\_delete/kategorie.php

## Popis

**vlozi funkci pro zjistení, jestli se muzou smazat polozky \$ids z tabulky Kategorie**

kategorii nemozno mazat v pripade, ze je evidovana u nejakeho zbozi;  
Vraci seznam tech, co nemuzou byt smazany, oddeleny teckami;  
(na rozdíl od kolegyn z tohoto adresare není volána z \_tab\_form.php, ale z vlastního \_kategorie.php)

- **zdroj. kod**: ke stazeni
- **see**: \_kategorie.php

### /is/data/check\_delete/odberatele.php

## Popis

**vlozi funkci pro zjistení, jestli muze tab\_form.php smazat polozky \$ids z tabulky Odberatele**

Vraci seznam tech, co nemuzou byt smazany, oddeleny teckami

- **zdroj. kod**: ke stazeni
- **see**: tab\_form.php

Documentation generated on Sun, 28 Jan 2007 00:24:39 +0100 by [phpDocumentor 1.3.1](#)

### /is/data/check\_delete/pobocky.php

## Popis

**vlozi funkci pro zjistení, jestli muze tab\_form.php smazat polozky \$ids z tabulky Pobocky**

- **zdroj. kod**: ke stazeni
- **see**: tab\_form.php

Documentation generated on Sun, 28 Jan 2007 00:24:41 +0100 by [phpDocumentor 1.3.1](#)

### /is/data/check\_delete/vyrobci.php

## Popis

**vlozi funkci pro zjistení, jestli muze tab\_form.php smazat polozky \$ids z tabulky Vyrobci**

vyrobce nemozno mazat v pripade, ze jsou evidovani u nejakeho zbozi;  
Vraci seznam tech, co nemuzou byt smazany, oddeleny teckami

- **zdroj. kod**: ke stazeni
- **see**: tab\_form.php

Documentation generated on Sun, 28 Jan 2007 00:25:25 +0100 by [phpDocumentor 1.3.1](#)

### /is/data/check\_delete/zbozi.php

## Popis

**vlozi funkci pro zjistení, jestli muze tab\_form.php smazat polozky \$ids z tabulky Zbozi**

zbozi nemozno mazat v pripade, ze se už objevuje v polozkach;  
Vraci seznam tech, co nemuzou byt smazany, oddeleny teckami;

- **zdroj. kod**: ke stazeni
- **see**: tab\_form.php

Documentation generated on Sun, 28 Jan 2007 00:25:26 +0100 by [phpDocumentor 1.3.1](#)

### /is/data/konekt.php

## Popis

**tenhle kod je vkladan vzdy kdyz je potreba se pripojit do DB**

- **zdroj. kod**: ke stazeni

Documentation generated on Sun, 28 Jan 2007 00:24:36 +0100 by [phpDocumentor 1.3.1](#)

### /is/data/session.php

## Popis

**overuje aktualnost sessiony,**

v pripade potreby presmeruje na prihlasovací stranku  
volana z kazde stranky ISu, kde je treba zarucit prihlaseneho navstevnika

■ zdroj. kod: [ke stazeni](#)

**Includes**

`include ('data/konekt.php')` (radek 46)

**pripoji se do DB (nebo skonci, v pripade neuspechu)**

`include ('data/store_posts.php')` (radek 36)

**jestli nebyla zadana bezna akce,ulozi do SESSION zadane udaje(budou mu vraceny)**

**Constants**

**AutoLogOff** = 2400 (radek 27)

**vteriny, po kterych bude uzivatel automaticky odlogovan**

**Functions**

**ma\_prava** (radek 19)

**overi jestli ma uzivatel/ka dane prava**

*boolean* **ma\_prava** (*string* \$jake)

- **string \$jake:** pozadovane prava prozene pres uzivatelovy prava v DB tvaru, které jsou uloženy v \$\_SESSION['prava'] .. případně jestli nemá "vsechna" prava

