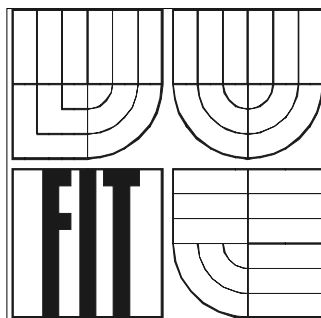


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



## **Addon – Evidence majetku pro SAP B1**

Ročníkový projekt



## **Prohlášení**

Prohlašuji, že jsem tento semestrální projekt vypracoval samostatně pod vedením Ing. Jaroslava Rába.  
Další informace mi poskytli zejména Ing. David MARTIN, BA a zaměstnanci firmy Inforis s.r.o.  
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Brně dne 1.května 2006

.....  
Roman MARTIN, BA

## **Abstrakt**

Cílem této práce je seznámit se s problematikou evidence investičního majetku firmy dle platné české legislativy, architekturou systému SAP® Business One, technologiemi pro tvorbu modulů do systému SAP® Business One a navrhnout a implementovat část modulu pro evidenci investičního majetku v systému SAP® Business One.

Na základě analýzy je proveden návrh systému, včetně výběru implementačního prostředí a použitých nástrojů. Část modulu je poté implementována podle návrhu, s důrazem na podobné uživatelské rozhraní a konvence, zavedené v systému SAP® Business One.

## **Klíčová slova**

SAP, Business One, SBO, SAP B1, databáze, informační systém, IS, ekonomický informační systém, ERP, síť, klient, server, MSSQL, .Net, evidence majetku, investiční majetek

## **Abstract**

The object of this project is to study issues around the fixed assets evidence according to valid Czech laws, architecture of SAP® Business One system, technology necessary to building add-ons for SAP® Business One and propose and implement part of add-on for fixed assets evidence in SAP® Business One.

The proposal of system is based on the analysis, including choosing the implementing environment and tools. The add-on is implemented based on proposal, with accent on similar user interface and conventions to them in SAP® Business One.

## **Keywords**

SAP, Business One, SBO, SAP B1, database, information system, IS, economic information system, ERP, net, client, server, MSSQL, .Net, assets evidence, fixed assets

## Seznam použitých zkratk

API	Application Programming Interface
COM	Component Object Model
DDL	Data Definition Language
ERP	Integrovaný podnikový systém
GUI	Graphical User Interface
HTML	HyperText Markup Language
SAP B1	Sap Business One
SQL	Structured Query Language
WWW	World Wide Web
XML	eXtensible Markup Language

# Obsah

Obsah.....	6
1 Úvod.....	7
1.1 Moderní ekonomický informační systém .....	7
1.2 Cíl projektu .....	8
2 Česká legislativa týkající se evidence majetku.....	9
3 Informační systém SAB Business One .....	11
3.1 Cenově dostupný informační systém.....	11
3.2 Pozice v rodině produktů firmy SAP .....	14
3.2.1 Přístup firmy SAP k trhu .....	14
3.2.2 Řešení pro firmy všech velikostí.....	15
3.3 Architektura systému .....	16
3.4 Funkcionalita a vlastnosti .....	17
3.5 Software Development Kit.....	18
3.5.1 Data Interface API.....	19
3.5.2 Data Interface Server .....	20
3.5.3 User Interface API.....	20
4 Návrh .....	21
4.1 Požadavky zákazníka.....	21
4.2 Strukturovaný návrh aplikace.....	21
4.2.1 Cílový operační systém .....	21
4.2.2 Diagram případů užití.....	21
4.2.3 Popis případů užití.....	23
4.2.4 Datový model aplikace .....	26
5 Implementace .....	28
5.1 Použité technologie .....	28
5.1.1 Přístup k datům .....	28
5.1.2 Integrace grafického uživatelského rozhraní .....	29
5.1.3 Vývojové prostředí.....	29
5.2 Vlastní implementace projektu.....	30
5.2.1 Soubory projektu .....	30
5.2.2 Popis aplikace .....	30
5.2.3 Omezení.....	31
6 Závěr .....	32
Literatura .....	34

# 1 Úvod

## 1.1 Moderní ekonomický informační systém

Přežít v dnešním těsně propojeném obchodním prostředí není snadné. Aby mohly malé a střední firmy v tomto prostředí uspět a konkurovat větším, potřebují integrované řešení, které umožní naplnění jejich potřeb. Potřebují cenově dostupné a výkonné řešení, poskytující přesné a kontrolní mechanismy, na základě kterých je každé firmě umožněno jednoduše přijímat potřebná rozhodnutí v reálném čase. Namísto oddělených finančních, skladových a prodejních aplikací požadují malé a střední firmy integrované řešení, které zaručuje jediný komplexní zdroj údajů, kontrolní manažerské nástroje a zaručí přehlednost prodejních procesů. Toto vše mohou získat díky řešení SAP<sup>®</sup> Business One, které je cenově dostupné a jednoduše implementovatelné, navržené s cílem splnit specifické požadavky malých a středních firem.

SAP<sup>®</sup> Business One je moderním ekonomickým informačním systémem určeným právě pro malé a střední firmy. Je to revoluční softwarové řešení od největšího dodavatele softwarových řešení na světě – společnosti SAP. Jde o řešení, které pomáhá uskutečňovat efektivnější rozhodnutí, zpřehlednit podnikání a zvýšit konkurenceschopnost na trhu.

I když je SAP<sup>®</sup> Business One velice komplexní a nabízí spoustu užitečných nástrojů, některé funkce jsou poskytovány pomocí oddělených, zvláště instalovaných modulů, které se v naprosté většině integrují do prostředí SAP<sup>®</sup> Business One a pro koncového uživatele jsou k nerozeznání od integrovaných funkcí systému. Jedním z takovýchto modulů je i modul s názvem Fixed Assets, který slouží pro evidenci investičního majetku firmy. Tento modul z legislativního hlediska vyhovuje pouze několika zemím, pro které je systém SAP<sup>®</sup> Business One určen. Mezi tyto země bohužel v současné době Česká republika nepatří a tak vznikla potřeba tento problém nějakým způsobem vyřešit. Nabízí se dvě možnosti. První možností je používat externí aplikaci pro evidenci majetku, nezávisle na systému SAP<sup>®</sup> Business One. Tato možnost ale skýtá řadu problému, kterými jsou například propojení mezi systémy, potřeba evidovat například faktury za nakoupený majetek ve dvou systémech, uživatelská náročnost obsluhovat dva různé systémy a v neposlední řadě se také setkává s nevolí koncových zákazníků něco podobného akceptovat. Druhou možností je navrhnout a implementovat modul pro evidenci majetku, který se integruje do systému SAP<sup>®</sup> Business One podobně jako originální modul Fixed Assets a bude vyhovovat české legislativě. Právě touto problematikou se zabývá tato práce.



## **1.2 Cíl projektu**

Cílem ročníkového projektu je seznámit se s českou legislativou pro evidenci investičního majetku firmy, architekturou informačního systému SAP® Business One, technologiemi pomocí kterých lze implementovat modul do tohoto systému, navrhnout a implementovat část modulu pro evidenci investičního majetku firmy, který se integruje do prostředí systému.

## 2 Česká legislativa týkající se evidence majetku

Evidence majetku slouží k evidenci účetních a daňových informací dlouhodobého investičního majetku firmy - účetní jednotky. Evidence jako taková je podpůrnou evidencí k účetní evidenci.

Evidence majetku je upravena zákonem 563/1991 Sb., o účetnictví a navazujícími prováděcími předpisy, zejména vyhláškou č. 500/2002 Sb., kterou se provádějí některá ustanovení zákona č. 563/1991 Sb., o účetnictví, ve znění pozdějších předpisů, pro účetní jednotky, které jsou podnikateli účtujícími v soustavě podvojného účetnictví a účetními standardy (zejména standardem č.013) a to v rozsahu nezbytném pro účetní účely. Daňové aspekty pak upravuje zákon 586/1992 Sb., o daních z příjmů zejména v § 26-33.

Účetní předpisy neupravují evidenci majetku zcela vyčerpávajícím způsobem, nýbrž určují zejména obsahové vymezení složek dlouhodobého majetku a principy oceňování, odpisování, účtování a vykazování dlouhodobého majetku. Obsah evidence je pak upraven zejména v bodu 6.3. výše uvedeného účetního standardu. Způsob vedení evidence (ruční papírová, softwarově řešená) či její podobu (design formulářů, sestav) účetní ani jiné předpisy neupravují. Zjednodušeně řečeno lze tedy při sestavování evidence majetku vyjít z obecných požadavků účetních předpisů na správnost, úplnost, průkaznost, srozumitelnost a přehlednost evidence (§8 odst.1 zákona o účetnictví) a z dalších vnitřních potřeb a specifických potřeb konkrétní účetní jednotky.

Zákon o daních z příjmů upravuje vymezení majetku, ocenění majetku (pokud se liší od účetního ocenění), způsoby daňového odpisování, daňové dopady s pohyby majetku (pořízení, vyřazení, technické zhodnocení, vklady atd.) a to vždy za účelem zjištění základu daně z příjmů daného poplatníka.

Dalšími předpisy, které upravují daňové aspekty majetku jsou okrajově zákon o dani silniční, o dani z nemovitostí, o dani dědické, dani darovací a dani z převodu nemovitostí (tzv. "zákon o trojdani") a zákon o dani z přidané hodnoty. Tyto předpisy se však do evidence majetku, která je vedena pro účely účetnictví a daně z příjmů, promítají jen přeneseně či okrajově. Jmenujme například ocenění majetku v souvislosti s tím, zda má plátcе daně z přidané hodnoty zcela, zčásti a nebo vůbec nemá nárok na odpočet DPH při pořízení majetku. Zákon o dani silniční, o dani z nemovitostí a "zákon o trojdani" se v podstatě do evidence majetku v praxi nikterak neprojeví a evidenční požadavky těchto předpisů plní poplatníci v drtivé většině vedením různých jednoduchých tabulek.

