



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Řešení inventarizace pro podnikový informační systém

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika

Studijní obor: 1802T007 – Informační technologie

Autor práce: **Bc. Martin Brotánek**

Vedoucí práce: doc. RNDr. Pavel Satrapa, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

The inventarization solution for corporate information system

Master thesis

Study programme: N2612 – Electrical engineering and informatics

Study branch: 1802T007 – Information technology

Author: **Bc. Martin Brotánek**

Supervisor: doc. RNDr. Pavel Satrapa, Ph.D.



ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Martin Brotánek**
Osobní číslo: **M15000158**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Informační technologie**
Název tématu: **Řešení inventarizace pro podnikový informační systém**
Zadávací katedra: **Ústav nových technologií a aplikované informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Proveďte rešerši dostupných inventarizačních systémů a jejich schopností.
2. Navrhněte systém pro inventarizaci umožňující práci v on-line i off-line režimu.
3. Implementujte navržené řešení - serverovou část i aplikaci pro čtečky kódů.
4. Vyvinutý systém otestujte.

Rozsah grafických prací: **dle potřeby**
Rozsah pracovní zprávy: **cca 60 stran**
Forma zpracování diplomové práce: **tištěná/elektronická**
Seznam odborné literatury:

[1] SCHNEIDER, Thorsten., Eric. WESTENBERGER a Hermann. GAHM. **ABAP development for SAP HANA. Boston: Galileo Press, 2014. ISBN 9781592298594.**


[2] MURRAY, Martin. **Materials management with SAP ERP: functionality and technical configuration. 3rd ed., updated and rev. Boston: Galileo Press, c2011. ISBN 9781592293582.**

[3] WILLIAM D. HASEMAN AND ROSS HIGHTOWER. **Mobile development for SAP. s.l.: SAP Press, 2013. ISBN 9781592294480.**


[4] KALESKE, Stephan. **SAP query reporting: practical guide. Boston: SAP Press, 2011. ISBN 9781592293650.**

Vedoucí diplomové práce: **doc. RNDr. Pavel Satrapa, Ph.D.**
Ústav nových technologií a aplikované informatiky

Datum zadání diplomové práce: **1. března 2018**
Termín odevzdání diplomové práce: **14. května 2018**


prof. Ing. Zdeněk Pliva, Ph.D.
děkan




Ing. Josef Novák, Ph.D.
vedoucí ústavu

V Liberci dne 1. března 2018

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 4. 9. 2018

Podpis: 

Poděkování

Rád bych poděkoval doc. RNDr. Pavlu Satrapovi, Ph.D. za odborné vedení. Dále děkuji firmě Pregis, a.s. za poskytnutí přístupu k hardwaru a softwaru nezbytného pro vznik této práce.

Abstrakt

Tato práce se zabývá procesem inventarizace z hlediska podnikového informačního systému. Popisuje obecné principy procesu a jejich řešení v systému SAP. Stěžejním tématem je rozšíření systému o práci se čtečkami čárových kódů. Práce obsahuje rešerši dostupných řešení následovanou návrhem a implementací vlastního systému. Zdrojové kódy aplikací pro systém SAP i mobilní terminály jsou součástí CD přílohy.

Klíčová slova: Inventarizace, inventura, ERP, SAP, čárové kódy, čtečky čárových kódů, mobilní terminály

Abstract

This thesis deals with the inventory process from the point of view of the enterprise information system. It describes the general principles of the process and its solution in the SAP system. The key issue is the extension of the system by using barcode readers. The work contains a search of available solutions followed by design and implementation of own system. Application source codes for both SAP and mobile terminals are included in the attachment CD.

Keywords: Inventory, ERP, SAP, barcodes, barcode readers, mobile terminals

Obsah

Seznam zkratek	12
1 Úvod	13
2 Inventarizace majetku	14
2.1 Inventura	15
2.1.1 Fyzická inventura	15
2.1.2 Dokladová inventura	15
2.1.3 Inventurní soupis	16
2.2 Značení a identifikace	16
2.2.1 Čárové kódy	17
2.2.2 RFID	20
3 Podnikový ERP systém	21
3.1 SAP	21
3.1.1 Programovací jazyk ABAP	22
4 Dostupná řešení	24
4.1 Standardní inventarizace v ERP SAP	24
4.2 Inventura pomocí mobilních terminálů	25
4.2.1 Řešení od společnosti EPRIN	26
4.2.2 Systém LES společnosti Sabris	26
4.2.3 mStock od KCT Data	27

5	Návrh inventarizace pomocí mobilních terminálů	28
5.1	Návrh procesu	29
5.1.1	Založení inventury	31
5.1.2	Příprava inventurních seznamů	31
5.1.3	Fyzické počítání a záznam výsledků	32
5.1.4	Zanesení výsledků počítání do SAP	33
5.1.5	Přehled stavu a kontrola výsledků počítání	33
5.1.6	Označení položky pro opětovné přepočítání	34
5.1.7	Zaúčtování rozdílů	35
5.1.8	Nálezy	35
5.2	Uživatelské účty pro mobilní terminály	38
5.2.1	Datový model pro uživatelské účty	38
5.3	Oprávnění v systému SAP	40
5.4	Aplikace v SAP ERP	41
5.4.1	Správa uživatelských účtů	41
5.4.2	Řídicí program	42
5.4.3	Rozhraní pro RFC volání	44
5.5	Aplikace pro mobilní terminály	45
6	Implementace navrženého systému	47
6.1	Aplikace v SAP ERP	47
6.1.1	Datový model	47
6.1.2	Správa uživatelských účtů	52
6.1.3	Pomocná třída ZMBCINV	53
6.1.4	Řídicí aplikace	56
6.1.5	Rozhraní systému	61
6.2	Aplikace pro mobilní terminály	62
6.2.1	Návrhový vzor MVVM	63
6.2.2	Struktura aplikace	64

6.2.3	Konfigurační soubor	64
6.2.4	Databáze a datový model	65
6.2.5	Snímání čárových kódů	67
6.2.6	Přístup do SAP	70
6.2.7	Uživatelské rozhraní	70
7	Testování	74
7.1	Testy procesu neproškolenými subjekty	74
8	Závěr	77
	Literatura	79
A	Obsah přiloženého CD	82

Seznam obrázků

2.1	Kód EAN-13	18
2.2	Kód EAN-128 s aplikačními identifikátory	18
2.3	QR kód	19
5.1	Proces inventarizace	30
5.2	Zjednodušený datový model uživatelských účtů pro terminály	39
6.1	Datový model v systému SAP	51
6.2	Výběrová obrazovka programu pro správu uživatelů	52
6.3	Hlavní obrazovka správy uživatelů	52
6.4	Ukázka kontroly objektu oprávnění	53
6.5	Výběrová obrazovka řídicí aplikace	57
6.6	Ukázka implementace výběrové obrazovky	57
6.7	Hlavní obrazovka řídicí aplikace	58
6.8	Řídicí program - historie zadávání	59
6.9	Ukázka použití BAPI pro zaúčtování rozdílů	60
6.10	Návrhový vzor MVVM	63
6.11	Ukázka implementace repositáře	66
6.12	Ukázka vytvoření instancí generického repositáře	66
6.13	Ukázka uložení dat pomocí repositáře	66
6.14	Algoritmus snímání čárových kódů s využitím parserů	68
6.15	Ukázka parserů čárových kódů	69
6.16	Ukázka použití generické zadávací obrazovky	73

Seznam tabulek

4.1	SAP transakce pro inventarizaci zásob	25
5.1	Přehled navržených rozšíření SAP inventarizace	31
6.1	Specifikace terminálu Zebra MC3200	63

Seznam zkratek

SAP	Název firmy, akronym - Systems, Applications and Products
ERP	Enterprise Resource Planning, Plánování podnikových zdrojů
EAN	European Article Number, Evropské číslo artiklu
RFID	Radio-Frequency Identification, Identifikace objektů s využitím radiofrekvenčních vln
MM	Material Management, Materiálové hospodářství
RFC	Remote Function Call
BAPI	Business Application Programming Interface
DB	Databáze
BC	BarCode, čárový kód

1 Úvod

Inventarizace je proces, kterým by si dle aktuálního zákona měla každá firma alespoň jednou za rok projít. Kromě splnění zákonné povinnosti je přínosem i pro samotný subjekt, který si tímto způsobem ověří reálný stav svého majetku a napraví případné nesrovnalosti v účetnictví.

Ze zákona jsou sice definovány náležitosti a požadavky, které má proces splňovat, nicméně detaily provedení se mohou podnik od podniku lišit. Malé subjekty mohou vést záznamy firmy v papírové či elektronické formě, avšak bez využití robustních nástrojů. Větší podniky vzhledem k objemu spravované agendy většinou využívají některého z komerčních systémů, který zajišťuje jistou jednotnou formu vedených záznamů a přístupu k nim.

Tato práce se zabývá řešením inventarizace v podniku, který využívá pro vedení agendy informační systém, konkrétně od společnosti SAP. Tento systém sice již obsahuje sadu nástrojů usnadňující proces inventarizace, nicméně při zjišťování reálného stavu majetku je nutné si průběžně zapisovat zjištěné skutečnosti na papír a následně celkový výsledek zanést opět do systému.

Cílem práce je nahrazení zápisu údajů do papírových podkladů a jejich následného přepisu do systému využitím zařízení pro snímání čárových kódů. Takové řešení by mělo eliminovat možné chyby vzniklé při zadávání údajů do počítače a zároveň by při průběžném zasílání dat usnadnilo průběžný přehled o aktuálním stavu procesu.

2 Inventarizace majetku

Inventarizaci majetku a závazků definuje zákon o účetnictví č. 563/1991 Sb. Účetní jednotky zjišťují skutečný stav majetku a ověřují, zda zjištěný stav odpovídá stavu majetku v účetnictví. Inventarizaci účetní jednotky provádějí k okamžiku, ke kterému sestavují účetní závěrku [1].

Inventarizace zahrnuje:

- Inventuru – zjištění a zaznamenání skutečného stavu do inventurních soupisů,
- porovnání skutečného stavu majetku a závazků se stavem v účetnictví,
- zjištění, vyčíslení a zaúčtování případných rozdílů.

Inventarizace je tedy označením souboru prací potřebných k ověření správnosti účetnictví, který končí až v momentě vyrovnání případných nesrovnalostí. Oproti tomu **inventura** je pouze jednou z částí tohoto procesu a končí zjištěním skutečného stavu majetku k určitému dni [2].

Zákon o účetnictví vymezuje dva druhy inventarizace:

- Periodická inventarizace – provádí se k okamžiku, ke kterému se sestavuje účetní závěrka jako řádná nebo mimořádná. Obvykle jednou za rok.
- Průběžná inventarizace – smí se provádět pouze u zásob, u nichž se účtuje podle míst jejich uložení nebo hmotně odpovědných osob, a dále u dlouhodobého hmotného movitého majetku, jenž je v soustavném pohybu a nemá stálé místo, kam náleží.[1]

2.1 Inventura

Proces zjišťování skutečného množství majetku se nazývá inventura. Jedná se o stěžejní část procesu inventarizace, na jejíž přesnosti či nepřesnosti závisí průkaznost celého procesu inventarizace.

Inventuru dělíme mezi dva základní druhy definované jako inventura **fyzická** a **dokladová**.