Documentation generated on Sun, 28 Jan 2007 00:24:59 +0100 by [phpDocumentor 1.3.1](#)

**/is/data/store\_posts.php****Popis**

**ulozi od SESSION tech POST promenne,**

které chtel uzivatel submitnout, ale z nejakych duvodu(obnoveni loginu, nektery spatny udaj) mu je vracime pouzivano(a volano) v modulu tab\_form.php

■ zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:25:01 +0100 by [phpDocumentor 1.3.1](#)

**/is/data/tab\_form.php****Popis**

**kus kodu("modul") vkladan jako procedura na vykresleni updatovaci tabulky**

zanechava za sebou (podle potreby) naplnene - avsak nevykreslene - boxy s help zpravami ve stringu \$help\_tab - muzou se vypsat bez overovani nastaveni helpu, to se overuje uz tady pokracuje - taky univerzalne - volanim \_tab\_form.php - a pak redirect zpatky

"parametry" :

- \$tab\_tab - tabulka ve které se bude stourat, s případným omezením jako WHERE, GROUP BY atd.
- \$default\_sort - nepovinný - jak se bude defaultně radit(první místo jestli user neklikal, dříve pak)
- \$read\_only - nepovinný - jména sloupce (pole stringu), které budou readonly
- \$trans - nepovinný - asociativní sloupec->pole hodnot,pro selectboxy/transformaci klicu na hodnoty
- \$hidden - nepovinný - asociativní pole sloupec -> hodnota pro sloupce, které nemá zobrazovat(ale doplnit tam "tu" hodnotu)
- \$default\_val - nepovinný - asociativní pole sloupec -> hodnota pro sloupce, které mají mít nějakou tu předvyplněnou hodnotu
- \$no\_insert - boolean - jestli true, není možno vkládat nové řádky
- \$filtr - nepovinný - něco jako trans,ale pro určení filtrovatelných sloupců,vždy název sloupce a
  - pole - pak se filtruje pomocí selectboxu z pole(jako trans ale NULL by mělo být definováno explicitně)
  - 'M' - bude se operovat s matematickými výrazy
  - 'T' - textové pole, přes které se poze klasicky LIKE
- \$limit - nepovinný - jestli zadán,objeví se toolbar na omezování počtu řádků;hodnota \$limit je výchozí hodnotou
- \$head\_name - nepovinný - pole oficiální\_jméno\_sloupce->to\_co\_zobrazí\_tady .. nezadaným zůstane samozřejmě oficiální
- \$input\_width - nepovinný - asociativní pole sloupec -> hodnota pro sloupce(resp.inputy nad ním), které mají pevnou šířku..hodnota půjde CSSkovyho width:
- \$special\_input - nepovinný - asociativní pole sloupec -> hodnota pro sloupce(resp.inputy nad ním), které mají být nějakým způsobem "speciální", t.j. hodnota bude místo <input> (pozor, tabindex se počítá až počas kreslení tabulky - "resení" je dat speciálnímu inputu předpokladanou hodnotu(která tam pak bude dvakrát, to ale moc nevadí)
- \$input\_up - nepovinný - jestli je nenulový, bude "vkládací řádek" nad datovými řádky(jinak pod)
- \$check\_delete - nepovinný - soubor, ve kterém se nachází funkce check\_delete(id), která vrátí 1, pokud může být řádek s prim.klicem 'id' smazán

pripojeni do databaze se odehrava v data/session.php

■ zdroj. kod: [ke stazeni](#)

**Includes**

`include_once ('funkce_put_get.php')` (radek 37)

**vlozi funkce na reprodukci PUT/GET parametru**

`include_once ('funkce_filtr.php')` (radek 42)

**vlozi funkce na praci s filtrovanim dat**

Documentation generated on Sun, 28 Jan 2007 00:25:02 +0100 by [phpDocumentor 1.3.1](#)