V praxi je účetní evidence téměř vždy vedena v softwarové podobě za pomoci nejrůznějších tabulkových procesorů či nejlépe přímo k tomu určených modulech informačních systémů. V

současné době se již zcela vytrácí ručně vedené evidence. V případě menších firem je evidence vedena obvykle v univerzálních, nenáročných a levných programech, obvykle - jak je již výše řečeno - v modulu používaného informačního systému. V případě větších firem (např. mobilní operátoři, nadnárodní koncerny, rozsáhlé výrobní podniky) či firem s odlišnou oblastí výroby (např. firmy operující v oblasti zbrojního průmyslu, leasingové společnosti atd.) je obvykle evidence majetku řešena modulárně ve vzájemné vazbě na účetní evidenci, na obchodní evidenci či další evidence, které vyžadují zvláštní předpisy - prvotním úkolem takových řešení již není splnění povinností vyplývajících z účetních a daňových předpisů, nýbrž zajistit všechny potřebné informace nutné pro vedení podniku a obchodní využití majetku.

Evidence majetku u malých a středních firem vychází vždy z požadavků účetních a daňových předpisů. Evidence musí být schopna nejen evidovat veškeré údaje stanovené účetními a daňovými předpisy, správně počítat účetní a daňové odpisy, ale i poskytovat veškeré relevantní informace pro pracovníky firmy, počínaje účetních pracovníků, vedoucích pracovníků, manažerů konče auditory, daňovými poradci a státními orgány. V dnešní době musí být evidence zpracována tak, aby umožnila jednoduché přizpůsobení změnám ve firemním a legislativním prostředí. Jinými slovy - evidence musí uživateli nabídnout prostředí, v kterém bude uživatel ve spolupráci se servisní firmou schopen přizpůsobit evidenci běžným legislativním změnám a vnitřním požadavkům firmy, musí uživateli umožnit přijatelný způsob opravy chyb v evidenci způsobených jím samým a zejména musí uživateli umožnit libovolně definovat strukturu a formu výstupů.

## 3 Informační systém SAB Business One

Přežít v dnešním těsně propojeném obchodním prostředí není snadné. Aby mohly malé a střední firmy konkurovat větším, potřebují stejně kvalitní řešení, avšak cenově dostupné se specifickou funkcionalitou.

SAP takové řešení nabízí. Díky kombinaci plného rozsahu obchodní funkcionality v jediném, jednoduchém, cenově dostupném, integrovaném řešení se informační systém SAP® Business One stává ideálním nástrojem pro malé a střední firmy. Jednoduše se používá a přizpůsobuje. Toto řešení bylo vytvořeno nejlepším světovým dodavatelem řešení pro globální společnosti. SAP® Business One (dále jen SAP B1) otevírá konkurenční příležitosti i pro menší společnosti, a to nejen dnes, ale i do budoucnosti.

### 3.1 Cenově dostupný informační systém

V dnešních dynamických obchodních podmínkách již nemohou firmy déle fungovat bez efektivního informačního systému. Aby byly schopny zaručit efektivní řízení a konkurenceschopnost bez ohledu na jejich velikost, jsou čím dál tím více závislé na informační technologii.

Stejně jako velké korporace, tak i malé a střední firmy potřebují integrovaný podnikový systém (ERP), jenž jim pomůže dosáhnout jimi definované obchodní cíle efektivněji, potřebují software se kterým mohou počítat i do budoucnosti.

Aby byl ERP software pro malé a střední firmy efektivní, musí splnit široký rozsah požadavků a očekávání. Například musí být rychle implementovatelný, musí nabídnout dostatečný rozsah funkcionality a umožnit odpovídající integraci mezi samostatnými aplikačními komponentami. Musí být také cenově dostupný, jednoduše použitelný a kompatibilní s budoucími technologickými aktualizacemi. A konečně, musí využívat nejnovější technologie, včetně internetu, podporovat stávající standardy a mít zajištěnu místní podporu od zkušeného IT dodavatele na instalaci, údržbu a školení.

SAP B1 je integrované řešení vytvořené přímo pro malé a střední firmy. Neposkytuje jen rychlý a snadný přístup ke všem vnitropodnikovým informacím, ale také dodává reporty a dokumenty nutné pro přijímání rozhodnutí pro všechny oblasti vaší firmy. Stačí jen o málo více než stisknutí tlačítka a máte k dispozici užitečné nástroje pro efektivní podporu každodenních procesů v oblasti prodeje, nákupu, účetní oblasti a dalších. SAP B1 dodává rozhodující informace pro to, aby bylo vedení firmy úspěšné.

Implementace SAP B1 může proběhnout velmi rychle. Doba implementace závisí na složitosti výchozích obchodních transakcí a pohybuje se v rozmezí od několika dnů až po několik týdnů. Jednoduchost implementace je garantována strukturou a konstrukcí samotného řešení, které využívá výhod speciálních SAP nástrojů pro migraci dat jakými jsou například API na import dat, import účetní osnovy a dalších nástrojů včetně možnosti využití XML.

SAP B1 poskytuje všechny funkce a vlastnosti potřebné pro efektivní, pokrokové a na růst orientované řešení. V neposlední řadě řešení odpovídá vysokým standardům SAP, které se budou vyvíjet společně s požadavky zákazníků.

Hlavními prvky dostupného a hodnotného řešení jsou:

#### **- Uživatelsky příjemné rozhraní, procesně provázaná data**

SAP B1 je jednoduše ovladatelný, zjednoduší běžné pracovní úkony. Inovační funkčnost „Drag&Relate“ umožní intuitivně pracovat s daty. Například z obrazovky určené pro objednávky se můžete zanořit do údajů o zákazníkovi, jedním kliknutím myši najít všechny obchodní příležitosti a s Drag&Relate vytvořit analytickou sestavu v podstatě ke všem údajům na obrazovce. Tímto způsobem můžete získat potřebné stěžejní informace k tomu, aby mohlo být vaše rozhodování efektivnější a rychlejší.

#### **- Výstražné funkce založené na Workflow**

Každý uživatel SAP B1 může nakonfigurovat rozmanité výstražné funkce pro mimořádné události, např. upomínky, schvalovací procesy, zasilání dokumentů jiným uživatelům, atp. V souladu s konfigurací nastavení zasílá řešení automaticky hlášení zodpovědnému zaměstnanci. Tyto činnosti mohou být spouštěny výstražnými funkcemi obsaženými v řešení samém, nebo také pomocí e-mailu či faxu.

#### **- Flexibilní průvodce pro vytváření sestav**

Vedle širokého rozpětí standardně dodávaných sestav můžete využít intuitivní nástroj „Query Wizard“ ke shromažďování informací ze všech oblastí do specifických zákaznických výstupních sestav. Je podporován prakticky každý typ analýzy.

#### **- Snadná integrace s Microsoft Office**

Snadné používání SAP B1 zajišťuje to, že pracuje v homogenním prostředí s Microsoft Windows, v jeho dobře známém standardním uživatelském rozhraní a funkcionalitě. Například můžete jediným klikem myši konvertovat sestavu do personalizovaného dopisu nebo do Microsoft Excelu a odeslat ji e-mailem.

#### **- Specifické firemní šablony dokumentů**

Každá šablona dokumentů může být v řešení SAP B1 přizpůsobena požadavkům vaší firmy. Zákaznické prvky, které vyberete, jakými jsou data, obrázky produktů, náklady, mezisoučty a další, mohou být formátovány do úhledných dokumentů.

### **- Podpora zahraničních měn a jazyků**

SAP B1 pro účetnictví podporuje současné zpracování všech zaúčtování ve třech měnách: lokální měně, obchodní měně a systémové měně. Můžete lehce vytvářet sestavy v jakékoli mezinárodní měně. Řešení SAP B1 také podporuje více jazyků a umožňuje uživatelům, aby si vybrali jejich osobní jazykové rozhraní a kdykoliv jej mohli změnit.

### **- Funkce Internal sales force automation (SFA)**

Zakotvená funkcionalita SFA vám umožní spravovat obchodní příležitosti, sledovat plán prodeje a řídit aktivity spojené se zákazníky. Máte k dispozici úplný pohled na informace o zákaznících včetně hrubého zisku a stavu zásob. Po krátkém zaškolení vám tyto informace mohou pomoci optimalizovat vaše obchodní příležitosti a zlepšit vaše celkové výsledky.

### **- Standardní rozhraní k zajištění komplexní integrace**

SAP B1, jako centrální řídicí aplikace, zahrnuje standardní rozhraní pro interní a externí datové zdroje. Tyto rozhraní zahrnují počítače do dlaně, prodejní internetové aplikace a analytické programy třetích stran.

### **- Aktualizace uživatelských polí a změny „za běhu“**

Toto řešení může být jednoduše přizpůsobeno zákaznickým potřebám. Integrovaný, uživatelsky příjemný systém správy polí umožní jednoduše vkládat nová pole do již existujících formulářů. Lze také nalinkovat tzv. roletová pole, analýzy a procesy téměř k jakémukoli poli. Pro zvýšení funkcionality lze se stejnými nástroji přidávat i dodatečné tabulky.

Firma SAP ví, že systémy pracující v reálném čase vyžadují bezprostřední aktualizace a změny, a že uživatelé musí mít možnost vkládat tabulky a pole „za běhu“ systému. Proto veškeré změny provedené v rámci SAP B1 vstoupí v platnost okamžitě. Nejčastěji požadované změny mohou být snadno definovány uživatelem bez toho, aby musel absolvovat nějaké technické školení. Tato funkcionalita je významným zdokonalením oproti konkurenčním řešením, která umožňují aktualizace pouze v dávkách.

### **- Komplexní vývojové prostředí (SDK) k zajištění bezproblémových aktualizací**

Na komplikovanější technické úpravy mají vývojáři přístup k jednoduchému, nicméně výkonnému nástroji SDK. Ten je postaven na osvědčených odvětvových standardech, které jsou softwarovým vývojářům důvěrně známé a poskytuje v rámci SAP B1 úplný přístup k funkcionalitě řešení.