2.1.1 Fyzická inventura

Fyzickou inventurou se zjišťují skutečné stavy převážně majetku hmotné povahy, např. materiál na skladě, zboží v prodejně, hotovost, stroje atd. Zjištění skutečného stavu probíhá spočítáním, vážením, měřením apod., přičemž se vychází z jednotek množství použitých v účetnictví. Ve výjimečných případech může být využit i technický přepočít, pokud zjišťování klasickým způsobem není z technických, ekonomických či bezpečnostních důvodů možné - například volně uložený materiál jako uhlí, písek atd. Veškerá zjištěná skutečnost a případné okolnosti ovlivňující přesnost musí být doložená v inventurních soupisech viz 2.1.3. [2]

2.1.2 Dokladová inventura

V případech, kdy povaha předmětu inventarizace nedovoluje fyzický způsob inventury, je přistoupeno k dokladové inventuře. Důvodem může být například materiál na cestě, zboží na cestě, pronajatý hmotný majetek, jiný majetek, ke kterému není možný fyzický přístup atd. Dokladovou inventurou se rovněž zjišťuje stav majetku nehmotné povahy, u něhož fyzický přístup není možný.

Při této inventuře se skutečný stav prokazuje doložením písemností, jakými jsou například účetní doklady, spisy, smlouvy atd. Dokladová inventura je využitelná i pro hromadně uložený materiál, u kterého není technický přepočít možný [2].

2.1.3 Inventurní soupis

Jelikož je dle zákona o účetnictví nutné v případě potřeby prokázat provedení inventarizace, musí mít účetní jednotka k dispozici patřičné důkazy v písemné podobě. Mezi takové patří například inventurní soupisy, do kterých je v průběhu inventury zaznamenán skutečný stav majetku. Zákon o účetnictví neurčuje jednotnou podobu soupisů, neexistuje povinná šablona. Nicméně účetní jednotka by měla uspořádat soupisy tak, aby umožňovaly snadné porovnání stavů skutečných a v účetnictví, a aby byla zaručena jejich průkaznost. Správnost skutečností na těchto soupisech potvrzuje svým podpisem pracovník, který je odpovědný za provedení inventarizace [2].

Inventurní soupisy jsou tedy účetními záznamy, které musejí obsahovat:

- a) jednoznačné určení majetku,
- b) podpis osoby, která prováděla inventuru,
- c) podpis odpovědné osoby za provedení inventarizace,
- d) způsob provedení inventury,
- e) ocenění majetku k okamžiku ukončení inventury,
- f) okamžik zahájení a ukončení inventury.

2.2 Značení a identifikace

Hmotný majetek bývá mnohdy doplněn pomocným značením, které usnadňuje správnou identifikaci položky. Nezřídka obsahuje i další povinné či volitelné dodatečné informace, jakým může být označení výrobní šarže, které je v některých odvětvích ze zákona vyžadováno (např. v potravinářství) či informaci o množství, počtu účetních jednotek obsažených v balení atd. Takovéto značení bývá mnohdy využíváno pro

další strojové zpracování s přenosem informací do podnikového informačního systému. Mezi značné přínosy zmíněného řešení patří zejména výrazné zrychlení procesu načítání položek a eliminace chyb způsobených lidským faktorem, např. překlepů při zadávání identifikačních čísel uživatelem na klávesnici.

Popsaná identifikace přitom neslouží pouze pro proces inventarizace, resp. inventury, ale je využitelná i pro další procesy ve firmě využívající některého z podnikových ERP systémů. Pod takovými procesy si lze představit například příjem zboží na sklad nebo expedici, kdy je nutné vše zaznamenávat do systému a udržovat patřičné údaje a doklady pro potřeby účetnictví [3].

2.2.1 Čárové kódy

Hlavním účelem čárových kódů je jednoznačná identifikace položek, kterými může být vyráběný produkt, ale i například nakupovaná surovina. Čárové kódy obvykle nenesou komplexní informace o produktu, slouží spíše pro jednoznačnou identifikaci.

Kódy lze obecně rozdělit do dvou základních skupin, kterými jsou:

- jednorozměrný (1D),
- dvourozměrný (2D).

1D kódy

Jednorozměrný čárový kód je tvořen sekvencí čar a mezer s definovanou šířkou. Nositelem informace je obvykle nejen vytištěná čára, ale i mezera mezi nimi [4].

Běžný spotřebitel nejčastěji přijde do styku s kódem **EAN-13** (European Article Number), případně jeho kratší verzí EAN-8. Tento kód slouží k identifikaci zboží (např. v obchodě, identifikace sušenky XY výrobce Z) a nese zpravidla pouze informaci v podobě čísel, jak lze vidět v ukázce na obrázku 2.1.

Číslice v kódu EAN-13 jsou rozděleny do následujících částí:



Obrázek 2.1: Kód EAN-13

- Systémové číslice - počáteční dvě až tři číslice identifikují zemi, kde je zaregistrovaný výrobce, respektive autorizovaného národního registrátora kódů (v ČR jím je společnost GS1 Czech Republic, identifikátor 859),
- kód výrobce - čtyři až pět číslic v závislosti na počtu systémových číslic,
- kód výrobku - 5 číslic,
- kontrolní číslice, jelikož se jedná o samodetekující kód. Její výpočet je proveden sečtením číslic na lichých pozicích spolu s trojnásobkem číslic na sudých pozicích. Kontrolní číslicí je hodnota, kterou je nutné k výsledku přičíst pro zaokrouhlení na celé desítky (směrem nahoru).

Dalším častým zástupcem je kód **CODE 128**. Tento kód umožňuje díky své délce kromě základní identifikace zakódovat i další podstatné informace o výrobku, např. datum výroby, výrobní dávku, rozměry atd. To je možné díky délce až 96 alfa-numerických znaků. Na rozdíl od kódu EAN nemá jasně daná pravidla a lze si teoreticky do kódu zapsat informace libovolně [4].



Obrázek 2.2: Kód EAN-128 s aplikačními identifikátory

Lze se setkat i s označením **UCC/EAN128**, což je ve skutečnosti speciální varianta CODE 128, kdy každá informace obsahuje svůj aplikační identifikátor (viz obr. 2.2). Tento identifikátor říká, jaký druh informace za ním následuje (číslo šarže,

datum trvanlivosti atd.). Výhodou tohoto druhu kódu je jeho standardizace, tudíž různé systémy, které jej podporují, by měly být schopny údaje správně identifikovat [5].

2D kódy

Jak již označení tohoto druhu kódu napovídá, jedná se o dvourozměrné kódy, někdy též nazývané jako datové matice. Jedná se většinou o čtvercový útvar vyplněný kostičkami, které reprezentují jednotlivé bity zakódované informace, ohraničené útvary, které umožňují určit odkud kam sahají informace.

Dvourozměrné provedení umožňuje uložit na malé ploše větší množství informací než klasické 1D čárové kódy. Většina standardů umožňuje uložení přibližně 3000 ASCII znaků, jeden ze zástupců - QR kód (obr. 2.3) - například umožňuje uložení až 4296 znaků nebo 7089 číslic.



Obrázek 2.3: QR kód

QR kódy bývají často používané pro uložení odkazu na webovou stránku, která je otevřená po rozpoznání kódu aplikací v mobilním telefonu využívající kameru přístroje. Další uplatnění mohou najít například při označování hardware, kdy se do kódu mohou zakódovat další informace, kterými mohou být například výrobní číslo, základní technické údaje, odkaz na stránky výrobce atd. [6]

2.2.2 RFID

RFID využívá malých značek (RFID tagů), které obsahují integrovaný obvod a anténu. Tyto tagy jsou schopné reagovat na radiové vlny vysílané RFID čtecím zařízením. Dokáží odesílat, zpracovávat a ukládat informace [7].

Hlavní přednosti RFID:

- absence nutnosti přímé viditelnosti mezi čtecím zařízením a značkou,
- detekce na větší vzdálenost ve srovnání s čárovými kódy,
- možnost zápisu dat do paměti čipu,
- zpracování více položek současně, čtečky čárových kódů mohou načítat položky pouze po jedné,
- vyšší odolnost v náročném prostředí.

Ačkoliv byla zmíněna řada výhod, nelze označit RFID tagy jako obecné ideální řešení. Je nutné přihlídnout i k nevýhodám, kterými jsou:

- rušení způsobené prostředím - např. přítomností kovů, kapalin, zdrojů elektromagnetického záření,
- mezinárodní standardy ohledně používaných frekvencí, kdy existují rozdíly např. mezi standardy v Evropě a USA,
- vyšší cena v porovnání s čárovými kódy,
- případná porucha na straně čtecího zařízení nebo samotného tagu znamená nemožnost identifikace položky, jelikož tag nenese žádnou informaci čitelnou pro lidské oko [3].

3 Podnikový ERP systém

ERP (Enterprise Resource Planing) systém je nástroj, který pokrývá plánování a řízení klíčových interních podnikových procesů. Jedná se například o oblasti výroby, logistiky, personalistiky, účetnictví atd. Smyslem takového systému je zejména zefektivnění řízení podniku a přístup ke klíčovým informacím v reálném čase.

Základním principem ERP systému je vytvoření jednotného datového úložiště, které obsahuje veškerá klíčová podniková data, ke kterým má současně přístup libovolný počet zaměstnanců. Přístup k informacím je v takovém systému dále řízen na základě přidělených oprávnění, kdy by každý zaměstnanec měl mít možnost prohlížet a upravovat pouze jemu příslušná data [8].

3.1 SAP

Podnikový informační systém SAP, kterého se tato práce týká, je nejrozšířenějším zástupcem existujících ERP systémů. Dle dat dostupných k 19. červenci 2018 má po celém světě více než 404 tisíc zákazníků ve více než 180 zemích [9].

Systém SAP je rozdělen do jednotlivých modulů, kdy každý zastává určitou oblast podnikového systému. Takových existuje celá řada a výhodou tohoto systému je možnost zakoupit si licence jen pro vybrané oblasti, které jsou v konkrétní firmě požadovány.

Pro přibližnou představu, co si lze pod pojmem modul představit, uvedu například:

- HR - Human Resources - personalistika,

- FI - Financial accounting - finanční účetnictví,
- CO - Controlling - kontroling,
- SD - Sales and Distribution - odbyt a prodej,
- MM - Material Management - materiálové hospodářství,
- WM - Warehouse Management - řízené sklady. Nadstavba nad logistickými sklady modulu MM o další členění s detailnějšími informacemi o uskladněných položkách - např. přesné umístění, počet jednotek materiálu v konkrétním balení, rozšíření o skladové příkazy atd.

Inventarizace v této práci je zaměřená právě na předposlední zmíněný modul MM (Materiálové hospodářství). Obecně bychom tento modul mohli charakterizovat tak, že se v tomto modulu řeší vše okolo nakládání s materiály (materiál v SAP terminologii je například i produkt, služba atd.), nákupu, inventarizace, logistických skladů a dalších. [10]

3.1.1 Programovací jazyk ABAP

S ohledem na zadání této práce, jejíž součástí je i vyvinutí vlastního řešení inventarizace v podnikovém systému, je vhodné zmínit programovací jazyk ABAP. Jedná se o jazyk vyvinutý společností SAP, který slouží pro rozšiřování ERP systému, v němž je jeho většina napsaná. Pomocí tohoto jazyka je možné doplňovat a upravovat funkcionalitu existujících SAP programů i vytvářet zcela nové programy a reporty, rozhraní pro RFC volání z jiných systémů atd.

