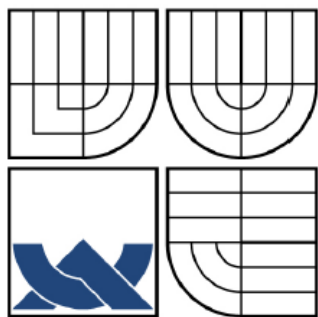
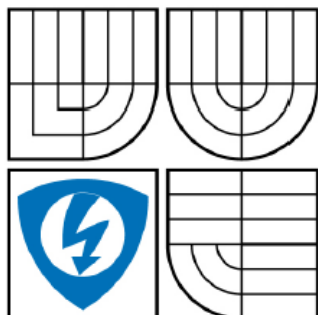




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNologiÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

FIREMNÍ INFORMAČNÍ SYSTÉM

COMPANY INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

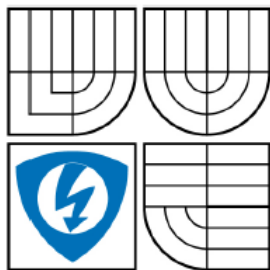
AUTHOR

PAVEL HRONEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. TOMÁŠ MACHO, Ph.D.



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav automatizace a měřicí techniky

Bakalářská práce

bakalářský studijní obor
Automatizační a měřicí technika

Student: Pavel Hronek

Ročník: 3

ID: 70360

Akademický rok: 2008/2009

NÁZEV TÉMATU:

Firemní informační systém

POKYNY PRO VYPRACOVÁNÍ:

1. Seznamte se s požadavky na informační systém pro firmu S. N. O. P. zabývající se výrobou autosoučástek.
2. Navrhněte koncepci informačního systému.
3. Proved'te analýzu databázového systému a nakreslete E-R diagramy. Zabývejte se problematikou životních cyklů.
4. Informační systém naprogramujte a odlaďte. Použijte databázové prostředí MySQL.

DOPORUČENÁ LITERATURA:

Maslakowski M., Naučte se MySQL za 21 dní. Praha: Computer Press, 2001. 478 s. ISBN 80-72226-448-6.

Brázda J., PHP 5 začínáme programovat. Praha: Grada Publishing a.s., 2006. 244 s. ISBN 80-247-1146-X.

Dále dle vlastního výběru.

Termín zadání: 9.2.2009

Termín odevzdání: 1.6.2009

Vedoucí práce: Ing. Tomáš Macho, Ph.D.

prof. Ing. Pavel Jura, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Abstrakt

Předložená práce obsahuje analýzu, návrh a implementaci informačního systému pro firmu S.N.O.P, který umožňuje udržovat informace o zařízeních pracujících s výlisky. Dovoluje evidovat závady, denní a pravidelné kontroly, poskytuje podporu pro plánování údržby zařízení. Pro implementaci byl použit databázový systém MySQL 5.0 a skriptovací jazyk PHP 5.2. Informační systém obsahuje zabezpečení proti neoprávněnému přístupu a proti schopnosti uživatele modifikovat sql příkazy.

Klíčová slova: Informační systém, MySQL, PHP, návrh databáze, analýza IS,

Abstract

The present work includes analysis, design and implementation of information system for the company S. N. O. P., which allows to maintain information about working with moldings. Allows to record defects, daily and periodic inspections, providing support for the planning of maintenance equipment. For the implementation used database system MySQL 5.0 and scripting language PHP 5 The information system includes security against unauthorized access and the ability of users to modify the sql commands.

Keywords: Information system, MySQL, PHP, database design, analysis of IS

HRONEK, P. *Firemní informační systém*. Brno: Vysoké učení technické v Brně,
Fakulta elektrotechniky a komunikačních technologií, 2009. 48 s.
Vedoucí bakalářské práce Ing. Tomáš Macho, Ph.D.

Prohlášení

„Prohlašuji, že svou bakalářskou práci na téma Firemní informační systém jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne: **1. června 2009**

.....
podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Tomáši Machovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne: **1. června 2009**

.....
podpis autora

OBSAH

1. ÚVOD	8
2. POPIS PROBLEMATIKY VÝROBY DÍLŮ A POŽADAVKŮ FIRMY NA IS	9
2.1 Popis požadavků firmy na informační systém	10
3. NÁVRH KONCEPCE IS	13
3.1 Použité technologie, vývojové nástroje a programy	13
3.2 Návrh koncepce IS	16
3.2.1 Rozdělení kategorií podle obsahu.....	18
3.2.2 SQL dotazy	21
4. E-R DIAGRAM A ANALÝZA DATABÁZOVÉHO SYSTÉMU.....	22
4.1 Základní pravidla pro návrh databáze	22
4.1.1 Typy tabulek	22
4.1.2 Normalizační pravidla	23
4.1.3 Vztahy.....	24
4.2 Analýza databázového systému	25
4.2.1 Použité datové typy.....	25
4.2.2 Popis a vazby mezi tabulkami	25
4.3 Životní cykly	27
4.3.1 Životní cykly vybraných entit.....	27
5. POPIS IMPLEMENTOVANÉHO SYSTÉMU	31
5.1 Popis systému z pohledu uživatele IS	31
5.1.1 Popis jednotlivých modulů	31
5.2 Popis systému z pohledu programátora	36
5.2.1 Zabezpečení a autentizace uživatelů.....	36
5.2.2 Popis hlavních funkcí souboru function.php pro práci s formuláři	38
5.2.3 Popis hlavních funkcí souboru upload.php pro práci s uploadem souborů ..	42
5.2.4 Popis hlavních funkcí souboru db_class.php pro práci s databází.....	44
6. ZÁVĚR.....	47
7. LITERATURA	48

SEZNAM OBRÁZKŮ

- Obr.č.1:** Lisy Balconi
- Obr. č.2:** Ukázka prostředí phpMyAdmin
- Obr. č.3:** Ukázka grafické podoby IS
- Obr. č.4:** Životní cyklus tabulky zařízení
- Obr. č.5:** Životní cyklus tabulky malé opravy
- Obr. č.6:** Životní cyklus tabulky velké opravy, údržba
- Obr. č.8:** Životní cyklus tabulky sklady náhradních dílů
- Obr. č.8:** Vývojový diagram funkce pro relaci
- Obr. č.9:** Vývojový diagram funkce pro filtraci
- Obr. č.10:** Vývojový diagram funkce pro update nd ze SAPu
- Obr. č.11:** Vývojový diagram funkce pro kontrolu uploadu
- Obr. č.12:** Vývojový diagram funkce pro kontrolu práv

SEZNAM ZKRATEK

- IS** Information system - Informační systém
- E-R** Entity-relationship – Entita-relace
- PLC** Programmable Logic Controller– Programovatelný logický automat
- PC** Personal computer – Osobní počítač
- PHP** Hypertext Preprocessor – hypertextový preprocesor
- SQL** Structured Query Language – strukturovaný jazyk pro dotazy na relační databáze
- HTML** HyperText Markup Language – hypertextový značkovací jazyk
- OS** Operating System – operační systém
- ODBC** Open DataBase Connectivity – systém pro propojení s databází
- GPL** General Public Licence – všeobecná veřejná licence GNU
- CSS** Cascading Style Sheets – tabulky kaskádových stylů
- HW** Hardware – technické vybavení počítače
- DB** Database - databáze

1. ÚVOD

Firma S. N. O. P., která vyrábí lisované auto-součástky má velké množství strojů na zpracování výlisků. Papírová evidence závad, denních a pravidelných kontrol a údržeb je při takovém množství strojů problém. Zaměstnanci, kteří se zabývají opravou a údržbou strojů stráví velké množství času vyplňováním papírových formulářů, které se navíc mohou snadno ztratit. Navíc nemají možnost si zjistit aktuální stav, vše závisí jen na jejich komunikaci. Proto vyvstal požadavek na nasazení informačního systému, který by usnadnil práci jak zaměstnancům, tak i vedení firmy, které bude mít aktuální přehled o stavu zařízení. Návrh a implementace takového informačního systému je předmětem předložené bakalářské práce.

V první části práce je uveden popis výrobního procesu a seznámení se s požadavky firmy na informační systém. Druhá část je věnována návrhu informačního systému. Poslední část práce obsahuje popis implementovaného systému.

2. POPIS PROBLEMATIKY VÝROBY DÍLŮ A POŽADAVKŮ FIRMY NA IS

Nejdříve se seznámíme s celým postupem výroby dílů. Do firmy přivezou pásy plechu, které se uskladní. Poté se pás portálovým jeřábem přemístí do podavače, ten pás rozvíjí a předává vysokotlakému hydraulickému lisu. Tento lis má výměnné lisovací hlavy (ve firemní řeči jsou nazývány jako nástroje), které mění obsluhu lisu podle toho, který díl chce lisovat. Po výstupu z lisu se díly umístí do plechových beden, které se převezou do skladu. Podle projektu se pak oddělení výroby rozhodne, co s danými výlisky udělá. Buď se tyto polotovary dále zpracovávají v automatických robotech, nebo je upravují zaměstnanci. Ti pracují u poloautomatických svářecích systémů, kde se na díly přivařují např. matky nebo jiné součásti. Poté se díly znovu umístí do kovových beden a jdou na výstupní kontrolu, kde je zaměstnanci všechny přeskládají a zkontrolují sváry. Bedny zabalí a jsou uskladněny, než si pro ně přijede nějaká spediční firma. Každá série dílů je testována na trhačím stroji, kde se testují hlavně manuálně dělané operace jako např. přivařené matky. Tyto informace se musí archivovat po dobu 10 let od ukončení výroby série, z důvodu případné pozdější zkoušky .



Jedním z nejdůležitějších zařízení ve firmě jsou hydraulické lisy od italské firmy Balconi. Těchto kolosů, vážících okolo 200 tun, je zde asi dvacet a jsou plně automaticky řízené pomocí PLC od firmy Siemens. Tyto lisy jsou schopny zvládnout 100 - 1200 tun materiálu za hodinu.

Obr.1: Lisy Balconi

Lisy mají před sebou ještě válcové podavače, které rozvíjí pásy plechu.

Dále jsou zde automatické svařovací roboty od firem ABB a KUKA. Tito roboti mají otočné podávací a svařovací kleště, které jsou schopné rychlého a zároveň přesného pohybu. Také jsou zde poloautomatické bodové svářečky od firmy SERRA, které jsou rovněž řízeny pomocí PLC Siemens. Pak jsou zde ostatní zařízení jako různé podavače matic, nýtovací stroje, balící a montážní stroje apod.

O tato zařízení se starají tři oddělení. Jednak oddělení technologie, které má na starosti komplexní udržování strojů z hlediska automatizace, jednak oddělení údržby strojů, které se stará o fyzický chod zařízení. Opravují prasklé hadice, ložiska a vše, co se týká opotřebení dílů. Poté je zde oddělení expertů automatizace, kde jsou odborně vyškolení pracovníci, kteří se specializují přímo na jedno dané zařízení.

2.1 POPIS POŽADAVKŮ FIRMY NA INFORMAČNÍ SYSTÉM

Informační systém (dále také IS) by měl být dostupný z jakéhokoliv počítače ve firemní síti. V systému by měly existovat uživatelské účty, které budou rozděleny podle rolí jednotlivých zaměstnanců. Systém by měl být uživatelsky přívětivý, ale není kladen důraz na grafickou podobu systému.

Zařízení

Úkolem je vytvořit systém, který bude obsahovat databázi všech zařízení. U každého z nich musí být uveden jeho název, kód, používaný ve firmě pro přesné určení daného stroje, typ zařízení tj. typová specifikace od výrobce a popis, kde se upřesní v případě dvou stejných zařízení ve firmě co které dělá, dále inventární a výrobní číslo. Měl být odkaz na případnou literaturu pro preventivní údržbu. Každému typu zařízení se přiřadí, na kterém úseku ve firmě pracuje, do jaké skupiny zařízení patří a kde je stroj umístěn. Musí být možné rozlišit, zda se jedná o tzv. rozhodující zařízení, které pak dále určuje svými parametry údržbu dalším stejným zařízením nebo zda-li je to nástroj. Zařízení může být ze seznamů vyřazeno, ale data musí zůstat archivovaná v databázi.