## 3.2 Pozice v rodině produktů firmy SAP

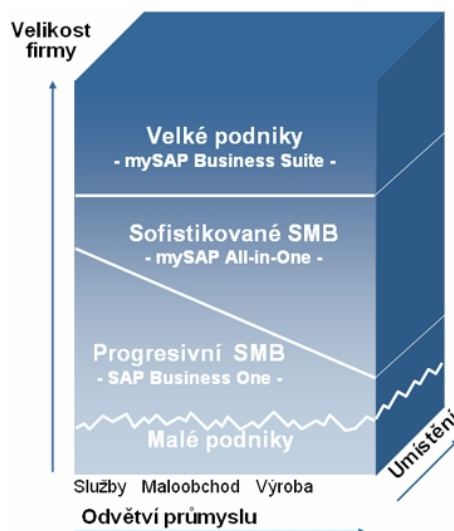
### 3.2.1 Přístup firmy SAP k trhu

Neexistuje jednotný a uniformní trh. Firmy působící na trhu se podstatně liší svou velikostí, potřebou komplexních informačních systémů, průmyslovým odvětvím a umístěním.

Firma SAP proto rozděluje trh následovně (Obrázek 3.1):

- Malé firmy, které nevytvářejí rozvahu
- Progresivní malé a střední podniky
- Sofistikované malé a střední podniky
- Velké podniky

Druhá a třetí skupina potom tvoří trh malých a středních podniků, v angličtině nazýván jako Small and Medium Business (SMB).



Obrázek 3.1: Rozdělení trhu podle SAP

Malé firmy nevytvářející rozvahu obecně nemají zvláštní potřebu týkající se podnikového software. Ve většině případů mohou být potřeby uspokojeny nejnovějšími aplikacemi Microsoft Office a jednoduchým účetním programem. Trh vytvořený těmito firmami je tedy z hlediska komplexních informačních systémů nezajímavý a firma SAP pro něj nenabízí žádný produkt.

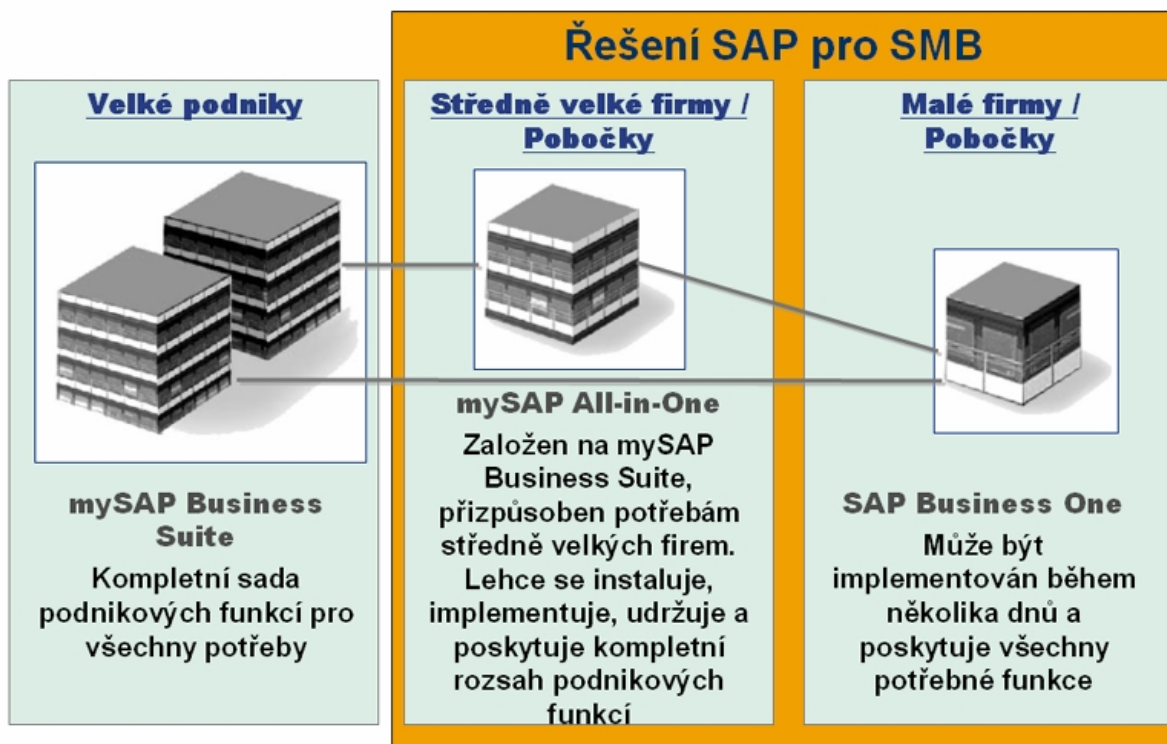
Progresivní malé a střední podniky naproti tomu musí být schopny modelovat a řídit své podnikové procesy pomocí softwarového systému. Speciálně pro takovéto firmy vyvinula firma SAP produkt SAP<sup>®</sup> Business One. Lze říci, že se systém SAP<sup>®</sup> Business One zaměřuje na firmy s pomyslnou hranicí maximálně 250 zaměstnanců.

Jelikož podnikové procesy sofistikovaných malých a středních podniků jsou komplexnější, potřebují tyto firmy softwarový systém, který lze snadno implementovat a který nabízí široký rozsah podnikových funkcí. Firma SAP vyvinula systém mySAP All-in-One speciálně pro tyto firmy.

Velké podniky představují tradiční základnu zákazníků firmy SAP a je pro ně určen systém mySAP Business Suite.

### 3.2.2 Řešení pro firmy všech velikostí

Firma SAP nabízí řešení pro firmy všech velikostí. Rozdělení cílového trhu spolu s nabízenými produkty pro tyto segmenty trhu a velmi stručnou charakteristikou zobrazuje Obrázek 3.2:



Obrázek 3.2: Přehled produktů SAP pro jednotlivé trhy

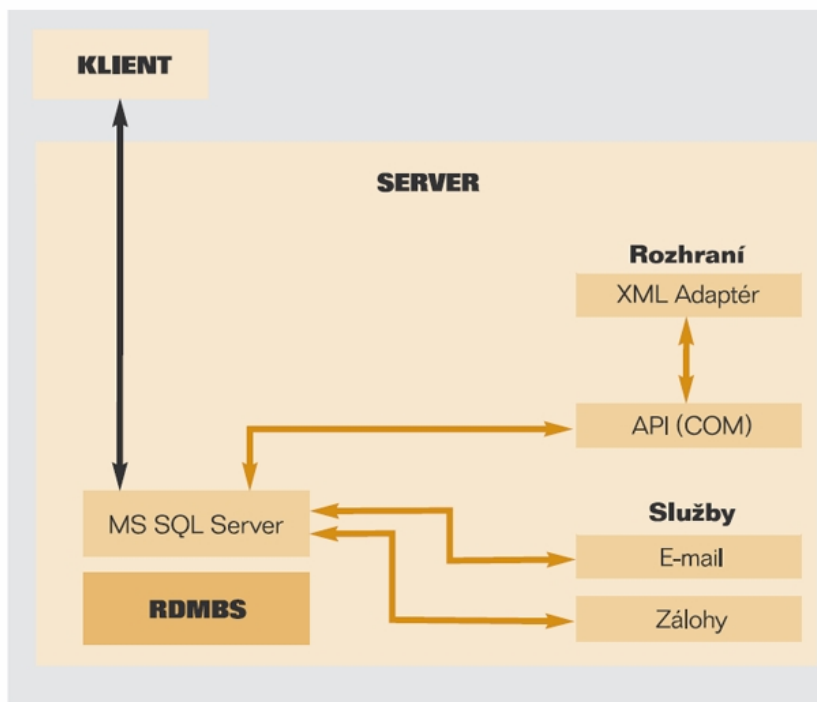
- **mySAP Business Suite:** Tato komplexní řada podnikových řešení splňuje požadavky všech hlavních průmyslových odvětví a poskytuje plný rozsah funkčnosti pro všechny podnikové procesy.
- **mySAP All-in-One:** Systém mySAP-All-in-One přináší výhody sady mySAP Business Suite malým a středně velkým podnikům (sofistikované malé a střední firmy) v balíčku speciálně upraveném podle jejich potřeb. Řešení mySAP All-in-One byla navržena partnery SAP jako systémy přednastavené podle konkrétního průmyslového odvětví, založené na zjednodušené systémové architektuře s nízkými náklady na údržbu.
- **SAP<sup>®</sup> Business One:** Systém SAP<sup>®</sup> Business One obsahuje všechny funkce požadované malými a středními podniky (progresivní malé a střední firmy) a lze je implementovat během několika dnů. Nabízí velkou šíři inovovaných funkcí při minimální údržbě a provozních nákladech.

Všechny tři produkty mohou být vzájemně integrovány.



### 3.3 Architektura systému

SAP B1 je plně integrované řešení s jednotným, intuitivním vzhledem a vlastnostmi napříč všemi funkcemi. Jeho moderní architektura mu umožňuje plně využít zvolenou databázi, výsledkem čehož je výkonné řešení, které se snadno udržuje a spravuje.



Obrázek 3.3: Architektura SAP B1

Aplikace SAP B1 je umístěna na jednom serveru s operačním systémem Microsoft Windows Server a je plně integrována v síti Windows NT. SAP B1 pro uživatele zajišťuje špičkový výkon využitím dvouvrstvé klient-server architektury (viz Obrázek 3.3). Takzvaný tlustý klient je tedy umístěn na klientské stanici a zajišťuje téměř veškerou business logiku (validace dat, apod.). Kromě aplikace, je na serveru umístěn ještě databázový server a licenční server. Tyto servery mohou, ale nemusí běžet na fyzicky stejném serveru jako aplikace SAP B1. Podporované databázové servery jsou: Microsoft SQL Server 2000, IBM DB2 Universal Database verze Express Edition 8.1 a Sybase Adaptive Server Enterprise 12.5.1. Licenční server je aplikace SAP B1 pro online přidělování pojmenovaných licencí aktuálně připojeným uživatelům.

