

MORAVSKÁ VYSOKÁ ŠKOLA OLOMOUC

Ústav informatiky a aplikované matematiky

Jiří Jemelka

**Návrh lokální firemní databáze typu Access ovládané
přes VB.NET**

A proposal local company controlled by the Access type
VB.NET

Bakalářská práce.

Vedoucí práce : PhDr. Jan Lavrinčík, Ph.D.

Olomouc 2015

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a použil jen uvedené informační zdroje. Prohlašuji, že odevzdaná tištěná verze bakalářské práce se shoduje s elektronickou verzí vloženou do IS/STAG.

Olomouc

Rád bych poděkoval vedoucímu své práce PhDr. Janu Lavrinčíkovi, Ph.D. za věcné připomínky, cenné rady a jeho podporu, kterou mě věnoval při zpracování mé bakalářské práce.

Moravská vysoká škola Olomouc
Akademický rok: 2013/2014

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Jiří JEMELKA
Osobní číslo: M12133
Studijní program: B6208 Ekonomika a management
Studijní obor: Podnikové informační systémy
Název tématu: Návrh lokální firemní databáze typu Access ovládané přes VB.NET.
Téma anglicky: A proposal local company controlled by the Access database type VB.NET.
Zadávající katedra: Ústav informatiky a aplikované matematiky

Zásady pro vypracování:

Cíl práce:

Cílem této bakalářské práce je vytvořit návrh lokální firemní databáze typu Access, která je ovládaná aplikací zaměřenou na problematiku vyhodnocování dodavatelů dle vybraných metodik managementu partnerství s dodavateli vytvořenou v programovacím prostředí VB.NET.

Předmět práce :

Vytvoření návrhu obecného konceptu aplikace pro vyhodnocování dodavatelů podle vybraných metodik managementu partnerství s dodavateli, která bude rozčleněna do několika interních modulů. Každý interní modul návrhu této aplikace bude zaměřen na určitou oblast managementu partnerství s dodavateli. Aplikace je koncipována jako samostatné softwarové řešení pracující nad lokální databází, které bude ve své datové struktuře uchovávat pouze data nutná pro své vyhodnocovací procesy. Následně bude provedeno hlubší rozpracování vybrané části návrhu aplikace na praktickém příkladu fiktivní firmy a implementace této části bude provedena pomocí programovacího prostředí VB.NET a lokální databáze typu Access. Na závěr bude vypracováno doporučení zaměřené na další rozvoj návrhu této aplikace, která může sloužit jako finální aplikace nebo jako dočasné řešení pro firmy, které uvažují o začlenění managementu partnerství s dodavateli do svého firemního IS.

Metodologie práce :

- Stanovení cíle bakalářské práce.
- Při tvorbě bakalářské práce postupovat dle metodiky pro tvorbu bakalářské práce na MVŠO.
- Základní analýza dané problematiky podle stanoveného cíle bakalářské práce s vytvoření harmonogramu bakalářské práce do kterého jsou zahrnuty pravidelné konzultace s vedoucím práce.
- Tvorba teoretické části, která je zaměřená na vysvětlení základních pojmů řešené problematiky.
- Návrh konceptu pro vybrané metodiky vyhodnocování dodavatelů dle managementu partnerství s dodavateli.
- Analýza lokálních databází typu Access a programovacího prostředí MS Visual Basic.NET z pohledu zamýšleného návrhu aplikace pro vyhodnocování dodavatelů.
- Návrh konceptu aplikace pro vyhodnocování dodavatelů dle vybraných metodik managementu partnerství s dodavateli.
- Implementace vybrané části navrhnutého konceptu aplikace s využitím lokální databáze typu Access a programovacího prostředí MS Visual Basic.NET na praktickém příkladu.
- Závěr s doporučením zaměřeným na další rozvoj řešeného návrhu konceptu aplikace pro vyhodnocování dodavatelů.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

JOACHIM FUCHS, ANDREAS BARCHFELD. Visual Basic Velká kniha řešení. 1. Vyd. Brno: Computer Press,a.s., 2010, 722 s. ISBN 978-80-251-2212-9.

PAVEL KOCICH, ONDŘEJ SPILKA. 1001 tipů a triků pro Microsoft Visual Basic. 1. Vyd. Brno: Computer Press,a.s., 2010, 520 s. ISBN 978-80-251-2118-4.

PETER BELKO. Access 2013 Podrobná uživatelská příručka. 1. vyd. Brno: Computer Press,a.s., 2014, 392 s. ISBN 978-80-251-4125-0.

JAROSLAV NENADÁL. Management partnerství s dodavateli. 1. vyd. Praha: Management Press, s.r.o., 2006, 323 s. ISBN 80-7261-152-6.

JAROSLAV NENADÁL A KOL. Moderní management jakosti. Principy, postupy, metody. 1. vyd. (dotisk) Praha: Management Press, s.r.o., 2011, 377 s. ISBN 978-80-7261-186-7.

MILOSLAV SYNEK A KOLEKTIV. Manažerská ekonomika. 5., aktualizované a doplněné vydání. Praha: Grada Publishing, a.s., 2011, 480 s. ISBN 978-80-247-3494-1.

Vedoucí bakalářské práce:

PhDr. Jan LAVRINČÍK, DiS.

Ústav informatiky a aplikované matematiky

Datum zadání bakalářské práce:

10. února 2014

Termín odevzdání bakalářské práce:

31. března 2015

Podpis studenta:

Datum:

Podpis vedoucího práce:

Datum:

Mgr. et Mgr. Michaela VANĚČKOVÁ, Ph.D.
prorektorka

L.S.

Mgr. Veronika RAINZOVÁ, Ph.D.
manažer ústavu

V Olomouci dne 14. dubna 2014



OBSAH

ÚVOD	09
1 MANAGEMENT PARTNERSTVÍ S DODAVATELI	11
1.1 Nákup v organizaci a dodavatelský řetězec (SCM)	12
1.2 Standardy a principy partnerství s dodavateli.....	13
1.3 Politika a strategie vztahů s dodavateli	14
1.4 Definování požadavku na dodávky a dodavatele.....	14
1.5 Výběr vhodných dodavatelů.....	14
1.6 Společné plánování s dodavateli	15
1.7 Posuzování stavu vyzrálosti systému managementu u dodavatelů.....	16
1.8 Ověřování shody dodávek.....	16
2 HODNOCENÍ VÝKONNOSTI DODAVATELŮ	18
2.1 Přezkoumání harmonizace strategií cílů – vlastní versus dodavatelé	18
2.2 Definice vhodných ukazatelů a metrik.....	18
2.3 Návrh sběru dat a systému hodnocení dodavatelů	19
2.4 Implementace systému pro hodnocení dodavatelů	19
3 DATABÁZE TYPU ACCESS	20
3.1 Datové typy	21
3.2 Návrh tabulek databáze	21
3.3 Vztahy mezi tabulkami databáze(Relace).....	22
3.4 Připojení k databázi.	23
3.4.1 Databázové rozhraní OLE DB.....	24
3.4.2 Databázové rozhraní ADO.NET.	24
3.5 Připojení k databázi Access typu *.accdb.	27
4 VISUAL.BASIC .NET	28
4.1 Charakteristika VB .NET.....	28
4.2 .NET Framework.....	29
4.2.1 Architektura .NET Framework.....	29
4.2.2 Kompilace .NET aplikací (VB .NET).....	30
5 METODIKA NÁVRHU APLIKACE.....	31
5.1 Specifikace požadavků	32
5.2 Analýza	32
5.3 Návrh a implementace	33

5.4 Testovací režim	35
6 APLIKACE – PRAKTICKÁ ČÁST	36
6.1 Nedostatky současného řešení.	36
6.2 Navržené řešení.	37
6.3 Obecný popis funkcionalit navrhované aplikace.	37
6.4 Obecné uživatelské znázornění struktury aplikace.	39
6.5 Návrh validační metodiky a hodnotících kritérií.	40
6.5.1 Návrh validační metodiky.	40
6.5.2 Návrh hodnotících kritérií.	44
6.6 Obecný popis procesů hodnocení výkonnosti dodavatele	46
6.7 Reporting aplikace.	48
6.8 Elektronická nápověda aplikace.	48
ZÁVĚR	49
ANOTACE	50
SEZNAM LITERATURY	51
SEZNAM OBRÁZKŮ	52
SEZNAM TABULEK	52
SEZNAM PŘÍLOH.....	52

ÚVOD

Dominantní roli téměř ve všech oblastech současného tržního prostředí má poptávka nad nabídkou a to vyžaduje, aby cíle podniků byly efektivnější a rychle reagovaly na změny trhu. Konkurence mezi prodejci je velmi vysoká, ale u mnoha komodit již nelze produkty příliš diferencovat, protože rozdíly v kvalitě mezi produkty od různých výrobců jsou téměř zanedbatelné a technický náskok pomocí výrobních inovací bývá konkurencí velmi rychle eliminován obdobným produktem.¹ Zákazníci v dnešní době neočekávají jen, že produkt bude fungovat tak jak má, ale požadují, aby veškeré firemní procesy byly soustředěny na co nejvyšší užitek zákazníka. Proto interní a externí hodnototvorný řetězec musí být flexibilní k zákazníkovi s maximálním využitím marketingové koncepce podniku.² Jde o procesně orientovaný pohled, který sleduje tok výkonných objektů (materiál, informace, finance) pomocí sítě podniků, které jsou součástí tohoto procesu a vytvářejí tak komplexní hodnototvorný řetězec, jenž je označován jako dodavatelský řetězec.³ Koordinování a řízení podnikatelských procesů v dodavatelském řetězci mezi různými partnery (jako jsou např. dodavatelé, logistické služby, kooperující výrobci, zprostředkovatelé odbytu, konečný zákazník) označujeme jako řízení dodavatelského řetězce (supply chain management dále jen SCM)⁴.

Problematika řízení dodavatelského řetězce výrobní organizace je velmi obsáhlá a zahrnuje procesy od vyhledávání dodavatelů, nákupu, způsobu plánování a realizace výroby, až po prodejní procesy vyrobeného produktu a procesy marketingu. V mé bakalářské práci jsem se zaměřil na problematiku z oblasti nákupu, která je specifikována managementem partnerství s dodavateli a standardizována pomocí norem ISO z oblasti systému managementu jakosti (QMS). Protože se domnívám, že kvalitní dodavatel může výrazně přispět k růstu efektivity a konkurenceschopnosti výrobní organizace a je proto důležité se této oblasti systematicky věnovat. Management partnerství s dodavateli zahrnuje mimo jiné problematiky jako je tvorba firemní politiky a strategií vztahů s dodavateli, která následně vyspecifikuje požadavky na dodavatele a jejich dodávky, kdy tyto procesy směřují k výběru vhodných dodavatelů. Další důležitou problematikou managementu partnerství s dodavateli je pravidelný monitoring dodávek

¹ Srov. TOMEK, Gustav, a VÁVROVÁ, Věra. *Řízení výroby a nákupu*. s.335

² Srov. TOMEK, Gustav, a VÁVROVÁ, Věra. *Integrované řízení výroby*. s.14

³ Srov. SYNEK, Miroslav, a kol., *Manažerská ekonomika*. s.243

⁴ Srov. FIALA, Petr. *Modelování dodavatelských řetězců*. s.14

s validací sledovaných parametrů a následným hodnocením dodavatelů, posuzování úrovně systému managementu dodavatelů dle standardů ISO (např. QMS, EMS, OHSAS a atd.). Nedílnou součástí je motivace dodavatelů ohledně jejich neustálého zlepšování dle modelu Excellence EFQM, metodik ISO a managementu partnerství s dodavateli, až po vznik firemního programu partnerství s dodavateli.

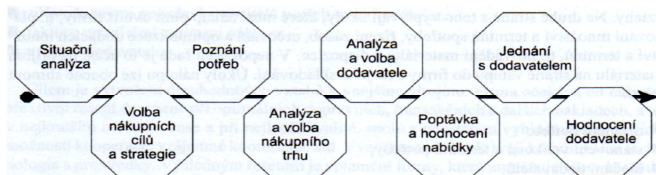
Téma mé bakalářské práce je návrh lokální firemní databáze typu Access ovládané přes VB.NET. Dané softwarové řešení bude vytvořené v vývojovém prostředí VB.NET - Visual Studia 2013 pracující nad lokální databází typu MS–Access 2013, které v svých datových strukturách uchovává pouze data, která jsou nutná pro správnou funkčnost dané aplikace.

Mým hlavním cílem je navrhnout problémově orientovanou lokální databázi, která je ovládána aplikací zaměřenou na hodnocení výkonnosti dodavatelů, kdy se v počáteční fázi zaměřuji na monitoring dodávek vstupních materiálů pro výrobní procesy dané organizace, které jsou validovány u sledovaných parametrů podle předdefinovaných metodik i pro substituční skupinu materiálů s následným monitoringem vzniku neshod v průběhu výroby a reklamačním řízení od odběratelů, které je ukončeno následným hodnocením výkonnosti dodavatelů za sledované období. Mezi dílčí cíle patří analýzy procesů, které slouží k vytvoření datového modelu, s požadovanou úrovní integrity dat dané databáze a specifikací validačních a hodnotících kritérií pro zamýšlené řešení. Dále mezi dílčí cíle patří analýza vhodnosti použité databáze a softwarového prostředku pro tvorbu dané aplikace. V této práci jsem se zaměřil na výběr vhodného modelu životního cyklu vývoje softwaru a jeho použití při uplatnění objektově orientovaného programování a modelu specifikace požadavků. Jehož cílem je popsat odsouhlasené funkcionality softwarové aplikace. Pro tvorbu návrhu aplikace jsem využil modelovací jazyk UML a popsal jsem pomocí use case diagramu funkční model aplikace, pomocí sekvenčního diagramu dynamický pohled na funkcionality aplikace a pomocí diagramu tříd statický pohled na strukturu a vazby tříd aplikace. Tyto informace jsem doplnil datovým modelem dle zamýšleného řešení rozsahu v této bakalářské práci a doplnil je standardním třívrstevným pohledem na architekturu zamýšleného výsledného systému a to na uživatelské rozhraní, distribuci a persistenci. Kdy takto vytvořené řešení bude sloužit k testování procesů sledovaných problematik a výsledky budou následně využity při implementaci této problematiky do podnikového informačního systému (např. SAP).

1 MANAGEMENT PARTNERSTVÍ S DODAVATELI

1.1 Nákup v organizaci a dodavatelský řetězec (SCM).

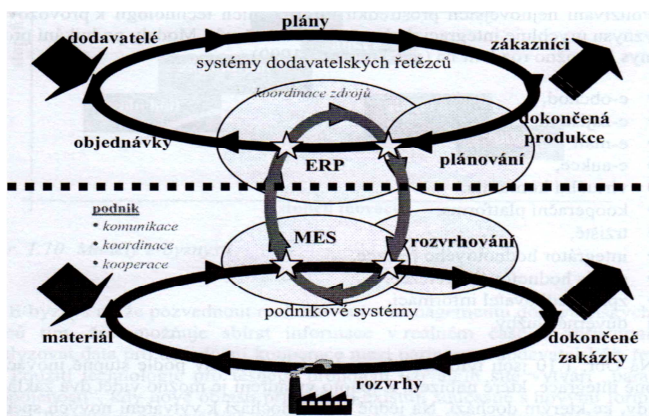
Nakupování je systematický proces, který je zabezpečován útvarem nákupu (zásobování), ve kterém s využitím nákupního marketingu odběratelé zabezpečují dodávky jako vstupy pro své vlastní plánované firemní procesy.⁵



Obr. 1. Princip zjednodušeného modelu nákupního marketingu.

Zdroj : SYNEK, Miroslav, a kol., *Manažerská ekonomika*. s.208.

Pro dosažení integrace mezi podnikovými systémy a systémy dodavatelských řetězců jsou v podnikových IS využívány systémy typu MES (Manufacturing Execution Systems, které propojují materiálové a informační toky podniku) a systémy typu ERP (Enterprise Resource Planning, které jsou sofistikovaně zaměřeny na plánování a rozvrhování výroby a to od dodání materiálu, až po expedici finálního produktu k odběrateli).⁶ Výhodou ERP systémů je využívání transakčních databází, které umožňují zvýšit integraci aktivit dodavatelských řetězců do podnikového IS. Současným trendem je integrace ERP systémů napříč celým hodnotovým řetězcem, kde je účastnická výměna informací brána jako základní kritérium optimalizace celkového zdroje, který je určen pro dosažení maximální spokojenosti spotřebitele (odběratele, konečného zákazníka).



Obr. 2. Schéma integrace podnikových systémů a systémů dodavatelských řetězců.

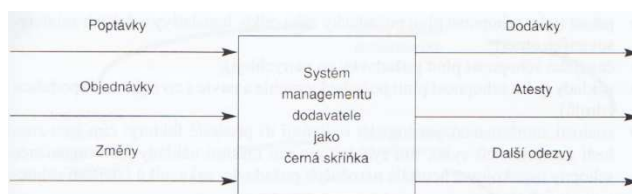
Zdroj : FIALA, Petr. *Modelování dodavatelských řetězců*. s.33.

⁵ Srov. SYNEK, Miroslav, a kol., *Manažerská ekonomika*. s.207-208

⁶ Srov. FIALA, Petr. *Modelování dodavatelských řetězců*. s.32-33

1.2 Standardy a principy partnerství s dodavateli.

Tradiční pojetí nákupu je zaměřeno na komunikaci zástupců dodavatelů a odběratelů, kteří se zaměřují pouze na procesy spojené s dodávkami, jako jsou poptávky, objednávky, vlastní dodávky, odezvy od dodavatelů a atd., ale již se nezabývá úrovní implementace QMS u dodavatele, který je pro spolehlivost a jakost dodávek velmi důležitý.⁷



Obr. 3. Systém managementu dodavatele jako černá skříňka.