Kontroly, prevence a údržba

Se zařízením souvisí kontroly, prevence a údržba zařízení. Tyto operace mohou být shodné pro více různých strojů. Mělo by být specifikováno čeho se prohlídka týká, jaký je její přibližný čas, popis toho, co se má udělat a určení, jak často tuto prohlídku dělat.

Denní kontroly

Další část jsou denní kontroly. Ty se týkají vždy úseků, kterým je daný stroj přidělen. Zde stačí jen popis kontroly, která se má provést a její upřesnění.

Záznamy expertů

Dalším oddílem by měla být část, kam by se ukládaly záznamy od expertů. Ty se váží k danému zařízení a jejich součástí by mělo být datum, popis projevu závady, popis jak závadu odstranili a typ závady – ty budou v předem stanoveném seznamu.

Malé opravy

Na zařízení se každý den vyskytují nějaké malé závady, jako např. nesvítí žárovka, nejde motor apod.. Vždy se specifikují pro konkrétní zařízení. Součástí by měl být začátek a konec opravy, popisy závad, popisy a typ oprav. Typy oprav jsou opět přesně specifikovány podle podnikových standardů. Mělo by být také možno zjistit, které opravy ještě nejsou dokončeny.

Plány údržby, větší opravy

Další částí jsou plány údržby a generálních oprav. Úkolem je přesné naplánování údržby každého stroje. Součástí je číslo plánu, typ údržby (typy jsou přesně specifikovány), datum a čas začátku a konce údržby, její přesný popis a kdo údržbu vložil a kdo jí naopak dokončil. Dále by se mělo zadávat období nebo datum, kdy by měla být údržba provedena a fotky dokumentující průběh údržby. Množství fotografií je kolem deseti u jedné údržby.

Náhradní díly

Poslední důležitou částí jsou náhradní díly. Ty mohou být společné pro více zařízení, lišit se však budou počtem kusů vázaných ke každému zařízení. Součástí je reference dílu v SAPUu. Tu by měl systém generovat podle aktuálního čísla položky v databázi a specifického prefixu. Dále by měl obsahovat reference náhradního dílu, zařazení do skupiny náhradních dílů, které jsou předem specifikovány, jeho výrobce a dodavatele (ti mohou být odlišní), popis náhradního dílu a jeho přesné umístění a fotografie. Skladové položky jsou rozděleny na počet kusů, které jsou celkově aktuálně na skladě, počet minimálních položek a počet objednaných ks. Počet objednaných položek je aktuální množství ks, na které se poslala objednávka. Systém zadávání náhradních dílů by měl být takovýto: do informačního systému se zadá nový díl, vyplní se u něj všechny potřebné informace jako skupina, reference apod. IS pro skladovou položku vygeneruje referenční číslo. Uživatel pak musí tento náhradní díl zadat také do SAPu s příslušnou vygenerovanou referencí. Ze SAPu se každý den budou exportovat informace do samostatného souboru. Z tohoto souboru se pak budou aktualizovat skladové položky u jednotlivých náhradních dílů.

Podniky

Systém je zatím primárně určen pro dva různé závody v České republice, ale měla by být možnost v případě potřeby rozšířit tento systém na více závodů. Zařízení musí být rozlišeny podle podniku. To samé u náhradních dílu, ale s podmínkou, že v databázi bude náhradní díl s určitou referencí zadán jen jednou (reference jsou shodné pro všechny závody). Rozlišeny budou jen ostatní položky, tj. počty kusů náhradních dílů ve skladu, výrobci a dodavatelé (ty si může každý závod zvolit podle svého) a umístění.

3. NÁVRH KONCEPCE IS

Zvolil jsem informační systém implementovaný do webového rozhraní. Důvodem je jeho síla, která se týká kompatibility a možnosti přístupu do IS na kterémkoli PC bez zásahu správce sítě.

3.1 POUŽITÉ TECHNOLOGIE, VÝVOJOVÉ NÁSTROJE A PROGRAMY

Systém je postaven na databázovém systému MySQL a programovacím jazyku PHP. Databázi MySQL jsem si vybral pro její možnosti, které se vyrovnají konkurenčním, draze placeným databázovým systémům. Pro návrh a vývoj IS jsem používal notebook s OS Windows XP - SP3, MySQL 5.0.45, PHP 5.2.4, phpMyAdmin 2.11.1, Apache a návrhový program DBDesigner 4.

PHP

Název tohoto skriptovacího jazyka, šířeného jako Open Source, je odvozen od slov Hypertext Preprocessor. Původně byl název odvozen ze slov Personal Home Page, ale to bylo později změněno z důvodu nedostatečného vystihnutí jazyka. Velká část jeho syntaxe pochází z jazyků C, Perl a Java. Jedná se tedy o skriptovací jazyk, tzn. že pro spouštění skriptů je nutný externí program (tzv.interpreter), který jej vykoná. Nejčastěji se tento jazyk používá na straně serveru k tvorbě dynamických webových aplikací. Mezi jeho výhody patří nezávislost na platformě či operačním systému a hlavně to, že je zdarma. Tento jazyk je masově rozšířený po celém světě a využívají ho desítky milionů serverů.

Vznik jazyka PHP sahá až do roku 1994, kdy jej jeho autor, Rasmus Lerdorf, použil na osobních stránkách jako určitou náhradu jazyka Perl. V roce 1995 uvolnil pro veřejnost verzi 2, aby urychlil vývoj a hledání chyb.

Zajímavostí v této verzi je implementace cyklu while. Ručně napsaná lexikální čtečka procházela skript a pokud narazila na slovo while, zapsala si jeho pozici v souboru do paměti. Na konci cyklu vyhledala uloženou pozici a celý cyklus byl

znovu vykonán. O dva roky později došlo k uvolnění verze 3, ve které bylo přepsáno celé jádro programu. V roce 2000 vyšla verze 4. V poslední době se však začaly množit požadavky na objektově orientovaný přístup. S verzí 5 přišlo rozšíření o mnoho funkcí, umožňujících objektově orientované programování. Nyní je ve vývoji verze 6, která by měla už příští rok snad oficiálně vyjít.

Největšími přednostmi jazyka PHP je jeho snadnost práce s databázovými systémy (např. MySQL, PostgreSQL, Oracle) a dobrá podpora protokolů (např. HTML, POP3), což mu umožňuje snadné vytváření aplikací.

Základním určením PHP je skriptování na straně serveru, které vytvoří dynamické webové stránky. Na podobném principu pracují jazyky jako ASP.NET či JavaServer Pages. Nejprve byl jazyk PHP určen pouze pro jednoduché skripty, ale s příchodem PHP 5 a rozvojem objektového programování se PHP spolu s databázovým systémem stará o chod celých webů.

MySQL

MySQL je výkonný databázový relační systém s architekturou klient/server. Je dostatečně stabilní, bezpečný a má výborný poměr cena/výkon (pro jeho šíření zdarma a také pro jeho nenáročnost na HW). Jádro databázového systému je napsáno v jazyce C a C++.

V roce 1995 byla vydaná první verze, o tři roky později byla vydaná verze pro MS Windows. S každou vydanou verzí se přidávaly funkce, které zlepšovaly tento databázový systém. Např. ve verzi 3.23 přibyla funkce replikace, ve verzi 4.1 poddotazy, ve verzi 5.0 pohledy atd.. Od roku 2005 je k dispozici verze 5, nyní konkrétně 5.1, která umožňuje např. plánování úloh, zapisování historie operací do tabulek apod..

Mezi jeho výhody patří nezávislost na platformě a OS (aplikace klienta i serveru mohou běžet např. pod MS Windows, Apple, Linux), snadnost instalace, ODBC rozhraní, umožňující v prostředí MS Windows spolupráci se spoustou programovacích jazyků. Pro programování klientů se může použít široká škála jazyků (např. C, C++, java, Perl, php). Další výhodou je kompatibilita se SQL (

jazyk, který se používá na dotazy a aktualizaci dat), trigger (příkazy SQL, které server spouští automaticky při určitých operacích) apod..

Naopak mezi jeho nevýhody patří mimo jiné nemožnost řešení online záloh (tabulka se při záloze musí zamknout), nemožnost definování vlastních datových typů, omezení možnosti použití pro aplikace v reálném čase atd..

MySQL používají desítky milionů serverů a velkým způsobem konkurují komerčním databázovým systémům.

phpMyAdmin

Tento program je jedním z nejoblíbenějších nástrojů na správu MySQL. Můžeme v něm vytvářet, upravovat a mazat databáze, vkládat a měnit datové záznamy, provádět export a import záznamů, vkládání a úpravu práv uživatelů atd.. Pro funkčnost tohoto programu je třeba mít nainstalován server Apache včetně jazyku PHP. U všech programů napsaných v PHP platí, že skripty se spouští na serveru, ale výsledné HTML stránky je možno zobrazit odkudkoliv. To je velká výhoda tohoto programu. Můžeme jej totiž použít nejen na správu databáze na lokálním počítači, ale i na vzdáleném serveru. Mezi jeho nevýhody patří jeho menší uživatelská přívětivost a složitost v případě nějaké systémové změny programu.

DBDesigner

DBDesigner je vizuální program pro návrh, tvorbu a správu MySQL databází. Umožňuje návrh databáze pomocí grafického rozhraní, nebo zpětné vygenerování tabulek a jejich relací. Při návrhu může generovat příkazy SQL a tím online navrhovat tabulku a rovnou informace importovat do databáze. Je Open Source pod licencí GPL, tudíž je zcela zdarma.

Server: localhost ▶ Databáze: udrzba_stroju ▶ Tabulka: nahradni_dily "InnoDB free: 5120 kB"

Projít Struktura SQL Vyhledávání Vložit Export Import Úpravy Vyprázdnit Odstranit

i Zobrazeny záznamy 0 - 5 (~6¹ celkem, Dotaz zabral 0.0006 sekund)

SQL-dotaz:

```
SELECT *
FROM 'nahradni_dily'
LIMIT 0 , 30
```

☐ Profilování [Upravit] [Vysvětlit dotaz] [Vytvořit PHP kód] [Obnovit]

i ¹Může být nepřesné, viz FAQ 3.11

Zobrazit: 30 řádků začínající od 0

ve vodorovném režimu a opakovat hlavičky po 100 řádcích.

Setřídít podle klíče: Žádná

↕		id_nahradni_dily	reference	popis	reference_sap	poznamka	vytvoreno	id_skupiny_nahradni_dily	stav
☐		☐	1	US00001	CU040A25N	NULL	2009-05-27 23:02:57	13	1
☐		☐	2	US00002	HP1352A10AN	NULL	2009-05-27 23:03:55	3	2
☐		☐	3	US00003	CU040M25N	NULL	2009-05-27 23:04:09	3	2
☐		☐	4	US00004	KD-H-12896	NULL	2009-05-27 23:04:19	13	2
☐		☐	5	US00005	CR40/A	NULL	2009-05-27 23:04:32	1	2
☐		☐	6	US00006	CU250A25N	NULL	2009-05-27 23:04:42	13	2

☐ Zaškrtnout vše / ☐ Odškrtnout vše Zaškrtnuté: ☐ ☐ ☐

Obr.2: Ukázka prostředí phpMyAdmin

3.2 NÁVRH KONCEPCE IS

Na IS se budou uživatelé moci dostat jen z lokální sítě, poběží na interním firemním serveru. Nebudou tedy kladeny tak vysoké nároky na zabezpečení dat proti průniku hackerů.

Grafická podoba

Grafickou podobu stránky jsem navrhl podobnou jako u firemního intranetu, který jsem také vyvíjel, ale s tím rozdílem, že místo vertikálního menu jsem zvolil horizontální, aby se případné velké tabulky přehledně vešly na jednu obrazovku. Barvy jsou firemní, převládá modrá a šedá. Navrchu je nadpis spolu s částí firemního loga, pod ní horizontální „vysouvací“ menu. V dolní části je patička, na které jsou copyright informace a případný kontakt na administrátora. Celá stránka je ohraničena černými okraji a má bílé pozadí. Stránka má pevně definovanou šířku, zbytek je

podle rozlišení monitoru doplněn šedou barvou. Nadpisy hlavních kategorií jsou v modrém pruhu s bílým písmem, podnadpisy v šedém pruhu taktéž s bílým písmem.

s. n. o . p. cz a.s.