To, že je řešení optimalizováno pro Microsoft Windows Server, umožňuje plně využít existující síť k maximálnímu výkonu. Toto zahrnuje zabezpečení, zálohy a protokoly síťových přístupů. Přístup je udělován přes veřejnou síť (WAN), terminálové služby nebo přes modemové připojení.

V prostředí SAP B1 lze využívat standardní procedury na zálohování databází a jednoduše ukládat a přenášet databázi na jiné zařízení, čímž lze zajistit okamžitý přístup ke kritickým obchodním informacím.

Aby zůstala údržba jednoduchá, je systém zabezpečení postaven na funkčním modelu. Administrátor může prohlížet seznam všech funkcí pro každý modul a může specifikovat, který typ přístupu k nim který uživatel má: čtení, zápis nebo obojí.

Všechny komponenty jsou postaveny na základní filosofii, že řešení musí přirozeně fungovat a reagovat v souladu s potřebami zákazníka. Nesmí být příliš umělé a těžké na naučení. Protože SAP B1 a jeho komponenty jsou přizpůsobeny logice uživatele, jsou požadavky na školení minimální a osvojení uživatelem je velmi rychlé.

## 3.4 Funkcionalita a vlastnosti

SAP B1 je založen na jednoduše ovladatelném rozhraní. Používá se jako hlavní firemní ERP aplikace s možností využití dalších standardních rozhraní. Je komplexním a univerzálním podnikatelským nástrojem, který využijí manažeři i ostatní zaměstnanci v celém podniku.

Řešení poskytuje všechny běžné administrativní funkce, které umožňují upravovat a zálohovat data, definovat kurzy přepočtu měn, parametrizovat oprávnění i přístupy a přistupovat k údajům z produktů třetích stran. Skládá se z jednotlivých modulů, které rozšiřují schopnosti řešení SAP B1, zjednodušují podnikové procesy a nabízejí daleko více, než jen základní administrativní funkce.

Jedná se o následující moduly:

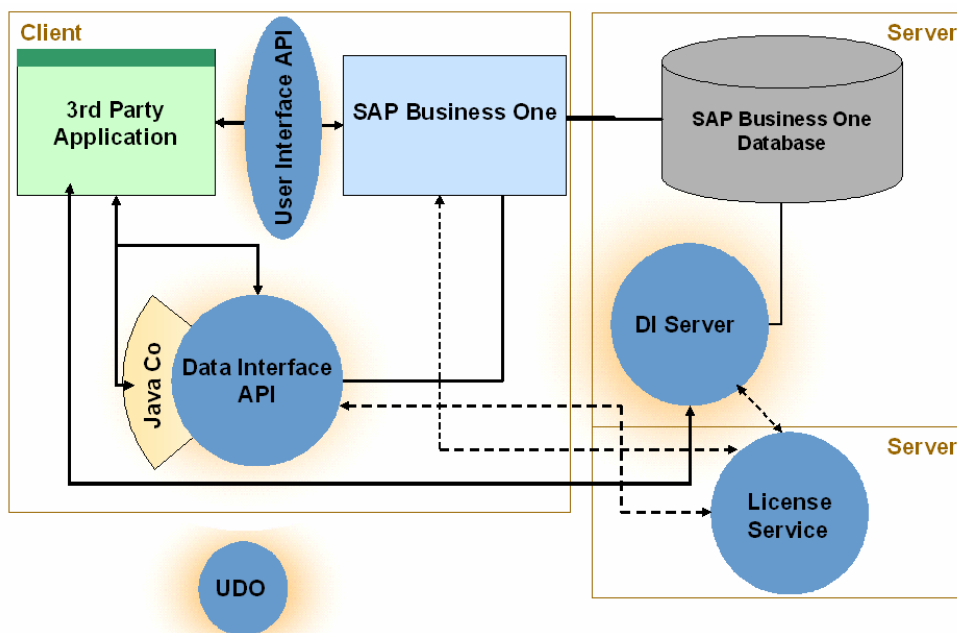
- **Finanční účetnictví** - podporuje všechny běžně využívané finanční operace, definici a správu účtů, účetní deníky, účtování v cizí měně, rozpočtování atd.
- **Prodej a distribuce** - pomáhá definovat ceny výrobků, zpracovávat zakázky odběratelů, řídit procesy dodání výrobků, aktualizovat skladové zásoby a řídit procesy fakturace a pohledávek.
- **Nákup** - řízení a správa dodavatelských smluv a operací, jako jsou vystavení nákupní objednávky, aktualizace interních zásob, kalkulace hodnot importovaných položek, zpracování plateb a řízení dobropisů a záloh.
- **Obchodní partner** - umožňuje kontrolu odpovídajících informací o odběratelích, prodejcích a dodavatelích, včetně jejich vlastností, stavech na jejich účtech, sumarizaci jejich kontraktů a analýzy odbytového plánu.
- **Bankovní operace** - zabezpečují zpracování finančních operací, jako jsou pokladní operace, depozitní operace, operace s šeky, zálohy, platby kreditní kartou a také zpracování bankovních výpisů.
- **Řízení skladu** - řízení stavu skladu, správa položek, příprava ceníků, specifické cenové podmínky, přesuny mezi sklady, pohyby materiálu.

- **Výroba** - nabízí nástroje, které umožňují definování víceúrovňových kusovníků, vytváření výrobních zakázek, a které poskytují potřebné informace o výrobku a dostupnosti jednotlivých materiálů.
- **Controlling** - umožňuje definici nákladových středisek a pravidel pro rozdělení nákladů, generování výkazů zisků a ztrát pro každé středisko.
- **Výkaznictví** - poskytuje sestavy pro každou oblast podnikání, včetně dodavatelských a odběratelských operací, cashflow, sumarizaci kontaktů se zákazníky, přehledy účtování, skladové zásoby, zůstatky na účtech, výkazy o cenách a další je možné vytvořit pomocí předdefinovaných výkazů nebo navržených sestav.
- **Řízení servisu** - optimalizuje činnosti oddělení servisu, poskytuje podporu pro servisní operace, správu servisních kontraktů, plánování servisních aktivit, sledování komunikace se zákazníky, podporu zákazníků a řízení obchodních příležitostí.
- **Personalistika** - umožňuje řízení lidských zdrojů, včetně správy zaměstnaneckých údajů, kontaktních informací a výkazů docházky.
- **Administrace** - umožňuje konfigurovat základní systémové nastavení pro všechny komponenty. To zahrnuje definování kurzu měny, inicializaci systému, nastavení autorizačních parametrů, nastavení licencí k produktu, vytvoření interní pošty, nastavení e-mailu, nastavení funkcí importu a exportu dat.

## 3.5 Software Development Kit

SAP B1 je dodáván jako robustní základní systém se silnou funkcionalitou. Spolu se systémem je partnerům dodáván Software Development Kit (SDK). Jedná se o balíček několika knihoven, které umožňují vyvíjet moduly pro systém SAP B1, které mohou přidávat nebo upravovat standardní chování systému. SDK je určen pouze pro obchodní partnery firmy SAP, nikoliv pro koncové zákazníky.

SDK se skládá ze tří různých programovacích aplikačních rozhraní (API) (viz Obrázek 3.4.)



Obrázek 3.4: Komponenty Software Development Kit

Jedná se o tyto rozhraní:

- Data Interface API (DI API)
- Data Interface Server (DI Server)
- User Interface API (UI API)

Další součástí SDK je User Defined Object (UDO), což je uživatelsky definovaný objekt, který se připojí k ostatním obchodním objektům v databázi systému a programátor jej může ve svých modulech používat.

### 3.5.1 Data Interface API

DI API nabízí přístup k systému na datové úrovni. Je založeno na COM (Component Object Model) technologii firmy Microsoft. Z toho plyne, že programovací prostředí, ve kterém budeme modul programovat, musí tuto technologii podporovat. Navíc obsahuje nadstavbu v podobě Java Connector (Java Co), který umožňuje přistupovat k tomuto aplikačnímu rozhraní pomocí jazyka Java.

Pomocí DI API lze číst, zapisovat a měnit objekty systému na datové úrovni. Nabízí přístup ke kmenovým a transakčním datům. Jehož pomocí lze také připojit již existující aplikace třetích stran k systému SAP B1.

DI API není závislé na klientu SAP B1. Z toho plyne, že aplikace vytvořené s jeho využitím můžeme provozovat i na počítači, kde není nainstalovaný klient systému. Obchodní logiku potom zajišťuje samotné DI API, které nabízí stejnou logiku a tedy i stejné validace jako klient systému.

### **3.5.2 Data Interface Server**

DI Server nabízí přístup k systému na datové úrovni. Je založen na technologii DCOM a SOAP. DI Server je spuštěn na serveru, kde nabízí stejné možnosti jako DIAPI s tím rozdílem, že se jedná o serverovou službu. Na rozdíl od DIAPI je tedy schopen obsloužit velké množství klientů v jednom časovém okamžiku a neomezuje technologie použité pro vývoj klientů na technologie podporující COM.

### **3.5.3 User Interface API**

UIAPI nabízí přístup k systému na uživatelské úrovni. Stejně jako DIAPI je založeno na technologii COM. Nabízí přístup k běžící aplikaci, u které může programátor měnit nebo přidávat formuláře, ovládací prvky, měnit jejich vlastnosti, přidávat okna a nabídky menu, obsluhovat události, atd.

Pomocí UIAPI lze vytvořit modul, jehož uživatelské rozhraní bude integrováno v uživatelském rozhraní SAP B1 se stejným vzhledem. Toto je pro uživatele obrovská výhoda, protože nemusí spouštět externí aplikaci, ale vše mají po ruce v jednom prostředí.