Zdroj : NENADÁL, Jaroslav. *Management partnerství s dodavateli*. s.25.

Současná koncepce QMS implementovaná dle standardů ISO se již nezaměřuje pouze na jakost dodávek, ale i na tvorbu oboustranně prospěšných vazeb mezi dodavatelem a odběratelem, jejichž úroveň je zárukou trvalejší schopnosti dodavatele plnit požadavky odběratele. Tyto vazby jsou označovány pojmem partnerství. Partnerství s dodavateli je pracovní vztah mezi odběratelem a dodavatelem budovaný na bázi vzájemné důvěry a přináší oběma obchodním partnerům přidanou hodnotu, která je rovněž přenášena v finálním produktu k konečnému zákazníkovi jak uvádí popis EFQM Modelu Excellence věnovaný této problematice.⁸ ISO věnovaná této problematice požaduje, aby odběratelské organizace řídili své procesy směrem k dodavatelům, své dodavatele hodnotily, vybíraly je podle předem známých kritérií, sdělovaly dodavatelům své požadavky, realizovaly vhodné metody ověřování zda dodávky plní požadavky a o takto nashromážděných aktivitách vedly záznamy dle standardů řízení dokumentace QMS.⁹ Základním strategickým cílem programu partnerství s dodavateli je vytváření neustále se zlepšující soustavy vztahů vzájemné důvěry mezi dodavateli a odběrateli. Od takto specifikovaného základního cíle lze odvodit i cíle dílčí jako je zlepšení pružnosti dodavatele k plnění požadavků odběratele, redukce ztrát způsobených nízkou kvalitou jakosti u dodavatele, zlepšení vzájemné komunikace, snížení celkových nákladů na procesy zásobování u odběratele,

⁷Srov. SYNEK, Miroslav, a kol., *Manažerská ekonomika*. s.207-208

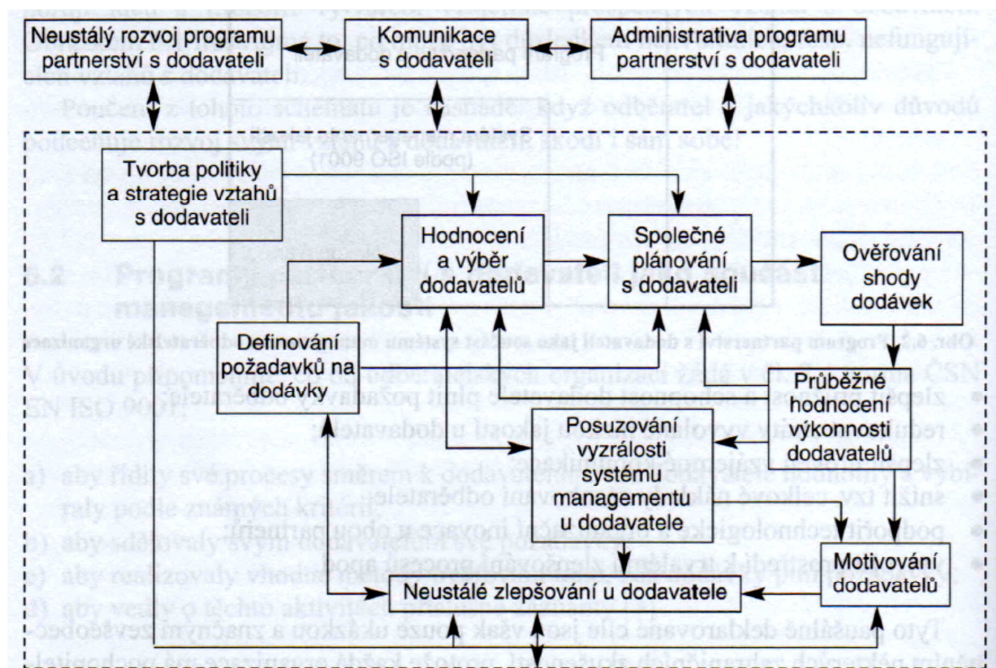
⁸Srov. NENADÁL, Jaroslav. *Management partnerství s dodavateli*. s.24 -25

⁹Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 132

podpora technologických a organizačních inovací u obou partnerů a atd..¹⁰

Management partnerství s dodavateli lze chápat jako množinu vzájemně provázaných procesů a činností, které v standardu obsahují :¹¹

- Tvorbu politiky a strategie vztahů s dodavateli.
- Definování požadavků na dodávky.
- Hodnocení a výběr dodavatelů.
- Společné plánování s dodavateli.
- Posuzování stavu vyzrálosti systému managementu u dodavatele.
- Ověřování shody dodávek.
- Průběžné hodnocení výkonnosti dodavatelů.
- Motivování dodavatelů k neustálému zlepšování.
- Komunikace s dodavateli.
- Administrativa procesů s dodavateli.
- Neustálé zlepšování a rozvoj programu partnerství s dodavateli.



Obr. 4. Základní rámec procesů programu partnerství s dodavateli.

Zdroj : NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 134.

¹⁰Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 132-133

¹¹Srov. tamtéž, s. 133

1.3 Politika a strategie vztahů s dodavateli.

Základním kritériem je rozvíjení vztahů důvěry po opakovaném prokázání spolehlivosti dodavatele, kdy je nutné vybalancovat dvě základní kritéria a to co nejnižší cenu a co nejvyšší kvalitu (neboli dostatečnou úroveň jakosti) za dodávku.¹² Mezi další kritéria patří stanovení strategické významnosti dodavatele s využitím např. Paretovy analýzy nebo ABC metody. Neméně důležité je stanovení zda u daného materiálu využít strategii dodavatelského vějíře nebo služeb jediného dodavatele. U dodavatelů, kteří opakovaně prokazují spolehlivost přistupovat k zavádění individuálních dlouhodobých obchodních vztahů, které mají mít motivační charakter u dodavatelů a pro odběratele budou přínosné v tom, že dodavatel bude v širší míře investovat do zlepšování vlastních procesů a tak i do zvýšení jakosti dodávek. Neméně důležité je stanovení rozsahu technické podpory a sdílení informací s dodavateli, které by mělo odpovídat úrovni vztahů mezi odběratelem a dodavatelem s přijetím společenské odpovědnosti za své podnikání u obou stran nákupního procesu.

1.4 Definování požadavků na dodávky a dodavatele.

Pravomocí odběratele je stanovení požadavků na budoucí dodávky a dodavatele v těchto oblastech. Pro vlastní nakupované produkty nebo služby, procesy a systém managementu u dodavatelů, ostatní služby a činnosti spojené s dodáváním.¹³

- Vytvoření materiálové specifikace s úplnou nomenklaturou měřitelných znaků jakosti s jejich platností v časové ose.
- Pro kontrolu uvedených znaků jakosti mít definované kontrolní postupy dle metodik QMS.
- Volba vhodné metodiky pro posuzování shody s definováním kritérií přijatelnosti dodávek.
- Vytvoření plánu dodání s upřesněním dodávaného množství a očekávaných nákladů, které se vztahují k dané dodávce.
- Požadavky na identifikaci pro účely zpětné sledovanosti dodávek.

Z pohledu služeb a procesů managementu u dodavatelů by neměli chybět požadavky dle standardů QMS, EMS a OHSAS a atd.. Důležitá je i úroveň vstu-

¹²Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 134-135

¹³Srov. tamtéž, s. 136-137

pní kontroly u odběratele a výstupní kontroly u dodavatele, její přejímací plány pokud je využívána statistická přejímka. Požadavky na způsobilost výrobních procesů dodavatele s způsoby jeho zabezpečování požadované úrovně jakosti u jeho subdodavatelů z důvodu eliminace neshod u procesů v celém dodavatelském řetězci. Důležité je stanovení odpovědností za, jednotlivé činnosti při zabezpečování jakosti dodávek, způsobu komunikace s dodavateli, včetně změn obchodních smluv s definicí rozsahu spolupráce v průběhu plnění obchodních smluv.

Z pohledu požadavků na služby a činnosti spojené s dodáváním jako je návaznost na legislativu, rozsah povinně vedených a udržovaných záznamů včetně jejich doby uchovávání. Dalším požadavkem mohou být certifikáty vytvořené třetí stranou, požadavky na obaly, způsob skladování a atd..

Pro specifikování požadavků na dodávky a dodavatele je vhodné využít metody QFD (Quality Function Deployment – dům jakosti), která může využívat i váhové ohodnocení sledovaných parametrů.¹⁴

1.5 Výběr vhodných dodavatelů.

Tato činnost je součástí procesu nakupování a jejím cílem je vybrat z mnoha potencionálních dodavatelů alespoň jednoho, který bude plnit požadovaná kritéria stanovená odběratelem. Vytvoření rámce vlastního hodnocení a výběru dodavatele je plně v kompetenci odběratele, ale ten by měl obsahovat audit u potencionálního dodavatele, ověření jeho spolehlivosti na testovacích dodávkách, analýzu referencí z dodávek jiného materiálu a atd.. Pro výběr vhodných dodavatelů je využíváno vícekriteriální hodnocení s zaváděním vah jednotlivých kritérií, dotazníků a atd.. Metody a metodiky jsou v procesech standardizovány, ale jde o prvotní nastavování kritérií pro danou problematiku, která mají svá žádoucí specifika a z tohoto důvodu jejich standardizované zpracování pomocí znalostní databáze nepřináší tak velkou přidanou hodnotu na rozdíl od standardizovaného zpracovávání již zavedeného a standardně dodávajícího dodavatele.¹⁵

1.6 Společné plánování s dodavateli.

Společné plánování s dodavateli zahrnuje plánování jakosti dodávek, ekonomické plá-

¹⁴Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 136-137

¹⁵Srov. tamtéž, s. 137-138

nování, procesně orientované plánování a plánování manažerských aktivit. Plánování jakosti dodávek je zaměřeno na dosažení konsensu mezi požadavky odběratele a dodavatele v oblasti přijatelné úrovně jakosti. Společné ekonomické plánování je zaměřeno na optimalizaci nákladů dodavatele s omezením nežádoucích výkyvů cen jeho finálního produktu. Společné procesní plánování se zaměřuje na zlepšování a projektování procesů u dodavatele jako jsou podmínky skladování, přepravy, identifikovatelnosti dodávek a atd. s využitím např. metody FMEA. Plánování manažerských aktivit je soustředěno na způsoby vzájemné komunikace, definice rozsahu technické podpory a pomoci, způsoby sdílení dobré praxe a atd..

1.7 Posuzování stavu vyzrálosti systému managementu u dodavatelů.

Toto posuzování je zaměřeno na hodnocení vyzrálosti systému managementu u dodavatelů v průběhu realizovaného obchodního vztahu a ukazuje nám úroveň rozvoje systémů managementu dodavatelské organizace, která nám signalizuje podmínku i záruku dlouhodobější schopnosti daného dodavatele poskytovat výrobky a služby s vysokou jakostí a spolehlivostí dodávek dle uzavřených obchodních podmínek.¹⁶ Tyto činnosti jsou prováděny z důvodu verifikace, že takto vybraní dodavatelé jsou způsobilí pro uzavírání dlouhodobých kontraktů.

1.8 Ověřování shody dodávek.

Protože nežijeme v dokonalém světě, nedá se předpokládat, že by se nikdy v dodávaných výrobcích a službách nevyskytly žádné odchylky od odběratelem specifikovaných požadavků, které označujeme jako neshody. Proces ověřování shody dodávek je validací, která stanoví zda naměřené nebo zjištěné odchylky jsou v akceptovatelném rozsahu specifikovaných požadavků odběratelem nebo jsou mimo tento rozsah a jsou neshodou.¹⁷ Ověřování shody je proces , který nepřidává hodnotu, naopak je poměrně nákladný.

Standardně odběratel po obdržení dodávky provede vstupní kontrolu a to jak množstevní tak jakostní dle předdefinovaných metodik, které jsou součástí řízené dokumentace dané firmy. Jako nejefektivnější se v této oblasti jeví využití systému statistické přejímky, která může velmi zefektivnit finální náklady na

¹⁶Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 140-141

¹⁷Srov. tamtéž, s. 131-143

ověřování shody dodávek s dodržáním požadované úrovně jakosti.

Ve vztazích založených na vzájemné důvěře, je možné přejít k méně finančně náročným způsobům ověřování shody dodávek, kdy je možné fyzickou kontrolu přenést na stranu výstupní kontroly dodavatele a vstupní kontrolu nahradit evidencí certifikátů jakosti dodaných dodavatelem.

Varianta	Činnosti dodavatele	Činnosti odběratele	Míra prevence vůči výskytu a odhalení neshod v dodávkách
1.	Bez výstupní kontroly jakosti	Přijímá vše, 100% kontrola až ve výrobě	Téměř žádná
2.	Bez systému zabezpečování jakosti	100% kontrola na vstupu	Minimální
3.	100% kontrola na výstupu (před expedicí)	100% kontrola na vstupu	Malá
4.	100% kontrola na výstupu	Výběrová kontrola na vstupu	Malá, ale s nižšími náklady odběratele
5.	100% kontrola ve výrobě a výběrová kontrola na výstupu	Výběrová kontrola na vstupu	Střední
6.	SPC ve výrobě, výběrová kontrola na výstupu	Namátková kontrola na vstupu	Poměrně vysoká
7.	SPC ve výrobě, namátková kontrola na výstupu	Namátková kontrola na vstupu	Vysoká zásluhou dlouhodobé a vysoké způsobilosti procesů u dodavatele
8.	SPC ve výrobě bez výstupní kontroly	Přechod na tzv. akceptovanou kontrolu	Maximální

Obr. 5. Varianty ověřování shody dodávek..

Zdroj : NENADÁL, Jaroslav, a kol., *Moderní management jakosti*.s. 142.

2 HODNOCENÍ VÝKONNOSTI DODAVATELŮ

Dle EFQM Modelu Excellence je výkonnost dodavatelů vyjádřena jako jejich schopnost plnit požadavky na dodávky specifikované ve smlouvě o dodávkách, uzavřené mezi odběratelem a dodavatelem.¹⁸ Pravidelně opakovaným procesem hodnocení výkonnosti dodavatelů pak mají odběratelé možnost posuzovat okamžitou schopnost svých dodavatelů dostát svým závazkům, které přijaly a podepsaly v příslušných obchodních kontraktech. Mezi základní prvky hodnocení výkonnosti dodavatelů, na které se obvykle soustředí pozornost odběratelů patří, jakost dodávek, termíny dodávek a náklady spojené s dodávkami.

2.1 Přezkoumání harmonizace strategií cílů – vlastní versus dodavatelé.

Pokud odběratel nemá vyjasněnu vlastní strategii cílů je systematické měření a vyhodnocování výkonnosti dodavatelů vždy problematické, protože měření výkonnosti je vždy měření reality s následnou validací oproti stanoveným hodnotám, které by měli být logicky odvozeny od strategických cílů odběratelů.¹⁹ Protože nevyjasněné strategie cílů můžou následně vést k vysokým nákladům na jakost, nebo k problémům s dodržováním termínů dodávek u odběratele.

2.2 Definice vhodných ukazatelů a metrik.

Pro stanovení vhodných ukazatelů a metrik lze volit variantní přístupy, např. z oblastí výsledků statistické přejímky dodávek, z výdajů na odstraňování neshod v dodávkách, z údajů pro dodržování harmonogramu dodávání, z výsledků procesně vedených auditů, z údajů o rozsahu a efektech procesu zlepšování u dodavatelů a z výsledků měření spokojenosti odběratelů.²⁰

Důležité jsou společné charakteristiky pro metodiky hodnocení výkonnosti dodavatelů jako je jednoduchost aplikace, možnost opakovaného nasazení v krátkých časových intervalech, s nízkou náročností na zdroje a vlastní pracnost, maximální využívání informačních vstupů, které byly získány jako výstupy z jiných procesů, s požadavky na vysokou míru objektivitu získaných dat a výsledků a s provázaností na informační systém.

¹⁸Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 185-188

¹⁹Srov. tamtéž, s. 191

²⁰Srov. tamtéž, s. 189-192

²¹Srov. tamtéž, s. 189-192

2.3 Návrh sběru dat a systému hodnocení dodavatelů.

Po stanovení ukazatelů a metrik pro opakované hodnocení výkonnosti dodavatelů s stanovením jejich případných cílových hodnot je nutné navrhnout systém sběru dat, který bude zaznamenávat veškerá potřebná a relevantní data. Důležité je nastavení způsobu sběru dat tak, aby byl snadno dohledatelný např. podle identifikací zpětné sledovatelnosti a tak přiřaditelné podle materiálu k hodnocenému dodavateli. Do systému sběru dat budou zařazeni pracovníci nejen z oddělení nákupu, oddělení kontroly jakosti (vstupní kontrola, kontrola výrobních procesů, výstupní kontrola), ale také vybraní pracovníci z výrobních středisek, kteří mají za úkol sledování technologické kázně a efektivity výroby (technologové, vedoucí výrobních středisek, vybraní referenti a atd.).