## /is/data/tab\_view.php

### Popis

**kus kodu vkladan jako procedura na vykresleni tabulky zobrazujici libovolny select**

zanechava za sebou (podle potreby) naplnene - avsak nevykreslene - boxy s help zpravama v \$help\_tab - muzou se vypsat bez overovani nastaveni helpu, to se overuje uz tady

"parametry" :

- \$tab\_sql - dotaz na vytahnuti dat - pozor, musi byt bez ORDER BY a HAVING!
- \$default\_sort - nepovinný - jak se bude defaultne radit(prvni misto jestli user neklikal, druhe pak)
- \$filtr - nepovinný - neco jako trans,ale pro urceni filtrovatelnych sloupce,vzdy nazev sloupce a
  - pole - pak se filtruje pomoci selectboxu z pole(jako trans ale NULL by melo byt definovano explicitne)
  - 'M' - bude se operovat s matematickymi vyrazy
  - 'T' - textove pole, pres ktere se pozene klasicky LIKE
- \$tabindex - nepovinný - jaky tabindex maji mit sortovaci klikatka(default:1)
- \$limit - nepovinný - jestli zadan,objevi se toolbar na omezovani poctu radku;hodnota \$limit je vychozi hodnotou
- \$head\_name - nepovinný - pole oficialni\_jmeno\_sloupce->to\_co\_zobrazi\_tady .. nezadany zustane samozrejme oficialni

pozor, (oproti tab\_form) chybi parametr trans(tady je rozumnejsi udelat JOIN)

pripojeni do databaze se odehrava v data/session.php

- **zdroj. kod:** [ke stazeni](#)

### Includes

`include_once ('funkce_put_get.php')` (radek 29)

**vlozi funkce na reprodukci PUT/GET parametru**

`include_once ('funkce_filtr.php')` (radek 34)

**vlozi funkce na praci s filtrovanim dat**

Documentation generated on Sun, 28 Jan 2007 00:25:09 +0100 by [phpDocumentor 1.3.1](#)

## /is/dodavatele.php

### Popis

**manipulace s dodavateli - vytvoreni, nastavovani....**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem nadefinovane promenne pouzije ve tab\_form.php

- **zdroj. kod:** [ke stazeni](#)
- **see:** [tab\\_form.php](#)

### Includes

`include ("data/tab_form.php")` (radek 84)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

Documentation generated on Sun, 28 Jan 2007 00:24:28 +0100 by [phpDocumentor 1.3.1](#)

## /is/doklad.php

### Popis

**vydejka nebo prijemka ve tvaru pro tisk**

volano z main.php?co=vydejka nebo prijemka (jsou hodne podobne, proto spolu .. faktura je ve faktura.php)

ocekava pres \_GET :

- co=vydejka nebo co=prijemka
- id=neco pro cislo dokladu
- **zdroj. kod:** [ke stazeni](#)

### Includes

*include ('data/session.php')* (radek 25)

**overi jestli je uzivatel prihlasen, aktualizuje session**

*Documentation generated on Sun, 28 Jan 2007 00:24:28 +0100 by phpDocumentor 1.3.1*

## /is/dph.php

### Popis

#### Includes

**manipulace se sazbami DPH - vytvoreni, nastavovani**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

*include ("data/tab\_form.php")* (radek 57)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

*Documentation generated on Sun, 28 Jan 2007 00:24:28 +0100 by phpDocumentor 1.3.1*

## /is/export\_csv.php

### Popis

**MySQL dotaz, který dostava v \$\_SESSION['export'] nabidne na download jako CSV soubor**

jestli dostane parametr v \$\_GET['co'], soubor se bude jmenovat \$\_GET['co'].csv, jinak export\_dnes.csv

je volan z ucetnickych modulu prehled\_neco.php

- zdroj. kod: [ke stazeni](#)

### Includes

*include ('data/session.php')* (radek 19)