## **4 Návrh**

### **4.1 Požadavky zákazníka**

Jde o modul pro evidenci investičního majetku firmy, který bude vyhovovat platné české legislativě a příslušným účetním standardům. Modul by měl být do jisté míry flexibilní vůči reformám legislativy, či odlišným potřebám některých zákazníků. Modul se bude integrovat do SAP B1. Jeho uživatelské rozhraní by mělo být velmi podobné jako mají okna v SAP B1, to samé platí o stylu ovládnání modulu. Modul by měl umožňovat definovat různá uživatelská oprávnění alespoň na účetně a daňově citlivých místech, jako jsou např.: editace některých číselníků, storno odpisů, výpočet odpisů, zaúčtování pohybů, atd. Ideální by však bylo umožnit definovat uživatelská oprávnění na úrovni jednotlivých menu. Modul by měl umožnit účtovat tzv. dávkově. To znamená, že po výpočtu účetního odpisu se tento nezaúčtuje ihned jak je v SAPu zvykem, ale zapíše se do pohybů. Pohyby lze pak později ručně hromadně zaúčtovat. Mezitím lze v pohybech dělat stornování některých zápisů.

### **4.2 Strukturovaný návrh aplikace**

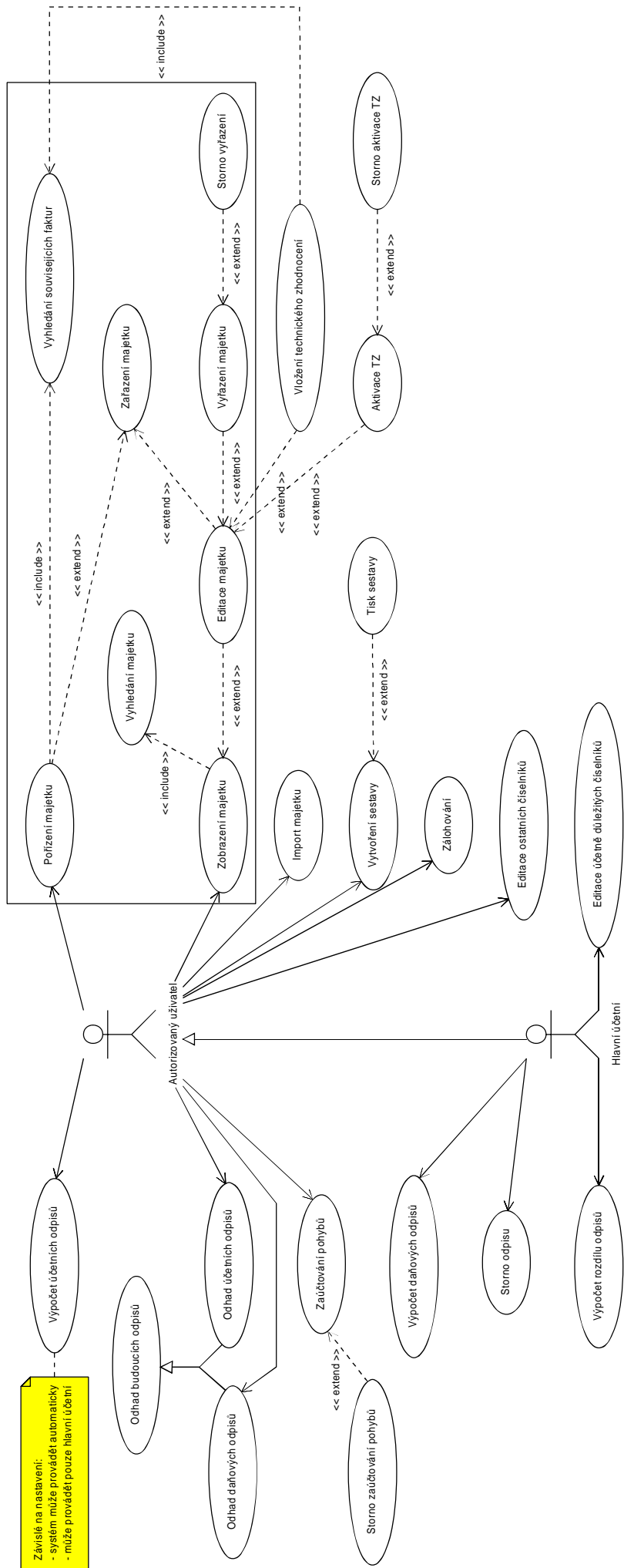
#### **4.2.1 Cílový operační systém**

- Microsoft Windows XP
- Microsoft Windows 2000
- Microsoft Windows 2003 Server
- Microsoft Windows 2000 Server

#### **4.2.2 Diagram případů užití**

Viz následující strana

Diagram případu užítí



## 4.2.3 Popis případů užití

Pro práci s evidencí majetku, se uživatel musí nejprve přihlásit k systému SAP B1. Pokud má oprávnění spustit modul evidence majetku, musí mu systém umožnit následující případy užití s ohledem na uživatelské oprávnění.

### Pořízení majetku

Aktérem je autorizovaný uživatel, který může vložit nový majetek do evidence. Musí zadat všechny povinné údaje majetku, včetně výběru související faktury v účetnictví. Systém vygeneruje inventární číslo majetku. V případě zadání všech povinných údajů pro zařazení majetku, lze tento ihned na přání zařadit.

Zde spadají následující operace:

- Vyhledání souvisejících faktur
- Zařazení majetku

### Vyhledání souvisejících faktur

Aktérem je autorizovaný uživatel, který chce vyhledat fakturu v účetnictví. Systém zobrazí formulář pro vyhledání faktury a umožní výběr zboží vyhledané faktury, které souvisí s danou operací.

### Zobrazení majetku

Aktérem je autorizovaný uživatel, který chce zobrazit evidovaný majetek firmy. Systém mu zobrazí formulář inventární karty majetku s naposledy zobrazenými daty. Mezi daty se lze pohybovat vpřed a vzad obvyklými způsoby.

Zde spadají následující operace:

- Vyhledání majetku
- Editace majetku

### Vyhledání majetku

Aktérem je autorizovaný uživatel, který chce vyhledat majetek odpovídající zadaným kritériím. Systém mu to musí umožnit obvyklým způsobem na formuláři inventární karty majetku.

### Editace majetku

Aktérem je autorizovaný uživatel, který si vyhledal a zobrazil konkrétní záznam majetku a chce jej změnit. Systém mu musí obvyklým způsobem umožnit změnu povolených údajů na formuláři inventární karty majetku. Povolenými údaji jsou v případě pouze pořízeného majetku všechny údaje, vyjma inventárního čísla, v případě zařazeného majetku všechna účetně nedůležitá data, případně vybraná účetně důležitá data a v případě vyřazeného majetku to nejsou žádná data.



Sem patří následující operace:

- Zařazení majetku
- Vyřazení majetku
- Vložení technického zhodnocení
- Aktivace TZ

### **Zařazení majetku**

Aktérem je autorizovaný uživatel, který chce daný majetek zařadit do užívání. Systém tuto operaci zaznamená do pohybů a označí majetek za zařazený.

### **Vyřazení majetku**

Aktérem je autorizovaný uživatel, který chce daný majetek vyřadit z užívání. Systém tuto operaci zaznamená do pohybů a označí majetek za vyřazený.

Sem patří následující operace:

- Storno vyřazení

### **Storno vyřazení majetku**

Aktérem je autorizovaný uživatel, který zjistil, že udělal chybu a chce vyřazený majetek znovu zařadit do používání. Systém tuto operaci zaznamená do pohybů a označí majetek jako zařazený.

### **Vložení technického zhodnocení**

Aktérem je autorizovaný uživatel, který chce u existujícího majetku vložit technické zhodnocení. Systém tuto operaci zaznamená na inventární kartu majetku.

Sem patří následující operace:

- Vyhledání souvisejících faktur

### **Aktivace TZ**

Aktérem je autorizovaný uživatel, který chce aktivovat dříve vložené technické zhodnocení. Systém tuto operaci zaznamená do pohybů a zvýší cenu majetku.

Sem patří následující operace:

- Storno aktivace TZ

### **Storno aktivace TZ**

Aktérem je autorizovaný uživatel, který zjistil, že udělal chybu při aktivaci technického zhodnocení a chce toto vrátit zpět. Systém mu to umožní a zaznamená tuto operaci do pohybů.

### **Import majetku**

Aktérem je autorizovaný uživatel, který chce importovat majetek v určitém pevně daném formátu z jiného systému. Systém zkontroluje integritu a validitu těchto dat a v případě žádného problému importuje data do evidence.

### **Vytvoření sestavy**

Aktérem je autorizovaný uživatel, který chce vytvořit určitou sestavu majetku. Systém mu umožní vytvořit vlastní sestavu, nebo mu nabídne z již vytvořených sestav.

Sem patří následující operace:

- Tisk sestavy

### **Tisk sestavy**

Aktérem je autorizovaný uživatel, který chce vytisknout určitou sestavu. Systém pošle na tiskárnu příslušná data.

### **Zálohování**

Aktérem je autorizovaný uživatel, který chce zálohovat data evidence majetku. Systém musí uživateli umožnit automatickou zálohu v zadaných časových intervalech a manuální okamžitou zálohu.

### **Editace ostatních číselníků**

Aktérem je autorizovaný uživatel, který chce editovat účetně nedůležité číselníky. Systém mu to umožní.

### **Editace účetně důležitých číselníků**

Aktérem je uživatel s rolí hlavní účetní, který chce editovat účetně důležité číselníky. Systém mu to umožní po ověření zda má na tuto operaci uživatel právo.

### **Výpočet účetních odpisů**

Aktérem je autorizovaný uživatel, který chce vypočítat účetní odpisy zadaného majetku za stanovený časový interval. Systém ověří systémové nastavení a pokud na to má uživatel právo, jsou odpisy vypočítány a zaznamenány do pohybů.