2.4 Implementace systému pro hodnocení dodavatelů.

Při návrhu systému hodnocení dodavatelů je nutné dbát na to, že jde o kvantifikované výsledky, které by měly být stanoveny na základě měření, pokud je to možné. Nelze proto vycházet jen z expertních odhadů, které mohou být velmi subjektivní. Je nutné zachovat atribut jednoduchosti systému s minimální zdrojovou náročností a vysokou opakovatelností jeho využití, ale současně je nutné dosáhnout vysoké objektivity získaných výsledků, které jsou základem systematického hodnocení dodavatelů.²²

Po zpracování finální verze návrhu systému hodnocení výkonnosti dodavatelů je možné zahájit implementační fázi, která může odhalit, že některá data, která jsou důležitá pro výpočet některých ukazatelů výkonnosti dodavatelů, nejsou bohužel dostupná. Proto bude nutné provést korekce návrhu. Neméně důležitou částí implementace tohoto systému bude dostatečné zaškolení pracovníků a to jak v části získávání dat a informací, pro správný chod tohoto systému, tak pracovníků odpovědných za vyhodnocování a následnou komunikaci s zákazníky. Je nutné, aby organizace při implementaci finální podoby systému hodnocení dodavatelů zapracovala veškeré činnosti, které s provozem tohoto systému souvisí do řízené dokumentace organizace dle QMS.

²²Srov. NENADÁL, Jaroslav, a kol., *Moderní management jakosti*. s. 193-194

3 DATABÁZE TYPU ACCESS

Databáze typu Access je relační databáze, která má převážně desktopové využití, protože s nástupem nových databázových technologií jako je SQL atd., již její síťové využití není v praxi příliš rozšířené. Výhodou databází Access, které jsou určené pro souborové aplikace je jejich nízká pořizovací cena, vlastní poměrně jednoduchá správa a nízké hardwarové nároky pro jejich provoz.

Relační databáze typu Access je fyzické provedení implementace schématu dat, které bude v našem případě složené pouze z tabulek případně pohledů a dotazů, které budou využity databázovým systémem pro zajištění validace a integrity dat. Kdy základním krokem při tvorbě schématu relační databáze zůstává tvorba datového modelu, který se zabývá konceptuálním popisem řešené problematiky. V této části definujeme entity s jejich omezení, atributy entit a vztahy mezi entitami. Kdy tyto závislosti mezi entitami jsou popisovány jako relace, které je možné znázornit velmi přehledným způsobem pomocí E/R diagramu, jenž je základem pro převod datového modelu do implementační podoby databázového schématu.

Mezi obecné možnosti relačních databází patří plná kontrola nad definováním údajů, manipulování s údaji a řízení údajů.

Definování údajů je důležité při návrhu databáze, protože nám přesně stanoví jakým způsobem budeme údaje ukládat, jejich datové typy a vzájemné vztahy. Kdy na základě takto vyspecifikovaných požadavků definujeme, jak jsou data při ukládání do databáze formátována a validována.

Manipulace s údaji je neméně důležitá, protože je žádoucí nashromážděná data v databázi řadit, filtrovat a vybírat jen její části pro potřeby aplikace, která pracuje nad databázovou strukturou. Mezi standardní možnosti patří spojování dat do různých pohledů a zobrazovat tak více dat z různých tabulek, přidat, editovat, odstranit, přesouvat data nebo vytvořit novou tabulku jako rozšíření databázové struktury nebo jen jako dočasnou tabulku a to vše podle požadavků dané programové aplikace.

Řízení dat v databázových systémech je důležitou činností, která nám stanoví pomocí uživatelských oprávnění, který uživatel může konkrétní data, zobrazovat, editovat, rušit a stanoví přesná pravidla při víceuživatelském režimu s strukturami databáze.

3.1 Datové typy .

Správné navrhnutí datových typů je velmi důležité při vytváření tabulek, protože tak můžeme velmi efektivně ošetřit a zefektivnit validační procesy používané aplikací, která s danou databází pracuje, při ukládání dat do databázových struktur a zjednodušit tak některé další, převážně výpočetní operace, které provádí aplikace nad databázovou strukturou. Neméně důležité je správné navržení datových typů pro primární a cizí klíče tabulek dané databáze.

3.2 Návrh tabulek databáze.

Hlavním cílem při návrhu databáze je vytvořit pouze potřebné tabulky, jejich obsah a vzájemné propojení, které odpovídá všem požadavkům zadavatele databáze. Tabulky jsou dle terminologie používané u MS-Access tvořeny poli (sloupci) a v řádcích jsou evidované záznamy, kdy záznam představuje konkrétní údaje.²³ Při tvorbě tabulek je nutno postupovat dle teorie relačních databází a vytvářet tabulky podle principů normalizace. Kdy normalizace databáze je proces, který databázi zjednodušuje, odstraňuje redundance a zvyšuje efektivitu při práci s tabulkami databáze.²⁴

První pravidlo normalizace (Jedinečnost polí) je splněno pouze jsou-li všechny atributy atomické (dále již nedělitelné). Pro atributy platí, že nesmí obsahovat více druhů dat, musí obsahovat pro každý záznam skalární hodnotu, kdy nesmí být relací. V databázi typu Access to znamená, že každé pole má představovat jedinečný typ informace, proto je nutné odstranit veškeré údaje generující sloučené údaje v jednom poli a opakující se údaje.

Druhé pravidlo normalizace (Primární klíč) může být splněno v případě, že je splněna první forma normalizace a každý sloupec tabulky mimo primární klíč je zcela závislý na primárním klíči (i pro více primárních klíčů). V databázi typu Access musí mít, každá tabulka jednoznačný identifikátor neboli primární klíč, jenž může být vytvořen z jednoho nebo více polí tabulky. Význam druhého pravidla normalizace je v tom, že každý záznam v tabulce musí být jednoznačně identifikovatelný. Databáze typu Access požadují vytvoření primárního klíče, standardně jako automatické číslo, ale lze jej zadat manuálně i pro jiný datový typ.

²³Srov. BELKO, Peter. *Microsoft Access 2013 Podrobná uživatelská příručka*. s. 34

²⁴Srov. RIORDAN, Rebecca M.. *Vytváříme relační databázové aplikace*. s. 23-25

Třetí pravidlo normalizace (Funkcionální závislost) platí v případě, že je tabulka v druhé normální formě a současně neexistují závislosti neklíčových sloupců tabulky. V databázi Access pro každou jedinečnou hodnotu primárního klíče musí datové sloupce tabulky popisovat plně předmět tabulky. Neboli po vydefinování primárního klíče je nutné zjistit, zda je vztažen ke všem údajům, které jsou předmětem tabulky, pokud ne je nutné definovat jiný primární klíč, který tuto podmínku splňuje nebo přepracovat tabulku. Primární klíč tabulky musí jednoznačně identifikovat předmět dané tabulky.

Čtvrté pravidlo normalizace (Nezávislost polí) nastává, pokud platí třetí forma normalizace, v které popisujeme jen jedinou souvislost nebo fakt. V databázi Access toto pravidlo platí pokud je nutné realizovat změnu údajů v libovolném poli tabulky mimo pole primárního klíče bez jakéhokoliv ovlivnění údajů v jiném poli tabulky. Čtvrté pravidlo normalizace ověřuje zda mohou vzniknout problémy při změnách údajů v tabulkách dané databáze.

3.3 Vztahy mezi tabulkami databáze (Relace).

Správně navržená databázová struktura je základem výkonnosti a efektivnosti každého databázového systému. Proto dalším krokem, který následuje po normalizaci tabulek a nastavení datových typů tabulek databáze je definování vhodných vztahů mezi tabulkami. Pro správnou funkci vztahů mezi tabulkami, potřebujeme primární klíč primární tabulky a cizí klíč v jiné tabulce (sekundární), kdy oba dva klíče musí být stejného datového typu. V databázích jsou standardně využívány tyto vztahy mezi záznamy tabulek M:N, 1:1, a 1:N, které jsou nazývány relace.

Relace jedna ku jedné (1:1, one-to-one) nám vytváří vztah, kdy je každý záznam z primární tabulky svázán s jedním záznamem z sekundární tabulky pomocí jedinečných klíčů v obou tabulkách (primární a sekundární klíč).²⁵

Relace jedna ku více (1:N, many-to-one) nám vytváří vztah, kdy každý záznam primární tabulky je možno svázat s jedním nebo více řádky sekundární tabulky pomocí jedinečných klíčů v obou tabulkách.²⁶

Relace více ku více (M:N, many-to-many) nám vytváří vztah, kdy více řádků primární tabulky může být svázáno s více řádky sekundární tabulky po-

²⁵Srov. BELKO, Peter. Microsoft Access 2013 Podrobná uživatelská příručka. s. 38-39

²⁶Srov. tamtéž.

mocí jedinečných klíčů v obou tabulkách. V praxi je toto řešení realizováno pomocí vazební (propojovací) tabulky kde je vztah N:M rozložen na dva vztahy 1:N:M:1.²⁷

Dalším druhem relace je **unární relace**, která vytváří relaci pouze v jedné tabulce a tak slouží k vytváření hierarchických vztahů mezi záznamy tabulky.

3.4 Databázové rozhraní.

Problematiku získání dat z databáze daného typu lze obecně vyřešit třemi způsoby a to pomocí rozhraní API dané databázové platformy, použitím vhodného obecného databázového rozhraní nebo použitím nativního databázového rozhraní.

Rozhraní API (aplikační programové rozhraní) je z pohledu historie nejstarší způsob přístupu k datům dané databáze, který nikdy nebyl plně standardizován a i proto, je v současnosti využíván jen velmi ojedinele.

Obecná databázová rozhraní dokáží pracovat s více databázovými platformami (i současně), pro které poskytují společnou obvykle jen základní množinu služeb. Nevýhodou obecných rozhraní bývá často nižší výkon a dostupnost na straně klienta pouze pro určitý operační systém. Patří sem ODBC, OLE DB, ADO, BDE, JCBD, JDO a atd..

Nativní databázová rozhraní mají širší množinu služeb a jsou vždy přizpůsobeny konkrétnímu databázovému stroji a existují pro daný vývojový nástroj. Výhodou je vyšší výkon oproti obecným rozhraním a využitím všech možností databázového stroje, ale nevýhodou je jejich neuniverzálnost. Typickými zástupci jsou ADO.NET, dbExpress (efektivní nástupce BDE od Borlandu) nebo ObjectSpaces pro prostředí Javy.

Vzhledem k rozsahu této práce se již budu věnovat pouze rozhraním, které budou využity, pro získání dat z databází typu Access. Velmi využívané obecné databázové rozhraní ODBC (Open Database Connectivity) jsem se rozhodl nevyužít, protože pro správnou funkci řešené aplikace by musel být nainstalován příslušný ODBC driver a to by využívání aplikace s lokální databází zbytečně komplikovalo.

²⁷Srov. BELKO, Peter. Microsoft Access 2013 Podrobná uživatelská příručka. s. 38-39

3.4.1 Databázové rozhraní OLE DB .

OLE DB je obecné databázové rozhraní pro přístup k datům od firmy Microsoft, které je navrženo pro práci s relačními i nerelačními zdroji dat. Kdy základem OLE DB je komponentová technologie COM (Component Object Model) a její definice otevřených kolekcí rozhraní, které zapouzdřují databázové funkce s využitím vrstveného modelu architektury OLE DB. Kdy základní popis architektury OLE DB je zaměřen na tři části a to poskytovatele dat, konzumenty dat a služby.

OLE DB poskytovatel dat, přistupuje přímo fyzicky k datům nebo k dalšímu rozhraní, které přistupuje fyzicky k datům. Obvyklý poskytovatel je součástí databázové technologie a funkcionality dané databáze využívá pomocí rozhraní COM, které je součástí OLE DB.

OLE DB konzument dat, je obvykle aplikace, která využívá OLE DB rozhraní, kdy aplikace konzumováním OLE DB rozhraní dodávané OLE DB poskytovatelem, pracuje s daty nebo je spravuje a klade datové požadavky.

OLE DB služba je hybrid, který na jedné straně OLE DB rozhraní konzumuje a na druhé straně OLE DB rozhraní poskytuje. Kdy nám tyto služby umožní provést implementaci rozšířených vlastností jako společnou indexaci nebo v jednom čase využít více datových zdrojů, kdy samotná aplikace pracuje s jedním poskytovatelem = vytvoření OLE DB služby.

Model OLE DB je tvořen pomocí základní struktury abstraktních komponent, které mají sady rozhraní jako např. DataSource (spojení s datovým zdrojem), Session (transakční zpracování v rámci jednoho klienta), Command (vykoná databázové příkazy např. v SQL) nebo Rowset, který zpřístupňuje data v formě tabulek.

Programování v OLE DB není příliš vodné pro přímé volání z jazyků jako je Visual Basic, protože OLE DB není v základu objektové, ale existuje jeho objektová nadstavba ADO.NET .

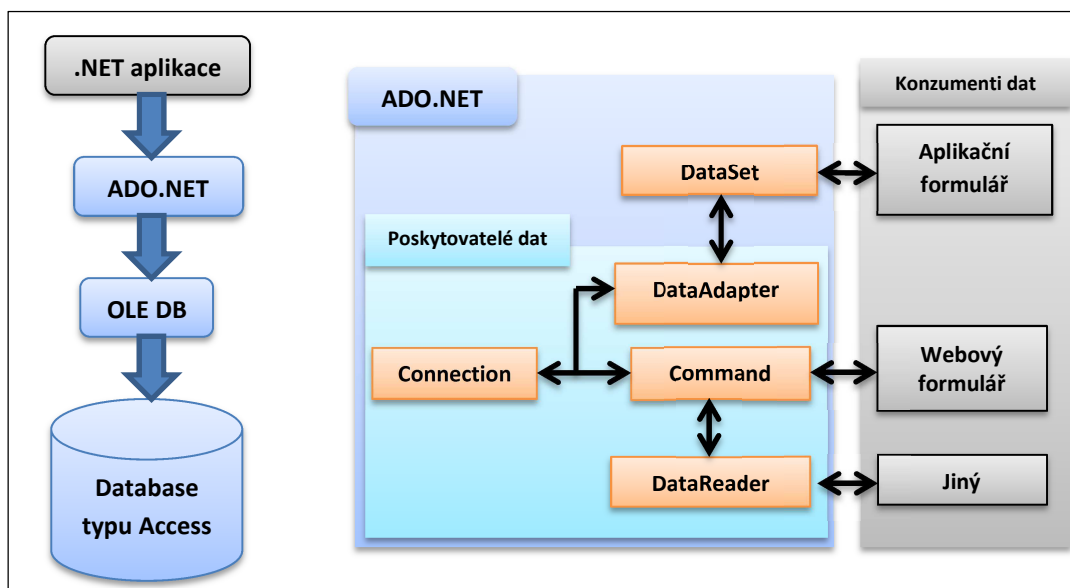
3.4.2 Databázové rozhraní ADO.NET.

ADO.NET patří mezi nativní databázová rozhraní, je součástí vývojové platformy .NET Framework pro přímý přístup k datům a umí pracovat s relačními i nerelačními zdroji dat. ADO.NET využívá i metavrstvu OLE DB pro zajištění zpětné datové kompatibility, kdy tuto vlastnost využíváme při připojení data-

bází typu Access k objektově orientovanému prostředí v programovacím jazyku Visual Basic .NET.

ADO.NET jako součást .NET Framework je složeno z množiny tříd, které jsou v vzájemné interakci. Tyto třídy můžeme rozdělit na dvě základní skupiny. První skupinu vytváří poskytovatelé dat (Data Provides), kteří mají za úkol komunikovat s datovým zdrojem (fyzické úložiště). Druhá skupina je vytvářena sadou dat (DataSet) reprezentující skutečná data. Obě skupiny jsou schopny komunikovat s konzumenty dat. Konzumenta dat zde může představovat libovolná třída např. aplikační formulář vytvořený v Visual Basic .NET a atd..

.NET Framework obsahuje v základní instalaci poskytovatele dat OLE DB metavrstvy, který je zde pouze z důvodu zpětné kompatibility a je také využíván pro připojení k databázím typu Access. Mezi další poskytovatele patří poskytovatel pro MS SQL server nebo Oracle a DB2 a další databáze, které z důvodu rozsahu této práce již nebudu dále rozebírat. Všichni poskytovatelé obsahují stejné objekty, které mohou mít mírně odlišné metody a vlastnosti.



Obr. 6. Možný přístup .NET aplikace k datům a zjednodušený objektový model ADO.NET.

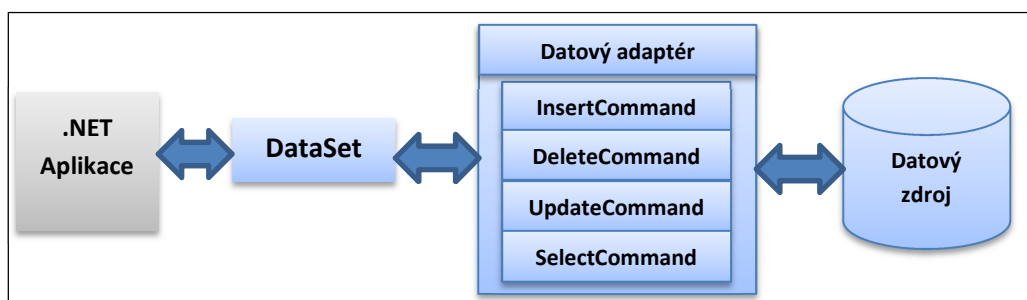
Zdroj : Vlastní tvorba.

Objekt připojení (Connection) je zaměřen na fyzické připojení ke zdroji dat, kdy vlastní volby příslušného zdroje dat jsou závislé na nastavení parametrů tohoto objektu. Může otevírat nebo zavírat toto připojení, měnit zdrojovou databázi a spravovat transakce.