Počátky jazyku sahají do roku 1992, kdy byl uvolněn spolu se systémem SAP R/3. V roce 1999 se dále dočkal dalšího podstatného rozšíření, kterým byla možnost objektově orientovaného programování [11]. Ačkoliv vývoj jazyka probíhá neustále, například v podobě již zmíněného rozšíření o objekty, je stále zachována zpětná kompatibilita.

Základní syntaxe jazyku ABAP je poměrně prostá a podobná jiným známým jazykům. Umožňuje používání klasických IF - ELSE konstrukcí, smyček, víceroz-

měrných polí (nazývané interní tabulky), objektů, obsahuje vlastní SQL jazyk atd. Cílem této práce není popis programovacího jazyka, proto se zde nebudu věnovat detailnímu popisu syntaxe a principů obecně známých z jiných jazyků.

Pro snadnější porozumění části o implementaci je vhodné zmínit několik stěžejních stavebních prvků, v práci hojně využívaných.

- **Funkční modul.** Jedná se o samostatně stojící a spustitelný podprogram, obsahující vstupní a výstupní parametry, kterých může obsahovat teoreticky libovolný počet. Funkční moduly lze volat i z jiných systémů, kterými nemusí být pouze SAP, podporují tedy takzvané RFC volání [12].
- **Skupina funkcí.** Dá se říci, že se jedná o povinnou obálku funkčního modulu. Obvykle sdružuje více funkčních modulů logicky patřících k sobě. Typicky například funkční moduly pro inventarizaci sloučíme do jedné skupiny funkcí [12].
- **BAPI** (Business Application Programming Interface). Jedná se o rozhraní od SAP, jež je používáno pro interakci s aplikační vrstvou systému, implementované jako funkční moduly. BAPI je ucelené řešení, které mimo jiné zajišťuje správný přístup k datovým objektům, kontroly oprávnění a konzistenci dat. Zároveň SAP zaručuje jejich správnou funkčnost i v budoucích verzích systému [13]. Příkladem BAPI může být například v této práci použitý funkční modul s názvem „BAPI_MATPHYSINV_CREATE“, který slouží pro vytvoření inventurního soupisu.
- **Objekt oprávnění.** Jedná se o speciální objekty v SAP sloužící pro přidělování specifických oprávnění jednotlivým uživatelům. Může obsahovat až 10 polí, podle kterých se poté v programech testuje oprávnění. Dále typicky obsahuje pole ACTVT, určující druh aktivity - změna, prohlížení atd. Tyto objekty se dále mohou pro přehlednost a snazší přidělování uživatelům sdružovat do rolí [14] [15]. Teoretickým příkladem může být role skladník, obsahující 10 různých objektů oprávnění, kterou poté lze jednoduše přidělit více uživatelům.

4 Dostupná řešení

4.1 Standardní inventarizace v ERP SAP

Systém SAP v modulu MM již od základu umožňuje inventarizaci zásob. Podporuje periodické i mimořádné, popsané v kap. 2. Jednotlivé kroky procesu jsou rozdělené do dílčích transakcí, jejichž spouštění a úkony, které umožňují, jsou řízeny systémem objektů oprávnění a rolí zmíněných v kap. 3.1.1.

Tento systém inventarizace je sám o sobě dostačující. Jeho velkou nevýhodou je ovšem fakt, že je založen na ručním zaznamenávání výsledků počítání do vytištěných inventurních soupisů s nutností následného přepsání hodnot do počítače. To, kromě negativního vlivu na rychlost celého procesu a možný vznik chyb způsobených „překlepy“ při ručním zadávání, znamená absenci možnosti průběžného sledování stavu probíhající inventury.

Provedení inventarizace skladových zásob v ERP SAP se skládá z následujících kroků:

1. Vytvoření inventurního dokladu,
2. zobrazení nebo změna inventurního dokladu,
3. vytištění inventurního dokladu,
4. vlastní fyzické počítání a zápis výsledků do vytištěných podkladů,
5. zanesení spočítaného množství do inventurního dokladu v SAP,
6. kontrola výsledků počítání,

7. případná oprava počítání,
8. zaúčtování inventurních rozdílů.

Pro každý z těchto kroků (kromě bodu 4. - fyzické počítání) existují v SAP samostatné transakce (tab. 4.1), ke kterým může správce systému libovolně dle potřeb podniku přidělit uživatelům oprávnění k užívání. Nahrazení bodů tři až pět je hlavním předmětem řešení založeného na využití mobilních terminálů, kterému se tato práce věnuje v kapitolách 5 a 6.

Tabulka 4.1: SAP transakce pro inventarizaci zásob

Transakce	Popis
MI24	Přehled inventurních dokladů a výsledků inventury. Tisk
MI31	Hromadné vytvoření inventurních dokladů
MI01	Vytvoření inventurního dokladu pro jednotlivé materiály
MI21	Tisk inventurního dokladu
MI02	Změna / výmaz inventurního dokladu (pokud již neproběhlo počítání)
MI03	Zobrazení inventurního dokladu
MI04	Zápis počítaného množství do dokladu
MI05	Oprava počítání
MI20	Kontrola výsledků inventury
MI07	Zaúčtování inventurních rozdílů

4.2 Inventura pomocí mobilních terminálů

Řešení inventarizace zásob pomocí čteček čárových kódů, respektive mobilních terminálů, existuje na trhu celá řada. Tato řešení jsou dodávaná různými společnostmi a v této části práce shrnu několik jejich zástupců.

Bohužel se mi nepodařilo porovnat ceny těchto nabízených řešení, jelikož dodavatelé neuveřejňují cenovou nabídku na svých stránkách. Je to vcelku pochopitelné, a to nejen z hlediska konkurenčního boje mezi jednotlivými dodavateli řešení,

ale i z důvodů technických. Každá implementace se totiž může lišit svou náročností, ať již z důvodu rozdílných nastavení systémů, požadovaných úprav na míru (např. vlastních tiskových výstupů) či nutnosti uzpůsobení podnikové bezdrátové sítě. Proto je pochopitelné, že společnosti cenovou nabídku před provedením analýzy neuveřejňují.

4.2.1 Řešení od společnosti EPRIN

Řešení společnosti EPRIN spočívá ve vybavení operátorů skladů mobilními terminály Unitech PA962 s operačním systémem Windows 4.2NET, vybaveným snímačem čárového kódu.

Na terminálu je spuštěn upravený internetový prohlížeč, který komunikuje se serverem EprinComm (middleware). Ten poskytuje sadu funkcí pro samotné terminály s přehledem připojených terminálů a dále realizuje volání příslušných BAPI (3.1.1) v SAP. Systém je založen na okamžitém předávání informací mezi terminálem a SAP, což má výhody v aktuálnosti zpracovaných dat na straně terminálu i okamžitým přehledem o stavu skladu v SAP. Tento systém neumožňuje práci v offline režimu.

Systém společnosti EPRIN umožňuje provádění přeskladnění mezi výrobou a skladem, expedici, interní přeskladnění a provedení inventury [16].

Bohužel na stránkách není uvedeno o jaký druh skladů se jedná - zda logistické v modulu MM, kterými se tato práce zabývá, nebo o řízené v modulu WM.

4.2.2 Systém LES společnosti Sabris

Společnost Sabris vyvinula nadstavbu nad řízenými sklady ERP systému SAP. Systém je založen na využití mobilních terminálů, dodavatel na svých stránkách neuvádí konkrétně kterých a s jakým operačním systémem.

Systém pracuje pouze v online režimu. Rovněž jako v předchozím případě umožňuje provádění vícero operací ve skladovém hospodářství, kterými jsou:

- Příjem materiálu k objednávce,

- vychystávání, nebo-li vyskladňování, položek podle různých strategií,
- přeskladnění mezi výrobou a skladem,
- inventuru [17].

4.2.3 mStock od KCT Data

Aplikace mStock obdobně jako konkurenční produkty obsahuje balík nástrojů, které slouží pro řešení celé řady procesů skladového hospodářství podniku.

Aplikace je na rozdíl od předchozích určena nejen pro mobilní terminály se snímači čárových kódů, ale lze využít i tzv. „chytrých“ mobilních telefonů nebo tabletů s kamerou. Kromě obvyklých 1D čárových kódů lze navíc využít i značení pomocí QR kódů. Vyvinuta byla na platformě SAP Mobile a podporuje běžné mobilní OS systémy chytrých telefonů, např. Android, iOS atd.

Dodavatel na svých stránkách neuvádí, zda systém pracuje pouze v online režimu, nebo jestli podporuje i offline práci s daty [18].

5 Návrh inventarizace pomocí mobilních terminálů

Jak již bylo zmíněno v kapitole 4, systém SAP již obsahuje funkcionalitu pro řešení inventarizace zásob. Způsob, kterým ji řeší, může být pro většinu podniků dostačující, nicméně samotná inventura spočívá v průběžném ručním zapisování výsledků počítání do papírových formulářů a jejich následném přepisování odpovědným pracovníkem do počítače.

Za značné nevýhody standardní inventury v systému SAP formou zápisu tužkou na papír a následného přepisování do počítače považují:

- **Rychlost** - pracovník musí dohledat materiál v seznamu položek pro inventarizaci, který může čítat stovky až tisíce záznamů.
- **Chybovost** - jeden materiál může být na inventurním seznamu vícekrát v různých šaržích, které se mohou jen nepatrně lišit. Lehce se může zdát, že se osoba splete a zapíše množství k nesprávné položce.
- **Průběžný přehled** - ve smyslu přehledu o stavu počítání, jaké množství položek již bylo spočítáno a kolik toho ještě chybí. Pokud chce pracovník zjistit, jak velkou část má již hotovou, případně které materiály mu ještě zbývají spočítat, nezbude mu nic jiného, než prolistovat zápisový arch. Zároveň nemá nadřízený pracovník možnost nezávisle sledovat aktuální stav počítání, pokud přímo neosloví osobu provádějící inventuru a neprojde s ní záznamy počítání.

- **Ztráta dat** - jelikož jsou veškeré výsledky počítání zaznamenávány do podkladů v papírové formě až do chvíle, kdy je inventurní počítání hotové a předané k zanesení do systému SAP. V případě ztráty či poškození (např. polítkou tekutinou, která způsobí nečitelnost) zápisových listů není jiný způsob jak zjištěné informace obnovit, než opětovné provedení fyzického počítání.