### **Odhad budoucích odpisů**

Aktérem je autorizovaný uživatel, který chce odhadnout budoucí odpisy určitého majetku ve stanoveném časovém období. Systém zobrazí výsledek odhadu. Tato operace se nikam nezaznamenává.

Sem patří následující operace:

- Odhad daňových odpisů

- Odhad účetních odpisů

### **Zaúčtování pohybů**

Aktérem je autorizovaný uživatel, který chce zaúčtovat zaznamenané pohyby. Systém zaúčtuje zaznamenané pohyby podle aktuální dat do účetnictví.

Sem patří následující operace:

- Storno zaúčtování pohybů

### **Storno zaúčtování pohybů**

Aktérem je autorizovaný uživatel, který si uvědomil, že udělal chybu v zaúčtování pohybů. Systém mu umožní vybrat pohyby, které chce stornovat a tyto zaúčtuje do účetnictví na protiúčty pro stornování.

### **Výpočet daňových odpisů**

Aktérem je uživatel s rolí hlavní účetní, který chce vypočítat daňové odpisy za stanovené období. Systém vypočte daňové odpisy majetku za stanovené období a označí toto období u majetku za daňově odepsané s datem vytvoření odpisu. Odpis se zapíše do pohybů.

### **Storno odpisu**

Aktérem je uživatel s rolí hlavní účetní, který si uvědomil, že udělal chybu při stanovení data kdy provedl daňové odpisy a chce tyto stornovat. Systém mu umožní stornovat všechny daňové odpisy ze zadaného data.

### **Výpočet rozdílu odpisů**

Aktérem je uživatel s rolí hlavní účetní, který chce vypočítat rozdíl mezi daňovými a účetními odpisy za rok. Systém mu toto umožní a zobrazí výsledek na obrazovku, operace se nikam nezaznamenává.

## **4.2.4 Datový model aplikace**

Viz následující strana

<< table >> <b>Systémové nastavení</b> << PK >>=syst.nastavení ID/VARCHAR(20) -nastavení SQL_VARIANT -datový typ/VARCHAR(20) -poznámka NVARCHAR(100)
---

<< table >> <b>SKP</b> << PK >>=SKP ID/INT -kód/VARCHAR(20) -daňová odpis skupina NVARCHAR(3) -popis NVARCHAR(100) -poznámka TEXT
---

<< table, SAP >> <b>Sčít (OCRY)</b> << PK >>=Code NVARCHAR(3)
---

<< table >> <b>Sřídisko</b> << PK >>=sřídisko ID/INT -popis NVARCHAR(100) -účet NVARCHAR(100) -místo NVARCHAR(100) -PSC/VARCHAR(20) -poznámka NVARCHAR(200)
--

<< table, SAP >> <b>Odpovědná osoba (OHEM)</b> << PK >>=empID/INT
---

<< table >> <b>Způsob pořízení</b> << PK >>=způsob pořízení ID/INT -název NVARCHAR(20)
---

<< table >> <b>Způsob vyřazení</b> << PK >>=způsob vyřazení ID/INT -název NVARCHAR(20)
---

<< table >> <b>Kód účtování</b> << PK >>=kód účtování ID/INT -kód účtování INT << FK >>=řídisko INT -typ ceny/VARCHAR(1) -účet MID/VARCHAR(20) -známenko MID/VARCHAR(1) -popis NVARCHAR(100)
--

<< table >> <b>Kód pohybu</b> << PK >>=kód pohybu ID/INT -kód pohybu INT << FK >>=kód účtování INT -popis NVARCHAR(100) -poznámka NVARCHAR(200)
---

<< table >> <b>Typ pohybu</b> << PK >>=typ pohybu ID/INT -název NVARCHAR(20)
---

<< table >> <b>Inventurní karta majetku</b> << PK >>=inventurní číslo/VARCHAR(20) -číslo souboru NVARCHAR(20) << FK >>=typ majetku INT -popis NVARCHAR(100) -zařazení/VARCHAR(1) -výběz/VARCHAR(1) -DPH NUMERIC(19,6) -vstupní cena daňově NUMERIC(19,6) -opravný daňově NUMERIC(19,6) -opravný účetní NUMERIC(19,6) -zůstatková cena daňově NUMERIC(19,6) -zůstatková cena účetní NUMERIC(19,6) -datum porizení SMALLDATETIME -datum zařazení SMALLDATETIME << FK >>=způsob pořízení INT -výrobní číslo NVARCHAR(100) << FK >>=SKP INT -odpovídá se daňově/VARCHAR(1) -způsob daňového odpisování/VARCHAR(1) << FK >>=daňová odpisová skupina INT -zůškový odpis v 1. roce/VARCHAR(1) -datum zahájení daň odpisování SMALLDATETIME -technické zhodnocení celkem NUMERIC(19,6) -počet let odpisovaných ze ZVC INT -počet let odpisovaných ze ZVC INT -poslední daň. odpis v roce INT -odpovídá se účetně/VARCHAR(1) << FK >>=Účetní odpisová skupina INT -měsíční odpisové procento NUMERIC(7,4) << FK >>=kód pohybu účetních odpisů INT -datum zahájení účet. odpisování SMALLDATETIME -datum vyřazení SMALLDATETIME -důvod vyřazení/VARCHAR(100) << FK >>=způsob vyřazení INT << FK >>=sřídisko INT << FK >>=vibízí uživatel INT -umístění NVARCHAR(200) << FK >>=řídisko INT -rok výroby INT -záruka dc SMALLDATETIME -poznámka TEXT
---

<< table >> <b>Sřídisko</b> << PK >>=sřídisko ID/INT -popis NVARCHAR(100) -účet NVARCHAR(100) -místo NVARCHAR(100) -PSC/VARCHAR(20) -poznámka NVARCHAR(200)
--

<< table, SAP >> <b>Odpovědná osoba (OHEM)</b> << PK >>=empID/INT
---

<< table >> <b>Způsob pořízení</b> << PK >>=způsob pořízení ID/INT -název NVARCHAR(20)
---

<< table >> <b>Způsob vyřazení</b> << PK >>=způsob vyřazení ID/INT -název NVARCHAR(20)
---

<< table >> <b>Kód pohybu</b> << PK >>=kód pohybu ID/INT -kód pohybu INT << FK >>=kód účtování INT -popis NVARCHAR(100) -poznámka NVARCHAR(200)
---

<< table >> <b>Typ pohybu</b> << PK >>=typ pohybu ID/INT -název NVARCHAR(20)
---

<< table >> <b>Účetní odpisová skupina</b> << PK >>=účetní odpis skupina ID/INT -popis NVARCHAR(100) -dílše skupiny INT << FK >>=typ majetku INT -počet měsíců INT
--

<< table >> <b>Daňová odpisová skupina</b> << PK >>=daňová odpis skupina ID/INT -číslo skupiny NVARCHAR(5) -rok plátování INT -počet let odpisování INT -popis NVARCHAR(100) -procento zvýš. odpisu v 1. roce NUMERIC(4,2) -sazba rov odpisu 1.rok NUMERIC(5,2) -sazba rov odpisu další roky NUMERIC(5,2) -sazba rov odpisu ZVC NUMERIC(5,2) -sazba zrych odpisu 1.rok NUMERIC(5,2) -sazba zrych odpisu další roky NUMERIC(5,2) -sazba zrych odpisu ZVC NUMERIC(5,2)
---

<< table >> <b>Typ majetku</b> << PK >>=typ majetku ID/INT -zkratka NVARCHAR(5) -název NVARCHAR(50)
---

<< table >> <b>Třída</b> << PK >>=řída ID/INT -řída INT -název NVARCHAR(100)
--

<< table >> <b>Dodavatel (OCRD)</b> << table, SAP >> -Dodavatel NVARCHAR(15) << PK >>=CardCode NVARCHAR(15)
---

<< table >> <b>Doklady</b> << PK >>=inventurní číslo INT << PK >>=faktura INT
--

<< table >> <b>Účetní odpisová skupina</b> << PK >>=účetní odpis skupina ID/INT -popis NVARCHAR(100) -dílše skupiny INT << FK >>=typ majetku INT -počet měsíců INT
--

<< table >> <b>Daňová odpisová skupina</b> << PK >>=daňová odpis skupina ID/INT -číslo skupiny NVARCHAR(5) -rok plátování INT -počet let odpisování INT -popis NVARCHAR(100) -procento zvýš. odpisu v 1. roce NUMERIC(4,2) -sazba rov odpisu 1.rok NUMERIC(5,2) -sazba rov odpisu další roky NUMERIC(5,2) -sazba rov odpisu ZVC NUMERIC(5,2) -sazba zrych odpisu 1.rok NUMERIC(5,2) -sazba zrych odpisu další roky NUMERIC(5,2) -sazba zrych odpisu ZVC NUMERIC(5,2)
---

<< table >> <b>Typ majetku</b> << PK >>=typ majetku ID/INT -zkratka NVARCHAR(5) -název NVARCHAR(50)
---

<< table >> <b>Třída</b> << PK >>=řída ID/INT -řída INT -název NVARCHAR(100)
--

<< table >> <b>Doklady</b> << PK >>=inventurní číslo INT << PK >>=faktura INT
--

<< table, SAP >> <b>Doklady</b> << PK >>=inventurní číslo/VARCHAR(20) << PK >>=dodavatel NVARCHAR(15)
--

<< table >> <b>Účetní odpisová skupina</b> << PK >>=účetní odpis skupina ID/INT -popis NVARCHAR(100) -dílše skupiny INT << FK >>=typ majetku INT -počet měsíců INT
--