Objekt příkazu (Command) je příkazem SQL nebo odkazem na uloženou proceduru, která se má na datovém zdroji vykonat, např. aktualizování nebo získání dat a atd.. Objekty příkazu je možno nezávisle vytvářet na objektu připojení, kdy tyto objekty pracují také s objekty datového adaptéru.

Objekt čtenář dat (DataReader) je velmi rychlým objektem, který je schopen získat velké množství dat z datového zdroje, ale pouze pro čtení a lze jej procházet pouze směrem dopředu. V paměti se nachází v daném okamžiku vždy pouze jen jeden řádek, kdy tento objekt nelze vytvořit přímým způsobem, ale lze vytvořit voláním metody ExecuteReader objektu příkazu.

Objekt datový adaptér (DataAdapter) je komunikačním mostem mezi objektem připojení a objektem příkazu. Kdy datový adaptér získá data od objektu připojení a předá je do sady dat a následně předá změny dat, pro aktualizaci dat ve zdroji dat. Pro tyto činnosti využívá čtyři objekty příkazu pro výběr, vkládání, mazání a aktualizaci dat. Kdy výběr slouží pro naplnění sady dat a ostatní jsou vykonávány dle požadavků na přenosy změn do zdroje dat.



Obr. 7. Spolupráce mezi sadou dat (DataSet) a datovým adaptérem (DataAdapter).

Zdroj : Vlastní tvorba.

DataSet neboli sada dat je paměťovou prezentací dat, které jsme získaly pomocí datového adaptéru. Sadu dat lze také chápat jako zjednodušenou relační databázi, která je složena s libovolného počtu tabulek a relací mezi nimi. Kdy každá tabulka obsahuje seznam sloupců a řádků s integritními omezeními.

Pomocí kolekce sloupců můžeme nastavovat jak standardní vlastnosti jako název sloupce, jeho datový typ tak i výraz pro dopočítávání hodnot daného sloupce, popřípadě zda bude obsahovat hodnotu NULL a atd..

Řádky tabulek obsahují reálná data, podle definice kolekce daného sloupce. Každý řádek je uchováván v několika verzích hodnot a to jako původní, aktuální

a navržené hodnoty, kdy tato funkcionality zjednodušuje vlastní ukládání dat do datového zdroje i s detekcí na změny původních hodnot jiným uživatelem.

Integritní omezení datové sady mají obdobné funkce jako integritní omezení v relačních databázích a to zachování integrity dat, kontrolu primárních a cizích klíčů, kontrolu duplicity dat a atd..

Sada dat může obsahovat data z různých datových zdrojů. Sada dat je samostatná entita bez jakýchkoliv vazeb na zdroj dat a proto je označována jako odpojená (disconnected). Změny v datové sadě se následně přenesou do zdroje dat pomocí explicitního volání metod datového adaptéru. Obsah sady dat je ukládán do XLM souboru, který nám zajistí zjednodušenou komunikaci se všemi softwary a aplikacemi, které formát XLM podporují.

3.5 Připojení k databázi Access typu *.accdb .

Databázi *.accdb typu Access připojíme k aplikaci vytvořené v vývojovém prostředí Visual Basic .NET pomocí datové sady. Kdy je nutné mít vytvořenou formulářovou aplikaci **Windows Forms Applications** v uvedeném vývojovém prostředí a databázi vytvořenou v aplikaci Access 2013, nebo Office 365, popřípadě Access 2010 a Access 2007.

Po otevření formulářové aplikace Visual Basic .NET v vývojovém prostředí MS Visual Studio 2013 Professional v nabídce menu vybereme položku **Zobrazení (VIEW)** v její nabídce vyhledáme položku **Ostatní okna (Other Windows)**. Po kliknutí na tuto nabídku je zobrazena sub-nabídka, v které vybereme položku **Zdroje dat (Data Sources)**.

V záložce tvorby projektu v Visual Studiu s názvem **Zdroje dat (Data Sources)** klikneme na **Přidat nový zdroj dat (Add New Data Source)** vybereme možnost **Databáze (Database)** a následně vybereme nové připojení databáze, kde nastavíme zdroj dat na ostatní a poskytovatele dat nastavíme na **.NET Framework Data Provider for OLE DB**. V následujícím okně pro takto upřesněný zdroj dat doplníme poskytovatele dat na **Microsoft Office 12.0 Access Database Engine OLE DB Provider**. Dalším krokem je nastavení cesty k databázovému souboru typu Access a nastavení připojovacího řetězce konfiguračního souboru aplikace. Následuje rozbalení tabulek a volby databázových objektů, které budou součástí dané sady dat a bude s nimi možno přes **Edit DataSet with Designer** pracovat (nastavovat indexy, relace a atd.).

4 VISUAL BASIC .NET

Visual Basic .NET (dále jen VB.NET), je plnohodnotným objektově orientovaným programovacím jazykem, který je založen na platformě .NET Framework a má obdobné možnosti jako C#. VB.NET byl vyvinut z programovacího jazyka Visual Basic, který byl původně zaměřen na výuku programování. Visual Basic zaostával za svou konkurencí pro pomalejší běh jeho aplikací a pro omezenější možnosti vývoje tohoto programovacího prostředí, kdy poslední verzí byl databázově zaměřený Visual Basic 6.0. Po zavedení platformy .NET firma Microsoft přišla na trh s první .NET verzí Visual Basic 7 na platformě .NET Framework.

4.1 Charakteristika VB.NET.

VB.NET patří mezi jazyky s virtuálním strojem, kdy virtuální stroj představuje spojení výhodnějších stránek kompilovaných a interpretovaných programovacích jazyků s platformou .NET Framework. Kdy kompilace zdrojového kódu do souborů s příponou EXE, které můžeme spouštět je nevratným procesem.

Nové funkcionality VB.NET vycházejí z využití platformy .NET Framework, která umožňuje vytvářet aplikace bez závislosti na cílové platformě. Např. webové aplikace na platformě ASP.NET nebo mobilní aplikace na platformě Windows Mobile přes rozhraní .NET Compact Framework, které je součástí .NET Framework. Rychlejší běh aplikací je zajištěn díky kompilaci do intermediálního jazyka CIL v .NET Frameworku. Důležitá je i kompletní podpora objektově orientovaného modelu (třídy, dědičnost, rozhraní generické typy). K dispozici je velké množství funkcí v integrovaných knihovnách s implementací pokročilých funkcionalit pro práci s databázemi a podporou integrovaného dotazovacího jazyku LINQ, který dokáže pracovat téměř nad jakýmikoliv daty.

VB.NET zachovává i kompatibilitu s rozhraním COM a WinAPI, ale jeho kód již není zcela zpětně kompatibilní s aplikacemi, které byly vytvořeny ve Visual Basic 6.

Od verze VB.NET 7 je povinná deklarace proměnných, která v dřívějších verzích nebyla povinná a pro proměnné, které neměli přiřazen datový typ pro-

gramové prostředí automaticky přiřazovalo datový typ Variant. Datový typ Variant byl v VB .NET zrušen, protože .NET Framework pro tyto účely využívá datový typ Objekt.

Názvy veškerých slov kódu jsou nezávislé na velikosti písmen a od verze 10 byla zavedena automatická detekce konce řádku kódu.

Stálou výhodou VB .NET je jeho snadnější srozumitelnost pro začínající programátory, kteří mají zájem vyvíjet aplikace pomocí objektově orientovaného vývojového prostředí, který podporuje třídy, dědičnost, rozhraní a generické datové typy.

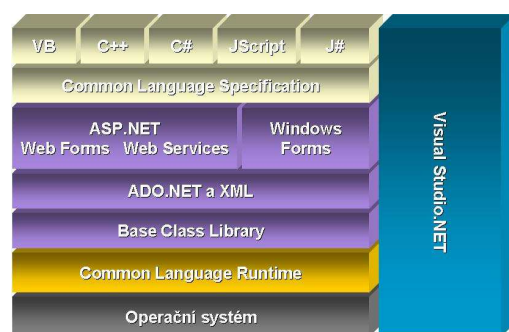
4.2 .NET Framework.

.NET je soubor technologií v softwarových produktech vytvářejících standardizovanou platformu, kde je základní komponentou Microsoft .NET Framework. .NET Framework je prostředí, které je důležité pro vlastní běh aplikací, protože nabízí jak spouštěcí rozhraní tak i potřebné programátorské knihovny pro spouštění a testování aplikací a umožňuje tak programátorům vyvíjet .NET aplikace v jakémkoliv programovacím jazyce, jenž podporuje základní vrstvu .NET Frameworku CLR (Common Language Runtime).

4.2.1 Architektura .NET Framework.²⁸

Komponenta .NET Framework zajišťuje pomocí základní vrstvy CLR automatizovanou správu paměti, rušení a vytváření objektů. Spouštění a zastavování vláken kódu, podporu multithreadingu, zabezpečení kódu a oprávnění k provádění daných operací. Linkování potřebných knihoven a komponent do paměti atd.. Architektura .NET Framework je složena s několika vrstev, které zde jen v stručnosti popíši. **Operační systém** je standardně pouze Windows. Funkcionalitu **CLR (Common Language Runtime)** popíši v části věnované kompilaci kódu. **Base Class Libray (BCL)** je knihovna, která obsahuje nejčastěji používané funkce pro práci s souborovým systémem, třídění a atd.. **ADO.NET** knihovna, která zpřístupňuje data pomocí formátu XML. **ASP.NET Web Forms Web Services** je knihovna pro vývoj webových aplikací a služeb. **Windows Forms** je knihovna, která je využívána pro desktop aplikace.

²⁸Srov. RICHTER, Jeffrey. *.NET Framework programování aplikací*. s. 30-39



Obr. 8. Architektura .NET Framework.

Zdroj : <http://www.cs.vsb.cz/behalek/vyuka/pcsharp/text/ch01s01.html>

Common Language Specification (CLS) definuje sadu základních vlastností, které musí mít programovací jazyky využívající .NET platformu. V nejvyšší vrstvě této architektury jsou již zmíněné programovací jazyky používané pro vývoj .NET aplikací.

Důležitou částí této architektury je vývojové prostředí od firmy Microsoft **Visual Studio .NET** (např. Visual Studio 2013), které dokáže operovat s všemi vrstvami .NET Framework a je tak nejkomplexnějším vývojovým nástrojem dané platformy.

4.2.2 Kompilace .NET aplikací (VB .NET).²⁹

Kompilace na této platformě neproběhne přímo do strojového kódu, ale do mezikódu označovaného CIL (Common Intermediate Language), který má strojový (binární) kód s jednodušší instrukční sadou přímo podporující objektově orientované programování. Kdy je tento mezikód zabalen do souboru EXE. Tento mezikód je následně při spuštění díky své relativní jednoduchosti rychle interpretovatelný virtuálním strojem neboli interpretem v našem případě jde o CLR (Common Language Runtime), který vytvoří pomocí JIT (just-in-time) kompilátoru strojový kód. Takto vytvořený strojový kód je maximálně optimalizován pro danou hardwarovou konfiguraci (procesor).

²⁹Srov. RICHTER, Jeffrey. *.NET Framework programování aplikací*. s. 38-43

5 METODIKA NÁVRHU APLIKACE

Vzhledem k rozsahu této práce se v metodice návrhu aplikace zaměřím jen na velmi obecný popis procesu vývoje softwaru metodikou Unified Process (dále jen UP), která se zabývá definicí otázek, kdo, co, kdy a jak, při vývoji software s pomocí unifikovaného modelovacího jazyka UML (dále je UML). Jazyk UML patří mezi otevřené průmyslové standardy určené pro vizuální modelování, který je schválený sdružením OMG (Object Management Group), ale není metodikou na rozdíl od UP, který metodikou je a jazyk UML velmi dobře doplňuje.

Modely v jazyku UML jsou zaměřeny na statickou strukturu a dynamické chování, které popisuje pomocí strukturní abstrakce modelu, relace pro propojení a diagramy, které vytvářejí zajímavé pohledy na model. UML je založen na pohledu 4+1 do kterého patří logický pohled, pohled procesů, pohled implementace a pohled nasazení, kdy jsou všechny tyto pohledy sjednoceny do pohledu případů užití, který je popisem požadavků uživatele.³⁰

Pro každý nový projekt je nutné vytvořit novou instanci metodiky UP, protože do ní musíme zapracovat mimo jiné i vnitropodnikové standardy. Software je v metodice UP vytvářen na rozdíl vodopádového modelu tvorby software pomocí iterací, kdy každá iterace je samostatný projekt který obsahuje tyto fáze - specifikaci požadavků, analýzu, implementaci, testování a nasazení software. Tyto iterace probíhají, až do finální podoby vznikajícího software.

5.1 Specifikace požadavků.

Většina práce, která je spojená s definicí a specifikací požadavků dle metodiky UP je provedena v fázích zahájení a rozpracování projektu. Specifikaci softwarových požadavků vyjadřujeme pomocí metamodelu, který obsahuje model požadavků (requirements model) a model případů (use case model). Kdy nám tyto modely umožní zaznamenat požadavky na aplikaci. Model požadavků je složen z funkčních požadavků, které určují, co by měla aplikace dělat a z nefunkčních požadavků, které specifikují vlastnosti nebo podmínky omezení aplikace. Model případů užití je složen z balíčků užití, které obsahují případy užití (specifikace funkce aplikace), aktéry (externí role s přímou interakcí s aplikací) a relace.

³⁰Srov. ARLOW, Jim, a NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací*. s. 28-69

Požadavky uspořádáme do taxonomie neboli do hierarchie požadavků, kdy dokážeme efektivněji využívat takto vytvořené kategorie požadavků (systematika požadavků). Pro vyhledávání požadavků můžeme využít uživatele systému a pohovory s nimi, dotazníky a výsledky dílen požadavků.

5.2 Analýza.

Analýza zachycuje většinu aktivit, které se týkají tvorby modelů a požadovaného chování dané aplikace. Analýza obvykle probíhá v obdobném čase, kdy se specifikují požadavky. Záměrem analýzy je tvorba analytického modelu, který nám popisuje, co musí aplikace udělat, ale neřeší přitom způsob jak to aplikace udělá, kdy tuto část řeší návrh. Vlastní hranice mezi analýzou a návrhem závisí vždy na autorovi aplikace.

Mezi aktivity analýzy v UP patří architektonická analýza, analýza případů užití, analýzu tříd a analýza balíčků. Kdy výstupem analýz případu užití jsou analytické třídy z kterých je následně vytvořen analytický model a realizace případů užití. Podle metodiky RUP (komerční varianta UP od IBM) je nutné se v analýze zaměřit na tři základní třídy a to třídu, která je prostředníkem mezi systémem a jeho prostředím (boundary), třída jenž zapouzdřuje chování pro daný případ užití (control) a třída, která je určena pro modelování persistentních informací o něčem (entity).

Pro tvorbu objektově orientovaného systému je nutné samostatné objekty propojit a vytvořit tak vztah mezi objekty a to specifikaci vztahů mezi třídami objektů, které označujeme jako asociace. Tyto vztahy jsou označovány pojmem relace, kdy relace jsou popisovány v jazyce UML jako sémantické vazby mezi modelovanými prvky, které spojují předměty a abstrakce.

Analýza balíčků je metodou stavebních bloků jazyka UML složená s předmětů a abstrakcí, relací a diagramů, kdy každý balíček má vlastní jmenný prostor v kterém musí být všechny názvy jedinečné.³¹ V jazyku UML je každý modelovaný prvek aplikace vlastněn právě jedním balíčkem, kdy balíčky vytvářejí hierarchickou stromovou strukturu s kořenovým balíčkem.

Pro modelování veškerých typů procesů v navrhované aplikaci využíváme diagramy aktivit, které patří mezi objektově orientované vývojové digramy.

³¹Srov. ARLOW, Jim, a NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací*. s. 232-244

5.3 Návrh a implementace

Návrhový model lze obecně považovat za upřesnění a rozpracování analytického modelu o konkrétní technická řešení. Návrhový model musí být úplnější s obsahem implementačních detailů, kdy např. analytický balíček může být v návrhovém modelu rozčleněn do několika podsystémů.

Návrhové modely jsou obvykle složeny z návrhových podsystémů, návrhových tříd, rozhraní, návrhových realizací případů užití a diagramů nasazení.

Pro návrh aplikace je vhodné využít tyto základní náhledy s UML digramy. Funkční náhled (digram případů užití), logický náhled (digram tříd nebo objektový diagram), dynamický náhled popisující chování (stavový diagram, diagram aktivit a interakční diagramy – sekvenční diagram nebo diagram spolupráce) a implementační náhled (diagram komponent nebo diagram rozmístění).³²

Pracovní postup implementace začíná v fázi rozpracování, ale hlavní důraz pro tuto část vývoje software je kladen ve fázi konstrukce. Implementace je proces, který se zabývá převodem návrhového modelu do spustitelného kódu. Proto jsou procesy implementace velmi závislé na volbě programovacího jazyka a hardwarových prostředků, na kterých budeme aplikaci provozovat. Procesy implementace jsou ukončeny nasazením software, kdy tento proces je specifikován digramy nasazení, které nám upřesní konečnou podobu vytvořeného software pro dané softwarové a hardwarové prostředí konkrétního finálního nasazení.

5.4 Testovací režim.

Cílem testování je ověření kvality vývoje dané aplikace, kdy se obvykle zaměřujeme na funkčnost, spolehlivost, výkonnost a případ užití dané aplikace. Testování je nutné provádět u všech fází vývoje software a ne až v průběhu procesu implementace, kdy je tvorba software téměř dokončena. Vlastní testování probíhá podle různých úrovní testování jako jsou testy jednotek software, integrační testy, testy systému, akceptační testy a atd. Dále mohou být použity různé techniky testování, které obecně členíme na black-box testování white-box testování. Na závěr je provedeno vyhodnocení testů s příslušným reportingem a software může být uvolněn pro běžné využití.