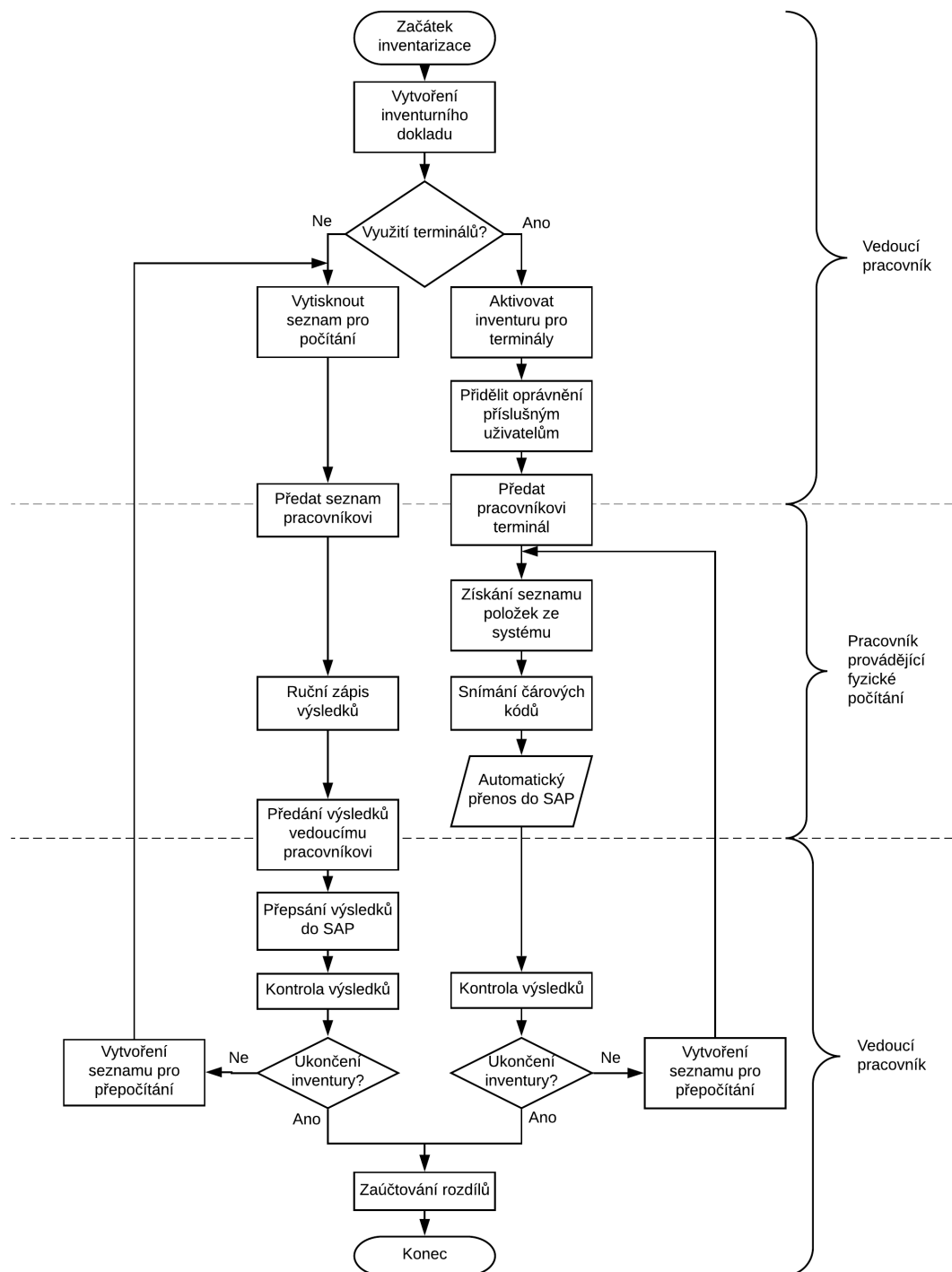
Hlavním cílem této práce tedy je odstranění zmíněných nedostatků rozšířením procesu inventarizace zásob v ERP SAP o možnost provedení inventury použitím mobilních terminálů umožňujících snímání čárových kódů.

5.1 Návrh procesu

Z důvodu již existujícího robustního řešení inventarizace se navržený systém snaží využívat maximum již existujících nástrojů a transakcí, datového modelu, systému oprávnění, existujících BAPI atd.

V systému jsou sice přístupné veškeré zdrojové kódy existujících programů a lze je modifikovat. Nicméně pro rozšíření stávajícího systému jsem se rozhodl vytvořit samostatnou nadstavbu, která nijak nezasahuje do standardních programů SAP z důvodu možných komplikací při budoucích aktualizacích systému, kdy by mohlo docházet ke konfliktům s provedenými úpravami.

V tabulce 5.1 je uveden stručný přehled částí inventarizace, které jsou využité v původní nezměněné formě, využitelné volitelně, či nahrazeny kompletně. Jejich detailnějšímu popisu se poté věnují následující podkapitoly. Na obrázku 5.1 je pomocí základního schématu procesu inventarizace ilustrováno porovnání původního systému a systému v této práci nově vytvořeného.



Obrázek 5.1: Proces inventarizace

Tabulka 5.1: Přehled navržených rozšíření SAP inventarizace

Činnost	Nahrazeno novým řešením
Vytvoření dokladu	Ne
Úprava dokladu	Ne
Vytištění dokladu pro zaznamenání počítání.	Ano, není nutné tisknout. Položky se přenášejí do terminálu.
Fyzické počítání, zápis do archu	Ano, počítání v terminálu
Zanesení počítání do SAP	Ano. Terminál odesílá data do SAP, v SAP nová transakce.
Kontrola výsledků	Volitelně standard nebo nová transakce
Oprava počítání	Volitelně - terminály + nová transakce nebo standardní transakce
Zaúčtování rozdílů	Volitelně - nová transakce nebo standard

5.1.1 Založení inventury

Proces inventarizace začíná vytvořením inventurního dokladu, od kterého se všechny následující kroky odvíjí. Pro založení dokladu již v systému existují velmi propracované transakce (MI01, MI31), obsahující celou škálu nastavení a možností pro výběr materiálů. S ohledem na to, že jsou současné nástroje pro založení dokladů více než dostatečné, tento bod bude ponechán na standardním řešení SAP. Kromě zbytečnosti případného znovu vytváření nového nástroje pro zahájení inventarizace má toto řešení i tu výhodu, že není při nasazení nového procesu do podniku nutné opětovně školit personál, který již standardní nástroje zná. Ze stejných důvodů ponechám na standardu i možnost případných modifikací dokladů.

5.1.2 Příprava inventurních seznamů

Tisk inventurních seznamů pro zápis výsledků fyzického počítání v tomto procesu nahrazují mobilními terminály, do kterých je daný seznam stažen pomocí RFC volání

funkčního modulu v SAP. Místo podpisu pracovníka na záznamovém archu bude do SAP přenášeno společně s daty počítání přihlašovací jméno příslušného pracovníka. Pro případ, že by byl pro účetnictví nezbytný doklad s podpisem v papírové formě, lze opět využít standardní SAP transakce pro vytištění inventurního soupisu.

Pro zpřístupnění možnosti řešit inventuru pomocí mobilních terminálů je navržen nový obslužný program, ve kterém se inventura „aktivuje“, aby ji bylo možno řešit terminály. Tento nový program bude rovněž sloužit pro přidělení dokladu pracovníkovi / pracovníkům, kteří budou mít oprávnění řešit terminálem danou inventuru.

5.1.3 Fyzické počítání a záznam výsledků

Vlastní fyzické počítání bude probíhat snímáním čárových kódů umístěných na jednotlivých položkách a zadáváním nalezeného množství do terminálu.

Zároveň bude v terminálu možnost přidávat záznamy o tzv. „nálezech“ (podrobněji v kap. ??), tedy položkách, které byly na daném skladu nalezeny a nebyly obsaženy v daném inventurním seznamu. Rozhodnutí jak s takovými záznamy naložit bude ponecháno na straně obslužného programu v SAP.

Výsledky počítání jsou ukládány do mobilního terminálu a následně přenášeny do systému SAP, kde probíhá ukládání do nové pomocné databázové tabulky. Tato tabulka je tedy jakousi náhradou tištěného archu, kam by při standardní inventuře pracovník výsledky zapisoval. Odesílání dat probíhá průběžně v tzv. online režimu pokud je dostupné spojení, nebo „na povel“ zvolením příslušné možnosti v terminálu v offline módu.

Pro zabudování offline režimu jsem se rozhodl z toho důvodu, že v případě inventarizace nevidím důvody, které by mu bránily, snad jen nemožnost průběžného monitorování stavu inventury a validace případných nově nalezených položek. Příprava pro offline režim umožní kromě provádění inventury v místech s absencí pokrytí bezdrátovou sítí i nepřetržitou práci u online režimu v případě výpadku spojení, kdy se pouze zastaví synchronizace dat.

5.1.4 Zanesení výsledků počítání do SAP

V kap. 5.1.3 bylo řečeno, že se záznamy o počítání přenášejí do nové databázové tabulky v SAP. Tato přenesená data je nyní zapotřebí přenést do standardního datového modelu SAP inventur. Při standardní inventuře papír-tužka by bylo nutné spustit transakci MI04 a jednotlivé záznamy ručně přepsat k jednotlivým položkám. Dále pokud by zaměstnanec našel danou položku opakovaně na více místech ve skladu, na papíře (a rovněž i v nové tabulce) by se nacházelo vícero záznamů s nalezeným množstvím, které by následně pracovník zadávající množství do systému musel sečíst.

Pro účel přenesení dat do standardního datového modelu proto navrhuji použít nový program, který provede případné součty množství u jednotlivých položek. Toto množství následně bude možné jednoduše automaticky „na jeden klik“ přenést do standardu.

5.1.5 Přehled stavu a kontrola výsledků počítání

Nejdříve si je nutné ujasnit, co je myšleno kontrolou výsledků. Při standardní inventuře nemají pracovníci přehled o účetním množství v systému, provádí tedy tzv. inventuru „na slepo“. O rozdílu mezi spočteným množstvím a účetním je příslušná osoba zadávající hodnoty do systému uvědoměna až po prvotním zadání výsledků počítání do SAP. Poté mu systém zpřístupní údaje o stavu v účetnictví a pracovník má možnost výsledky porovnat a případně provést opětovně přepočítání s případným opětovným zadáním nových hodnot (v SAP je tato skutečnost zaznamenána).

Jak již bylo zmíněno v předešlých kapitolách, v novém návrhu řešení inventarizace je proces doplněn o nové nástroje. Mezi ně patří i program pro předání hodnot spočítaných pomocí čteček čárových kódů (kap. 5.1.4) do standardních datových struktur inventury SAP.

Do tohoto nástroje jsou zaneseny i následující možnosti přehledu o stavu počítání během inventury:

- Zobrazení účetního množství jednotlivých položek před samotným provedením počítání. Tato možnost podléhá speciálnímu oprávnění viz kap. 5.3. Pokud bude mít uživatel příslušnou autorizaci, umožní mu tato funkcionality okamžité porovnání výsledků počítání ještě před zadáním hodnot do standardu SAP. V případě chybějícího oprávnění bude účetní množství zobrazeno až po zadání výsledků počítání, stejně jako by tomu bylo při provádění standardní SAP inventarizace.
- Průběžný přehled aktuálního stavu počítání jednotlivých položek. Tedy co již bylo spočítáno a napočítané množství jako sumy za jednotlivé položky.
- Zobrazení individuálního seznamu postupně zjištěných stavů pro jednotlivé položky, tedy přehled každého zaznamenaného čárového kódu a zadaného množství. Díky takovému detailnímu přehledu bude mít nadřízený pracovník další přehled o tom, co bylo nasnímáno. Obdobně jako by tomu bylo v případě metody papír-tužka, kdy by zaměstnanec postupně na záznamový list zapisoval formou přírůstku nalezené množství (např. žárovka 40W: 10 ks + 8 ks + 3 ks + 15ks,...). Toto navržené řešení umožní zkušenému pracovníkovi jakýsi přibližný odhad chyb při zadávání, kdy bude zaznamenáno v dané situaci nereálné množství. Například kvůli překlepu při zadávání množství, kdy probíhá inventura objemných položek vyskytující se v rozmezí jednotlivých kusů, ale z terminálu přijde spočítané množství 40 kusů, což na první pohled budí podezření na překlep „o nulu“. Zároveň tato možnost může najít své uplatnění v případě opakovaného přepočítávání zmíněného v kap. 5.1.6, kdy umožní porovnání jednotlivých iterací počítání.