Údržba strojů

Ambice Rentabilita Respekt
Průhlednost Profesionalita Důvěra

Zařízení Kontroly Denní kontroly Seznamy Náhradní Díly Malé Opravy Údržba Experti Administrace Odhlásit

Seznam zařízení

Vyhledávání zařízení:

Typ zařízení: - vyberte - Skupina: - vyberte - Úsek: - vyberte - Umístění: - vyberte - Stav: - vyberte - Filtruj

Výsledek vyhledávání:

Kód	Skupina	Typ	Umístění	Úsek	ND	Opravy	Údržby	Denní kontroly	Kontroly	Záznamy expertů	Stav
geget	AUTO LIS	lis BALCONI 160t	DOB	SP	Zobrazit(0)	Zobrazit(1)	Zobrazit(0)	Zobrazit(2)	Zobrazit(0)	Zobrazit(0)	aktivní
grvqgv	AUTO LIS	lis BALCONI 160t	DOB	SP	Zobrazit(0)	Zobrazit(0)	Zobrazit(0)	Zobrazit(2)	Zobrazit(0)	Zobrazit(0)	údržba
gege	AUTO LIS	lis BALCONI 160t	DOB	AL	Zobrazit(0)	Zobrazit(1)	Zobrazit(0)	Zobrazit(4)	Zobrazit(0)	Zobrazit(0)	aktivní
testtt	AUTO LIS	lis BALCONI 160t	DOB	AL	Zobrazit(0)	Zobrazit(0)	Zobrazit(0)	Zobrazit(4)	Zobrazit(0)	Zobrazit(0)	aktivní

Vložit nové zařízení

s.n.o.p. cz a.s. © 2009 Všechna práva vyhrazena

Obr.3: Ukázka grafické podoby IS

Session

Přihlašování do systému bude vyřešeno pomocí samostatné tabulky s uživateli a session. To by mělo fungovat takto: na každé stránce IS bude funkce pro žádost o kontrolu session. Pokud php dostane žádost o kteroukoliv stránku, podívá se nejdříve, jestli už náhodou session neběží. Pokud ano, připojí se k ní, pokud ne, tak vytvoří novou. Session si vytvoří tzv. „session – identifikátor“, který se bude skládat z uživatelského jména zaměstnance, jeho oddělení, podniku odkud se přihlašuje a času platnosti. Pokud bude uživatel více jak 30 minut v systému nečinný, platnost session zanikne a php uživatele přesměruje na přihlašovací stránku. Zde zadá své přihlašovací údaje a systém mu vytvoří nové session – id. Platnost zanikne také po odhlášení se ze systému, popř. po zavření okna prohlížeče. Session identifikátor se

předává buďto v URL, kdy za adresu přiřadí id, nebo častěji pomocí cookie. Předávání id v URL bývá často zakázáno, takže budeme používat cookie. V lokální síti je cookie standardně všude povoleno, takže se tím nemusíme více zabývat. Možnost přidávání uživatelů do databáze bude mít jen administrátor.

3.2.1 Rozdělení kategorií podle obsahu

IS musí být obsahově rozlišen podle informací, které se v daném modulu budou nacházet. Související moduly by měly být provázány.

Kategorie zařízení

První nadřazenou kategorií je zařízení. Zde by měl být celkový přehled zařízení. V rámci, kde se nám ukazují informace, se nám zobrazí formulář, pomocí něhož vybereme zařízení, o které máme zájem. Zde bude kompletní přehled a odkazy na moduly, které se k němu váží (zobrazení denních kontrol, malých oprav, náhradních dílů apod.). Po kliknutí na detail zařízení zde budou kompletní informace jako např. název, inventární číslo, umístění, obrázek atp. V horní části dále mimo výběru budeme mít možnost vložit nové zařízení (pokud na to má daný uživatel právo). Tento odkaz nás přesměruje na formulář, kde budou kolonky pro vyplnění údajů.

Kategorie kontroly prevence

Další kategorií budou kontroly, prevence. Zde se zobrazí, co všechno se má u daného zařízení preventivně kontrolovat, s jakou periodicitou a jaký to zabere čas. Také zde bude možnost vložit nový záznam, ale s tím, že záznam je vázán na určitou skupinu zařízení, nejenom na jedno konkrétní.

Kategorie denní kontroly

Dále je zde seznam všech denních kontrol, který zobrazuje které kontroly se musí u zařízení provést každý den. Budou se rozlišovat podle toho, ke kterému úseku patří.

Kategorie seznamy

Další hlavní sekci jsou seznamy. Bude zde odkaz na přehled dodavatelů a výrobců, které bude možno filtrovat podle různých kritérií. Opět je zde možnost vložení nového dodavatele nebo výrobce.

Kategorie náhradní díly

Poté jsou zde náhradní díly, ve kterých bude přehled všech dostupných dílů v daném podniku. Ten se bude moci odfiltrovat ještě podle typu náhradních dílů, dodavatele a výrobce nebo podle náhradních dílů patřících k určitému zařízení.

Kategorie malé opravy

Ty jsou vázány na určitý typ zařízení, podle toho se také bude provádět jejich výpis. Bude zde přehled, které opravy jsou již dokončené, a které nikoliv. Samozřejmě je vložení nové opravy.

Kategorie údržba, větší opravy

Dále je zde plán údržby a větších oprav. Zde bude přehled všech údržeb, které se provedly. Bude zde možno prohlížet fotodokumentaci. Možnost filtrování bude podle mnoha kritérií, např. zda byla již oprava započata nebo dokončena, typ zařízení apod.

Kategorie záznamy expertů

Budou zde evidovány záznamy expertů, které bude možno filtrovat. Jsou to záznamy k určitému zařízení, u něhož prováděli nějaký zásah. Samozřejmě je zde zase možnost přidání dalšího záznamu, vázaného k určitému zařízení atp..

Kategorie administrace

Poslední hlavní kategorií je administrace. Bude zde seznam uživatelů s možností jejich přidání a přidělení rolí, také seznam všech číselníků s možností odstranění některého záznamu, popř. doplnění číselníků o další položky.

Uživatelé IS

Do systému budou vkládáni uživatelé pouze administrátorem z důvodu bezpečnosti a zjednodušení. V IS budou existovat uživatelské účty, které budou rozděleny na uživatele, kteří mohou jen zadávat údržby, uživatele, kteří mohou zadávat opravy apod. Ochrana proti neoprávněnému přístupu bude zajištěna pomocí Session (viz. 3.2.2) a práv na jednotlivé moduly.

Vstupní formuláře

Formuláře budou zpracovávány pomocí metody POST. Jejich součástí budou jak pouhá textová pole, tak i pole zaškrťovacích políček či rolovací seznamy. Ten bude nečastěji generovaný dynamicky podle dat v databázi tak, aby byl seznam vždy aktuální.

Všechny použité formuláře samozřejmě budou ošetřeny proti zadávání zakázaných vstupních znaků, odstraňování uvozovek atp. Mohlo by zde totiž dojít k tzv. SQL Injection, což je schopnost uživatele změnit SQL dotaz pomocí zadaných dat. Také se musí ošetřit, aby někdo nekládal jedno zařízení popř. záznam o náhradním dílu či opravě dvakrát. Toto bude ošetřeno pomocí porovnávání jedinečných dat u právě vkládaného záznamu např. u zařízení je to kód zařízení atp.

Vkládání dat ze SAPu

V kategorii náhradní díly jsou údaje, které jsou často proměnné. Jedná se především o stavy skladu. Tyto údaje jsou aktuální v SAPu, z něhož se každý den generuje textový soubor, který obsahuje reference náhradních dílů a skutečné počty dílů na skladě. Informační systém tento soubor každý den projde, porovná reference a zaktualizuje počty kusů skladových položek.

3.2.2 SQL dotazy

V IS se bude často používat spojení více tabulek. Dotaz SELECT proto skoro vždy bude i několikanásobně rozšířen o LEFT JOIN, který pro každý záznam z první tabulky generuje záznam ve výsledku. Často bude příkaz SELECT rozšířený o podmínky WHERE, AND a OR popř. LIKE, pomocí kterých zabezpečíme odfiltrování záznamů, o které nemáme zájem. Další častý příkaz bude INSERT pro vkládání dat do tabulek, UPDATE pro změnu určitých dat ve sloupcích (např. u náhradních dílů při importování počtu skladových položek) a popř. DELETE pro smazání záznamu. Také bude použit příkaz ORDER BY, který seřadí výstupní data a LIMIT, který nám omezí výpis záznamů. Pro komunikaci s databází bude použito objektově orientované prostředí mysqli. Všechny funkce, které budou použity pro práci s databází, budou uloženy do jednoho souboru třídy mysqli.

4. E-R DIAGRAM A ANALÝZA DATABÁZOVÉHO SYSTÉMU

Pomocí E-R diagramu můžeme snadno analyzovat navržený databázový systém. Obsahuje všechny tabulky uložené v databázi a jejich vzájemné relační vazby.

4.1 ZÁKLADNÍ PRAVIDLA PRO NÁVRH DATABÁZE

Návrh databáze je prvním krokem tvorby programu. Má velký vliv na efektivitu programu, obtížnost programování a údržbu.

4.1.1 Typy tabulek

Zvláštností databáze MySQL je to, že pokud vytváříme novou tabulku, můžeme specifikovat jakého má být typu. V MySQL jsou tři nejdůležitější typy – MyISAM, InnoDB a HEAP.

MyISAM je základním typem tabulek. Tabulky jsou stabilní a jednoduché na správu. Rozlišujeme dvě varianty těchto tabulek: MyISAM statický, který se používá v případě, že všechny sloupce tabulky mají pevně stanovenou velikost (přístup k této tabulce je velmi rychlý, ale nevýhodou je rezervace paměťového místa pro záznamy) a MyISAM dynamický, který se používá pokud je v tabulce některý z datových typů VARCHAR, TEXT či BLOB a ty nemají předem definovanou velikost (přístup je pomalejší než u statické varianty, ale má menší nároky na paměť). O výběr se starat nemusíme, to provede MySQL automaticky.

InnoDB je modernější alternativa MyISAM. Přináší nové funkce. Např. transakce, které umožňují několik logicky souvisejících příkazů, které se provedou všechny nebo žádný a pokud při provádění některého příkazu nastane chyba, budou výsledky všech příkazů odstraněny. Další novinkou je zamykání na úrovni záznamů, které znamená to, že při provádění transakce nemusíme blokovat přístup k tabulce dalším uživatelům, ale uzamkneme jen jednotlivé záznamy, triggerů apod. Mezi její

nevýhody patří téměř dvojnásobné požadavky na paměť oproti tabulce MyISAM, složitější a pomalejší zjištění počtu záznamů v databázi pomocí příkazu COUNT ap.. **HEAP** je speciální typ tabulek, který je ukládá do RAM, nikoli na disk. Jsou značně omezené, např. nesmíme používat datový typ TEXT nebo BLOB. Jsou určeny pro rychlé zpracování malého množství dat.

4.1.2 Normalizační pravidla

Normální formy pro relační databáze formuloval vědec E.F.Codd. Výzkum v oblasti formálního popisu matematických množin, zejména relačních databází, ale dále pokračuje. Normální formy jsou pravidla, která pomáhají při návrhu databází, ale neumí nahradit lidské myšlení. Normalizovaná databáze nabízí více možností různých dotazů a odstranění redundance. Naproti tomu není vhodné všude redundanci odebrat, s počtem tabulek totiž roste čas načítání dat.

První normální forma

První normální forma říká tři základní věci. Nejdříve musíme eliminovat sloupce se stejným obsahem a potom vytvoříme tabulku pro každou skupinu dat. A poslední věc – každý záznam musí být jednoznačně identifikovatelný primárním klíčem.

Druhá normální forma

Druhá normální forma říká to, že kdykoliv se opakují obsahy některých sloupců v různých záznamech, tabulka by se měla rozdělit na několik tabulek a ty by se měly propojit cizími klíči.

Třetí normální forma

Třetí normální forma má jen jedno pravidlo a říká, že sloupce, které nejsou přímo závislé na primárním klíči musíme odstranit (přesunout do samostatné tabulky).