**overi jestli je uzivatel prihlasen, aktualizuje session**

*Documentation generated on Sun, 28 Jan 2007 00:24:29 +0100 by phpDocumentor 1.3.1*

## /is/faktura.php

### Popis

**faktura ve tvaru pro tisk (volano z main.php?co=vydejka )**

ocekava pres \_GET : ?id=neco pro cislo dokladu

- zdroj. kod: [ke stazeni](#)

### Includes

*include ('data/session.php')* (radek 14)

**overi jestli je uzivatel prihlasen, aktualizuje session**

*Documentation generated on Sun, 28 Jan 2007 00:24:29 +0100 by phpDocumentor 1.3.1*

## /is/index.php

### Popis

**prihlasovací formular**

jestli je volan z ISu(vyprsel session), dostava informaci o tom, kam se ma vratit

volan z \_login.php, \_logout.php nebo data/session.php nebo primo jako uvodni stranka

pokracuje pres \_login.php

- zdroj. kod: [ke stazeni](#)

### Includes

*include ('data/konekt.php')* (radek 30)

**pripoji se do DB**

(nebo skonci, v pripade neuspechu)

*require ('data/html\_head.php')* (radek 18)

**vlozi (neukoncenou!) hlavicku html souboru**

*require ('data/html\_end.php')* (radek 63)

**ukonci body, html**

*require* ('data/html\_body.php') (radek 23)

**ukonci hlavicku, zapocne body**

Documentation generated on Sun, 28 Jan 2007 00:24:31 +0100 by [phpDocumentor 1.3.1](#)

## /is/kategorie.php

### Popis

**manipulace s kategoriemi - presunovani, prejmenovavani, vytvoreni, mazani...**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem na zmeny nepouziva tab\_form.php, vola si vlastni \_kategorie.php

- zdroj. kod: [ke stazeni](#)

### Includes

*include\_once* ('data/funkce\_put\_get.php') (radek 62)

**vlozi funkce na reprodukci PUT/GET parametru**

Documentation generated on Sun, 28 Jan 2007 00:24:31 +0100 by [phpDocumentor 1.3.1](#)

## /is/lide.php

### Popis

**vypise seznam aktivnich operatoru, pripadne strucne informace o nem/ni, bylo-li kliknuto...**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

*include* ("data/tab\_view.php") (radek 55)

**vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;**

zuzitkuje zde dosavadni parametry a nakresli tabulku

Documentation generated on Sun, 28 Jan 2007 00:24:36 +0100 by [phpDocumentor 1.3.1](#)

## /is/main.php

### Popis

**"hlavni" soubor - postara se o vykresleni spravneho menu, obsahu**

Jestli ma uzivatel v nastaveni zobrazovani help messagu, nacte prislusne javascripty

Udela 3 absolutne pozicovane DIV-y : leve menu, horni listu a hlavni screen, pak pripoji prislusny "modul" dany pres ?kde=lokalita

- zdroj. kod: [ke stazeni](#)

### Includes

*include* (\$kde.".php") (radek 110)

**vlozeni hlavniho obsahu**

jestli byl pozadovan nejaky neexistujici "modul", pripoji uvodni stranku

*include* ('data/session.php') (radek 15)

**overi jestli je uzivatel prihlasen, aktualizuje session**

*include* ('data/funkce\_help.php') (radek 22)

**vlozi funkce pro "napovedu"**

t.j. help\_icon a help\_text

*require* ('data/html\_head.php') (radek 27)

**vlozi (neukoncenou!) hlavicku html souboru**

*require* ('data/html\_body.php') (radek 35)

**ukonci hlavicku, zapocne body**

`require ('data/html_end.php')` (radek 122)

**ukonci body, html**

`require ("prvni.php")` (radek 117)

**vlozeni "uvodni stranky"**

jestli se main.php nepovedlo vlozit to, co uzivatel pozadoval pres \_GET, vlozi uvodni stranku