5.1.6 Označení položky pro opětovné přepočítání

V případě zjištění, že napočítané stavy položek neodpovídají účetnímu množství, je vhodné mít možnost iniciovat nové počítání pro vybrané položky.

Ve standardní SAP inventuře je viditelný stav účetního množství po prvotním zadání počítání a při zjištění rozdílu je možné opakovat počítání a zanést nové

výsledky do systému. V novém procesu, který v této práci navrhuji, je tato možnost oprav počítání zachována.

Novou součástí systému, kterou zavádím, je možnost provést i případné nové počítání pomocí mobilních terminálů. Pro tuto možnost bude využita na straně SAP nová transakce, respektive ta samá, jako je využita pro přehled inventarizace a zanesení výsledků počítání do standardu SAP, zmíněna v kapitolách 5.1.4 a 5.1.5. Tato nová transakce obsahuje možnost pro označení libovolných položek k opakovanému přepočítání, přehled minulého počítání a možnost propsat nové výsledky do SAP inventury.

5.1.7 Zaúčtování rozdílů

Zaúčtování účetních rozdílů je posledním krokem inventarizace a celý proces ukončuje. Standard SAP má samozřejmě i pro tuto část připravené hotové nástroje. Díky návrhu nového systému, který je nadstavbou nad standardní SAP inventurou a veškeré skutečnosti postupně přenáší do standardního systému, lze pro završení procesu využít standardních transakcí.

Jelikož byla pro většinu částí navržena nová transakce (nový řídicí program), ve kterém lze po založení inventury provádět veškeré následné kroky v postupu inventarizace, je do tohoto nástroje navíc přidána možnost provést zaúčtování inventurních rozdílů. Lze sice použít standard, na který jsou uživatelé zvyklí, ale možnost mít veškeré nástroje na jednom místě považuji za určité zvýšení uživatelského komfortu při práci se systémem.

5.1.8 Nálezy

Nejdříve je nutné definovat si pojem „nález“. Za nález je v této práci považována položka, která není obsažena na inventurním seznamu vystaveném pro určitý sklad, nicméně byla při provádění inventury ve skladu nalezena. Jelikož inventarizace slouží ke zjištění skutečného stavu majetku a napravení případných nesrovnalostí v účet-

nictví (viz kap. 2), je nutné se zabývat i takovýmto nově objeveným majetkem, který je nutné zanést do účetnictví.

Uvažujme situaci, kdy byl materiál *X* zaměstnancem fyzicky přemístěn ze skladu *A* do skladu *B* bez zanesení takovéto skutečnosti do podnikového systému, respektive do účetnictví. Po provedení inventury na skladu *A* by byl zjištěn chybějící materiál. Následně by byla provedena inventura na skladu *B*, kde by sice byl pracovníky materiál *X* nalezen, nicméně jelikož by chyběl na patřičném inventurním seznamu, nebyla by tato skutečnost zanesena do systému. Výsledkem po provedení inventarizace a zaúčtováním veškerých rozdílů by následně byl stav, kdy se sice daný majetek ve společnosti fyzicky nachází, ovšem z pohledu účetnictví by materiál ve firmě chyběl. Takovýto stav je v přímém rozporu s účelem inventarizace, jelikož jedním z cílů celého procesu je právě náprava takovýchto nesrovnalostí.

Řešení v SAP standardu

Při standardní inventuře formou papír-tužka je možné si případné nálezy ručně zaznamenávat na papír, kdy si zaměstnanec poznamená údaje potřebné pro identifikaci materiálu v ERP SAP společně s nalezeným množstvím. Samozřejmě je vysoce pravděpodobné, že nebude mít jistotu, v jaké účetní měrné jednotce je materiál vedený, takže si sám vhodnou jednotku zvolí dle vlastního uvážení dle povahy materiálu (kusy, gramy,...).

Následně pracovník, který má na starosti přepis zjištěných skutečností do transakcí v SAP, rozhodne, jak s takovýmito nálezy naložit. Ve standardním systému má tyto možnosti:

- Upravit inventurní doklad v systému a přidat nové položky, ke kterým následně doplní zjištěné množství,
- založit nový inventurní doklad, do kterého zanesese případné nálezy,
- nebrat takovýto nález v úvahu, jelikož po přezkoumání zjistí, že se nejedná o majetek firmy (takovýto majetek ve firmě nikdy neexistoval) - například

zaměstnanec si splete soukromý majetek jiného zaměstnance s majetkem firmy a nesprávně ho uvede jako nález.

Od chvíle, kdy je případný nález zanesen v systému do stávajícího, případně nového dokladu, je s ním možno provádět operace stejným způsobem jako s klasickými položkami. Takovéto položky je tedy možno znovu přepočítávat a samozřejmě zaúčtovat.

Řešení v novém procesu

Při návrhu nového procesu jsem měl z pochopitelných důvodů na paměti, že systém musí být schopen vypořádat se i s případnými nálezy.

Jednou z možností je využít „starého“ postupu a takovéto položky si ručně poznamenávat na papír, následné zpracování proběhne standardně.

Další možností je s nálezy pracovat přímo na straně terminálu. Při nasnímání položky, která není obsažena v inventurním dokladu, má obsluha možnost uložit takovouto položku jako nález a přenést takto do SAP. V řídicím programu bude následně možné takovéto položky identifikovat.

Pro zpracování takovýchto nálezů v nově vyvinutém programu jsem se rozhodl zabudovat následující možnosti zpracování:

- Uložit takovéto položky do nového inventurního dokladu, ve kterém bude udržována vazba na doklad původní. Takto bude při zpětné kontrole zřetelné, které položky byly obsaženy v původním dokladu, a které byly při procesu „nalezeny“. Takto nově zpracovaný doklad bude následně možné rovněž zpracovat čtečkami čárových kódů.
- Možnost vymazání nálezů. Tato volba poslouží v případě, že bude nález přidán ručně přes standardní transakce do inventurního dokladu, případně pro možnost vymazání neplatných položek.
- Odeslání položek k opětovnému přepočítání v případě pochybností o jejich správnosti.

5.2 Uživatelské účty pro mobilní terminály

S ohledem na fakt, že má přesnost a správnost provedení inventury přímý vliv na účetnictví podniku (kap. 2), je nutné zabezpečit přístup k terminálům uživatelskými účty.

Přístup do terminálu řízený uživatelskými účty jsem zvolil z následujících důvodů:

- Identifikace pracovníka, který inventuru provedl viz kap. 5.1.2, který rovněž nese odpovědnost za správnost výsledků,
- zabezpečení proti neoprávněnému používání terminálu.

Pro řízení účtů se od počátku nabízely hned dvě možné varianty. První variantou je využití uživatelských účtů v systému SAP. Druhou alternativou je vytvoření nového nezávislého systému speciálně pro mobilní terminály.

První varianta, tedy SAP účty, se jeví jako jednodušší. Znamenala by nižší náročnost programování nového rozhraní pro řízení uživatelů a zjednodušil by se datový model pro uživatelské účty znázorněný na obr. 5.2. Zároveň by se zjednodušila administrativa, kdy by nebylo nutné zakládat speciální účty pro pracovníky, kteří jsou již v SAP zavedeni. Zmíněné výhody jsou ovšem současně i nevýhodou, a to zcela zásadní. Při této variantě by totiž všichni uživatelé museli mít přístup do systému SAP, což by například při najmutí krátkodobé pracovní síly (brigádníků) v období inventur znamenalo nutnost všem zakládat zbytečné účty v systému, do kterého nepotřebují mít přístup. Otázkou je i možný dopad na cenu SAP licencí spojených s vyšším počtem uživatelských účtů.

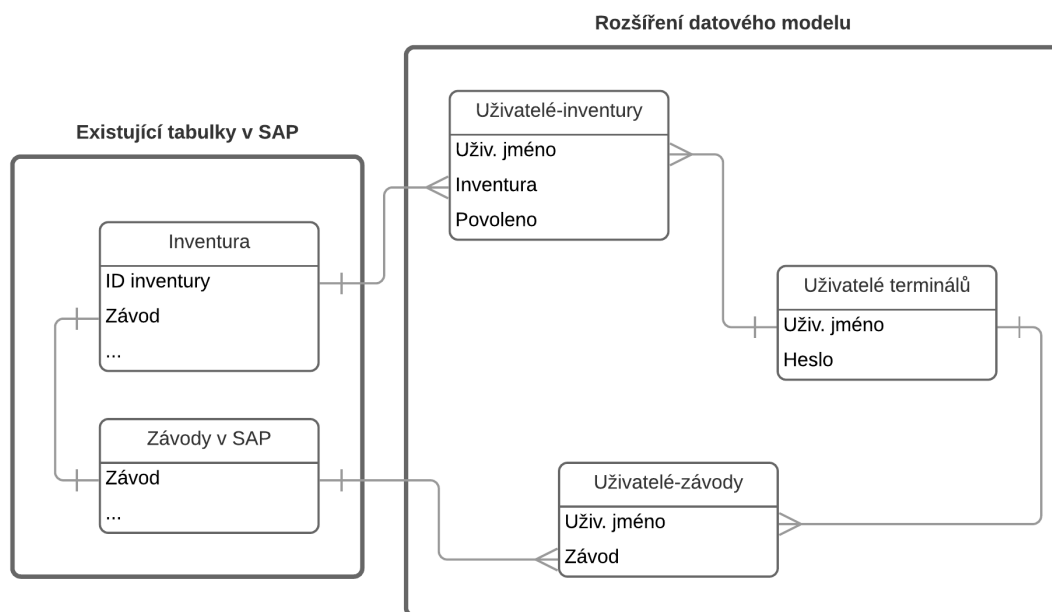
5.2.1 Datový model pro uživatelské účty

Při návrhu systému jsem se rozhodl vytvořit nový datový model pro uživatele mobilních terminálů. Z funkčního hlediska není nutné vytvoření nikterak složitýho datového modelu, jako jeho základ jsem se rozhodl využít jednoduchou tabulku, ve které by bylo uloženo pouze přihlašovací jméno a zašifrované heslo uživatele.

Datový model jsem se dále rozhodl rozšířit o další tabulku, která slouží k dalšímu rozdělení uživatelů pod jednotlivé závody. Takovéto dělení má výhodu v případě použití systému ve velkém podniku čítajícím větší množství závodů, kdy bude možné jednoduše filtrovat, který uživatel patří pod který závod. Sice by bylo možné tento údaj uložit do základní tabulky, to by ale znemožnilo vedení uživatelů pod více závody, což by komplikovalo situaci, kdy by si jednotlivé závody navzájem vypůjčovaly zaměstnance pro provádění inventur.

V kapitole 5.1.2 bylo dále zmíněno, že řešení konkrétní inventury bude vždy umožněno pouze určeným zaměstnancům. Pro umožnění vazby mezi inventurou a konkrétním uživatelem terminálu slouží další tabulka datového modelu, ve které bude uchovávaná vazba mezi uživateli a jim povolenými inventurami.

Zjednodušené schéma datového modelu popsaného v této kapitole je znázorněno v diagramu na obr. 5.2.



Obrázek 5.2: Zjednodušený datový model uživatelských účtů pro terminály

5.3 Oprávnění v systému SAP

Kapitola 5.2 pojednávala o účtech pro mobilní terminály, které je nutno důsledně spravovat. Dalším bodem zajišťující bezpečnost systému je řízení oprávnění na straně ERP SAP, kdy je nutné zabezpečit, aby jisté úkony mohly provádět pouze pověřené osoby.