4.1.3 Vztahy

Jestliže chceme normalizovat databázi pomocí normálních forem, musíme propojit několik tabulek. Tato propojení se nazývají vztahy. Mezi dvěma tabulkami existují tři vztahy, 1:1, 1:n a n:m.

Vztahy 1:1

U tohoto vztahu ke každému záznamu z první tabulky existuje záznam v druhé tabulce a naopak. Tyto druhy informace jsou velmi vzácné, protože by se informace vešly do jedné tabulky.

Důvody k rozdělení mohou být dva. Buď máme nějaké informace, které jsou důvěrné a nechceme, aby tam měl každý přístup, a nebo pokud má tabulka velmi mnoho sloupců a my využíváme jen některé z nich, je efektivnější vložit používané sloupce do jedné tabulky a ostatní do druhé. Tím se zvýší i rychlost práce s databází.

Vztahy 1:n

U těchto vztahů ke každému záznamu z první tabulky může existovat několik záznamů z druhé tabulky. Vztahy jsou realizovány přes hlavní sloupce, tzv. klíče. V hlavní tabulce je to primární klíč, v další tabulce cizí klíč, který odkazuje do hlavní tabulky. Opačným směrem toto neplatí, možná je někde zmínka o vztahu n:1, ale to je ve skutečnosti vztah 1:n z opačného pohledu.

Vztahy n:m

U těchto vztahů ke každému záznamu z první tabulky může existovat několik záznamů z druhé tabulky a naopak. Tyto vztahy je nutno pomocí vazební tabulky rozdělit na dva vztahy 1:n.

4.2 ANALÝZA DATABÁZOVÉHO SYSTÉMU

K navrženému databázovému systému byl pomocí programu DBDesigner vytvořen E-R diagram, který je umístěn v příloze (viz Příloha1). V databázi jsou použity tabulky typu InnoDB, znaková sada je nastavena na cp1250_czech_cs.

4.2.1 Použité datové typy

Pro primární klíče je použit datový typ INT, maximální počet číslic je deset (maximální rozsah typu INT). Všechny primární klíče používají speciální atribut AUTO_INCREMENT, který každému nově vkládanému záznamu přiřadí hodnotu o jednu větší, než v předchozím záznamu. Pro textové řetězce je použit datový typ VARCHAR. Délka textového řetězce je určena podle potřeby a přibližné velikosti vkládaného textu od 16 do 256 znaků. V MySQL 5.0 byl typ VARCHAR výrazně inovován, hlavní změna je v maximální šířce daného sloupce, které se změnila z 255 na 65535 znaků. Pro datum je použit typ DATETIME, což je kombinace data a času ve formátu rok-měsíc-den hodina:minuta:sekunda.

4.2.2 Popis a vazby mezi tabulkami

Názvy tabulek i sloupců byly voleny tak, aby bylo jednoznačně zřejmé, co je obsahem daného sloupce nebo tabulky. Primární klíče byly voleny s názvem id_název tabulky. Základní tabulkou databázového systému je tabulka „zarizeni“. Tato tabulka obsahuje všechny informace, které jsou potřebné k přesné identifikaci zařízení. Pomocí cizího klíče jsou na tuto tabulku navázány tabulky „typy_zarizeni“, skupiny_zarizeni a umisteni_zarizeni“, ve kterých jsou data, která se mohou pro různá zařízení opakovat. Také je zde připojena tabulka podniky, která danému zařízení udává, pod jaký podnik patří. Další tabulka spojená pomocí cizího klíče na tabulku „zarizeni“ je tabulka „useky_umisteni_zarizeni“, ke které se zase váží denní kontroly. Zde je popis denních kontrol, které se mají na zařízení provádět. Tato tabulka je také spojena pomocí cizího klíče s tabulkou podniky, protože kontroly mohou být v různých závodech odlišné. Tzv. spojovací tabulku tvoří „vazba_zarizeni_nahradni_dily“ která určuje, ke kterému zařízení a s jakým počtem

kusů náleží příslušný typ náhradního dílu. Tabulka „nahradni_dily“ obsahuje základní popisy a reference dílů. K ní je vázána tabulka „skupiny_nahradni_dily“, kde jsou záznamy, pod kterou skupinu daný díl patří a tabulka „sklady_nahradni_dily“, kde jsou informace o počtu kusů na skladě (ty se pak importují ze SAPu), dodavateli a výrobci apod.. K této tabulce se váže „umisteni_nahradni_dily“, kdy mohou být různé díly na stejném místě, „podniky“, kde může být jeden díl se stejnou referencí ve více závodech a „dodavatele_vyrobcu“, kde může být pro jeden díl různý dodavatel a výrobce. Na tabulku „dodavatele_vyrobcu“ jsou pak navázány tabulky „mesta_dodavatele_vyrobcu“, kde jsou všechna města v České republice včetně poštovních směrovacích čísel a „staty_dodavatele_vyrobcu“, kde jsou státy, ve kterých daná firma sídlí. Na tabulku „podniky“, kde jsou umístěny názvy jednotlivých závodů, se také váže tabulka „uzivatele“. Tady nalezneme seznam všech lidí, kteří systém používají. Na tuto tabulku je navázána sada systémových tabulek, které určují moduly a práva na ně. V tabulce „uzivatele_moduly“ je seznam všech modulů, dostupných v daném IS. Na ní je pak navázána tabulka „uzivatele_moduly_skupiny“, která určuje, do které skupiny daný modul patří. Dále zde máme „uzivatele_role“, což je tabulka obsahující seznam všech rolí, které jsou v IS k dispozici. Tabulka „uzivatele_role_moduly“ je svázána s tabulkami modulů a rolí a určuje, na jaký modul má uživatel v dané roli právo. Poslední tabulkou je zde „uzivatele_prava“. Umožňuje uživatelům přidělit jednu nebo i více rolí současně. S tabulkou „zarizeni“ je dále svázána tabulka „vazba_zarizeni_kontroly“, s tou pak „casy_kontrol“, kde se určuje, jak dlouho se daná kontrola provádí a „kontroly“, kde jsou specifikace a návody na jednotlivé kontroly prevence. S kontrolami souvisí „periodicity_kontrol“, kde je seznam všech dostupných periodicit. V „casti_zarizeni“ se určuje, na které části se kontrola provádí. Na „zarizeni“ se také váží „velke_opravy_udrzba“, kde evidujeme pravidelné kontroly stroje popř. velké opravy. Zde můžeme získat informace o datech údržby, kdo jí prováděl a její popis. Na ní jsou pak navázány tabulky „typy_udrzby“ a „obdobi_velke_opravy_udrzba“, které jsou společné pro více záznamů. Další tabulkou, vázanou na zařízení je „zaznamy_expertu“. Jsou zde uvedeny informace ohledně zásahů expertů na různých zařízeních, je zde kdo zásahy provedl, co bylo za

závadu, jak jí opravil apod.. Na tu je vázána tabulka „typy_oprav“, která určuje jaký typ opravy experti provedli. Tato tabulka je společná i pro „male_oprav“, což je tabulka, která se váže k danému typu zařízení. Zde je popis závady, jak byla odstraněna, kdo a kdy opravu provedl. Poslední dvě tabulky jsou samostatně stojící. První z nich je systémová tabulka „stavy“, kde je popsáno, v které entitě co který stav znamená, jakých stavů může entita nabývat a zda jsou to počáteční nebo koncové stavy. Poslední tabulkou je tabulka „foto“, kam jsou ukládány informace o uložených fotkách na server. Jsou zde např. informace o názvu souboru, cestě, entitě ke které je soubor přidělen a čísla řádku, ke které se daná fotka váže. Více informací je v E-R diagramu.

4.3 ŽIVOTNÍ CYKLY

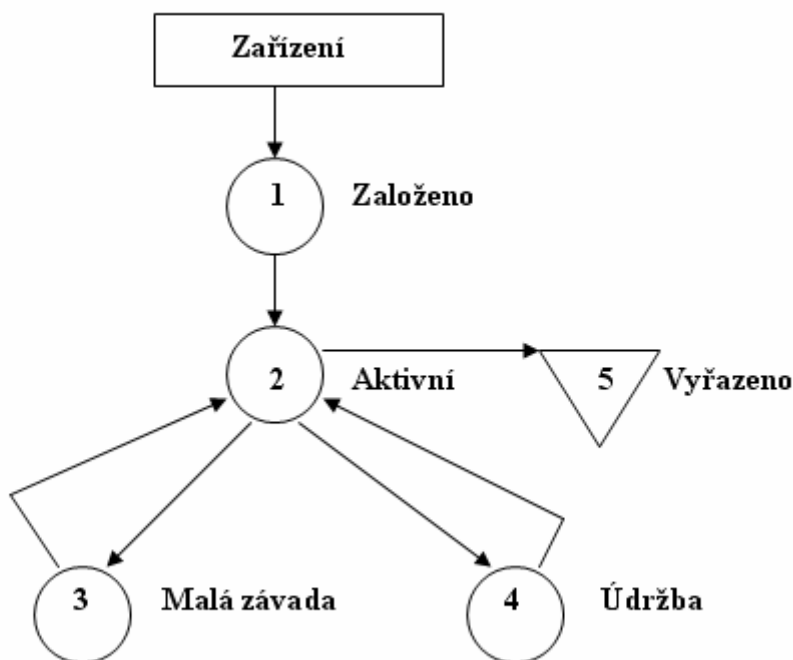
Životní cyklus entity ukazuje, jak se mění informace v entitě podle stavů, daných právě navrženým životním cyklem, jak se mění její atributy, vztahy apod.

4.3.1 Životní cykly vybraných entit

Popisují se zde hlavní vybrané entity. Ostatní entity mají životní cyklus buď úplně jednoduchý, nebo můžou nabývat jen stavu platný a neplatný.

Životní cyklus entity zařízení

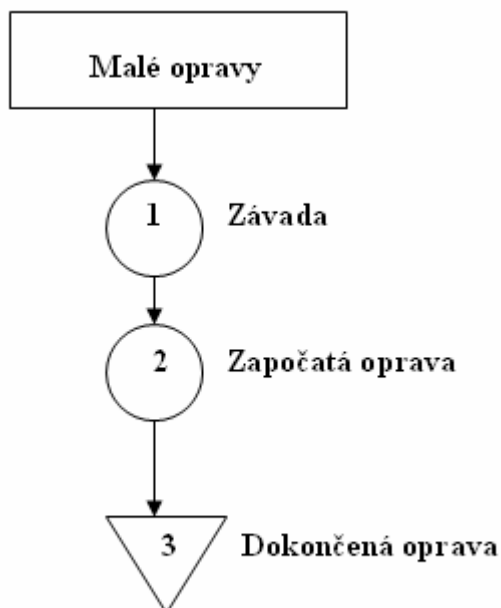
Záznamy v entitě „zarizení“ mohou nabývat pěti různých stavů. Pokud se vloží nové zařízení, tak je ve stavu 1 - Založeno a čeká se na schválení vedoucím oddělení. Ze stavu 1 do stavu 2 ho tedy může převést jen vedoucí. Pokud je ve stavu 2 - Aktivní, může přejít do stavu 3 - Malá závada, což je stav, kdy se na zařízení vyskytla nějaká závada a stroj je odstaven. Do tohoto stavu a zpátky do aktivního ho může dostat jen opravář. Dále se z aktivního stavu může dostat do stavu 4 - Údržba, což značí, že zařízení je odstaveno kvůli údržbě. Do toho stavu a zpátky do aktivního ho může posunout pouze údržbář. Poslední koncový stav je stav 5 - Vyřazeno kam ho může posunout jen vedoucí a to v případě, kdy se stroj z nějakého důvodu vyřadí, což je konečný stav.