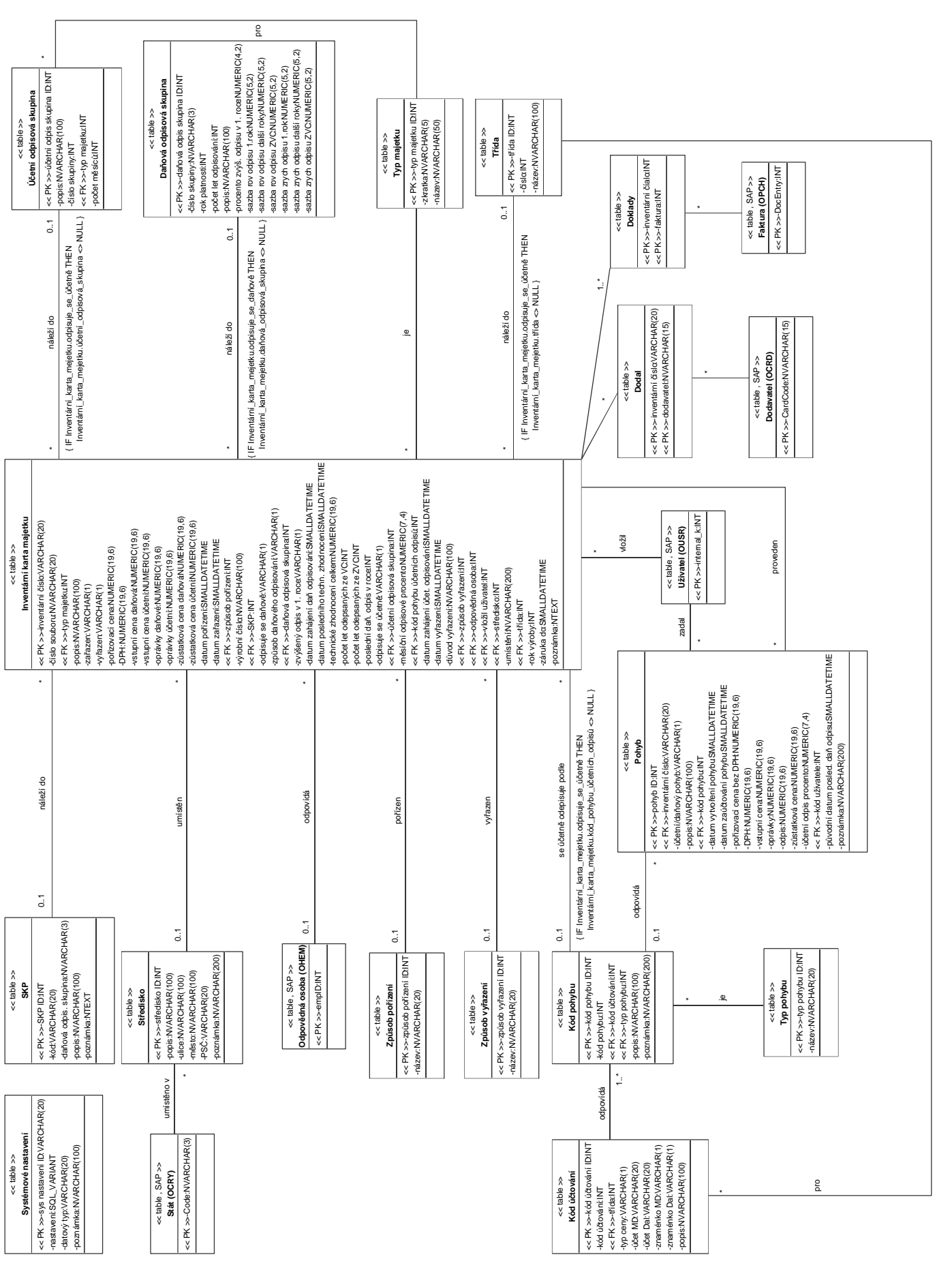
<< table >> <b>Daňová odpisová skupina</b> << PK >>=daňová odpis skupina ID/INT -číslo skupiny NVARCHAR(5) -rok plátování INT -počet let odpisování INT -popis NVARCHAR(100) -procento zvýš. odpisu v 1. roce NUMERIC(4,2) -sazba rov odpisu 1.rok NUMERIC(5,2) -sazba rov odpisu další roky NUMERIC(5,2) -sazba rov odpisu ZVC NUMERIC(5,2) -sazba zrych odpisu 1.rok NUMERIC(5,2) -sazba zrych odpisu další roky NUMERIC(5,2) -sazba zrych odpisu ZVC NUMERIC(5,2)
---

<< table >> <b>Typ majetku</b> << PK >>=typ majetku ID/INT -zkratka NVARCHAR(5) -název NVARCHAR(50)
---

<< table >> <b>Třída</b> << PK >>=řída ID/INT -řída INT -název NVARCHAR(100)
--

<< table >> <b>Doklady</b> << PK >>=inventurní číslo INT << PK >>=faktura INT
--

<< table, SAP >> <b>Doklady</b> << PK >>=inventurní číslo/VARCHAR(20) << PK >>=dodavatel NVARCHAR(15)
--



# 5 Implementace

## 5.1 Použité technologie

### 5.1.1 Přístup k datům

Při volbě přístupu k datům jsem měl na výběr ze tří možných možností:

- přímý přístup do databáze
- DI Server
- DIAPI

První možnost jsem ze začátku testoval, ale na poslední chvíli se od této možnosti odvrátil z následujících důvodů. Pokud bych chtěl, aby aplikace byla univerzální a fungovala na všech třech databázových technologiích podporovaných systémem SAP BI, musel bych naimplementovat přístup, založení databázových tabulek a operace nad nimi pro všechny tyto databáze, což vyžaduje minimálně nastudovat odlišnosti syntaxe SQL jazyka od jeho standardu u všech tří typů serverů. Dalším důvodem je, že firma SAP ve svých materiálech obecně nedoporučuje přímý přístup do databáze a to hned z několika důvodů. Například je nepřipustné, aby se tímto způsobem měnily systémové tabulky. Nejsou totiž kontrolována žádná integritní omezení a uživatelská oprávnění. Dále se v budoucích verzích může změnit struktura databáze a moduly, které budou na tomto principu postaveny najednou přestanou fungovat.

Druhou možnost jsem zamítl, protože je to spíše robustní nástroj určený pro aplikace typu server-klient, kde na jedné straně je jeden server a na straně druhé je mnoho klientů, kteří potřebují připojení současně, což není můj případ.

Po vyzkoušení a vyloučení předchozích možností tedy zbyla pouze poslední a tou je DIAPI. Přístup pomocí DIAPI lze implementovat poměrně elegantně i když jsme limitováni na vývojové prostředí, které podporuje COM technologii. Přístup k datům pomocí DIAPI lze implementovat dvěma způsoby. Prvním je vykonávání SQL dotazů na databázi pomocí jistých metod objektů a následné procházení vrácených záznamů. Tento způsob stále předpokládá znalost metody či odlišností dotazování ve třech různých databázových systémech. Druhým způsobem je použití metod objektu SBOB či jiných tzv. business objektů, které nabízejí přístup k vybraným, nejčastěji používaným objektům. Tyto metody nejsou závislé na typu databázového serveru, ani na změnách struktury uložených dat v budoucích verzích systému, jeví se tudíž jako nejlepší řešení. Bohužel ne ke všem objektům v systému existují objekty, které je zpřístupňují v DIAPI. Proto je v některých případech nutnost použít první způsob a zadávat SQL příkazy pro vyhledání požadovaných dat. Uložení dat tímto způsobem by se mělo používat jen opravdu v nutném případě, s jistotou že neporušíme žádná integritní omezení. Příkladem, kdy se tento způsob používá zcela běžně je zápis dat

do uživatelských tabulek, což jsou tabulky vytvořené uživatelem přímo v systému nebo programátorem pomocí DI-API pro vlastní data.

## 5.1.2 Integrace grafického uživatelského rozhraní

Pokud chceme integrovat GUI naší aplikace přímo do systému SAP B1, nezbyvá nám nic jiného než použít nástroje UI API. Celé UI API funguje na základě událostmi řízeného programování. Programátor má možnost odchytit jakoukoliv událost vyvolanou určitým prvkem GUI, na událost zareagovat a vrátit ji zpět systému ke zpracování nebo ji zablokovat. Pomocí kolekcí různých druhů a identifikací pomocí unikátních identifikátorů v rámci spuštěného klienta SAP B1, jsou zpřístupněny prvky GUI v tomto klientovi a lze je tak ovlivňovat, nebo přidávat vlastní. Vlastní prvky (např. tlačítko) můžeme přidávat na již existující formuláře, nebo na vlastní vytvořené formuláře. Vlastní formuláře lze vytvořit programově pomocí UI API, nebo pomocí modulu Screen Painter, který je dodáván spolu s SDK. Tento modul nám umožní nakreslit vlastní formulář a uložit jej do xml souboru, který pak lze programově v UI API načíst a otevřít v aplikaci. Se Screen Painterem se pracuje podobně, jako s jinými WYSIWYG editory pro tvorbu formulářů, avšak oproti programovým možnostem vytvoření formuláře je trochu omezený. Některé prvky v něm prostě nelze vytvořit a proto nám nezbyde nic jiného, než celý formulář vytvořit buď programově, nebo pomocí Screen Painteru, uložit jej do souboru, který programově načteme a programově přidáme zbývající prvky formuláře.

Používání UI API a událostmi řízeného programování v něm, by nám mělo připadat povědomé a intuitivní. Bohužel v některých případech toto neplatí. Jako příklad uveďme třeba tlačítko pro posuv vpřed a zpět v záznamech zobrazených na formuláři, které používá snad každý existující formulář v systému. V dokumentaci se dočteme, že tato tlačítka jsou ve skutečnosti položky menu a lze s nimi tak pracovat. Naše intuice nám říká, že bychom u vlastního formuláře mohli tlačítka použít následujícím způsobem. Při načtení formuláře tlačítka aktivovat (standardně jsou deaktivovaná) a zachytávat události vyvolané jejich stisknutím uživatelem a pomocí vlastní reže posouvat záznamy na formuláři. Bohužel tady narazíme. Používání těchto tlačítek není tak intuitivní a jednoduché jak se na první pohled mohlo zdát. Tlačítka lze používat pouze s využitím uživatelsky definovaného objektu (UDO) o kterém jsem se zmínil v kapitole o SDK a jeho použití již není tak jednoduché, jako naše domněnka o zachytávání událostí.