Ve standardních transakcích SAP je již připraven systém kontrol objektů oprávnění a není nutné provádět další úpravy. Stačí aby určeným osobám správce systému přidělil příslušná oprávnění. Tento systém kontrol je v SAP standardně zabudován i do existujících BAPI (kap. 3.1.1), které bude nový systém inventarizace využívat.

Přidělení objektu oprávnění, který je v programu kontrolován (kap. 3.1.1), není jediným způsobem pro řízení přístupů v systému SAP. Další možností je přidělení autorizace pro spouštění konkrétních transakcí, k čemuž jsem přihlédl při návrhu systému oprávnění pro proces inventarizace pomocí mobilních terminálů.

Při návrhu řízení oprávnění v SAP přicházely v úvahu dva možné scénáře:

1. Využití stávajících objektů oprávnění i pro nové nástroje, pouze pro správu uživatelských účtů vytvořit nové.
2. Pro veškeré nově přidané funkce vytvořit zcela nové objekty.

V této práci jsem se po pečlivé rozvaze přiklonil ke druhé variantě. Veškeré možné aktivity (např. aktivace inventury pro terminály, správu terminálových účtů, zadání počítání, účtování,..) v nových nástrojích pro inventarizaci tedy budou napojeny na sadu nově vytvořených objektů oprávnění, zatímco současně bude využito i standardních kontrol zmíněných v předchozích odstavcích. Díky takto postavenému řešení je možné dynamičtěji kombinovat oprávnění a tím i v jistém smyslu jednoduchým způsobem ovlivňovat možné využití systému.

Prakticky tak lze díky tomuto návrhu například zabránit uživatelům zpracování inventury pomocí standardních SAP transakcí (mimo fáze založení dokladu) a tím

je přinutit veškeré kroky provést v nově vytvořeném řídicím programu. Takovéto situace lze jednoduše docílit odebráním oprávnění pro spouštění standardních inventurních transakcí, přidělením oprávnění spustit nový program a dále přidělit příslušné nové objekty. Pokud by bylo zvoleno řešení oprávnění v první variantě, nebyl by takovýto detail řízení možný.

Dalším příkladem využití těchto oprávnění může být situace, kdy by bylo vyžadováno, aby pro některé akce bylo využito výhradně standardu, avšak pro jiné výhradně nového programu. V takové situaci tedy bude uživateli přiděleno oprávnění spouštět pouze konkrétní SAP transakce a zároveň přiděleny pouze nové objekty pro povolení provádět pouze konkrétní akce.

Standardní SAP objekty mají komplikovanější strukturu umožňující například i vymezení konkrétní oblasti podniku (závod, účetní okruh,...), pro který má objekt platnost. Pro objekty v nové nadstavbě systému takovýto detail nepovažuji za nutný, jelikož slouží pouze pro zpřístupnění konkrétních voleb v obslužných programech. O další detailní kontroly bude díky využití BAPI postaráno zmíněnými existujícími standardními objekty. Ponesou tedy pouze informaci o autorizaci na konkrétní akci (například provést účtování rozdílů).

5.4 Aplikace v SAP ERP

5.4.1 Správa uživatelských účtů

Kapitola 5.2 pojednávala o nutnosti vytvoření a datovém modelu systému uživatelských účtů pro mobilní terminály. Zvažoval jsem, jestli správu implementovat přímo do řídicího programu pro inventarizaci (kap. 5.4.2), ale po zvážení všech okolností jsem se rozhodl pro vyčlenění do nového programu a nové uživatelské transakce. Takovéto řešení považuji za vhodnější pro případné budoucí rozšiřování systému, kdy by tyto účty mohly být využity i pro další případné aplikace a začlenění správy účtů přímo do programu inventarizace by bylo nelogické.

Datový model (kap. 5.2.1 obr. 5.2) pro terminálové účty sestává celkem ze tří nových tabulek, ve kterých jsou uložena základní data uživatelů, jejich přiřazení ke konkrétním závodům podniku a přidělené inventuře. Tento obslužný program slouží pro správu dat v prvních dvou zmíněných tabulkách, přiřazování uživatelů ke konkrétním inventurám bude provedeno v řídicím programu pro inventury. Tímto způsobem zůstane neporušena hlavní myšlenka, která vedla k vyčlenění správy uživatelů do zvláštního programu, zmíněna v předchozím odstavci.

Program pro správu účtů mobilních terminálů umožní následující operace:

- Založení uživatelského účtu se zašifrovaným heslem,
- změnu hesla,
- zobrazení existujících uživatelských účtů s možností filtrace podle závodu,
- přiřazení uživatele k jednomu či více závodům.

5.4.2 Řídicí program

Řídicí program obsáhne veškerou novou funkcionalitu rozšířeného procesu inventarizace, který byl popsán v kapitole 5.1.

Při návrhu řídicího programu jsem uvažoval o vyčlenění jednotlivých funkcionalit pod různé transakce, podobně jako tomu je u standardní SAP inventarizace. Nakonec jsem se rozhodl veškerou funkcionalitu zahrnout do jednoho jediného programu, ve kterém budou daným uživatelům zpřístupněné pouze jim příslušné funkce na základě nových oprávnění (5.3).

Řešení, kdy veškeré funkce soustředím na jedno místo, považuji za vhodné z hlediska uživatelského komfortu, kdy nebude nutné při zpracování inventarizace postupně procházet více různými transakcemi, ale vše bude přehledně na jednom místě.

Funkce řídicího programu:

- Aktivace inventarizace pro zpracování nadstavbou systému pro mobilní terminály,

- přidělení pověření konkrétním uživatelům terminálů ke zpracování inventury,
- zobrazení aktuálního přehledu stavu inventurního počítání (včetně historie zadávání u každé položky),
- zapsání zjištěných stavů do standardní SAP inventury,
- porovnání účetního množství se stavem zjištěným během inventury,
- označení položek pro opakované přepočítání,
- opětovný zápis přepočítaných stavů do SAP standardu,
- založení nového inventurního dokladu pro případné nálezy,
- zaúčtování rozdílů.

Pro veškeré přenosy dat do SAP standardu (počítání, založení dokladů,...) program využije BAPI (kap. 3.1.1), které mají již zabudované standardní kontroly oprávnění (5.3).

Další motivací pro zvolení varianty, kde bude vše na jednom místě, bylo zjednodušení následující implementace programu a jeho případná údržba. Pro většinu funkcí je totiž vhodné shodné uživatelské rozhraní, které by v případě rozdělení do více programů bylo nutné implementovat stále dokola, což nepovažuji za vhodný přístup. Dále se domnívám, že stejně vyhlížející rozhraní po puštění různých transakcí by mohlo být pro uživatele matoucí.

Uživatelské rozhraní jsem se snažil navrhnout pokud možno co nejjednodušší a intuitivní. Sestává jen z několika málo jednoduchých obrazovek, jejichž funkce jsou:

První obrazovka

- Výběr konkrétní inventury,
- volby:
 - aktivace pro zpracování mobilními terminály,

zobrazení,
zpracování.

Druhá obrazovka

- Detailní výpis položek a stavů počítání,
- možnost označení konkrétních položek a provedení příslušné akce (započítání, účtování,...),
- skok na obrazovku s historií počítání konkrétní položky,
- skok na obrazovku pro správu uživatelů.

Pomocné obrazovky

- Správa uživatelů - zobrazí uživatele (mobilních terminálů) pro závod, kterého se inventura týká, a umožní autorizaci pro zpracování inventury.
- Historie počítání položky - zobrazení detailů každého zadaného vstupu. Například čas, přihlášený uživatel, množství, případná poznámka.

5.4.3 Rozhraní pro RFC volání

Pro zpracování inventury pomocí mobilních terminálů je nutné mít možnost výměny dat mezi podnikovým systémem a mobilními terminály. Pro tento účel využijí možnosti vytvoření funkčních modulů na straně SAP umožňující volání z jiných systémů (viz kap. 3.1.1), které následně budou přístupné jako webová služba.

Základem rozhraní budou funkční moduly, které umožní:

- Získání uživatelských účtů pro mobilní terminály. V online režimu práce (kdy je stále dostupné spojení) by sice stačilo provádět na straně SAP pouze autorizaci zadaných přihlašovacích údajů na terminálu, nicméně by v tomto případě nebylo možné přihlášení při nedostupné síti. Proto budou staženy

všechny uživatelské účty, aby bylo umožněno přihlášení do terminálu i v offline módu,

- získání dat (položek) inventury,
- odeslání dat počítání do systému,
- validaci nálezů v případě práce v online režimu.

5.5 Aplikace pro mobilní terminály

Stěžejní součástí celého procesu, kterým se tato práce zabývá, je aplikace pro mobilní terminály, které poslouží jako náhrada za ruční vyplňování vytištěných inventurních seznamů.

Přístup do aplikace pro inventury bude uživateli umožněn pouze v případě úspěšného přihlášení. Příslušné uživatelské účty aplikace získá pomocí RFC volání do systému SAP (kap. 5.4.3), ve kterém jsou účty zároveň spravovány (viz kap. 5.4.1).

Po úspěšném přihlášení uživatel zadá příslušné číslo inventury (pod kterým je vedena v SAP), případně nasnímá čárový kód s číslem inventury. Pokud bude mít odpovídající pověření pro její zpracování, bude do terminálu stažen inventurní seznam pro počítání.

Vlastní inventura dále spočívá ve snímání čárových kódů na položkách a zadávání nalezeného množství. Pro případy, kdy by se na položce mohl nacházet poškozený čárový kód, který nebude možné rozpoznat čtečkou čárového kódu, umožní terminál i ruční zadání příslušného kódu. Za další užitečnou vlastnost považuji možnost nastavení pouze předem určených druhů čárových kódů, které čtečka kódů rozpozná a ignorování případných jiných typů. Tím se omezí případy, kdy bude sejmuto nesprávný kód, což může nastávat například v případech, kdy balení výrobku obsahuje větší množství čárových kódů (typicky např. kód dodavatelské firmy, kód který použila přepravní firma, kód podniku ve kterém probíhá inventura,...).

Aplikace rovněž umožní zobrazení historie zadávání počítání pro konkrétní položku a případnou opravu zadaných dat přímo na mobilním terminálu.

Při online režimu budou data okamžitě přenášena do podnikového systému. Pokud bude terminál pracovat offline, data počítání budou kumulována v paměti zařízení a přenos do systému nastane až ve chvíli, kdy bude dostupné spojení. Pro přenosy dat jsem se rozhodl dále zabudovat do systému možnost volby, zda k přenosu dojde vždy při dostupném spojení, případně zda budou data odesílána pouze na vyžádání.

V části věnované návrhu samotného procesu (kap. 5.1) jsem dále popisoval práci s tzv. nálezy, což jsou položky nacházející se v místě prováděné inventury, které nejsou obsaženy v inventurním seznamu. Aplikace umožní zadání počítání i pro takovéto položky. V offline režimu pozbude možnosti validovat načtená data (např. v případě využití kódů EAN-13 je možné naskenovat i naprosto nesmyslnou položku, jelikož se tyto kódy běžně vyskytují na obchodním zboží) a získat ze SAP příslušnou měrnou jednotku, ve které má být množství zadáváno. Proto aplikace umožní pracovníkovi mimo nalezeného množství zadat i měrnou jednotku, ve které údaje zadává. Pro alespoň částečnou možnost validace takovéhoho vstupu bude v aplikaci načtena sada měrných jednotek existujících v podnikovém systému. Při práci v online režimu se případná validace nálezů usnadňuje, aplikace se přímo dotáže do systému, zda v něm daný kód existuje, a získá příslušné doplňkové údaje.