Documentation generated on Sun, 28 Jan 2007 00:24:37 +0100 by [phpDocumentor 1.3.1](#)

## /is/odberatele.php

### Popis

**manipulace s odberateli - vytvoreni, nastavovani....**

odberatele jsou v systemu jenom pro potreby tisku, neprirazujou se ke zbozi, ale k vydejce

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_form.php")` (radek 78)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

Documentation generated on Sun, 28 Jan 2007 00:24:40 +0100 by [phpDocumentor 1.3.1](#)

## /is/pobocky.php

### Popis

**manipulace s pobočkami - vytvoreni, nastaveni stavajících**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_form.php")` (radek 70)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

Documentation generated on Sun, 28 Jan 2007 00:24:42 +0100 by [phpDocumentor 1.3.1](#)

## /is/prehled\_dph.php

### Popis

**prehled DPH za vybrane obdobi - pro potreby ucetnictvi**

moznost exportu vysledku do CSV souboru

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_view.php")` (radek 89)

**vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;**

zuzitkuje ze dosavadni parametry

Documentation generated on Sun, 28 Jan 2007 00:24:43 +0100 by [phpDocumentor 1.3.1](#)

## /is/prehled\_zisku.php

### Popis

**prehled zisku za vybrane obdobi - pro potreby ucetnictvi**

moznost exportu vysledku do CSV souboru

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_view.php")` (radek 91)

**vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;**



zuzitkuje zde dosavadni parametry a nakresli tabulku

Documentation generated on Sun, 28 Jan 2007 00:24:45 +0100 by [phpDocumentor 1.3.1](#)

## /is/presuny.php

### Popis

#### presuny polozek mezi sklady

da se tady bud iniciovat presun (vola `_presklad.php`) nebo presun ukoncit(stornem nebo dorucenim, oboje `_doruceni.php`)

je to vlastne 3 krat `tab_view`; ty ale nejsou staveny tak, aby jich mohlo bezet vic na jedne strane(braly by si vzajemne parametry).  
tim padem se klika pro pozadovanou oblast - je to tak i prehlednejsi (pres parametr `?akce=k_nam` nebo `=od_nas` nebo nic z toho)

je nutne aby byl volan jako modul z `main.php`, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_view.php")` (radek 107)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a nakresli tabulku s presuny

`include ("data/tab_view.php")` (radek 143)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a nakresli tabulku s presuny

`include ("data/tab_view.php")` (radek 183)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a nakresli tabulku se zbozim

Documentation generated on Sun, 28 Jan 2007 00:24:46 +0100 by [phpDocumentor 1.3.1](#)

## /is/prijemka.php

### Popis

#### detaily o prijemce

.. a zaroven moznost doplnit ci uzavrit, da-li se.  
muze zde, v zavislosti od stavu objednávky a prav, volat `_nasklad.php` pro naskladneni, `_potvrd_prij.php` pro potvrzeni prijemky, `_poznamka_prij.php` pro nastaveni poznamky a nebo `_smaz_prij.php` pro smazani/stornovani prijemky

do pozornosti : sazby DPH v nabídce radi podle ID(ne nazvu); momentalne prvni : snizena

je nutne aby byl volan jako modul z `main.php`, v jeho menu se ale nenachazi(bezna cesta sem je `pres ?co=prijemky`, alternativni taky kdyz je nekde pozadovano info o zbozi)

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_form.php")` (radek 144)

#### vlozi kod na vykresleni 'univerzalni editacni tabulky'

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

`include ("data/tab_view.php")` (radek 168)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a vypise polozky na prijemce

Documentation generated on Sun, 28 Jan 2007 00:24:50 +0100 by [phpDocumentor 1.3.1](#)