Obr.4: Životní cyklus tabulky zařízení

Životní cyklus entity malé opravy

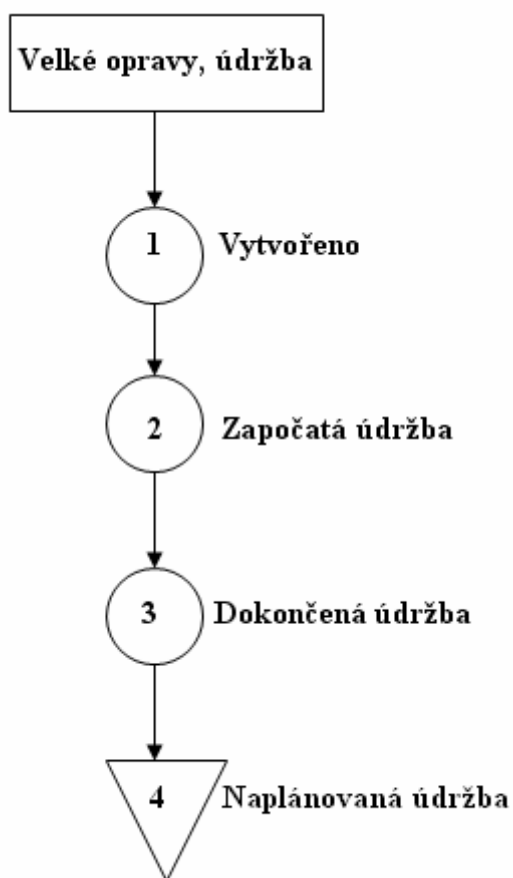
Záznamy v tabulce „male_opravy“ nabývají tří stavů. Po vložení záznamu do tabulky má záznam stav 1 - Závada. V tomto stavu si ji převezme opravář a převede ji do stavu 2 - Započatá oprava. Až závadu odstraní, převede záznam do konečného stavu 3 - Dokončená oprava.



Obr.5: Životní cyklus tabulky malé opravy

Životní cyklus entity velké opravy/údržba

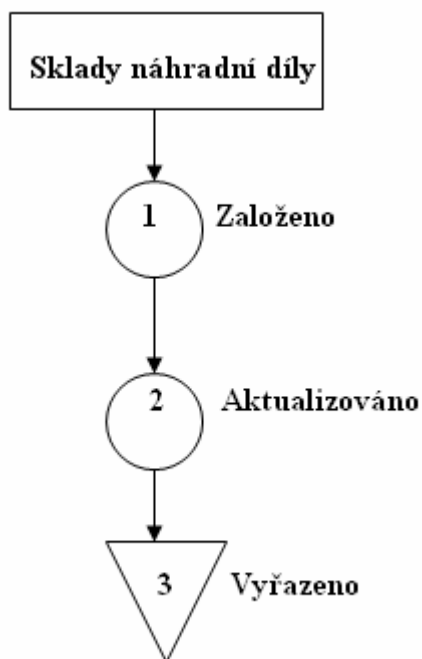
Po vložení záznamu do tabulky „velke_opravy_udrzba“ má záznam stav 1 - Vytvořeno. Do dalších stavů ho může převést údržbář. Další stav je 2 - Započatá údržba, kdy údržbář zaznamená, kdy opravu začal, jak probíhala. Pokud ji chce převést do stavu 3 - Dokončená údržba tak musí být stanoveny typy údržby, datumy ukončení a ostatní informace. Konečný stav je 4 – Naplánovaná údržba.



Obr.6: Životní cyklus tabulky velké opravy, údržba

Životní cyklus entity sklady náhradní díly

Po vložení záznamu do tabulky „sklady_nahradni_dily“ má záznam stav 1 - Založeno. To znamená, že náhradní díl, byl vložen do tabulky a čeká se na aktualizaci jeho skladových položek pomocí importu ze SAPu. V tomto stavu není možné s náhradním dílem jakkoli manipulovat. Pokud se provede update skladových položek se SAPu, funkce převede záznam do stavu 2 aktualizováno. V tomto stavu je možné s dílem manipulovat. Konečný stav - smazání dílů - z důvodu např. nepoužívání náhradního dílu je stav 3 - Vyřazeno.



Obr.7: Životní cyklus tabulky sklady náhradních dílů

5. POPIS IMPLEMENTOVANÉHO SYSTÉMU

Uživatel se na IS dostane pomocí svého internetového prohlížeče. Systém je odladěn pod OS Windows XP SP3 a prohlížečem Mozilla Firefox a Internet Explorer 7.

5.1 POPIS SYSTÉMU Z POHLEDU UŽIVATELE IS

Do systému se musí nejprve uživatel přihlásit pomocí svého loginu a hesla. Uživatelský účet mu vytvoří administrátor v sekci Vložení nového uživatele a přidělí mu patřičná práva v sekci Vložení nové role k uživateli. Po přihlášení se uživatel dostane na hlavní stránku, kde se mu zobrazí pod kým je přihlášen a kdy bylo naposled zaznamenáno jeho přihlášení. Uživatel si může vybrat z vertikálního menu do jakého modulu se chce dostat. První úroveň menu je statická z důvodu zjednodušení. Ve druhé úrovni se vypíší odkazy na moduly, na které má daný uživatel právo.

5.1.1 Popis jednotlivých modulů

V každém modulu, který slouží k náhledu dat, musíme nejprve vyplnit filtrační formulář, abychom se dostali k požadovaným informacím. Pokud je v modulu k náhledu více než třicet položek, zobrazí se uživateli stránkování. U oprav, záznamů expertů a údržeb se zaznamenává kdo a kdy závadu vložil a kdo ji pak následně opravil.

Zařízení

První položka v menu je Zařízení. Zde nalezneme volbu Přehled zařízení, po jejíž rozkliknutí se nám zobrazí vstupní filtr pro nalezení zařízení. Filtrovat můžeme podle různých položek jako např. podle Typu zařízení, skupiny, úseku a nebo podle stavu, ve kterém se dané zařízení nachází. Po úspěšném vyfiltrování se nám zobrazí tabulka s přehledem odfiltrovaných zařízení, které patří pod podnik ve kterém jsme přihlášení a nejsou ve stavu vyřazeno. V tabulce nalezneme základní údaje

k jednoznačné identifikaci zařízení a odkazy, přes které se dostaneme na související moduly. V prvním sloupečku po rozkliknutí kódu se dostaneme na detail zařízení, kde můžeme editovat jednotlivá pole (pokud tedy není zařízení ve stavu oprava nebo údržba), popř. vkládat k zařízením fotky a prohlížet si je. Když se vrátíme na předchozí tabulku, vidíme zde na co vše se můžeme u daného zařízení podívat. Pokud na daný související modul nemáme oprávnění, zobrazí se nám jen text bez možnosti přejít na další modul. U každého souvisejícího modulu máme v závorce počet záznamů, který souvisí s daným zařízením a na který se dostaneme kliknutím na odkaz. Další volba je Vložení nového zařízení. Zde se dostaneme na formulář, kde po vyplnění všech povinných položek vložíme nové zařízení. Poté ho však musí vedoucí převést do stavu aktivní, aby bylo možné se zařízením pracovat, přidávat k němu opravy, údržby atd.. Další volba je Přehled náhradních dílů k zařízení, kde se dostaneme na stránku s filtrem. Můžeme zde najít, které díly jsou přiřazeny k jakému zařízení a s jakým počtem kusů. Poslední volba je Přehled kontroly zařízení, kde si vyfiltrujeme jaké kontroly se váží k danému zařízení.

Kontroly

Druhá položka v menu je Kontroly, což jsou kontroly prevence u zařízení. První volba je Přehled kontroly prevence, kde se dostaneme nejdříve na vstupní filtr. Po vyfiltrování se nám zobrazí tabulka s přehledem kontrol, jaký čas je potřeba k dané kontrole a s jakou periodicitou se má kontrola provádět. Zde se můžeme po kliknutí na první položku id dostat na detail dané kontroly, kde jí popř. můžeme i editovat. Druhá volba je Vložení kontroly prevence, kde se dostaneme k formuláři, který nám umožní přidat kontrolu prevence do systému. Poslední volba je Vložení nové kontroly zařízení, kde si můžeme přidávat k zařízením různé kontroly, které jsme již do systému vložili.

Denní kontroly

Další položka v menu jsou Denní kontroly, kde je přehled denních kontrol, které se na zařízeních musí provádět. Denní kontroly jsou zatím podle požadavků firmy jen jakýsi přehled, časem by se ale tento modul měl změnit v systém

postupného zadávání s přehledem, co vše již bylo zkontrolováno a co ještě zbývá zkontrolovat. První volba je Přehled denních kontrol. Zde se dostaneme na stránku, kde je možno vyfiltrovat si podle zařízení nebo úseku které denní kontroly se mají kde provést. Denní kontrola se váže na úsek, kde je umístěna, takže položka zařízení je tam jen pro lepší orientaci, pokud se uživatel bude chtít podívat přesně na zařízení a ne na úsek. Po vyfiltrování dostaneme tabulku s popisem kontroly s informací, na kterém úseku se má kontrola provádět. V hlavičce tabulky je přehled všech zařízení, která s danou denní kontrolou souvisí.

Seznamy

Ve čtvrté položce menu, označené jako Seznamy je seznam dodavatelů a výrobců. První položka s názvem Přehled dodavatelů/výrobců nás dostane na stránku, kde po příslušném vyfiltrování máme přehled se základními údaji o výrobcí/dodavateli. Uvidíme zde jen ty dodavatele/výrobce, kteří se váží k podniku, pod kterým pracujeme. Po kliknutí na první položku Název firmy se dostaneme do detailu výrobce/dodavatele, kde můžeme editovat jeho údaje. Druhá položka je nazvána Vložení nového dodavatele/výrobce a přesměruje nás na formulář, kde můžeme vložit novou firmu.

Náhradní díly

Další položka v menu jsou náhradní díly. Zde je seznam náhradních dílů, které jsou v daném podniku a používají se nějakým způsobem u daných zařízení. Pod volbou Přehled náhradních dílů najdeme opět vstupní formulář, kde se nám po odfiltrování zobrazí tabulka se základními údaji o náhradním dílu. Pokud je položka náhradního dílu nově vložena a ještě není aktualizována, tak se nám u ní zobrazí jen základní informace, skladové položky totiž zatím nebyly vyplněny. Po kliknutí na první položku se dostaneme k detailu náhradního dílu, kde můžeme editovat povolené položky (jako například měnit umístění dílu atp.). Pokud aktualizace ještě neproběhla, nemůžeme editovat nic a zobrazí se nám hlášení o čekání na update ze SAPu. Další položka se jmenuje Vložit nový ND k zařízení. Zde se dostaneme na formulář, kde můžeme k danému zařízení přidělit náhradní díl s určitým počtem

kusů. Poslední položka je Vložení nového ND, kde vyplníme referenci dílu v SAPU, skupinu do které díl patří popř. její popis. Další povinnou položkou je reference, která se nám však vygeneruje automaticky podle aktuálního počtu položek v tabulce a podle podnikového prefixu. Po vložení náhradního dílu je díl ve stavu Založeno a není možné ho nějak editovat. Teď musí uživatel IS vzít vygenerovanou referenci ND a s touto referencí vložit díl do SAPu. Každý den v noci se pak provede update skladových položek ze SAPu a nový vložený díl přejde do stavu Aktualizováno.

Malé opravy

Šestou položkou v menu jsou Malé opravy. Pod touto volbou najdeme malé závady a opravy, které jsou systémem evidovány. První položkou je zde Přehled malých oprav. Po kliknutí se dostaneme na stránku, kde se nám po odfiltrování zobrazí tabulkový přehled všech evidovaných závad a oprav, kdy vznikly a v jakém jsou stavu. Po kliknutí na první položku id se dostaneme na detail opravy, kde jí podle jejího stavu můžeme editovat. Pokud opravu dokončíme, nelze editovat žádnou položku, můžeme se jen dívat na detaily popř. si prohlížet nebo ještě vložit nějakou fotografii dokumentující průběh opravy. Poslední volbou je Vložení malé opravy, kde se nám zobrazí formulář a zadáme o jakou závadu se jedná a odešleme ji do systému. Pak si ji převezme některý z opravářů a dokončí ji.