Za poslední důležitou součást systému považuji možnost zobrazení přehledu stavu inventury přímo v samotném mobilním terminálu. Pracovníkovi tak bude umožněno zobrazení počtu spočítaných položek a položek, pro které doposud nebylo zadáno žádné počítání. V případě, že bude pracovník považovat inventuru za kompletní a v seznamu se budou stále nacházet položky bez zadaného počítání, umožní aplikace těmto položkám zadat nulový počet. Toto poslouží pracovníkovi pro jakési „odškrtávání“ položek ze seznamu, které se například marně snažil dohledat při poslední kontrole zbývajících položek, ovšem bez úspěchu.

6 Implementace navrženého systému

6.1 Aplikace v SAP ERP

Veškeré programové vybavení pro inventarizaci s využitím mobilních terminálů jsem zahrnul do nově vytvořeného balíčku v systému SAP nazvaného ZM_BC_READERS. Tento název jsem zvolil z hlediska SAP konvencí, kdy první písmeno nových objektů (programů, funkčních modulů, tříd, DB tabulek,...), které nevytvořila společnost SAP, musí vždy začínat na písmeno Y nebo Z. Následující písmeno M představuje modul materiálového hospodářství (kap. 3.1). Část názvu BC_READERS jsem zvolil jakožto označení balíčku funkcionalit pro mobilní terminály, respektive čtečky čárových kódů. Obdobné jmenné konvence následně využívám i pro pojmenování ostatních částí vyvinutého systému.

6.1.1 Datový model

Za jednu z nejdůležitějších částí při vývoje nového procesu inventarizace považuji analýzu existujícího datového modelu a návrh jeho rozšíření, od čehož se bude následně odvíjet implementace celého řešení.

Pro proces inventur důležité databázové tabulky, které v systému již existují, jsou:

- **IKPF** - hlavička inventurního dokladu. Zde jsou uloženy základní údaje o inventuře, například závod, sklad, datумы (založení dokladu, posledního počítání, účtování), statusy probíhající inventury, odpovědná osoba atd.

Klíčem tabulky je číslo inventurního dokladu a fiskální rok.

- **ISEG** - položky inventurního dokladu. Zde je uloženo číslo materiálu v SAP, během postupu inventury se sem ukládají příslušné informace k položce - např. kdo provedl počítání, nalezené množství, statusy (spočteno, zaúčtováno), měrná jednotka pro počítání atd.

Klíčem je inventurní doklad, fiskální rok a číslo položky.

- **MARA** - všeobecná data materiálu. Jedná se o základní tabulku pro každý vedený materiál v SAP, obsahuje základní údaje - přidělené číslo materiálu (primární klíč, jedná se o znakové pole - může se jednat i o text), rozměry, druh materiálu, základní měrná jednotka atd.
- **MAKT** - krátké texty materiálu. Jedná se o doplnění značení libovolného materiálu o krátký text, které je možné vést v různých jazycích. Například bude do systému zaváděn nový materiál, kterému bude přiděleno číslo materiálu 40 (což je následně jeho identifikace napříč celým systémem). Pro snadnější identifikaci o co se ve skutečnosti jedná je možné do této tabulky uložit popis např. v českém jazyce „žárovka 40 W“ a v anglickém „lightbulb 40 W“.
- **MEAN** - EAN pro konkrétní materiál, případně měrnou jednotku.
- **T001W** - závody vedené v systému SAP.
- **T001L** - sklady v systému SAP. Každý sklad patří pod konkrétní závod.

Rozšíření datového modelu

V kapitole 5.2.1 jsem navrhl rozšíření modelu o uživatelské účty pro mobilní terminály. Pro příslušné tabulky jsem zvolil názvy **ZMBC_USERS** (uživatelé), **ZMBC_USERS_WERKS** (vazba uživatel - závod) a **ZMBC_USERS_INV** (uživatel - inventura).

Na začátku procesu inventarizace pomocí terminálů je nutné danou inventuru označit pro zpracování v novém procesu a tuto informaci uchovat v datovém modelu (kap. 5.1.2). Zvažoval jsem několik možných variant, jak k tomuto problému přistoupit:

1. Doplnit nové pole do hlavičky inventury.
2. Vytvořit novou tabulku, do které by se ukládaly informace o aktivaci inventury a případné doplňkové informace.
3. Vytvořit novou kopii existujících inventurních tabulek.

Vzhledem k tomu, že jsem se rozhodl nové rozšíření realizovat ve formě nadstavby bez zásahů do existujícího systému, od první varianty jsem upustil.

Při volbě mezi druhou a třetí variantou jsem nakonec upřednostnil variantu třetí, tedy vytvoření kopie existujících dvou tabulek, a to z následujících důvodů:

- při budoucím rozšiřování aplikace je možné libovolně rozšiřovat tabulky o potřebná pole bez zásahu do standardu a bez přílišného komplikování datového modelu,
- možnost porovnání aktuálních inventurních dokladů s jejich podobou v momentě založení inventury, pro případ zásahu uživatele přes SAP standard,
- rychlost vyhledávání v tabulkách, uvažujeme-li velký podnik s vysokým počtem závodů a skladů, který má zavedené mobilní terminály jen na některých z nich. Pokud bychom chtěli vyhledat data v originálních tabulkách, provádělo by se vyhledávání zbytečně v datech všech závodů, s následujícím joinem do nově založených doplňkových tabulek. Při využití kopií tabulek obsahujících pouze potřebné inventurní doklady se množství prohledávaných záznamů značně redukuje.

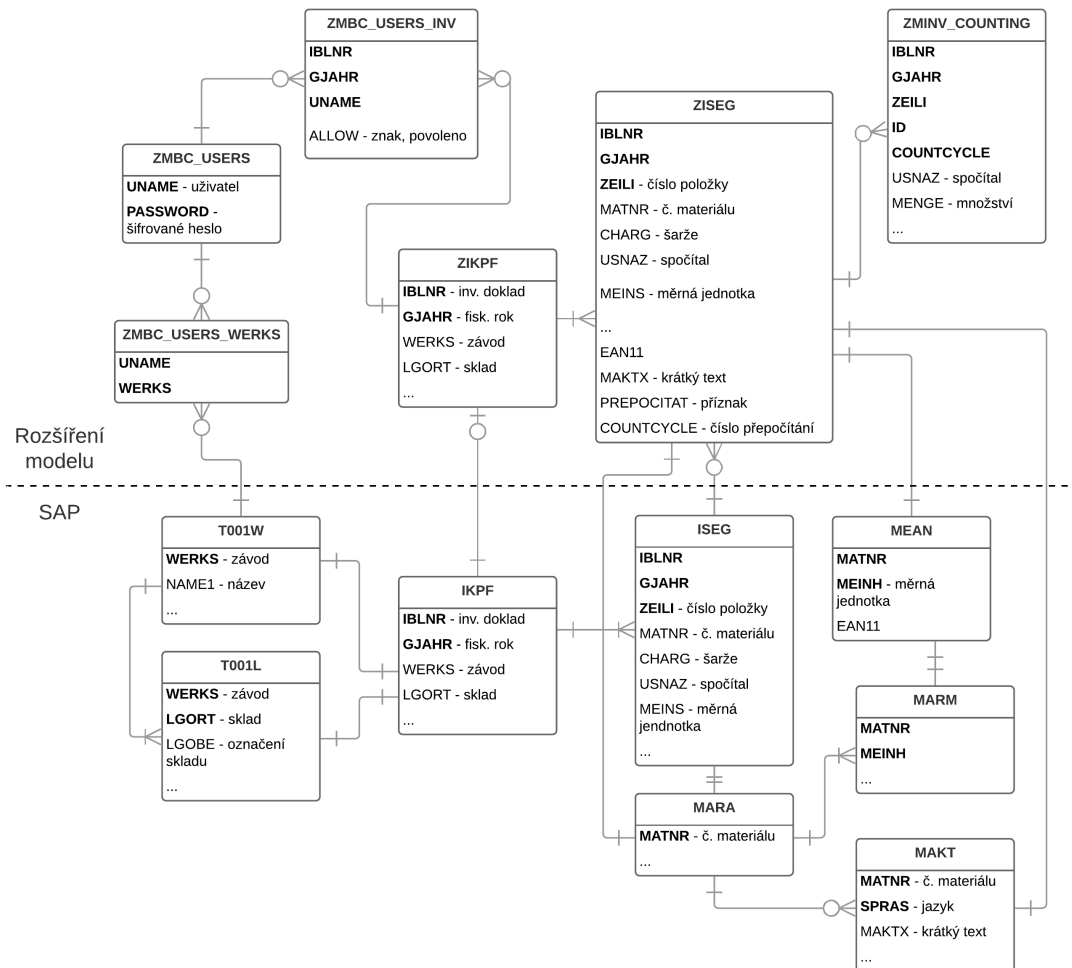
Z výše popsanych důvodů jsem založil nové tabulky ZIKPF a ZISEG. Tabulku s daty položek jsem se dále rozhodl rozšířit o příslušný EAN, krátký text materiálu, příznak vyžádaného přepočítání a číslo cyklu přepočítávání.

Poslední tabulkou rozšiřující datový model SAP je ZMINV_COUNTING. Tato tabulka slouží pro uložení jednotlivých záznamů počítání zaznamenaných mobilními terminály. Ukládají se do ní následující údaje:

- identifikace inventury (číslo inventurního dokladu a fiskální rok),

- číslo položky v inventurním dokladu,
- SAP číslo materiálu, případně i EAN,
- unikátní ID každého záznamu,
- číslo aktuálního cyklu počítání - pro případ, že byla položka odeslána k opakovanému přepočítání (kap. 5.1.6),
- množství, měrná jednotka
- datum a čas,
- pole pro případnou poznámku.

Schéma rozšířeného datového modelu pro inventarizaci je na obrázku 6.1.

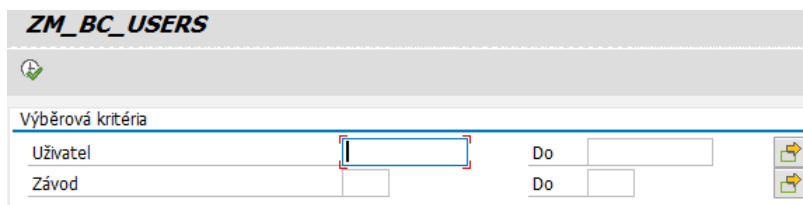


Obrázek 6.1: Datový model v systému SAP

6.1.2 Správa uživatelských účtů

Pro správu uživatelských účtů pro mobilní terminály (kap. 5.2) jsem vytvořil nový program ZM_BC_USERS, který lze spustit pomocí uživatelské transakce se stejným názvem.

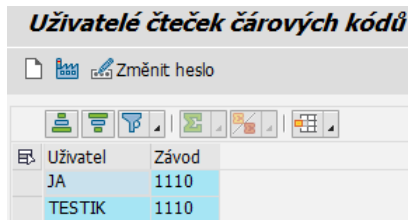
Tento program slouží pro přehled existujících uživatelů, založení nových, přiřazení ke konkrétnímu závodu a případně pro změnu hesla.