## /is/prijemky.php

### Popis

#### seznam prijemek, moznost jit na jejich detaily, pod tim odkaz na vytvoreni nove

`tab_view.php` vykresli tabulku, ta v sobe obsahuje linky na `?co=lide` a `?co=prijemka`

je nutne aby byl volan jako modul z `main.php`, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

### Includes

`include ("data/tab_view.php")` (radek 80)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje ze dosavadni parametry

Documentation generated on Sun, 28 Jan 2007 00:24:55 +0100 by [phpDocumentor 1.3.1](#)

## /is/prodej.php

### Popis

#### vytvoreni nove vydejky (primy prodej)

po odeslani vola \_nova\_faktura.php se splatnosti NULL a uradou v hotovosti  
je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:24:56 +0100 by [phpDocumentor 1.3.1](#)

## /is/prodej\_faktura.php

### Popis

#### formular pro vytvoreni nove faktury

po odeslani vola \_nova\_faktura.php

- var:  
predvolena splatnost faktury do formulare  
je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem
- zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:24:57 +0100 by [phpDocumentor 1.3.1](#)

## /is/prvni.php

### Popis

#### uvodni stranka, u zapnutych help-zprav zobrazi napovedu k napovede

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:24:59 +0100 by [phpDocumentor 1.3.1](#)

## /is/ucty.php

### Popis

#### manipulace s (cizimi) ucty - vytvoreni noveho, pripadne nastaveni udaju, prav

vola \_ucty.php pro zmenu nebo sam sebe pro zmenu usera

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:25:13 +0100 by [phpDocumentor 1.3.1](#)

## /is/vydejka.php

### Popis

#### detaily o vydejce/fakture .. a zaroven moznost doplnit ci uzavrit, da-li se

- tom, jestli je to faktura nebo vydejka se rozhoduje podle toho, jestli ma zadanou splatnost (jinak jsou, az na tiskovy vystup, v podstate stejne)

pro vetsi cast akci vola \_vydejka\_akce.php, pro nektere taky \_potvrd\_vydej.php, \_poznamka\_vydej.php ci \_odberatel\_vydej.php

je nutne aby byl volan jako modul z main.php, v jeho menu se ale nenachazi (bezna cesta sem je pres ?co=vydejky)

- zdroj. kod: [ke stazeni](#)

### Includes

*include ("data/tab\_view.php")* (radek 180)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a nakresli tabulku

*include ("data/tab\_view.php")* (radek 215)

#### vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;

zuzitkuje zde dosavadni parametry a nakresli tabulku s prehledem polozek

Documentation generated on Sun, 28 Jan 2007 00:25:16 +0100 by [phpDocumentor 1.3.1](#)

## /is/vydejky.php

### Popis

#### seznam vydejek ci faktur,

moznost jit na jejich detaily, pod tim odkaz na vytvoreni nove

muze teda jit na ?co=vydejka (detaily vydejky), ?co=lide (info o lidech) nebo ?co=prodej nebo ?co=prodej\_faktura (vytvoreni nove)

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

## Includes

`include ("data/tab_view.php")` (radek 91)

**vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;**

zuzitkuje zde dosavadni parametry a nakresli tabulku

Documentation generated on Sun, 28 Jan 2007 00:25:21 +0100 by [phpDocumentor 1.3.1](#)

## /is/vyrobci.php

### Popis

**manipulace s vyrobci - vytvoreni, nastavovani....**

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

## Includes

`include ("data/tab_form.php")` (radek 74)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

Documentation generated on Sun, 28 Jan 2007 00:25:24 +0100 by [phpDocumentor 1.3.1](#)

## /is/zasoby.php

### Popis

## zobrazí stav položek na skladech

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

## Includes

`include ("data/tab_view.php")` (radek 45)

**vlozi kod na vykresleni 'univerzalni tabulky' na prohlizeni;**

zuzitkuje zde dosavadni parametry a nakresli tabulku

Documentation generated on Sun, 28 Jan 2007 00:25:25 +0100 by [phpDocumentor 1.3.1](#)