Údržba

Další položkou v menu je Údržba. Zde nalezneme seznam údržeb, provedených na zařízení, popř. velké opravy, které se na zařízení musely udělat. První položka je Přehled údržby, kde se zase dostáváme ke vstupnímu filtru. Po odfiltrování záznamů se nám zobrazí tabulkový přehled základních údajů údržeb včetně jejich aktuálního stavu. Po kliknutí na první položku Číslo plánu se dostaneme na detail údržby, a podle jejího stavu na něm můžeme editovat některé položky. Pokud přejde zařízení do stavu naplánovaná další údržba, nelze se záznamem již nic provést. I zde máme možnost vložit k záznamu fotografie, které můžeme posléze prohlížet. Poslední položkou je Vložení nové opravy/údržby, kde se

nám zobrazí formulář a kde zadáme potřebné informace k tomu, abychom údržbu mohli uskutečnit.

Experti

Další položka v menu je Experti, kde jsou záznamy oddělení expertů automatizace. Uživatel vloží do systému nějakou závadu, která se nějak dotýká automatizace, expert si jí převezme a vyřeší. První odkaz je Přehled záznamů expertů. Zde vyplníme vstupní filtr a dostaneme se k aktuálnímu přehledu závad a oprav, které se týká oddělení expertů. Nalezneme zde základní identifikaci problému, po kliknutí na první položku id se dostaneme do detailu, kde můžeme podle toho ve kterém je záznam stavu editovat položky, přidávat popis k opravě atp.. Samozřejmě i zde je možno vkládat a prohlížet fotky. Poslední volba je Nový záznam expertů, pomocí které se dostaneme na formulář, kde Vložíme novou závadu, kterou si pak některý z expertů převezme.

Administrace

Předposlední položkou v menu je administrace. Zde nalezneme volby k vkládání nových uživatelů a editace jejich rolí, úpravu a vyřazování náhradních dílů a zařízení a samozřejmě také údržbu číselníků. První odkaz Přehled uživatelů nás zavede na stránku, kde se nám po odfiltrování zobrazí aktuální existující uživatelé. V tabulce máme základní informace o uživateli. Pokud chceme detail, klikneme na první položku login. Ta nás přesměruje na formulář, kde máme detailní informace o uživateli a můžeme měnit některé jeho údaje popř. mu nastavit nové heslo. Pokud se vrátíme zpět do přehledu uživatelů, máme zde pod sloupečkem Role odkaz na modul, který nám zobrazí, které role má daný uživatel přiděleny. Pod sloupečkem Akce nalezneme odkaz Smazat, pomocí kterého vymažeme uživatele ze systému. Další odkaz je Vložení nového uživatele, kde se dostaneme k formuláři, kterým vložíme nového uživatele do systému. Další volbou jsou Číselníky. Zde se dostaneme na přehled všech číselníků, které jsou v systému k dispozici. Po kliknutí na příslušný název číselníku jej můžeme editovat, tj. buď z něj položky mazat, nebo do něj položky přidávat. Čtvrtým odkazem je pak Zařízení. Zde se dostaneme po

odfiltrování na přehled zařízení, která můžeme buď, pokud jsou ve stavu Založeno, aktivovat nebo, pokud jsou ve stavu aktivním, je můžeme vyřadit. Předposlední volbou jsou Náhradní díly. Zde se stejně jako u zařízení dostaneme k přehledu všech náhradních dílů aktuálně založených v podniku a můžeme některý díl vymazat. Poslední položkou v administraci je Vložení nové role k uživateli, kde se dostaneme k formuláři, pomocí něhož můžeme vkládat k různým uživatelům různé role.

Odhlásit

Poslední položka v menu je Odhlásit. Po kliknutí na tento odkaz provedeme odhlášení ze systému, pokud by jsme chtěli dále pracovat v IS, musíme se znovu přihlásit.

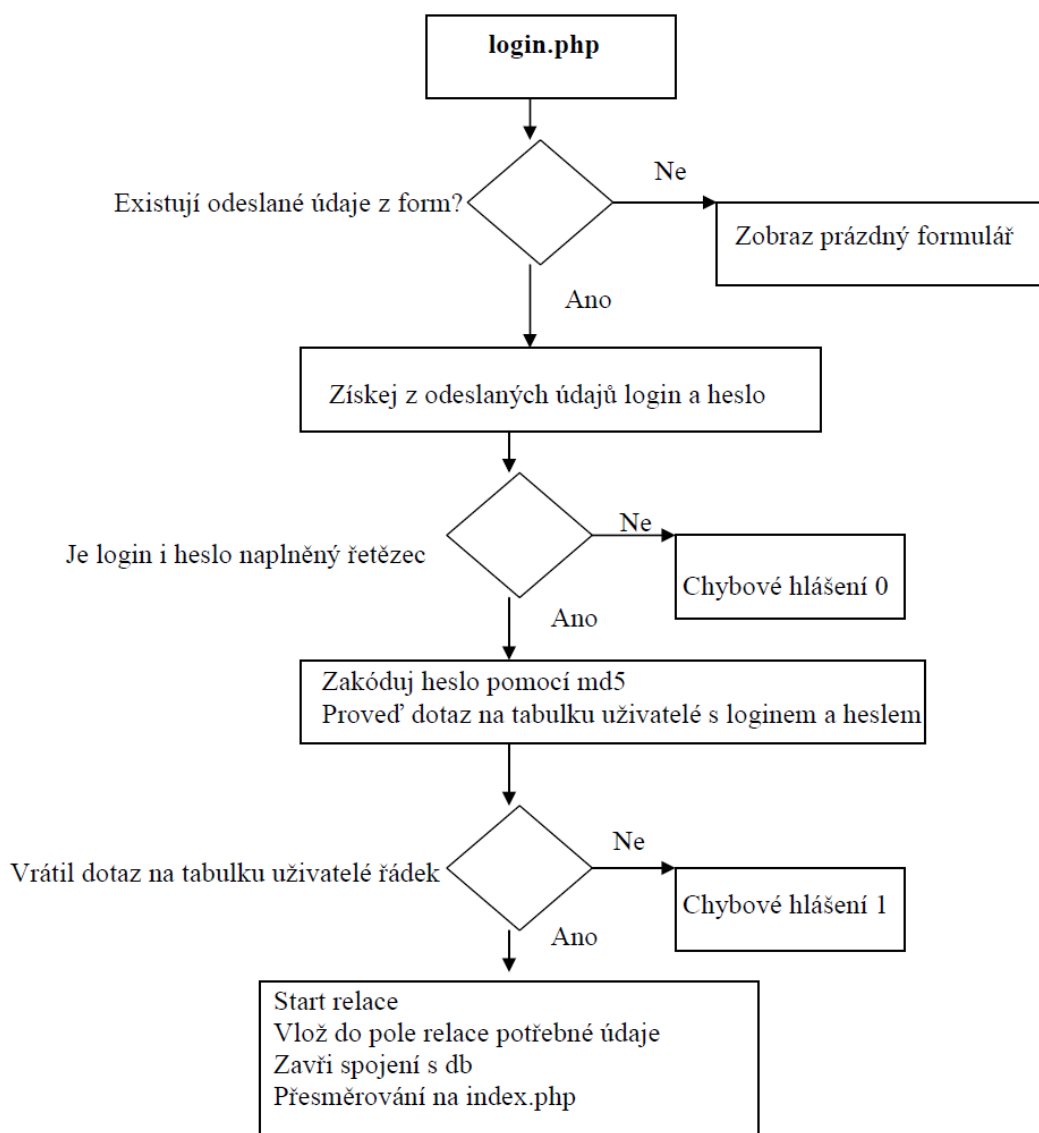
5.2 POPIS SYSTÉMU Z POHLEDU PROGRAMÁTORA

Každý modul IS je umístěn v samostatné složce a obsahuje základní typy formulářů (kompletní zdrojové kódy jsou na přiloženém CD viz. Příloha 2). Dále je vždy ve složce soubor formfunction.php, ve kterém jsou specifické funkce. Ty kontrolují a ukládají data do databáze. Všechna vkládaná data jsou ošetřena proti vložení zakázaných znaků, při ukládání a vkládání údajů do db je u unikátních záznamů kontrolováno, jestli už se v tabulce nevyskytují. Před každým uložením nebo vložem proběhne kontrola, zda je vše vyplněno, a zda jsou zadané údaje správného formátu.

5.2.1 Zabezpečení a autentizace uživatelů

Autentizace uživatelů je provedena v modulu login.php. Uživatel je identifikován jednoznačným loginem a svým heslem (heslo je zakódováno pomocí funkce md5), pomocí nichž je proveden dotaz na databázi a je zjištěno, zda má daný uživatel právo přístupu do IS. Pokud ne, zobrazí se mu hláška o špatně zadaných údajích. Pokud ano, nastaví funkce údaje do Session (popis viz. 3.2.2, dále relace) a pustí ho dále. Do relace jsou uloženy údaje o uživateli a údaje dále potřebné v systému (např. id_uživatele, podnik pod který patří, jméno a příjmení a také čas,

po kterou bude daná relace platná). Na začátku každého modulu, ještě před tím, než je cokoli odesláno, je kontrolována platnost relace. Pokud její platnost již vypršela je uživatel přesměrován na stránku login.php, kde se může znovu přihlásit a založit novou relaci.



Obr.8: Vývojový diagram funkce pro relaci

Každý modul, který v IS existuje, je zabezpečen proti neoprávněnému přístupu. Pomocí funkce `zkontroluj_prava`, která je popsána níže (viz 5.2.3), je poslán dotaz na db s id uživatele a je kontrolováno, jestli má na tento modul uživatel právo. Pokud ano, modul se mu zobrazí, pokud ne, zobrazí se mu jen hlavička stránky s menu a chybové hlášení o tom, že na daný modul nemá právo.

5.2.2 Popis hlavních funkcí souboru `function.php` pro práci s formuláři

V souboru `function.php`, který je umístěn v kořenovém adresáři, jsou napsány funkce pro vytváření a práci s formuláři, generování dat podle záznamů z db, ale také je zde funkce pro aktualizaci skladů náhradních dílů.

funkce `form_text($nazev, $hodnota, $velikost=40, $maxdelka=40, $sloupcunasirku=1, $radkunavysku=1, $skryty=FALSE)`

Je to funkce, která zobrazí vstupní textové pole pro zadávání informací do formuláře. Parametr string `$nazev` určuje název pole, parametr string `$hodnota` určuje základní zobrazenou hodnotu, parametr int `$velikost` určuje velikost pole, parametr int `$maxdelka` určuje maximální počet zadávaných znaků. Parametr int `$sloupcunasirku` určuje přes kolik sloupců a parametr int `$radkunavysku` přes kolik řádků se pole zobrazí. Parametr bool `$skryty` určuje, zda bude pole skryté nebo editovatelné. Funkce si zjistí, které hodnoty jsou vyplněny, sestaví výstup a vrátí standardní html vstupní pole včetně buňky tabulky.

funkce `form_seznam($nazev, $radky, $vybrano, $skryty=FALSE)`

Tato funkce vkládá do formuláře seznam hodnot v rolovacím seznamu.