³²Srov. ARLOW, Jim, a NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací*. s. 406-426

6 APLIKACE – PRAKTICKÁ ČÁST

V praktické části jsem se zaměřil na vypracování návrhu problémově orientované databáze, která je ovládána aplikací (vytvořenou v vývojovém prostředí VB .NET) zaměřenou na hodnocení výkonnosti dodavatelů. Kdy se v počáteční fázi na základě analýzy současného stavu dané problematiky v naší firmě zaměřuji na automatizaci vyhodnocování dodávek vstupních materiálů pro výrobní procesy, které jsou validované podle předdefinovaných kritérií hodnocení výkonnosti dodavatelů.

Proto jsem se pokusil na základě požadavků mých kolegů vytvořit návrh aplikace, která bude sloužit k testování možné automatizace vybraných, vhodných, sledovaných procesů a problematik z oblasti hodnocení výkonnosti dodavatelů, kdy výsledky takto provedeného testování budou využity při implementaci této problematiky do podnikového informačního systému.

Pro tvorbu návrhu dané problematiky v této bakalářské práci použiji fiktivní data jak pro dodavatele, materiály, substituční skupiny, dodávky, hodnotící kritéria a atd.. Kdy struktura problémově orientované databáze a funkcionality dané aplikace plně odpovídají implementačnímu záměru daného návrhu.

6.1 Nedostatky současného řešení.

Vyhodnocování výkonnosti dodavatelů je v naší firmě v současnosti řešeno speciální intranetovou aplikací, která nahradila souborový systém v MS-Excel, kdy hlavním přínosem bylo vytvoření standardizované WEB aplikace s využitím veškerých výhod takto zvoleného řešení, jako je například centrální datové úložiště vytvořených hodnocení, standardizovaný reporting nebo standardizované formulářové prostředí dané aplikace a atd..

Bohužel způsob pořizování dat do WEB aplikace je velmi podobný dřívějšímu souborovému systému, protože je založen na vložení finální hodnoty za dané dílčí kritérium do validačního formuláře, který slouží k výpočtu závěrečného hodnocení výkonnosti daného dodavatele za hodnocené období. Uživatel neboli expert na danou problematiku si musí nashromáždit a vyhodnotit dílčí kritéria mimo WEB aplikaci. V IS SAP je nastaveno jen automatizované hodnocení úplnosti a včasnosti dodávky, ale výsledky vytvořených hodnocení nejsou automaticky přenášeny do WEB aplikace.

6.2 Navržené řešení.

Mým cílem je na základě požadavků vytvořit návrh problémově orientované databáze (datový model) s aplikací tak aby pro hodnocení materiálů a jejich substitučních skupin podle dodaných dodávek byla vytvořena standardizovaná automatizovaná metodika validování, která pracuje dle předefinovaných kritérií, jejich vah důležitosti s nastavením limitů pro vybraná kritéria hodnocení.

Mezi důležité charakteristiky návrhu takto koncipované aplikace s problémově orientovanou databází patří, její vlastní jednoduchost, nízká náročnost na zdroje a vlastní pracnost, s maximálním využitím informačních vstupů, které byly získány jako výstupy z jiných podnikových procesů s kladením důrazu na objektivitu takto pořízených dat a jejich provázání na podnikový informační systém (dále jen IS).

Kdy základní struktura problémově orientované databáze vychází z využívaného podnikového IS (IS SAP a několik aplikací využívající datové zdroje SQL serveru) a je tvořena tabulkami : Dodavatel, Materiál, Dodávky, Vstupní kontrola, Výroba, Reklamace u dodavatele. Údaje v uvedených tabulkách bude proto možno pořizovat dvěma způsoby a to přímým pořízením přes formulář aplikace nebo importem přes *.CSV soubory a to pro jejich snadnou implementovatelnost jak na straně aplikace tak na straně podnikového IS.

Výstupy z této části, tedy indexy výkonnosti dodavatelů vztažených k dodávkám budou jedním z kritérií, kterých bude využito při výpočtu celkového indexu výkonnosti dodavatele do jehož výpočtu budou zařazeny další oblasti z managementu partnerství s dodavateli.

6.3 Obecný popis funkcionalit navrhované aplikace.

V této části bakalářské práce se budu věnovat obecnému záměru navrhované aplikace s popisem jejích základních funkcionalit jako zabezpečení dat, příprava datové struktury pro následné dávkové vyhodnocení za předem definované období, návrhu a specifikace hodnotících kritérií pro automatizované validace navrhované aplikace s vytvořením datové báze finálních výsledků a jejich následnou prezentaci pomocí vhodně zvoleného reportingu, kdy vlastní popis jednotlivých funkcionalit se pokusím popsat pomocí jazyka UML a budou k dispozici v příloze této bakalářské práce.

Zabezpečení dat je rozčleněno do dvou úrovní a to pomocí nastavení hesla pro uzamčení databáze typu Access, které uživatel nebude znát, protože tuto činnost bude řídit navrhovaná aplikace. Uživatel bude mít přístup do aplikace dle oprávnění nastavené pro jeho uživatelskou roli na základě zadání svého přihlašovacího jména (Login) a hesla. Roli, přihlašovací jméno a heslo pro první přihlášení nastaví uživateli administrátor aplikace, pokud uživatel nebude mít přiřazenu uživatelskou roli administrátor je oprávněn měnit pouze své uživatelské heslo.

Protože jde o lokální databázi tak jsem pro zjednodušení navrhl tři role uživatelů a to **USER**, **EDITOR** a **ADMINISTRATOR**. Vztah uživatelských rolí a funkcionalit navrhované aplikace popíši v dalších částech této práce.

Část navrhované aplikace, která je určena pro přípravu datové struktury (číselníků a atd.) pro následné validační procesy navrhované aplikace je rozčleněna, dle zdrojů dat do samostatných tabulek v databázi, které jsou zpřístupněny pomocí aplikačních formulářů. Data pořizuje oprávněný zaměstnanec s uživatelskou rolí EDITOR nebo ADMINISTRATOR. Do této části navrhované aplikace je zahrnuta i tvorba metodiky hodnotících kritérií, kterou popíši v další části této práce. Tato část zahrnuje definici období za které bude provedeno hodnocení výkonnosti dodavatelů.

Vlastní automatizovaná validace proběhne podle předdefinované metodiky a hodnotících kritérií, které jsou přiřazeny k vybranému materiálu nebo materiálové substituční skupině. Výsledky dílčích a celkového (dle rozsahu této aplikace) indexu hodnocení výkonnosti dodavatelů jsou uloženy do tabulky Výsledky. Procesy automatizované validace jsou přístupny pouze uživatelským rolím EDITOR a ADMINISTRATOR. Zde jsou výsledky hodnocení výkonnosti dodavatelů připraveny k vizualizaci pomocí předdefinovaných nástrojů reportingu navrhované aplikace.

Předdefinovaný vhodně zvolený reporting je k dispozici pro všechny uživatelské role navrhované aplikace.

Specifické funkcionality jako je správa uživatelů, import dat přes soubory typu CSV a ostatní specializované nastavení jsou k dispozici pouze uživatelské roli ADMINISTRATOR.

6.4 Obecné uživatelské znázornění struktury aplikace.

Obecné uživatelské znázornění struktury navrhované aplikace je určeno pouze pro vytvoření základní představy o aplikaci u běžného uživatele, který s ní bude v budoucnu pracovat a není tedy žádným nástrojem metodiky UP nebo unifikovaného modelovacího jazyka UML.

Hlavní struktura	Sub - struktura	Popis
ČÍSELNÍKY	DODAVATEL MATERIAL MATERIAL_SUBST_SKUP KRITERIA HODNOCENÍ OBDOBÍ HODNOCENÍ	Číselníky jsou rozčleněny pomocí záložek, v kterých jsou nadefinovány aplikační formuláře. Tyto formuláře slouží k vkládání, editací a odstranění záznamů datové struktury aplikace
DATA IS	DODÁVKY VSTUPNÍ KONTROLA VÝROBA REKLAMACE (u dodavatele)	Import datových struktur pomocí souborů typu CSV.
HODNOCENÍ	OBDOBÍ HODNOCENÍ MATERIÁL (dodavatel) MATERIAL_SUBST_SKUP DODAVATEL (materiál)	Po zpracování hodnocení výkonnosti dodavatelů pro stanovené období jsou zde k dispozici předdefinované pohledy na výsledky v záložkách Sub- struktury .
REPORTING	MATERIÁL (dodavatel) MATERIAL_SUBST_SKUP DODAVATEL (materiál)	Předdefinované reporty, které jsou obsahově shodné s předdefinovanými pohledy (formuláře) v záložce HODNOCENI.
IMPORT	DATA IS ČÍSELNÍKY	Import souborů typu CSV pro vybrané datové struktury je metodicky rozčleněn pomocí záložek. DATA IS : <ul style="list-style-type: none"> - Dodávky - Vstupní kontrola - Výroba - Reklamace u dodavatele ČÍSELNÍKY : <ul style="list-style-type: none"> - Dodavatel - Materiál (po importu je nutné pro každý materiál nastavit specifické parametry pro správnou funkci této aplikace)
SERVIS	UŽIVATELÉ SERVIS	Záložka UŽIVATELÉ je určena pro správu uživatelských účtů aplikace a záložka SERVIS je určena pro specifické nastavení aplikace.

Tab. č. 1 Obecné uživatelské znázornění struktury navrhované aplikace.

6.5 Návrh validační metodiky a hodnotících kritérií.

Velmi důležitou částí návrhu aplikace pro hodnocení výkonnosti dodavatelů je správná definice ukazatelů a metrik, která je v tomto návrhu specifikována jako návrh validační metodiky s definicí hodnotících kritérií. Pro správné fungování aplikace, je nutné přesně specifikovat strukturu tabulky pro kritériální hodnoty, která je v přímé souvislosti s algoritmem validační metodiky, kdy jsou výsledky validačního procesu za vyhodnocované období ukládány do tabulky výsledky a jsou identifikovány šarží a materiálovým číslem vstupní suroviny.

Na úvod bych chtěl upozornit, že v naší firmě jsou vstupní suroviny v podnikovém IS identifikovány pod jedinečným materiálovým číslem, které je vztaženo pouze k jednomu dodavateli, a proto tuto podmínku bude návrh aplikace obsahovat. Kdy povolená zaměnitelnost takto definovaných materiálů je vyjádřena pomocí substitučních skupin materiálů. Substituční skupina materiálů je definovaná jedinečným číslem a zahrnuje v sobě veškeré materiály od různých dodavatelů, které můžeme zaměňovat dle platných technologických postupů při výrobě daného produktu v naší firmě.

Zpracování výsledků hodnocení výkonnosti dodavatelů bude po diskuzích s kolegy, kteří jsou v současnosti zodpovědní za expertní vyhodnocování této problematiky v naší firmě realizováno v tomto návrhu dávkovým jednorázovým zpracováním za zvolený časový interval. Kdy jako neoptimálnější časový interval bylo zvoleno kvartální období s ponecháním současné stupnice hodnocení dodavatelů, která bude po počátečním provozu nového zpracování dle potřeby modifikována .

Současná stupnice hodnocení dodavatelů :

- 100 – 90 - Dodavatel typu A = odpovídá veškerým požadavkům
- 89 – 85 - Dodavatel typu B = odpovídá převážně požadavkům
- 84 – 80 - Dodavatel typu C = odpovídá podmíněně požadavkům
- < 80 - Dodavatel typu D = neodpovídá požadavkům

6.5.1 Návrh validační metodiky.

Po nastudování problematiky, která je zaměřena na používané metodiky vyhodnocování výkonnosti dodavatelů jsem po zralé úvaze a konzultaci s kolegy

(kteří by tuto navrhovanou aplikaci v budoucnu měli využívat) zvolil pro hodnocení výkonosti dodavatelů Scoring model, který je vztažen na dodávky.

Pro výpočet Scoring modelu je nutné uchovávat zpětnou evidenci veškerých potřebných dat, které jsou vztaženy k dodávkám za rozsah vyhodnocovaného období. Scoring model je koncipován jako bodový součet všech parciálních indexů sledovaných kritérií pro daný materiál a dodavatele, jejichž vypočtená hodnota je vynásobena vahami důležitosti (které v součtu musí nabýt hodnoty 1) daného kritéria pro danou hodnotící metodiku. Parciální hodnotící indexy a celkový index výkonosti dodavatele může nabývat hodnoty v intervalu od 0 do 100 (<0;100>).

Parciální indexy jsou vytvářeny na základě přímého expertního posouzení nebo automatizovanými výpočty pomocí předdefinovaného algoritmu a hodnotících kritérií. Algoritmus je nastaven podle používaných metodik expertů v naší firmě, kteří se specializují na problematiku hodnocení výkonosti dodavatelů.

Pro takto definovaný automatizovaný výpočet vybraných parciálních indexů jsou využity dva základní způsoby výpočtu a to výpočet pomocí poměru (R = ratio) a výpočet pomocí rozdílu (D = difference). Kdy použití výpočtu pro poměr nebo rozdíl u daných parciálních indexů odpovídá způsobu, vhodnosti výběru a porovnání hodnoceného kritéria.

Pro oba způsoby výpočtu jsem vytvořil jednotnou strukturu tabulky *Kritéria*, kdy rozlišení způsobu výpočtu je zapracováno v algoritmu, který provádí předdefinovaný výpočet parciálního kritéria hodnocení (Ratio = R, Difference = D, Zadání expertem = E) podle zadané hodnoty v atributu CTyp tabulky *Kritéria*.

Tvorba dané datové struktury kritérií není zcela univerzální, ale pro testovací záměr navrhované aplikace je dle mého názoru dostačující a dalším důvodem byl záměr z mé strany pro řešenou problematiku využít více standardizovaných prostředků VB .NET pro práci s daty lokální databáze než zabřednout do tvorby polí a matic, které by byly pro standardní práci s aplikací nutné. Samozřejmě při nasazení této problematiky do podnikového IS bude nutné tuto část více optimalizovat a unifikovat.

Způsob výpočtu pomocí poměru (R = Ratio) :

Atributy tabulky Kriteria	VK1	VK2	VK3	VK4	VK5	VK6	VK7	VK8	VK9
R (Ratio)	P1	P2	P3	P4	c1	c2	c3	c4	
Příklad :	105	110	115	120	2,5	2	1,5	1,25	

Tab. č. 2 Způsob výpočtu pomocí poměru.

Způsob výpočtu pomocí poměru (R=Ratio) je založen na poměru mezi požadovanou hodnotou a skutečnou hodnotou validovaného kritéria a podle vhodnosti datového typu (nevhodné je např. vytvářet poměr mezi datem požadovaného dodání a datem skutečného dodání dodávky zde je výhodnější využít způsob výpočtu pomocí rozdílu (D=Diference) pro tuto metodiku.

- **VK1 – VK4** ... slouží pro definování odchylky kritéria nad 100 % způsobem viz. příklad VK1 = 105 %, VK2=110 % .. atd. .
- **K každé odchylce je zde přiřazen koeficient c1 – c4 s předdefinovaným intervalem možných hodnot <1, 2,5>** . Hodnoty koeficientu jsou uloženy v **VK5 -VK8**. **VK9** je v tomto typu výpočtu nevyužíváno a je proto prázdné. **Implicitní hodnotou pro koeficienty je 1.**
- **Obecně platí :** VK1,VK5 (P1,c1); VK2,VK6 (P2,c2); VK3,VK7 (P3,c3); VK4, VK8 (P4, c4).
- **Kdy pro zadání koeficientu kritéria platí tato kriteriální omezení :**

$$2,5 \geq c1 \geq c2 \geq c3 \geq c4 = 1$$
- **Obecný popis výpočtu parciálního poměrového indexu R(Ratio) s využitím atributů VK1 – VK8 z tabulky Kriteria :**

Výpočet parciálního indexu popisují pomocí účelově vytvořeného pseudo jazyka, který obsahuje obecné příkazy používané v současných vývojových prostředcích.

$x1 =$ např. Objednané množství v ks

$x2 =$ např. Dodané množství v ks

$x3 = (x2/x1) * 100$

$c_x = c1 \dots c4$

$ind =$ parciální index za dané kritérium

V první části se zaměřujeme na stanovení polohy vypočtené hodnoty k struktuře předdefinovaných limitů hodnocení (VK1VK4) :

```

IF x3 > 100 AND x3 <=VK1 THEN
    cx = VK5
ELSE IF x3>VK1 AND x3 <=VK2 THEN
    cx = VK6
ELSE IF x3>VK2 AND x3 <=VK3 THEN
    cx = VK7
ELSE IF x3>VK3 AND x3 <=VK4 THEN
    cx = VK8
ELSE IF x3 > VK4 THEN
    cx=1
END IF

```

V druhé části je vypočtena hodnota parciálního indexu pomocí předdefinovaného koeficientu kritéria. Kdy hodnota koeficientu kritéria je stanovena podle polohy vypočtené hodnoty k struktuře předdefinovaných limitů hodnocení v první části, kdy pro výpočet R3 je využíván koeficient spolehlivosti Cspol.

Kdy koeficient Cspol poníží standardní výpočet parciálního indexu o konstantní hodnotu zadanou v servisní části předpokládané aplikace a je využíván při nedodržení dodaného množství dle objednávky, kdy nesplnění tohoto požadavku je bráno jako velmi závažné, kdy v počáteční fázi navrhujeme jeho hodnotu na **0,15**.