## /is/zbozi.php

### Popis

**definice zboží - editace, pridavani**

vyuziva tab\_form.php, ale definuje si vlastni vstup pro vyber kategorie

- a jelikoz tento zvetsuje vysku vstupoveho radku, vyuziva "vyhody"

a pro vstup "info" dava místo bezneho textoveho inputu vstup textarea

je nutne aby byl volan jako modul z main.php, v jeho menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

## Includes

`include ("data/tab_form.php")` (radek 246)

**vlozi kod na vykresleni 'univerzalni editacni tabulky'**

zuzitkuje zde dosavadni parametry a nakresli tabulku s formularem

`include_once ('data/funkce_put_get.php')` (radek 135)

**vlozi funkce na reprodukci PUT/GET parametru**

Documentation generated on Sun, 28 Jan 2007 00:25:27 +0100 by [phpDocumentor 1.3.1](#)

## /is/zmena.php

## Popis

### zmena vlastnich registracnich udaju

plus zapinani a vypinani otravne napovedy

pro veskere akce vola `_zmena_sebe.php`

je nutne aby byl volan jako modul z `main.php`, v jehoz menu se taky nachazi odkaz sem

- zdroj. kod: [ke stazeni](#)

Documentation generated on Sun, 28 Jan 2007 00:25:32 +0100 by [phpDocumentor 1.3.1](#)

## /zbozi.php

## Popis

### "Verejne" prohlizeni zbozi

Stoji mimo IS, resp. je doplnkem webu [www.fairtrade.cz](http://www.fairtrade.cz) pro prohlizeni zbozi.

Umoznuje hledat zbozi podle kategorie, nazvu, vysledky hledani filtrovat...

O nalezenem zbozi je mozno zobrazit detaily

Pozor! Na rozdil od zbytku FT stranek(cp-1250) se odehrava v kodovani UTF-8(stejne jako IS)

- **todo:** Bude vhodne zahrnout prohledavani zbozi i ve vyhledavaci FT webu;  
V `inc/actions.php` je momentalne zakomentovane prohledavani v nazvech zbozi, které pak odkazuje sem

## Includes

`include ("is/data/tab_view.php")` (radek 344)

**pouzije predchozi parametry a nakresli univerzalni tabulku na zobrazeni zbozi**

`include ("is/data/tab_view.php")` (radek 354)

**pouzije predchozi parametry a nakresli univerzalni tabulku na zobrazeni zbozi**

`include_once ("is/data/funkce_help.php")` (radek 329)

**Nadefinuje funkce na zobrazovani helpu**

Ty muzou byt pozadovany v dalsim kodu, ac se tady asi nepouzijou

`require_once ('is/data/konekt.php')` (radek 91)

**kvuli kodovani je treba pouzit prihlasovani z IS-u**

`require_once ('inc/menu.php')` (radek 82)

**pouzito pro vykresleni ostatnich veci pres web - menu atd.**

`require_once ('inc/login.php')` (radek 78)

**pouzito pro vykresleni ostatnich veci pres web - menu atd.**

`require_once ('inc/actions.php')` (radek 86)

**pouzito pro vykresleni ostatnich veci pres web - menu atd.**

## Functions

**callback** (radek 51)

**funkce, která vraci vstup z webu(cp1250) v našem kodovani(utf-8)**

*string* **callback** (*string* \$buffer)

- **string \$buffer:** zejména sem pujde presmerovany vystup

**udelej\_popisek** (radek 63)

**resi popisak podle toho, jestli je to odkaz na clanek**

ze zadaneho stringu zjistí jestli je to cista URL, podle toho vrati bud okaz na ni, nebo normalne text odkazu

*string* **udelej\_popisek** (*string* \$text)

- **string \$text**

Documentation generated on Sun, 28 Jan 2007 00:25:32 +0100 by [phpDocumentor 1.3.1](#)