Parametr string `$nazev` určuje název pole, parametr array `$radky` nese hodnoty, které budou zobrazeny v seznamu, parametr string `$vybrano` určuje vybranou hodnotu, parametr bool `$skryty` určuje zda bude seznam skrytý nebo editovatelný. Pokud má být seznam skrytý, přidá do html tagu `disabled`. V opačném případě vypíše normální select a pokračuje dál, vypíše první položku, která je vždy zvolte, definuje si pole a dokud je v poli `$radky` nějaká hodnota, naplňuje jimi pole `$vysledek_pole`. Poté prochází asociativní pole a vypisuje položky do selectu. Pokud narazí na stejnou

položku, které je poslána v proměnné \$vybrano, přiřadí jí parametr selected. Po skončení vypíše ukončení selectu a konec buňky tabulky.

funkce odkaz_tlacitko (\$modul, \$id_uzivatele, \$parametr, \$txt, \$cesta)

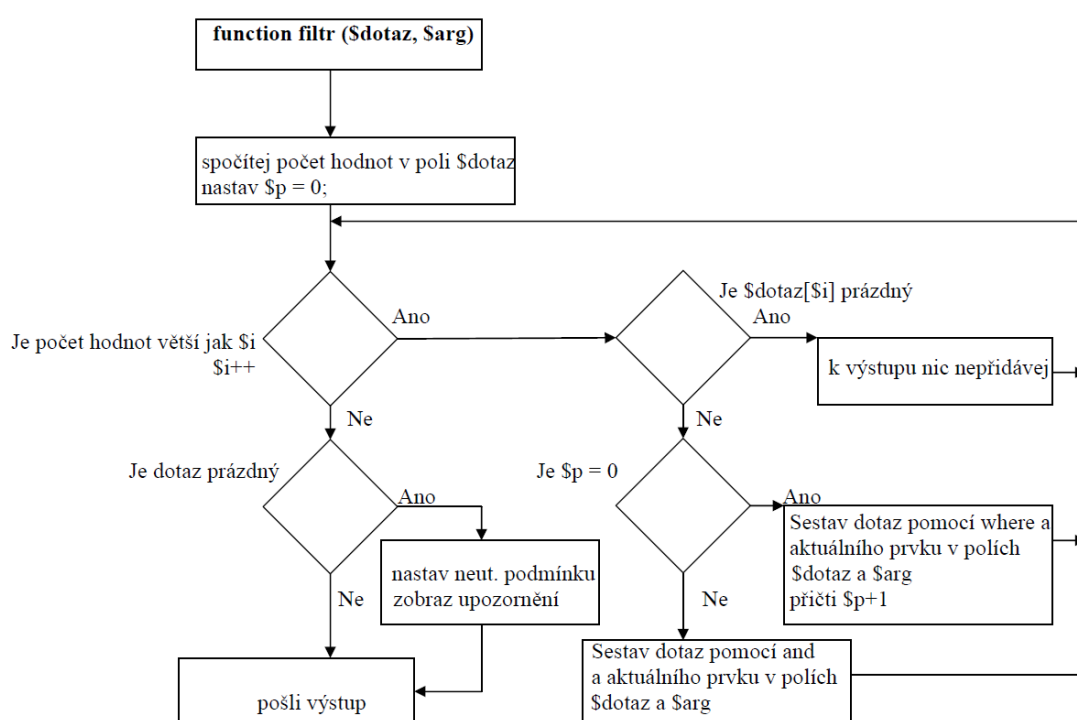
Funkce zobrazuje tlačítko, které nás odkazuje na jiný modul včetně ověření, zda na něj má daný uživatel právo. Pokud ne, tak se mu tlačítko vůbec nezobrazí. Parametr string \$modul určuje kompletní název modulu pro ověření přístupu. Parametr int \$id_uzivatele nese hodnotu id uživatele, který chce tlačítko zobrazit taktéž k ověření práv. Parametr int \$parametr určuje v případě více možností o jaké tlačítko půjde. Parametr string \$txt nese hodnotu textu, který bude zobrazen v tlačítku a parametr string \$cesta určuje aktuální cestu k odkazovanému modulu. Funkce nejprve provede dotaz na databázi, kde se zeptá, jestli daný uživatel má právo na požadovaný modul. Pokud dotaz vrátí záznam, zobrazí se mu tlačítko s daným textem a odkazem na modul, pokud ne, tak funkce nic nevrátí.

funkce vrat_prefix (\$tabulka, \$prefix, \$prefix_cislo)

Je to funkce, která vrací aktuální číselné hodnoty podle počtu záznamu v tabulce včetně prefixu. Parametr string \$tabulka určuje, ke které tabulce se bude záznam vázat, string \$prefix jaký prefix musí záznam mít a int \$prefix_cislo, kolik číslic má v záznamu být. Funkce nejprve podle zadaných parametrů sestaví dotaz na db, aby mohla zjistit počet záznamů. Potom se dotáže db kolik záznamů je právě v tabulce. Podle parametru \$prefix_cislo pomocí cyklu naplní prefix toliko nulami, kolik chceme mít v celkovém výstupu číslic. Ke stringu, kde jsou zatím jen nuly, přidá počet záznamů v tabulce a zjistí počet znaků. Dokud bude počet znaků větší než počet, kolik jsme předali v parametru funkce, bude odstraňovat první číslici. Až bude počet v pořádku, přidá k číslu stringový prefix předaný také v parametru funkce. Funkce vrací aktuální záznam podle počtu záznamu v tabulce včetně prefixu.

Funkce filtr (\$dotaz, \$arg)

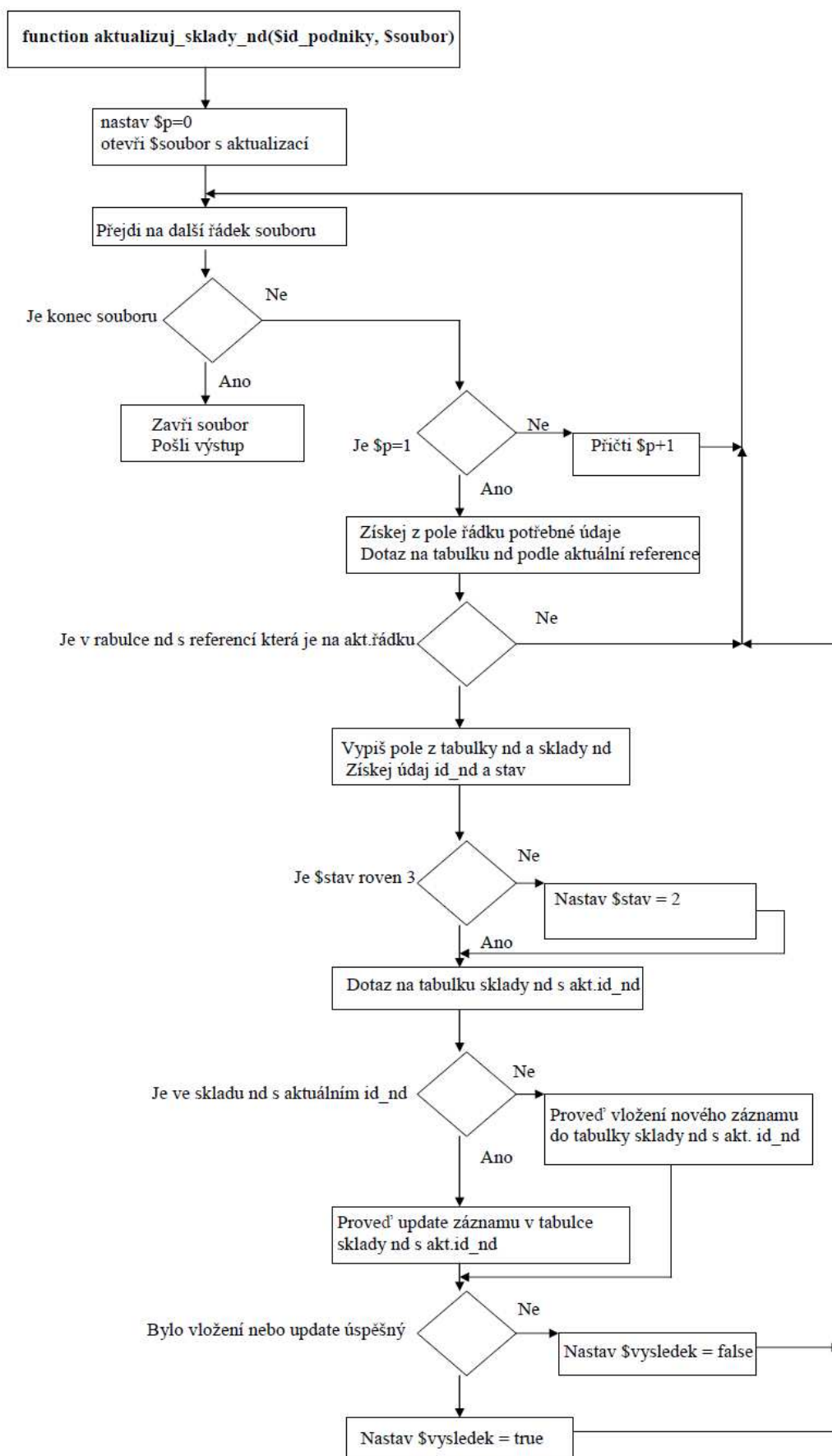
Funkci používáme pro filtraci pomocí více parametrů. Parametr array \$dotaz nese hodnoty testovaných záznamů. Parametr array \$arg nese názvy sloupců, které se budou v tabulce testovat. Funkce vrátí string, ve kterém je celá podmínka filtru pro dotaz sql.



Obr.9: Vývojový diagram funkce pro filtraci

Funkce aktualizuj_sklady_nd (\$id_podniky, \$soubor)

Tato funkce provede aktualizaci skladu náhradních dílů podle dat, vygenerovaných do csv souboru ze SAPu. Parametr int \$id_podniky určuje, který podnik se má podle daného souboru aktualizovat. Parametr string \$soubor cestu k csv souboru, podle kterého se bude aktualizovat. Funkce vrátí true, pokud byla aktualizace úspěšná. V opačném případě vrátí false.



Obr.10: Vývojový diagram funkce pro update skladů nd ze SAPu

5.2.3 Popis hlavních funkcí souboru upload.php pro práci s uploadem souborů

V souboru upload.php, který je také umístěn v kořenovém adresáři, jsou napsány funkce pro kontrolu, úpravu a následné nahrání obrázků na server.

Funkce vrat_nazev_souboru()

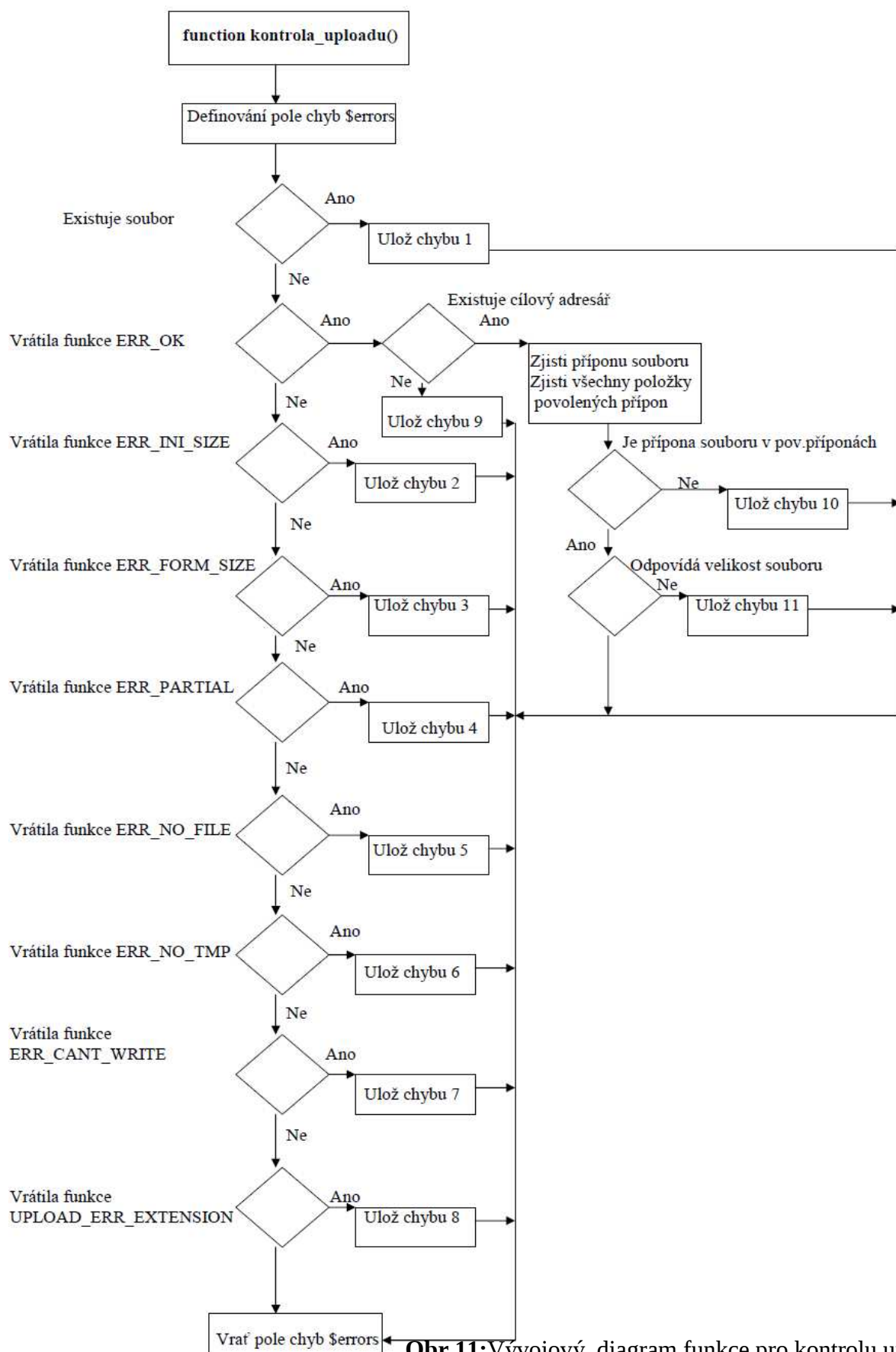
Funkce nám vrátí název souboru, s kterým pak bude ukládán na disk. Nejprve provede dotaz na tabulku foto, kde vybere sloupec id_foto a spočítá počet záznamů v tabulce. Poté k počtu přidá nulu a pomocí funkce implode složí dohromady id aktuálního vkládaného záznamu, dolní pomlčku a název původního souboru. Tím je s určitou jistotou jednoznačně definován unikátní název ukládaného souboru.