Výpočet R1 :

```

IF (x2/x1) > 1 THEN
    ind =1 - {(x2/x1)-1}/ cx} *100
ELSE
    ind =(x2/x1) * 100
ENDIF

```

Výpočet R2 :

```

IF (x2/x1) > 1 THEN
    ind =1 - {(x2/x1)-1}/(cx *10)} *100
ELSE
    ind =(x2/x1) * 100
ENDIF

```

Výpočet R3 :

```

IF (x2/x1) > 1 THEN
    ind =1 - {(x2/x1)-1}/(cx *10)} *100
ELSE
    ind =(x2/x1) -Cspol * 100
ENDIF

```

- Takto vypočtený výsledek poměrového parciálního indexu je vynásoben vahou důležitosti a uložen do tabulky výsledky k záznamu, který je jednoznačně identifikován šarží (dodávkou) vstupního materiálu.

Způsob výpočtu pomocí rozdílu (D = difference) :

Atributy tabulky Kriteria	VK1	VK2	VK3	VK4	VK5	VK6	VK7	VK8	VK9
D (Diff)	0	25	50	75	100	75	50	25	0
Příklad :	-20	-15	-7	-5	0	2	5	7	10

Tab. č. 3 Způsob výpočtu pomocí rozdílu.

Způsob výpočtu pomocí rozdílu (D=Difference) je založen na rozdílu mezi požadovanou hodnotou a skutečnou hodnotou validovaného kritéria a podle vhodnosti datového typu (tento způsob je vhodný pro např. zjištění odchylky mezi datem požadovaného dodání a datem skutečného dodání dodávky) pro tuto metodiku.

- Způsob výpočtu pomocí rozdílu je určen pro sledování parametrů , které mají nedefinovanou odchylku od požadované hodnoty, kdy **VK5** je odchylkou požadované hodnoty neboli $VK5=0$ a při jejím dosažení je parciální index = 100.
- **VK1 - VK4** .. nám specifikují meze pro zápornou odchylku od požadované hodnoty a to pomocí čtyř kontrolních bodů :
VK1 =0, VK2 = 25, VK3 = 50 a VK4 = 75 bodů pro parciální index.
- **VK6 – VK9** nám specifikují meze pro kladnou odchylku od požadované hodnoty a to pomocí čtyř kontrolních bodů :
VK6 = 75, VK7=50, VK8=25 a VK9 = 0 bodů pro parciální index.
- **Pro výpočet indexů validovaných hodnot, pokud se neshodují s limitními hodnotami kontrolních bodů bude použit velmi jednoduchý přepočet :**

$x1$ = např. Datum dodání dodávky dle objednávky

$x2$ = např. Datum dodání dodávky

ind = parciální index za dané kritérium

Kr = určuje bodový rozsah mezi limity validovaného kritéria. Podle použité metodiky je $Kr =25$

např. $(x2-x1) = 5$

Výpočet parciálního indexu popisují pomocí účelově vytvořeného pseudo jazyka, který obsahuje obecné příkazy používané v současných vývojových prostředcích. Kdy se v této části se zaměřujeme na stanovení polohy vypočtené hodnoty k struktuře předefinovaných limitů hodnocení (VK1VK9) :

Výpočet D (rozdíl) :

```
Kr = 25
x3 = x2-x1
IF x3 = VK5 THEN
    ind = 100
ELSE IF x3 = VK6 OR x3 = VK4 THEN
    ind =75
ELSE IF x3 = VK7 OR x3 = VK3 THEN
    ind =50
ELSE IF x3 = VK8 OR x3 = VK2 THEN
    ind =25
ELSE IF x3 = VK9 OR x3 = VK1 THEN
    ind =0
ELSE IF x3 > VK5 AND x3< VK6 THEN
    ind = [(x3-VK6) *(Kr / VK6)] + VK6
ELSE IF x3 > VK6 AND x3< VK7 THEN
    ind = [(x3-VK7) *(Kr / VK7)] + VK7
ELSE IF x3 > VK7 AND x3< VK8 THEN
    ind = [(x3-VK8) *(Kr / VK8)] + VK8
ELSE IF x3 > VK8 AND x3< VK9 THEN
    ind = [(x3-VK9) *(Kr / VK9)] + VK9
ELSE IF x3 > VK9
    ind =0
ELSE IF x3<VK5 AND x3 >VK4 THEN
    ind = [(x3-VK4) *(Kr / VK4)] + VK4
ELSE IF x3<VK4 AND x3 >VK3 THEN
    ind = [(x3-VK3) *(Kr / VK3)] + VK3
ELSE IF x3<VK3 AND x3 >VK2 THEN
    ind = [(x3-VK2) *(Kr / VK2)] + VK2
ELSE IF x3<VK2 AND x3 >VK1 THEN
    ind = [(x3-VK1) *(Kr / VK1)] + VK1
ELSE IF x3 < VK1
    ind =0
END IF
```

- Takto vypočtený výsledek rozdílového parciálního indexu je vynásoben vahou důležitosti a uložen do tabulky výsledky k záznamu, který je jednoznačně identifikován šarží (dodávkou) vstupního materiálu.

6.5.2 Návrh hodnotících kritérií.

Hodnotící kritéria použitá pro validační procesy této aplikace mají předdefinovanou strukturu a je možné je pomocí zadání hodnoty „0“ do váhy důležitosti z zpracování vyloučit. U hodnotících kritérií lze definovat pouze validační limity (pro způsob výpočtu poměrem i koeficienty z předdefinovaného rozsahu), váhu důležitosti a způsob výpočtu (poměr, rozdíl a expertní posouzení). Kdy způsob výpočtu je implicitně předdefinovaný a nelze uživatelsky měnit.

Datová struktura hodnotících kritérií je navržena tak, že lze vytvořit několik skupin hodnotících kritérií, které se mezi sebou odlišují validačními limity a váhami důležitosti.

Vyspecifikovaný seznam hodnotících kritérií :

Název kritéria hodnocení :	CTyp	Atr.Vysledky
Index úplnosti dodávky :	R3	Vysledky.HK01
Index úplnosti dodávky + Jakost :	R1	Vysledky.HK02
Index ceny dodávky:	R2	Vysledky.HK03
Index včastnosti dodávky :	D	Vysledky.HK04
Index kompletnosti průvodní dokumentace:	D	Vysledky.HK05
Index správnosti průvodní dokumentace:	E	Vysledky.HK06
Index stavu přepravy :	E	Vysledky.HK07
Index stavu ochranných obalů :	R1	Vysledky.HK08
Index znečištění materiálu:	R1	Vysledky.HK09
Index nákladů na kvalitu v vstupní kontrole :	R2	Vysledky.HK10
Index výrobnosti :	R1	Vysledky.HK11
Index nákladů na kvalitu v MEZIOP :	R2	Vysledky.HK12
Index výrobních nákladů dle podnikových THN:	R2	Vysledky.HK13
Index hodnocení výkonnosti dodavatele na dodávku :	S	Vysledky.HK14

Tab. č. 4 Vyspecifikovaný seznam hodnotících kritérií.

Atribut Kriteria.cTyp může nabývat tyto hodnoty :

CTyp	Krátký tetx	Popis
R1	Ratio	Způsob výpočtu pomocí poměru.
R2	Ratio	Způsob výpočtu pomocí poměru.
R3	Ratio	Způsob výpočtu pomocí poměru.
D	Difference	Způsob výpočtu pomocí rozdílu.
E	Expert	Zadání hodnoty parciálního indexu
S	Suma	Součet všech hodnot parciálních kritérií za dodávku.

Tab. č. 5 Povolené hodnoty atributu Kriteria.cTyp.

Stručný popis způsobu výpočtu parciálních kritérií pro hodnocení výkonnosti dodavatele za dodávky :

- **Index úplnosti dodávky** = $[Dodavky.Mnozstvi / Dodavky.ObjMnozstvi] * w$
- **Index úplnosti dodávky + Jakost** = $[(Dodavky.Mnozstvi - Reklamace.MnozReklam - VstupniKontrola.MnozUvVyhr - Dodavky.PoskozMatMnoz) / Dodavky.ObjMnozstvi] * w$
- **Index ceny dodávky** = $[Dodavky.ObjCenaBezDPH / Dodavky.CenaBezDPH] * w$
- **Index včasnosti dodávky** = $[Dodavky.DatPrijem / Dodavky.DatDodani] * w$
- **Index kompletnosti průvodní dokumentace** = $[Dodavky.DatKompletDok / Dodavky.DatDodani] * w$
- **Index správnosti průvodní dokumentace** = hodnota parciálního indexu zadaná expertem
- **Index stavu přepravy** = hodnota parciálního indexu zadaná expertem
- **Index stavu ochranných obalů** = $[(Dodavky.Mnozstvi - Dodavky.PoskozObalMnoz) / Dodavky.Mnozstvi] * w$
- **Index znečištění materiálu** = $[(Dodavky.Mnozstvi - Dodavky.PoskozMatMnoz) / Dodavky.Mnozstvi] * w$
- **Index nákladů na kvalitu v vstupní kontrole** = $[(VstupniKontrol.QNaklady_Plan / VstupniKontrola.QNaklady)] * w$
- **Index výrobnosti** = $[(Vyroba.VyrMnoz - Vyroba.VyrMnozNok - Vyroba.VyrMnozVyr) / Vyroba.VyrMnozPlan] * w$
- **Index nákladů na kvalitu v MEZIOP** = $[(Vyroba.QNaklady_Plan / Vyroba.QNaklady)] * w$
- **Index výrobních nákladů dle podnikových THN** = $[(Vyroba.QNakladyTHN / Vyroba.Naklady)] * w$
- **Index hodnocení výkonnosti dodavatele na dodávku** = $(HK01 + HK02 + + HK13)$

Kde w je váha důležitosti každého parciálního kritéria hodnocení v celkovém hodnocení výkonnosti dodavatele na dodávky dle expertního názoru specialisty v naší firmě, který se zabývá problematikou hodnocení dodavatelů.

6.6 Obecný popis procesů hodnocení výkonnosti dodavatele.

Hodnocení výkonnosti dodavatelů na dodávku (dále jen hodnocení) je prováděno za jednotlivý roční kvartál, kdy je takto zajištěn standardizovaný rozsah dat vymezený časovým intervalem o stejném rozsahu (dle přirozené tolerance kvartálů v kalendářním roce). Proto je nutné před vlastním procesem hodnocení založit v seznamu nové období pro hodnocení, protože do spuštění procesu hodnocení jsou veškerá data přiřazena k obecnému tzv. pracovnímu období, v kterém vytváříme datovou bázi (jako jsou např. informace z nákupního oddělení k jednotlivým dodávkám) a aktualizujeme číselníky navrhované aplikace a atd..

Před spuštěním procesu hodnocení je nezbytně nutné provést několik úkonů pro zajištění integrity dat a to kontrolu úplnosti a správnosti pořízených dat, specifikaci validačních kritérií do skupin kritérií s následným přiřazením takto vydefinované skupiny k materiálu. Proces hodnocení je aktivní, jen pro materiály, které mají v číselníku materiálů navrhované aplikace přiřazenu skupinu kritérií, podle které proběhne validace.

Pokud není požadovaná skupina kritérií založena je ji nutno založit do seznamu. Při založení skupiny kritérií jsou aplikací automatizovaně generovány jednotlivá kritéria pro hodnocení výkonnosti dodavatele dle předdefinovaného rozsahu s předdefinovanou hodnotou atributu CTyp, který určuje typ výpočtu daného kritéria. Odpovědný uživatel neboli expert na danou problematiku, zde vyspecifikuje dle typu zpracování příslušné validační limity předdefinovaných kritérií a stanoví pomocí váhy důležitosti kritéria jeho vliv na celkovou hodnotu indexu hodnocení výkonnosti dodavatele za dodávky.

V případě, že je hodnota váhy důležitosti u daného kritéria rovna nule, je toto kritérium automaticky vyloučeno z následného hodnocení a výsledek parciálního indexu hodnocení za toto kritérium má hodnotu nula (není zahrnuto do výsledného indexu).

Pokud je dle expertního názoru uživatele, který má v kompetenci hodnocení výkonnosti dodavatelů aktualizace datové báze problémově orientované databáze úplná je možné spustit proces vlastního hodnocení.

Základním nosným atributem celého procesu hodnocení je identifikátor dodávky, kterým je na základě zkušeností z naší firmy šarže vstupního materiálu. Každá šarže

vstupního materiálu je generovaná následnou řadou a začíná dvojčíslím platného roku, kdy byla zanesena do systému a tak je zajištěna její jednoznačnost. Vysoká spolehlivost využití identifikátoru vstupních šarží je zajištěna navíc podmínkou, že k číslu šarže vstupního materiálu může být přiřazeno jen jedno číslo materiálu a tak je zajištěn vztah mezi šarží vstupního materiálu, materiálem (s jedinečným identifikátorem) a dodavatelem (jde o jedinečnou vazbu mezi číslem materiálu a dodavatelem, kterou standardně používáme v naší firmě).

Po spuštění algoritmu, který provádí vlastní hodnocení je provedena kontrola, zda existuje záznam specifikující období hodnocení, pokud ne je algoritmus ukončen s hlášením o zjištěné skutečnosti.

V případě, že daný záznam existuje tak algoritmus následně provede kontrolu, zda tabulka *Material* obsahuje alespoň jeden záznam s přiřazenou skupinou kritérií, pokud ne je ukončen s hlášením o zjištěné skutečnosti.

Pokud daný záznam nebo záznamy existují, provede algoritmus následně vlastní vyhodnocení po jednotlivých materiálech a dodávkách (v cyklu od nejnižší hodnoty šarže po nejvyšší), kdy základním atributem pro identifikaci záznamu výsledků v tabulce *Vysledky* je šarže vstupního materiálu z tabulky *Dodavky*, která je zapsána současně s identifikátorem hodnoceného období (viz.výše) do tabulky *Vysledky* a vytvoří nový záznam hodnocení dodavatelů na dodávky (šarže), k kterému jsou vypočteny parciální indexy ostatních hodnocených kritérií (výpočet reálné hodnoty indexu vyhodnocovaného kritéria, které je vynásobeno vahou důležitosti), kdy data z ostatních tabulek problémově orientované databáze jsou relačně navázány na tabulku *Dodavky* přes atribut šarže.

Na závěr každého průběhu vyhodnocení dodávky je vypočten celkový index hodnocení výkonnosti dodavatele za dodávku podle materiálu, který je založen na sumarizaci parciálních indexů jednotlivých kritérií (reálná hodnota indexu vynásobená vahou důležitosti).

V každém prvním průchodu cyklu vyhodnocování dodávek vztažených k danému materiálu, je vytvořena nová varianta vyhodnocovaného záznamu materiálu. Nová varianta vyhodnocovaného materiálu je vytvořena replikou všech polí záznamu z tabulky *Material*, která je doplněna o hodnotu pole *Material.MatVerze*. Materiálové verzi (*Material.MatVerze*) je přiřazena hodnota identifikátoru právě vyhodnocovaného období (*HOdbobi.IdHodnObdob*), která zajistí jednoznačnost vytvořené varianty materiálu. Vytvoření varianty materiálu nám umožní jedinečné nastavení, kritérií a atd.

vyhodnocovaného materiálu v daném hodnoceném období a umožní vytvářet nové konfigurace kritérií a atd., v vyhodnocovacích obdobích, které budou následovat.

Na závěr je vypočten index výkonnosti dodavatele za hodnocené období, který je v současnosti vypočten aritmetickým průměrem z všech záznamů výsledků celkového indexu výkonnosti dodávky pro daný materiál a je následně uložen s identifikací hodnoceného období do tabulky *VysledekMat*. V budoucnu bude tento výpočet doplněn o další řešené metriky, které budou zařazeny do celkového výčtu charakteristik pro výpočet indexu hodnocení výkonnosti dodavatele.

6.7 Reporting aplikace.

Parciální výsledky indexů a celkový index hodnocení výkonnosti dodávek (šarží) jsou uloženy v tabulce *Vysledky* a index hodnocení výkonnosti dodavatele za sledované období jsou uloženy v tabulce *VysledekMat*, odkud jsou snadno dostupné pro prostředky reportingu. Reporting navrhované aplikace je řešen využitím formulářů pomocí ovládacího prvku DataGrid a standardními prostředky VB. NET pro tvorbu reportingu a tiskových sestav. Kdy je možné v dané aplikaci zobrazovat data s využitím relačních vztahů mezi tabulkami v problémově orientované databázi současně k materiálu, zákazníkovi a výsledkům hodnotících kritérií. Vlastní zobrazení je založeno na struktuře tabulce *Vysledky* a *VysledkyMat*, které je konstruována tak aby využití již zmíněných prostředků bylo co nejoptimálnější.

6.8 Elektronická nápověda aplikace

Pro potřeby uživatelů, kteří budou pracovat s navrhovanou aplikací bude důležité po vstupním zaškolení jak s aplikací pracovat, mít k dispozici manuál nebo elektronickou nápovědu. Kdyby se při využívání aplikace dostali do situace, že si budou muset ověřit např. správný postup při zadávání kritérií nebo atd.. Vzhledem k zamýšlenému rozsahu práce jsem se rozhodl vytvořit pouze elektronickou nápovědu, kterou jsem vytvořil v software Help & Manual Professiona Edition, který na tvorbu elektronických manuálů zaměřen. Elektronický manuál je standardně vytvořen v externím souboru *.chm, který je umístěn v adresáři aplikace a je zpřístupněn (spouštěn) přes ovládací prvek hlavního menu aplikace s názvem *Nápověda*.