## 5.1.3 Vývojové prostředí

Při zvolení DI a UI API pro přístup k datům a GUI systému jsem byl limitován na výběr vývojového prostředí, které podporuje technologii COM. Výběr byl poté jasný a vybral jsem Microsoft Visual Studio.Net v2003 Professional. Vybráno bylo zejména z následujících důvodů:

- v případě problémů s implementací pomocí SDK, poskytuje SAP podporu pouze k tomuto vývojovému prostředí
- jde o moderní, kvalitní a uživatelsky přátelské prostředí
- zadavatel projektu má k prostředí zakoupenou licenci
- prostředí podporuje technologii .NET, jejíž použití se jeví vzhledem k cílovým operačním systémům jako ideální.

## 5.2 Vlastní implementace projektu

Projekt jsem implementoval v jazyce Visual Basic.Net v prostředí Microsoft Visual Studio.Net 2003 Professional. Testoval jsem jej v operačním systému Microsoft Windows XP Professional s nainstalovaným Microsoft .Net Framework v1.1, SAP Business One 2004C, SAP Business One SDK Development v6.7.

### 5.2.1 Soubory projektu

Projekt obsahuje následující soubory:

- **Dokumentace.rtf, Dokumentace.pdf.** tento soubor s dokumentací
- **ReadMe.txt** .. textový soubor s instrukcemi pro přeložení a spuštění projektu a s kontaktem na prodejce použitých softwarových technologií
- **Evidence\_Majetku.sln** .. hlavní soubor projektu pro otevření ve Visual Studiu.Net
- **AppMain.vb** .. zdrojový kód pro inicializaci a spuštění aplikace
- **HIM.vb** .. zdrojový kód třídy pro evidenci majetku
- **UserTables.sql** .. SQL skript pro vytvoření uživatelských tabulek v databázi
- **IKM.srf** .. formulář inventární karty ve formátu XML
- ostatní .. ostatní soubory, vygenerované použitým vývojovým prostředím

### 5.2.2 Popis aplikace

Po spuštění aplikace se v zapnutém systému SAP B1 objeví v nabídce hlavního menu (standardně umístěné po levé straně) na konci položka menu s názvem Evidence majetku. Vybráním této položky se otevře podmenu s prozatím jednou položkou Inventární karta majetku. Po jejím vybrání se otevře jednoduchý formulář pro inventární kartu majetku. Formulář prozatím zobrazuje pouze základní data majetku, která nejsou nijak vázána na účetní či daňové operace s majetkem. Formulář by měl umožnit přidání nového, smazání nebo aktualizaci právě zobrazeného záznamu a pohyb mezi záznamy. Z tohoto funguje zatím bohužel pouze smazání zobrazeného záznamu a pohyb mezi záznamy s využitím tlačítek k tomu určených v dolní části formuláře.

### **5.2.3 Omezení**

Aplikace je zatím funkčně velmi omezena. Prakticky umožňuje pouze zobrazit existující majetek, listovat mezi jeho záznamy a smazat aktuálně zobrazený záznam. Aplikace je v tomto stavu samozřejmě prakticky zcela nepoužitelná a může sloužit pouze jako základ pro další práci na projektu.



## 6 Závěr

Cílem ročníkového projektu bylo seznámit se s problematikou evidence investičního majetku firmy dle platné české legislativy, architekturou systému SAP<sup>®</sup> Business One, technologiemi pro tvorbu modulů do systému SAP<sup>®</sup> Business One a navrhnout a implementovat část modulu pro evidenci investičního majetku v systému SAP<sup>®</sup> Business One.

V první fázi jsem nastudoval problematiku, českou legislativu a požadavky kladené na evidenci investičního majetku firmy. Krátké pojednání o problematice evidence majetku z hlediska české legislativy je v kapitole 2. Požadavky kladené na funkčnost jsou uvedeny v kapitole 4.1.

V druhé fázi jsem začal se studiem vývojového balíčku SDK (Software Development Kit), který je k produktu nabízen firmou SAP pro její obchodní partnery. Nastudoval jsem k čemu se jednotlivé komponenty SDK používají, rozhodnul které budu potřebovat pro splnění projektu a tyto nastudoval podrobněji.

Ve třetí fázi jsem analyzoval požadavky kladené na výsledný modul a navrhnul jeho model. Při analýze a návrhu modelu jsem bral inspiraci zejména z konkurenčních systémů a z rozhovorů s daňovým poradcem, který se mimo jiné zabývá i problematikou evidence majetku. Výstupem jsou zejména diagram případů užití v kapitole 4.2.2 a datový model databáze v kapitole 4.2.4.

Ve čtvrté fázi jsem se dohodl s vedoucím projektu Ing. Jaroslavem Rábem, že se pokusím implementovat část navrženého modelu. Tato část je vyznačena obdélníkem v diagramu případů užití v kapitole 4.2.2. Výsledkem je přiložená aplikace, u které se mi z časových důvodů nepodařilo dosáhnout kompletní požadované funkčnosti a je implementována pouze část, podrobnosti lze nalézt v kapitole 5.2.2. a 5.2.3.

Projekt bylo možné implementovat s dvěma různými přístupy k vlastním uživatelským datům.

První možností bylo použít vlastní přístup k databázi, například pomocí nástrojů v prostředí Visual Studio.Net jako jsou ODBCConnection a ODBCDataAdapter. V tomto případě by jsem pracoval s databází přímo pomocí SQL příkazů a pokud by jsem chtěl zajistit funkčnost modulu pro všechny databázové servery podporované systémem, musel by jsem nastudovat minimálně odlišnosti implementace jazyka SQL v jednotlivých systémech od standardu. Vlastní databázové tabulky vytvoříme pomocí DDL SQL příkazů přímo v databázi. Ze systému SAP tedy nebudou uživatelé o žádných tabulkách vědět a bez přímého zapisování SQL příkazů, nebo použití speciálních nástrojů pro správu databáze nebudou mít možnost zasahovat do těchto většinou citlivých dat. V případě potřeby přístupu k datům systému lze použít objektů obsažených v DI API.

Druhou možností bylo použít výhradně DI API a firmou SAP doporučený postup tvorby modulů. V tomto případě se tabulky vytvoří jako uživatelsky definované tabulky systému buď programem v DI API, nebo pomocí vizuálního nástroje přímo v systému SAP<sup>®</sup> Business One. K těmto

tabulkám lze poté přistupovat přes objekty v DI API a na první pohled je relativně jednoduše je s datovými poli na formuláři. Přímo s databází se tedy na první pohled vůbec nepracuje.

Jak jsem zjistil, asi všechny moduly, které jsem měl možnost do systému vidět, jsou implementovány podle druhé možnosti. Na poslední chvíli jsem se tedy rozhodl, že modul implementuji také pouze s využitím DI API. Bohužel jsem ještě tehdy netušil, že tady bude kámen úrazu mé práce. Práce v DI API s uživatelskými tabulkami totiž není tak jednoduchá, jak se na první pohled zdá. Už jen programové vytvoření tabulek je několikanásobně delší než podobný kód zapsaný v SQL jazyce. Pokud bychom chtěli vytvořit tabulky s pomocí vizuálních nástrojů v systému, nelze je zase jednoduše přenést do jiné databáze. Navíc nelze tabulky dost dobře propojovat pomocí primárních a cizích klíčů, což je jeden ze základů relačních databází a integritu dat je nutné zajišťovat vlastní režii v programu. Jak jsem následně zjistil, až na operaci čtení z uživatelských tabulek, je stejně třeba používat SQL dotazy. Jedna ze zdánlivých výhod tohoto přístupu se tedy ztrácí. Jako největší problém který v případě použití uživatelských tabulek vidím, je, že jsou přístupné každému uživateli systému v jedné z položek menu a každý uživatel tak může prohlížet záznamy v těchto tabulkách a co víc, může je dokonce měnit a mazat. Toto považuji za velké bezpečnostní riziko, hlavně co se týče fungování modulu.

V projektu bych chtěl dál pokračovat v diplomové práci. Hlavním cílem diplomové práce by mělo být dokončení implementace modulu, aby byl v praxi použitelný. Modul by měl podporovat všechny navržené funkce. Součástí dokončení implementace modulu by mělo být i přehodnocení použitého přístupu k vlastním datům s ohledem na bezpečnost, stabilní běh modulu a programovací komfort.

# Literatura

- [1] PELC, V.: *Daňové odpisy po novele 2005*. Praha : GRADA Publishing, a.s., 2005. ISBN 80-247-1071-4
- [2] PRUDKÝ, P.; LOŠŤÁK, M.: *Hmotný a nehmotný majetek v praxi*. Olomouc : ANAG, 2005. ISBN 80-7263-262-0
- [3] BRŮHA, L.: *Začínáme programovat v jazyce Visual Basic.net*. Praha : Computer Press, 2002. ISBN 80-7226-785-X
- [4] *SAP Business One: Profitabilita a růst pro malé a střední firmy* [online]. c2004 [cit. 2005-01-02]. Dostupný z WWW: <<http://www.sap.com/cz/solutions/smb/brochures/index.epx>>
- [5] *SAP Business One zvyšuje efektivitu malým a středním firmám* [online]. c2004 [cit. 2005-01-02]. Dostupný z WWW: <<http://www.sap.com/cz/solutions/smb/brochures/index.epx>>
- [6] *SAP Business One: TB1000* [online]. c2004 [cit. 2005-01-02]. Dostupný z WWW: <<https://websmp102.sap-ag.de/smb>>
- [7] *SAP Business One: TB1300* [online]. c2004 [cit. 2005-01-02]. Dostupný z WWW: <<https://websmp102.sap-ag.de/smb>>