Funkce upload_souboru()

Funkce nám provede konečné nahrání souboru do požadovaného adresáře. Nejprve zavolá podřízenou funkci vrat_nazev_souboru, s kterou pak soubor uloží. Poté již provede přesun souboru z dočasného adresáře na námi požadované místo již s aktuálním jménem a zavolá funkci zmenši_obrazek, která jednak, pokud je obrázek moc veliký, provede jeho zmenšení o jednu třetinu a jednak vytvoří nový malý obrázek, který bude sloužit k náhledu. Funkce vrátí true pokud se nahrání povedlo, jinak vrátí false.

Funkce zmenši_obrazek (\$cil)

Funkce nám zajistí zmenšení obrázku pokud je moc veliký. Parametr string \$cil určuje název souboru. Nejprve funkci nadefinujeme adresáře, odkud si má obrázek vzít a kam ho má uložit. Poté zjistíme velikost obrázku. Pokud je obrázek větší než 300px, tak výšku zmenšíme na 300px a šířku na 400px, vytvoříme v proměnné obsah obrázku a následně vytvoříme nový snímek. Tomu pak změníme velikosti stran a nakonec celý obrázek uložíme. Funkce vrátí true při dokončení, jinak false.



Obr.11:Vývojový diagram funkce pro kontrolu uploadu

Funkce kontrola_uploadu()

Funkcí kontrolujeme, zdali soubor k nahraní na server je v pořádku a jestli samotný přesun obrázku na server proběhl bez chyb. Vrací pole \$errors, kde je popsána chyba která nastala, v případě žádné chyby je pole prázdné.

5.2.4 Popis hlavních funkcí souboru db_class.php pro práci s databází

V tomto souboru je umístěná třída snopDb, ve které jsou definovány funkce pro komunikaci s databází.

Funkce vypis_pole_objektu (\$sql)

Funkce provádí dotaz na databázi a vrátí pole hodnot podle příkazu sql. Parametr string \$sql nese celý sql příkaz, který se má provést. Funkce vrací pole s výsledkem dotazu. V případě chyby vypíše varování a vrací false. Nejprve provede dotaz na databázi. Pokud je dotaz v pořádku a vrátí nějaká data podle zadaného příkazu, uloží je do proměnné \$vysledek. Pokud ne, vypíše chybu a vrací false. Dále testuje počet řádků ve výsledku. Pokud je v proměnné nějaký řádek, tak do pole \$vysledek_array ukládáme záznamy jako objekty až do doby, než ve výsledku není žádný záznam. K záznamům pak můžeme přistupovat pomocí příkazu \$radek->nazev_sloupce.

Funkce pocet_radku_n (\$sql)

Funkce provádí dotaz na databázi a vrátí počet řádků v tabulce. Parametr string \$sql nese celý sql příkaz, který se má provést. Funkce nejprve provede dotaz na databázi podle předaného parametru. Pak pomocí vlastnosti num_rows zjistí počet řádků. Funkce vrací int \$pocet kde je počet řádku v tabulce.

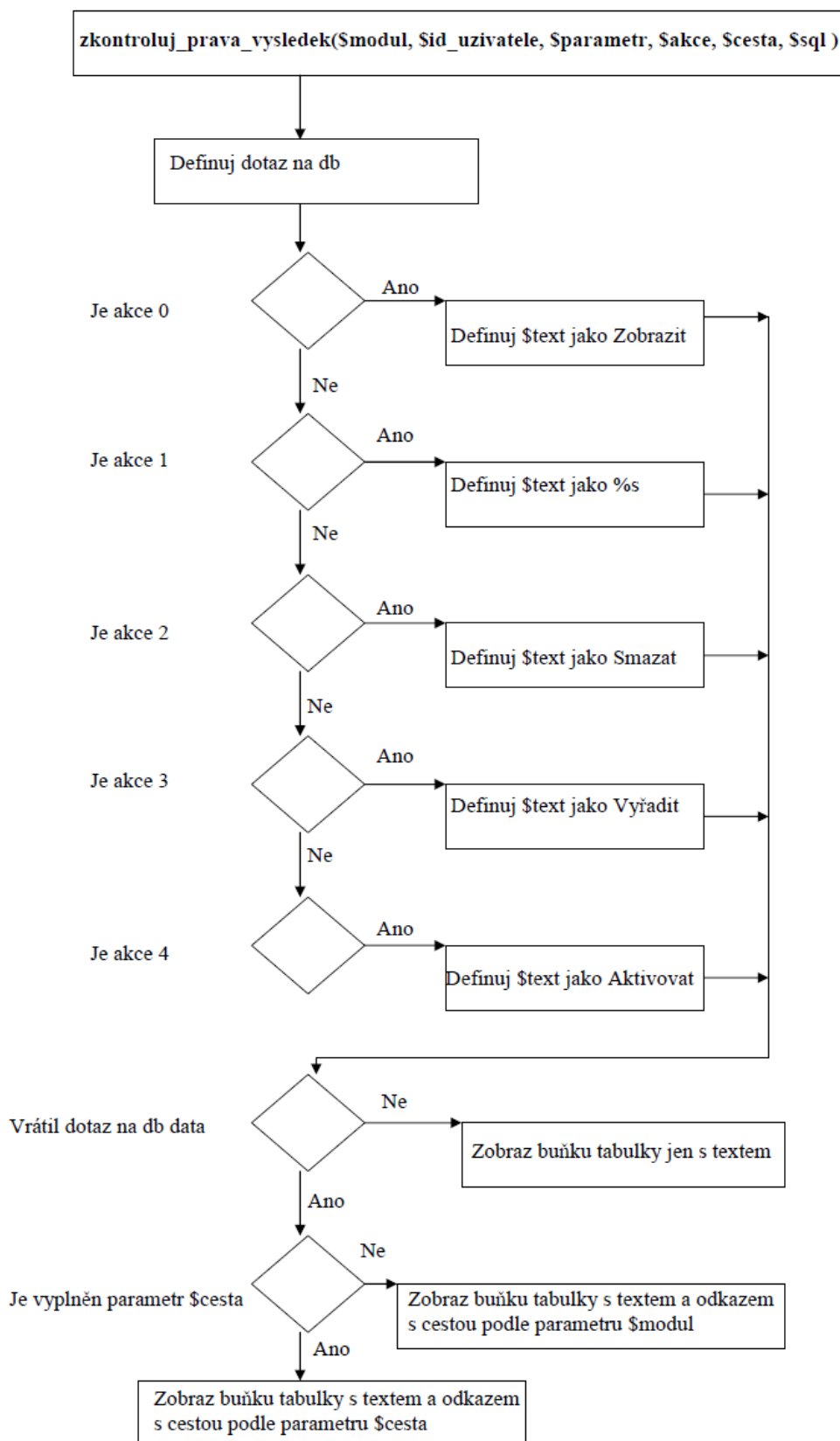
Funkce zkontroluj_prava (\$modul, \$id_uzivatele)

Funkce provádí kontrolu práv na daný modul a v případě, že uživatel nemá na modul oprávnění, tak mu zobrazí chybu. Parametr string \$modul nese celý název modulu. Parametr int \$id_uzivatele nese informaci o id_uzivatele pro kontrolu

oprávnění. Funkce provede dotaz na tabulku uživatele_role_moduly a uzivatele_prava s parametrem názvu modulu a id uživatele předaných funkci. Pokud funkce vrátí nějaký záznam, je vše v pořádku. Jinak vrací upozornění o chybějícím oprávnění.

Funkce zkontroluj_prava_vysledek (\$modul, \$id_uzivatele, \$parametr, \$akce, \$cesta, \$sql)

Funkce, která provede dotaz na databázi a vrátí řádek tabulky do výpisu, kde je odkaz s textem podle parametru. Pokud uživatel na daný modul nemá právo bude odkaz neaktivní. Parametr string \$modul určuje, na který modul je odkazováno. Parametr int \$id_uzivatele je id aktuálního uživatele. Parametr string \$cesta je aktuální cesta k souboru. Parametr string \$sql je sql příkaz pro výpis aktuálního počtu záznamu, ke kterému se pod odkazem dostaneme. Parametr int \$parametr - pokud je 0, vypíše se odkaz s názvem zobrazit, pokud 1 vypíše se odkaz s id, pokud 2 vypíše se odkaz s textem smazat, pokud 3 vypíše se vyřadit, pokud 4 tak se vypíše aktivovat.



Obr.12: V.diagram funkce pro kontrolu práv

6. ZÁVĚR

Cílem této bakalářské práce bylo navrhnout, naprogramovat a odladit informační systém (dále IS) pro firmu S. N. O. P., zabývající se výrobou autosoučástek.

Na základě seznámení se s chodem firmy a s požadavky na IS jsem navrhl a zvolil koncepci IS, navrhl databázi, vymodeloval E-R diagram který jsem analyzoval a vytvořil informační systém. IS je naprogramován ve skriptovacím jazyce PHP 5.2, který spolupracuje s databází MySQL 5.0 pomocí funkce třídy mysqli.

IS umí rozlišit uživatele podle podniku a podle jejich práv. Podle toho jim vygeneruje menu a vždy jim nabídne jen informace, které se týkají jejich podniku, ostatní jim zůstává skryto. Je ošetřen proti vkládání duplicitních záznamů a kontroluje správnost zadávaných informací. IS umožňuje shromažďovat informace o zařízeních. Uživatelé do něho zadávají závady, které se stanou na strojích. Opraváři si je převezmou a závadu opraví, tím stroj znovu uvedou do aktivního stavu. IS dále umožňuje uživateli, aby naplánoval údržbu stroje. Údržbář pak práci provede, vloží o údržbě záznam do IS a vrátí zařízení do aktivního stavu. Systém také sbírá a vyhodnocuje informace o denních či nepravidelných kontrolách, eviduje náhradní díly ve firmě, umožňuje rozlišovat, které díly patří k jakému zařízení. Skladové položky jsou automaticky aktualizovány každý den podle informací v SAPu.

IS je dostupný na adrese www.betonhronek.cz/us (testovací účet s plnými právy test/statnice09). Systém splňuje požadavky firmy S. N. O. P.. Ladění IS bude pokračovat do konce července 2009, kdy by měl být nasazen ve firmě k ostrému provozu.

7. LITERATURA

- [1] Kofler M.: Mistrovství v MySQL 5. Computer Press
ISBN 978-80-251-1502-02
- [2] Kolektiv autorů: Mistrovství v PHP 5. Computer Press
ISBN 978-80-251-1519-0
- [3] Kolektiv autorů: PHP a MySQL - rozvoj webových aplikací. Soft Press
ISBN 80-86497-60-7
- [4] Maslakowski M.: Naučte se MySQL za 21 dní. Computer Press
ISBN 80-72226-448-6.
- [5] Vývoj PHP aplikací - <http://interval.cz/vyvoj-aplikaci/php/>
- [6] Kurz PHP - <http://www.linuxsoft.cz/php/>

SEZNAM PŘÍLOH

Příloha 1 E-R diagram ve formátu A3

Příloha 2 CD-ROM se zdrojovými kódy a vypracovanou bakalářskou prací