ZÁVĚR

Cílem mé práce bylo vytvořit návrh lokální firemní databáze typu Access, která je ovládána aplikací přes VB .NET. Pro danou problematiku jsem si po dohodě s vedoucím práce vybral hodnocení výkonnosti dodavatelů, převážně z důvodu, že naše firma v tomto období přechází na novou verzi IS SAP a uvažuje o implementaci této prozatím externě řešené problematiky do tohoto IS. Pro řešení tohoto požadavku jsem po konzultacích s experty v naší firmě vyspecifikoval co nejrealnější kritéria hodnocení a vypracoval návrh metodiky s zohledněním požadavku expertů na automatizaci vyhodnocení dodávek. Kdy expert provádějící kvartální vyhodnocení není na několik dní touto činností paralyzován, ale navrhne a vyspecifikuje validační kritéria a systém automatizovaným zpracováním ušetří jeho drahocenný čas, který může využít pro jiné činnosti.

K řešení návrhu aplikace pro hodnocení výkonnosti dodavatelů na dodávku jsem využil lokální databázi typu Access. Pro daný záměr jsem vytvořil návrh tabulek, který je specifikován jako problémově orientovaná databáze. Pro ovládání databáze byl zvolen vývojový software VB. NET, protože vytvořená aplikace je uložena do spustitelného souboru typu *exe* na rozdíl od aplikace vytvořené v prostředí Access. Je pravdou, že z současného pohledu na vývoj a trendy software je využití VB. NET a databáze Access možno brát jako práci z již zastarávajícími nástroji, ale pro funkci dané aplikace je nutné aby si ji mohl daný expert rychle přesunout na svůj notebook, kde pokud možno bez dalšího doinstalování podpůrného software může s co nejnižšími hardwarovými nároky pracovat. Požadavek expertů zněl využít čas převážně na služebních cestách, a proto je takto koncipované řešení plně vyhovující, protože kvalita vzdáleného připojení (např. vlaky a atd.) na server v naší firmě není ani v zemích Evropy obvykle dostačující.

Po představení návrhu a části vytvořené aplikace expertům, bylo konstatováno, že se vývoj ubírá správným směrem a že přínos aplikace bude možné reálně posoudit, až po ostrém provozu s vyhodnocením zpracovaných dat.

Danou práci beru jako velký přínos pro svůj osobní rozvoj, kdy jsem nastudoval velké množství materiálů, které se týkají problematiky partnerství s dodavateli, QMS, řízení výroby a nákupu, a atd.. Dále jsem nastudoval materiály, které se týkají prostředí Framework a VB .NET, které v budoucnu jistě využiji v své profesní kariéře.

ANOTACE

Příjmení a jméno autora:	Jiří Jemelka
Instituce :	Moravská vysoká škola Olomouc
Název práce :	Návrh lokální databáze typu Access ovládané přes VB.NET.
Title :	A proposal local company controlled by Access type VB.NET.
Vedoucí práce :	PhDr. Jan Lavrinčík, Ph.D.
Počet stran :	62 (včetně 9 stran vevázaných příloh)
Počet příloh:	5
Rok obhajoby :	2015

Klíčová slova v českém jazyce : Řízení dodavatelského řetězce, Relace, Databázové rozhraní OLE DB a ADO.NET, .NET Framework, Visual.Basic.NET, Access, UML

Klíčová slova v anglickém jazyce : Supply chain management, Relation, Database interface OLE.DB and ADO.NET, .NET Framework, Visual Basic.NET Access, UML

Cílem této bakalářské práce je vytvořit návrh lokální firemní databáze typu Access, která je ovládána aplikací zaměřenou na problematiku vyhodnocování dodavatelů dle vybraných metodik managementu partnerství s dodavateli vytvořenou v programovacím prostředí VB.NET.

The aim of this thesis is to design local company Access type database, which is controlled applications focused on the issue of evaluating suppliers according to the selected methodologies management of partnerships with suppliers created in VB.NET programming environment.

SEZNAM LITERATURY :

- ARLOW, Jim, a NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací.*, Dotisk 1. vyd. Brno: Computer Press, a.s., 2011. 567 s. ISBN 978-80-251-1503-9.
- BELKO, Peter. *Microsoft Access 2013 Podrobná uživatelská příručka.*, 1. vyd. Praha: Computer Press, a.s., 2014. 392 s. ISBN 978-80-251-4125-0.
- FIALA, Petr. *Modelování dodavatelských řetězců.*, 1. vyd. Praha: Professional Publishing, 2005. 168 s. ISBN 80-86419-62-2.
- FUCHS, Joachim, a BARCHFELD, Andreas. *Visual Basic Velká kniha řešení.*, 1. vyd. Brno: Computer Press, a.s., 2010. 722 s. ISBN 978-80-251-2212-9.
- KOCICH, Pavel, a SPILKA, Ondřej. *1001 tipů a triků pro Microsoft Visual Basic.*, 1. vyd. Brno: Computer Press, a.s., 2010. 520 s. ISBN 978-80-251-2118-4.
- NENADÁL, Jaroslav, a kol., *Moderní management jakosti. Principy, postupy a metody.*, 1.Vyd.(dotisk) Praha: Management Press, s.r.o., 2011. 377 s. ISBN 978-80-7261-186-7.
- NENADÁL, Jaroslav. *Management partnerství s dodavateli. Nové perspektivy firemního nakupování.*, 1. vyd. Praha: Management Press, s.r.o., 2006. 323 s. ISBN 80-7261-152-6.
- RICHTER, Jeffrey. *.NET Framework programování aplikací.*, 1. vyd. Praha: Grada Publishing a.s., 2003. 552 s. ISBN 80-247-0450-1.
- RIORDAN, Rebecca M.. *Vytváříme relační databázové aplikace.*, 1. vyd. Brno: Computer Press, a.s., 2000. 280 s. ISBN 80-7226-360-9.
- SYNEK, Miroslav, a kol., *Manažerská ekonomika.*, 5. aktualizované a doplněné vydání. Praha: Grada Publishing, a.s., 2011. 480 s. ISBN 978-80-247-3494-1.
- TOMEK, Gustav, a VÁVROVÁ, Věra. *Integrované řízení výroby. Od operativního řízení výroby k dodavatelskému řetězci.*, 1. vyd. Praha: Grada Publishing, a.s., 2014. 368 s. ISBN 978-80-247-4486-5.
- TOMEK, Gustav, a VÁVROVÁ, Věra. *Řízení výroby a nákupu.*, 1. vyd. Praha: Grada Publishing, a.s., 2007. 384 s. ISBN 978-80-247-1479-0.

SEZNAM OBRÁZKŮ :

Obr. 1. Princip zjednodušeného modelu marketingu.	11
Obr. 2. Schéma integrace podnikových systémů a systémů dodavatelských řetězců.	11
Obr. 3. Systém managementu dodavatele jako černá skříňka.	12
Obr. 4. Základní rámec procesů programu partnerství s dodavateli.	13
Obr. 5. Varianty ověřování shody dodávek.	17
Obr. 6. Možný přístup .NET aplikace k datům a zjednodušený datový model ADO.NET.	25
Obr. 7. Spolupráce mezi sadou dat (DataSet) a datovým adaptérem (DataAdapter).	26
Obr. 8. Architektura .NET Framework.....	30

SEZNAM TABULEK :

Tab. č. 1 Obecné uživatelské znázornění struktury navrhované aplikace.	37
Tab. č. 2 Způsob výpočtu pomocí poměru.	40
Tab. č. 3 Způsob výpočtu pomocí rozdílu.	42
Tab. č. 4 Vyspecifikovaný seznam hodnotících kritérií.	44
Tab. č. 5 Povolené hodnoty atributu Kriteria.cTyp.	45

SEZNAM PŘÍLOH :

Příloha č. 1 - Popis vytvořených tabulek.	1
Příloha č. 2 - Příklad užití Use Case	2
Příloha č. 3 - Příklad doménového diagramu tříd.....	3
Příloha č. 4 - Příklad sekvenčního diagramu	4
Příloha č. 5 - Zápis databáze s využitím UML.	5

PŘÍLOHY

Příloha č. 1 - Popis vytvořených tabulek - Stránka 1 z 1.

Dodavatel				
Název	Klíč	Typ dat	N	Popis
IdDodavatel	PK	System.Int32	10	2000000001 – 2999999999
Nazev		System.String	25	Název dodavatele
Adresa		System.String	25	Ulice a popisné číslo
Misto		System.String	25	PSC a město
Stat		System.String	25	Název státu
Status		System.Boolean	1	0 = dodavatel není vázán na materiál a období zpracování – lze jej vymazat

Tab. 1. Dodavatel.

Material				
Název	Klíč	Typ dat	N	Popis
Material	PK	System.Int32	10	1000000001 - 9999999999
MatNaz		System.String	25	Název materiálu
IdDodavatel	FK	System.Int32	10	Vazba na tabulku dodavatel.
IdSkupKrit	FK	System.Int16	7	Vazba na tabulku skupina kritérii.
MatVerze	FK	System.Int16	7	MatVerze = Hobdobi.IdHodnObdob po zpracování hodnotícího období

Tab. 2. Material.

SubSkupMat				
Název	Klíč	Typ dat	N	Popis
IdSskupMat	PK	System.Int16	7	Generovaná následná celočíselná řada.
SskupNaz		System.String	25	Název substituční skupiny materiálů.
SskupPozn		System.String	25	Popis substituční skupiny materiálů.
Material	FK	System.Int32	10	1000000001 - 9999999999
MatVerze	FK	System.Int16	7	Vazba na tabulku Material.

Tab. 3. Substituční skupina.

HObdobi				
Název	Klíč	Typ dat	N	Popis
IdHodnObdob	PK	System.Int16	7	Generovaná následná celočíselná řada.
NazevHodObdob		System.String	7	Q1.2015
HodObdStatus		System.Boolean	1	True = Oddobí , které je aktivní pro vyhodnocování. False = Archivní období, již vyhodnocené.

Tab. 4. Hodnocené období.

Reklamac (u dodavatele)				
Název	Klíč	Typ dat	N	Popis
idRekl		System.Int32	7	
Sarze	PK	System.Int32	7	Vazba na tabulku Dodavka.
RNaklady		System.Double	9,3	Náklady na reklamaci včetně režijních.
MnozReklam		System.Double	9,3	Reklamované množství.

Tab. 5. Reklamacie u dodavatele.

Příloha č. 1 - Popis vytvořených tabulek - Stránka 1 z 2.

Výroba				
Název	Klíč	Typ dat	N	Popis
VyrBlok		System.Int32	7	Výrobní blok nebo výrobní šarže dle povahy výroby.
Sarze	PK	System.Int32	7	Vazba na tabulku Dodavka.
VyrJednotka		System.String	10	Vyráběná jednotka.
VyrMnozPlan		System.Double	9,3	Sumarizace veškerých záznamů za VyrSarze, které mají vazbu na materiálu a šarži z tabulky Dodavka.
VyrMnoz		System.Double	9,3	
VyrMnozNok		System.Double	9,3	
VyrMnozVyr		System.Double	9,3	
QNaklady_Plan		System.Double	9,3	
QNaklady		System.Double	9,3	
NakladyTHN		System.Double	9,3	
Naklady		System.Double	9,3	

Tab. 6. Výroba.

Výsledky				
Název	Klíč	Typ dat	N	Popis
Sarze	PK	System.Int32	7	Vazba na tabulku Dodavka.
IdSkupKrit	FK	System.Int16	7	Vazba na tabulku kritérií.
IdHodnObdob	FK	System.Int16	7	Vazba na tabulku Hodnotící období :
HK01		System.Double	3,2	Index úplnosti dodávky :
HK02		System.Double	3,2	Index úplnosti dodávky + Jakost :
HK03		System.Double	3,2	Index ceny dodávky:
HK04		System.Double	3,2	Index včasnosti dodávky :
HK05		System.Double	3,2	Index kompletnosti průvodní dokumentace:
HK06		System.Double	3,2	Index správnosti průvodní dokumentace:
HK07		System.Double	3,2	Index stavu přepravy :
HK08		System.Double	3,2	Index stavu ochranných obalů :
HK09		System.Double	3,2	Index znečištění materiálu:
HK10		System.Double	3,2	Index nákladů na kvalitu v vstupní kontrole :
HK11		System.Double	3,2	Index výrobnosti :
HK12		System.Double	3,2	Index nákladů na kvalitu v MEZIOP :
HK13		System.Double	3,2	Index výrobních nákladů dle podnikových THN:
HK14		System.Double	3,2	Index hodnocení výkonnosti dodavatele na dodávku :

Tab. 7. Výsledky.

Příloha č. 1 - Popis vytvořených tabulek - Stránka 1 z 3.

Dodavky				
Název	Klíč	Typ dat	N	Popis
Sarze	PK	System.Int32	7	RR00001 – RR99999; RR = kalendářní rok (2015 =1500001)
Material	FK	System.Int32	10	Vazba na tabulku Material.
IdObjednavka		System.Int32	10	4500000001 - 4599999999
DodaciList		System.String	15	Identifikace dodacího listu dodavatele.
DatumDodani		System.DateTime		Smluvně stanovené datum pro dodání.
ObjMnozsvi		System.Double	9,3	Objednané množství materiálu.
Jednotka		System.String	10	Jednotka dodávaného materiálu.
PovTolMnoz		System.Double	9,3	Povolená tolerance množství dodávky.
ObjCenaBezDPH		System.Double	9,3	Cena bez DPH uvedená v objednávce.
ObjTyp		System.Int16	1	Typ objednávky : 1 = standard, 2 = mimořádná dodávka
Mnozsvi		System.Double	9,3	Množství dodaného materiálu.
CenaBezDPH		System.Double	9,3	Cena dodaného materiálu bez DPH.
SazbaDPH		System.Double	2,2	Sazba DPH v %.
DatPrijem		System.DateTime		Datum příjmu (zaskladnění) materiálu.
TypPoriz		System.Boolean	1	Způsob pořízení : 1 = přes formulář 0 = importem *. CSV.
DatImportu		System.DateTime		Datum a čas provedeného importu.
KompletDok		System.Int16	2	Kontrola kompletnosti průvodní dokumentace.
DatKompletDok		System.DateTime		Datum od kdy je průvodní dokumentace kompletní.
StavPrepravy		System.Int16	3	Kontrola stavu přepravního prostředku. (0-100)
PoskozObalMnoz		System.Double	9,3	Poškozené množství jen na obalu.
PoskozMatMnoz		System.Double	9,3	Poškozený materiál přepravou.

Tab. 8. Dodávky.

VstupniKontrola				
Název	Klíč	Typ dat	N	Popis
Sarze	PK	System.Int32	7	Vazba na tabulku Dodavka.
QNaklady_Plan		System.Double	9,3	Plánované náklady vstupní kontrolu dle QMS. (odběry, analýzy a atd.)
QNaklady		System.Double	12,3	Skutečné náklady.
MnozUvVyhr		System.Double	12,3	Uvolněné množství s výhradou (neshodou)

Tab. 9. Data z vstupní kontroly.

VysledekMat				
Název	Klíč	Typ dat	N	Popis
Material	PK	System.Int32	10	1000000001 - 9999999999
IdHodnObdob	FK	System.Int16	7	Generovaná následná celočíselná řada.
IndexDod		System.Double	3,2	Index hodnocení výkonosti dodavatele za hodnocené období :

Tab. 10. Výsledky na materiál a hodnocené období.

Příloha č. 1 - Popis vytvořených tabulek - Stránka 1 z 4.

Kriteria				
Název	Klíč	Typ dat	N	Popis
IdSkupKrit	PK	System.Int16	7	
IdKrit	FK	System.Int16	7	
KritNazev		System.String	20	
Vaha		System.Double	3,2	
CTyp		System.String	1	R = Ratio - způsob výpočtu poměrem D = Difference - způsob výpočtu rozdílem E= Expert - zadání hodnoty expertem S = Summary - suma HK01 HK13
VK1		System.Double	9,3	Kritéria dle zvolené metodiky hodnocení.
VK2		System.Double	9,3	
VK3		System.Double	9,3	
VK4		System.Double	9,3	
VK5		System.Double	9,3	
VK6		System.Double	9,3	
VK7		System.Double	9,3	
VK8		System.Double	9,3	
VK9		System.Double	9,3	

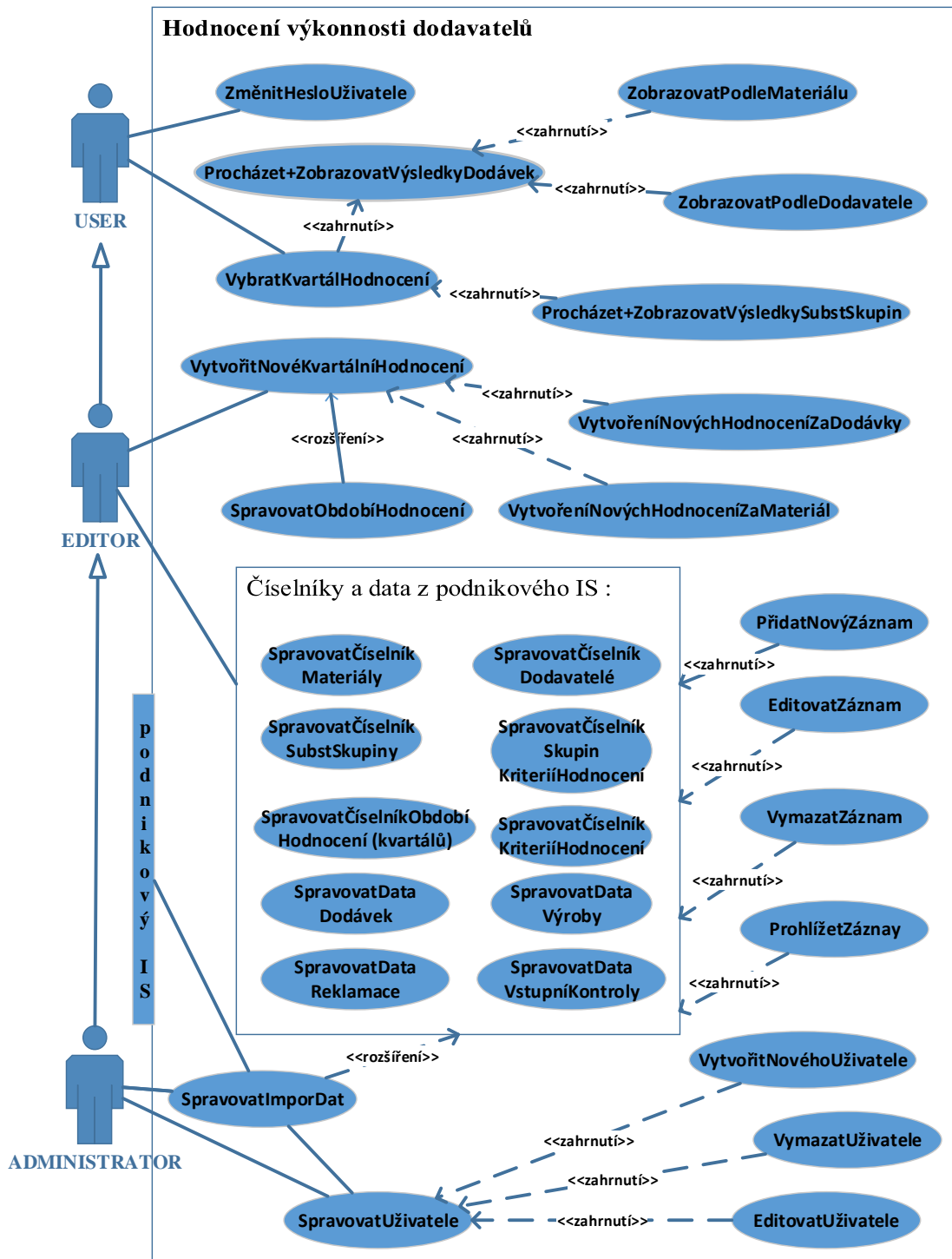
Tab. 11. Kritéria pro hodnocení dodávek .

KriteriaSkupiny				
Název	Klíč	Typ dat	N	Popis
IdSkupKrit	PK	System.Int16	7	
SkupKritNazev		System.String	20	
Status		System.Boolean	1	1 = Platná; 0 = archivní
DatZalozeni		System.DateTime		
Zalozil		System.Int16	2	Vazba na tabulku uživatel.

Tab. 12. Skupiny kritérií pro hodnocení dodávek.

User				
Název	Klíč	Typ dat	N	Popis
IdUser	PK	System.Int16	2	
Jmeno		System.String	8	
Heslo		System.String	8	
idRole		System.Int16	1	1=USER 2=EDITOR 3=ADMINISTRATOR

Tab. 13. Povolení uživatelé aplikace.

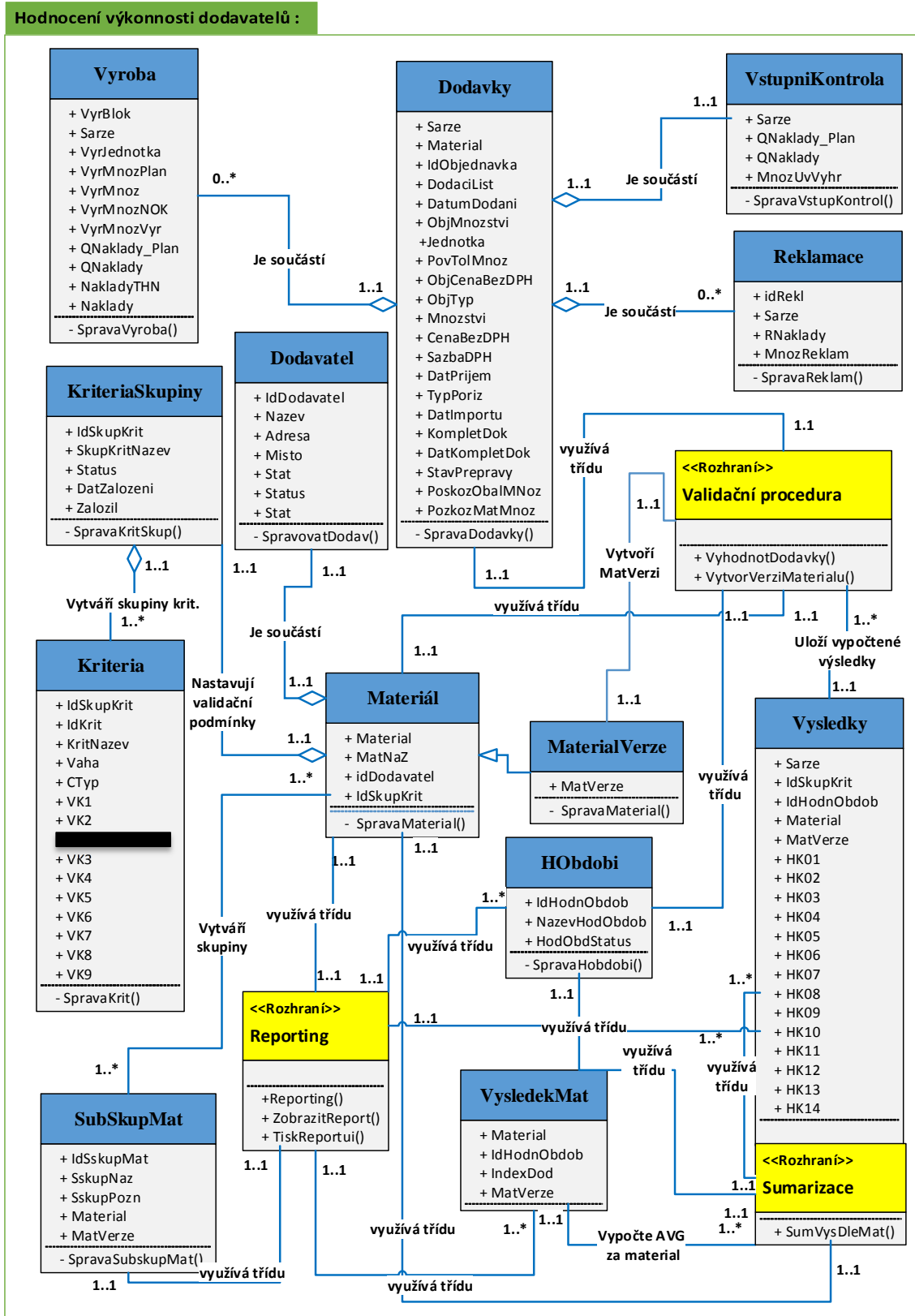


Obr. č. 1 UseCase: Hodnocení výkonnosti dodavatelů
Zdroj : Vlastní tvorba . MS-Visio 2013

Případ užití : Spravovat číselník materiály :
Účastníci : Editor, Administrator
Vstupní podmínky : Editor / Administrator je přihlášen v aplikaci a je v záložce číselníky.
Hlavní scénář : <ol style="list-style-type: none">1. Oprávněný uživatel se nachází v záložce číselníky v části číselník materiálů. 1. Systém zobrazí seznam uložených materiálů.2. KDYŽ oprávněný uživatel zvolí možnost přidání materiálu:<ol style="list-style-type: none">2.1. Systém aktivuje zobrazený formulář, pro přidání nového materiálu.2.2. Oprávněný uživatel vyplní veškeré údaje.2.3. Zadané informace potvrdí ovládacím prvkem pro uložení do databáze.2.4. Do databáze tabulky <i>Material</i> je přidán nový záznam materiálu.3. KDYŽ oprávněný uživatel zvolí možnost editace materiálu :<ol style="list-style-type: none">3.1. Oprávněný uživatel vybere ze seznamu materiálů záznam, který bude editovat.3.2. Provede požadovanou editaci dat vybraného materiálu.3.3. Provedené změny potvrdí ovládacím prvkem pro uložení dat do databáze tabulky <i>Material</i>.4. KDYŽ oprávněný uživatel zvolí možnost vymazat materiál :<ol style="list-style-type: none">4.1. Oprávněný uživatel vybere ze seznamu materiálů záznam, který bude odstraněn.4.2. Provede ovládacím prvkem jeho odstranění.4.3. Oprávněný uživatel je vyzván systémovým oknem aplikace k potvrzení požadavku na odstranění záznamu vybraného materiálu a po potvrzení je materiál odstraněn z databáze tabulky <i>Material</i>.
Rozšiřující bod : SpravovatImportDat
Výstupní podmínky : Žádné
Alternativní scénáře : Importování dat z podnikového systému přes *CSV.

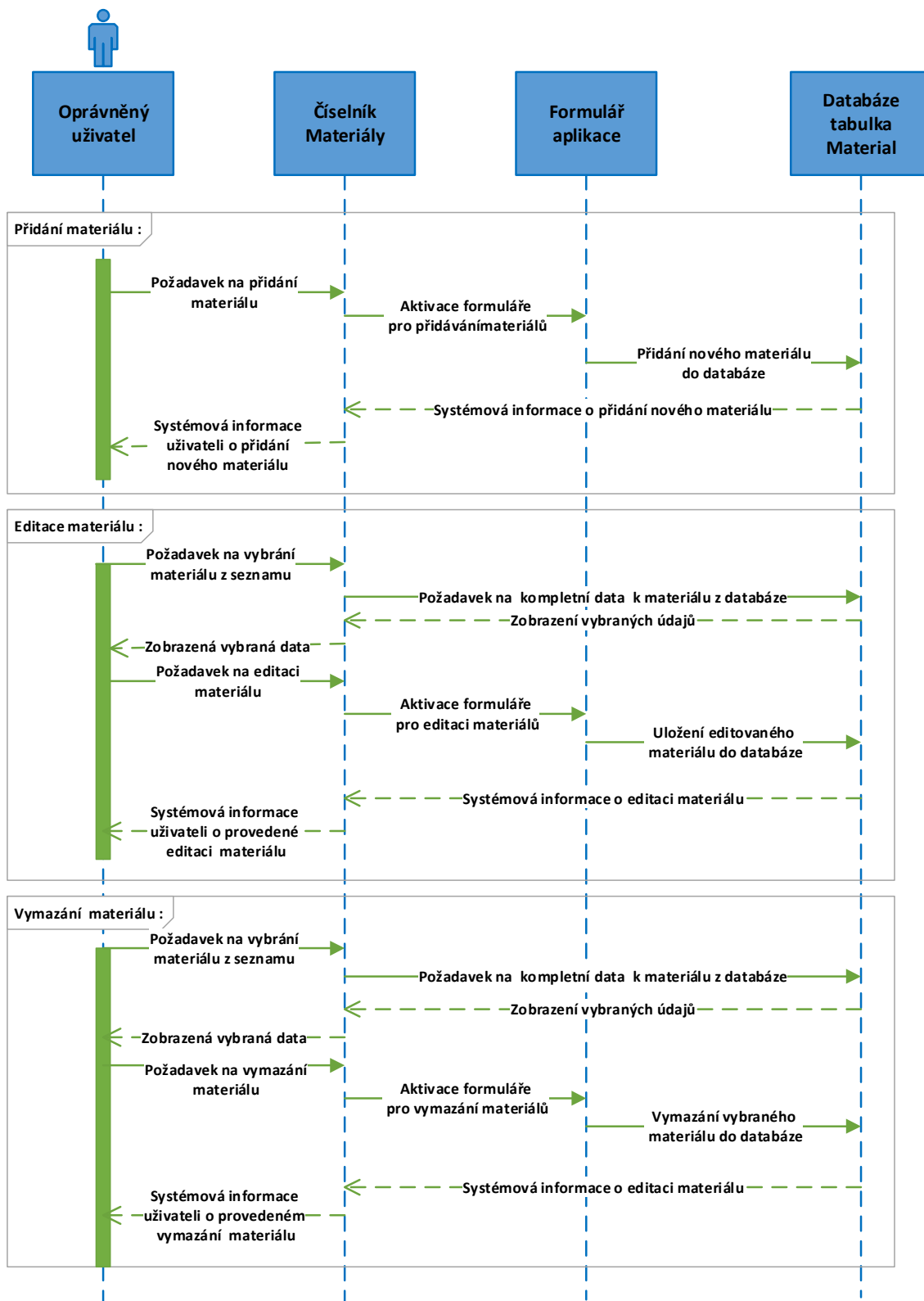
Obr. č. 2 UseCase: Spravovat číselník materiálů.

Zdroj : Vlastní tvorba .



Obr. č. 1 Velmi zjednodušený doménový model tříd – Hodnocení výkonnosti dodavatelů.
 Zdroj : Vlastní tvorba - MS-Visio 2013.

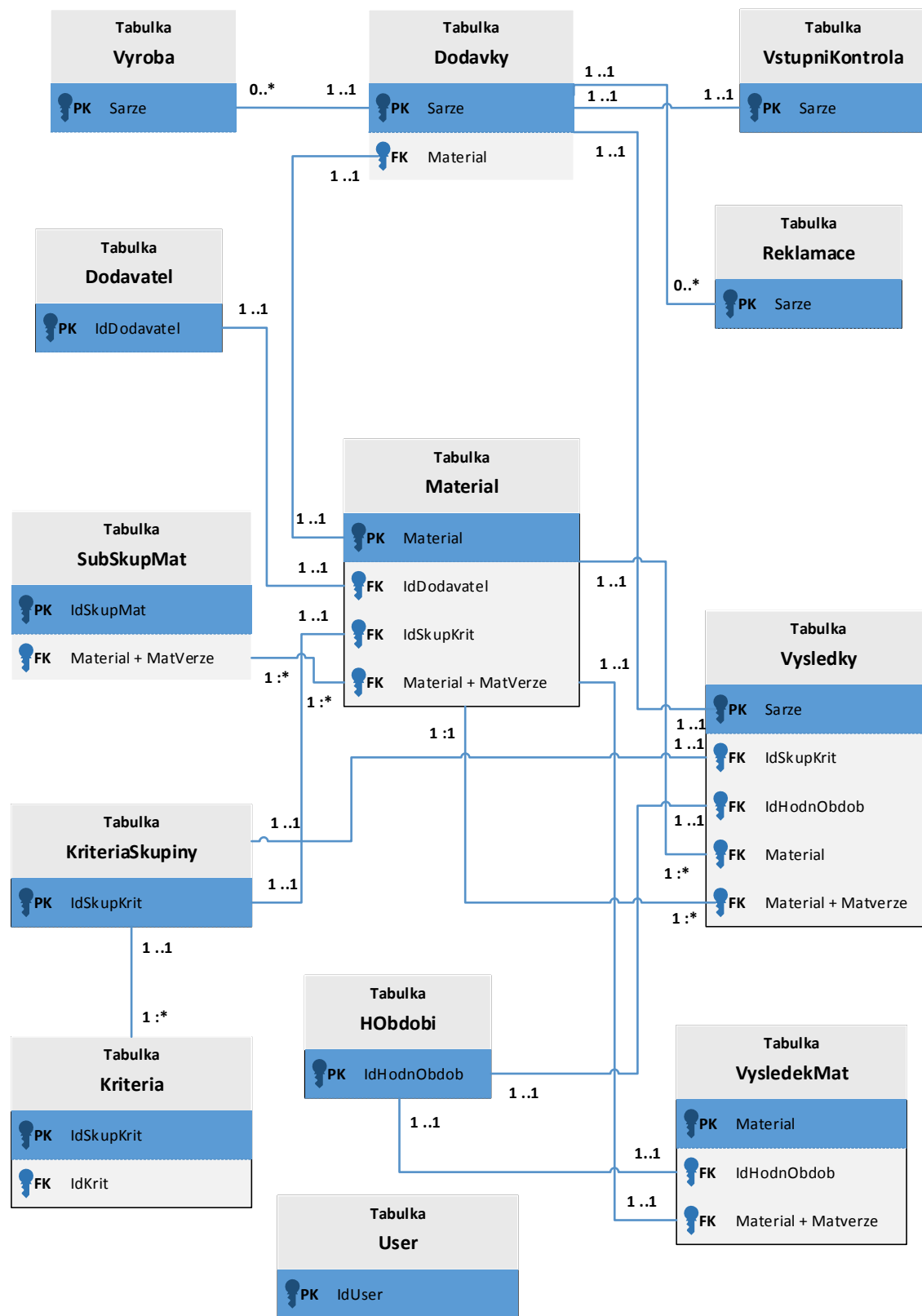
Příloha č. 4 - Příklad sekvenčního diagramu - Stránka 1 z 1.



Obr. č. 1 Sekvenční diagram pro UseCase: Spravovat číselník materiálů.
 Zdroj : Vlastní tvorba - MS-Visio 2013.

Příloha č. 5 - Zázpis databáze s využitím UML - Stránka 1 z 1.

Zázpis databáze s využitím UML jsem provedl v velmi obecné formě pouze s atributy, které mají primární nebo cizí klíč.



Obr. č. 1 Zázpis databáze s využitím UML v velmi obecné formě..

Zdroj : Vlastní tvorba - MS-Visio 2013.