Výběrová kritéria	
Uživatel	Do
Závod	Do

Obrázek 6.2: Výběrová obrazovka programu pro správu uživatelů

Po spuštění se zobrazí výběrová obrazovka (obr. 6.2), na které lze zadat případnou podmnožinu uživatelů pro zobrazení, nebo ji ponechat prázdnou pro zobrazení všech.



Uživatelé čteček čárových kódů	
Uživatel	Závod
JA	1110
TESTIK	1110

Obrázek 6.3: Hlavní obrazovka správy uživatelů

Na další obrazovce (obr. 6.3) se zobrazí seznam existujících uživatelů a jejich případné přiřazení ke konkrétním závodům. Na nástrojové liště jsou následně tlačítka pro zakládání uživatelů, přiřazení uživatele k závodu a změnu hesla.

Před uložením nového hesla do databáze je pro jeho zašifrování využit SAP funkční modul CALCULATE_HASH_FOR_CHAR využívající šifry SHA1.

Oprávnění

Při každé z činností je prováděna kontrola příslušného objektu oprávnění (kap. 3.1.1). Pro správu uživatelských účtů jsem vytvořil nový objekt oprávnění nazvaný ZMBC_U.

V tomto objektu je zahrnuto standardní SAP pole ACTVT s následujícími možnými hodnotami:

- 01 - založení uživatele
- 02 - změna uživatele
- 78 - přiřazení

Příslušná kontrola oprávnění je následně prováděna vždy ve vztahu k tomuto objektu a hodnotě pole ACTVT (uživatel jich smí mít přiřazeno víc). Na obrázku 6.4 lze vidět ukázkou implementace takovéto kontroly. Pokud byla kontrola úspěšná, je hodnota systémové proměnné sy-subrc rovna nule.

```
AUTHORITY-CHECK OBJECT 'ZMBC_U'  
  ID 'ACTVT' FIELD '01'.  
IF sy-subrc <> 0.  
  MESSAGE 'Nemáte oprávnění zakládat uživatele'  
    TYPE 'I' DISPLAY LIKE 'E'.  
EXIT.  
ENDIF.
```

Obrázek 6.4: Ukázkou kontroly objektu oprávnění

6.1.3 Pomocná třída ZMBCINV

Obdobně jako jsou v systému SAP implementovány BAPI (kap. 3.1.1), rozhodl jsem se pro základní úkony nad rozšířeným procesem inventarizace implementovat sadu univerzálních nástrojů. Ty jsem zahrnul do globálně (napříč celým SAP) viditelné třídy, kterou jsem nazval ZMBCINV. Vyčlenění do takovéto samostatné třídy redukuje množství stále dokola se opakujícího kódu, například některé části by bylo nutné implementovat současně v řídicím programu a ve funkčních modulech na rozhraní. Dále je díky tomuto řešení usnadněno případné další rozšiřování či údržba nového

systému, kdy v mnoha případech (např. při úpravě datového modelu) stačí pouze upravit příslušné metody.

Tato třída implementuje formou statických metod základní nástroje pro práci s rozšířenou inventurou pro mobilní terminály. Těmito metodami jsou:

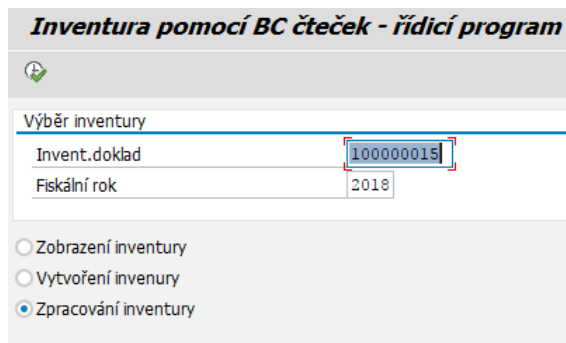
- `CHECK_BCINV_EXISTS` a `CHEKC_STINV_EXISTS` - kontrola, zda byla inventura aktivována pro zpracování mobilními terminály, případně zda byla založena ve standardu. Tuto kontrolu provádím jednoduchým selektem do DB tabulky `ZIKPF / IKPF`.
 - vstup: číslo inventury, fiskální rok
 - výstup: boolean
- `GET_HEADER` a `GET_HEADER_ST` - získání hlavičky inventurního dokladu z tabulky `ZIKPF`, respektive `IKPF`.
 - vstup: číslo inventury, fiskální rok
 - výstup: hlavička dokladu / výjimka pokud data nebyla nalezena
- `GET_ITEMS` a `GET_ITEMS_ST` - získání položek inventurního dokladu z DB tabulek `ZISEG / ISEG`. V případě metody `GET_ITEMS_ST` je zároveň proveden „join“ do dalších DB tabulek v SAP (6.1.1), ze kterých jsou načtena doplňková data (EAN, krátké texty atd.).
 - vstup: číslo inventury, fiskální rok
 - výstup: položky dokladu / výjimka pokud data nebyla nalezena
- `GET_COUNT_DATA` - načtení dat počítání pro příslušnou inventuru z tabulky `ZMINV_COUNTING`.
 - vstup: číslo inventury, fiskální rok
 - výstup: data počítání / výjimka pokud data nebyla nalezena

- `CREATE_INV` a `UPDATE_INV` - založení / aktualizace inventury pro mobilní terminály kopií dat ze standardních tabulek do nových. Tyto metody zároveň využívají předchozí zmíněné pro získání dat.
 - vstup: číslo inventury, fiskální rok
 - výstup: výjimka v případě, že byla zadána neexistující inventura
- `CHECK_INV_COMPLETE` - kontrola, zda již není inventura kompletně zpracována.
 - vstup: číslo inventury, fiskální rok
 - výstup: boolean
- `GET_BC_USERS` - získání seznamu uživatelů mobilních terminálů pro konkrétní inventuru
 - vstup: číslo inventury, fiskální rok
 - výstup: seznam uživatelů / výjimka pokud data nebyla nalezena
- `GET_UOM` - získání měrných jednotek. Tato metoda umožňuje volitelně získat měrné jednotky (MJ) / alternativní MJ pro zadaný jeden či více materiálů, případně získání seznamu všech vedených v SAP. Zároveň názvy jednotek převede na jejich české ekvivalenty.
 - vstup:
 - čísla materiálů - pokud není uveden žádný, dohledají se všechny existující MJ,
 - boolean příznak - dohledat jen základní MJ, nebo i alternativní,
 - výstup: tabulka (číslo materiálu, MJ)
- `GET_MATERIAL_INFO` - získání dodatečných dat pro materiál (zadané číslo materiálu nebo EAN), případně validace předaných hodnot. Tuto metodu jsem zpočátku vyvinul na rozhraní SAP pro kontroly v online módu mobilních terminálů, ale vzhledem k její univerzálnosti jsem se rozhodl ji začlenit do této pomocné třídy.

- vstup:
 - číslo materiálu
 - EAN
 - šarže
 - měrná jednotka
 - příznak - získat krátký text
 - příznak - získat alternativní měrné jednotky
- výstup:
 - boolean - materiál je/není validní
 - číslo materiálu - např. pokud byl zadán jen EAN, dohledá se příslušný materiál
 - tabulka měrných jednotek
 - krátký text materiálu
- CHECK_AUTH - kontroly oprávnění. V této metodě jsem implementoval kontroly konkrétních objektů oprávnění volané z různých míst řídicí aplikace. Díky tomu je vše přehledně na jednom místě, což usnadní případné budoucí úpravy.
 - vstup:
 - číslo inventury, fiskální rok
 - název akce - např. „UCTOVANI“
 - výstup: výjimka v případě chybějícího oprávnění

6.1.4 Řídicí aplikace

Pro správu nového procesu jsem vytvořil řídicí program (viz kap. 5.4.2) s názvem ZM_BC_INV spustitelný přes stejnojmennou transakci. Tento program umožňuje aktivaci pro zpracování mobilními terminály, přiřazení konkrétním pracovníkům, zobrazení stavu a samozřejmě řízení procesu.



Obrázek 6.5: Výběrová obrazovka řídicí aplikace

Po spuštění programu se zobrazí výběrová obrazovka (obr. 6.5), na které uživatel zadá konkrétní inventurní doklad vytvořený přes standardní transakce SAP (tab. 4.1). Dále lze pomocí radio buttonů zvolit případný režim, ve kterém se má aplikace spustit. Ukázkou implementace výběrové obrazovky lze vidět na obrázku 6.6.

```
SELECTION-SCREEN BEGIN OF BLOCK inv_selection WITH FRAME TITLE TEXT-001.
PARAMETERS: p_iblnr TYPE ikpf-iblnr MEMORY ID ibn OBLIGATORY,
             p_gjahr TYPE ikpf-gjahr MEMORY ID gjr OBLIGATORY.
SELECTION-SCREEN END OF BLOCK inv_selection.
PARAMETERS: p_show RADIOBUTTON GROUP 1 DEFAULT 'X',
             p_create RADIOBUTTON GROUP 1,
             p_proces RADIOBUTTON GROUP 1.
```

Obrázek 6.6: Ukázka implementace výběrové obrazovky

Třídy

Řídicí program je logicky rozdělen do čtyř tříd, kdy v každé z nich je implementován samostatný funkční celek. Dle SAP konvence začíná název třídy vždy předponou LCL_ (local class).

Těmito třídami jsou:

- LCL_ALV - třída implementující základní metody pro výstup do ALV (ABAP List View), tedy do tabulky na obrazovce. Příklad takového výstupu lze vidět na obr. 6.7.
- LCL_INVENTURE - přehled inventury s výstupem do tabulky (dědí od LCL_ALV) a metody pro zpracování inventury (účtování, označení k přepočítání atd.).

- LCL_COUNTING - přehled jednotlivých záznamů počítání, rovněž dědí od LCL_ALV.
- LCL_USERS - zobrazení uživatelů přiřazených k závodu, ve kterém probíhá inventura, a možnost jim povolit zpracování inventury pomocí terminálů.

X	Inv.doklad	Fiskál.rok	Položka	Mater.	Šarže	Krát.text.materiálu	Kód EAN/UPC	Účet.množ.	Mn. dokl.	N.	Poč. ČK MJ	Spočítal	Datum	Čas	Status
<input type="checkbox"/>	100000015	2018	1	1		FERT 01 MB	8590969134271	8,500	0,000		0,000	KS		00:00:00	
<input type="checkbox"/>	100000015	2018	2	2		FERT 02 MB	8590269134278	0,000	0,000		0,000	KS		00:00:00	
<input type="checkbox"/>	100000015	2018	3	3		FERT 03 MB	5901086000982	10,000	0,000		0,000	KS		00:00:00	
<input type="checkbox"/>	100000015	2018	4	4		FERT 4 MB	5901086001446	10,000	0,000		1,000	KS admin	08.08.2018	20:54:00	
<input type="checkbox"/>	100000015	2018	5	5		FERT 5 MB	8593894809740	10,000	0,000		12,000	KS admin	08.08.2018	21:26:57	PP
<input type="checkbox"/>	100000015	2018	6	6		FERT 6 MB	8590969414502	10,000	0,000		1,000	KS admin	08.08.2018	20:53:50	
<input type="checkbox"/>	100000015	2018	7	7		FERT 7 MB	8590269414509	10,000	0,000		13,000	KS admin	08.08.2018	20:56:37	
<input type="checkbox"/>	100000015	2018	8	8		FERT 8 MB	8590269422320	10,000	0,000		32,000	KS admin	08.08.2018	21:13:20	
<input type="checkbox"/>	100000015	2018	9	9		FERT 9 MB	8592766000513	10,000	0,000		31,000	KS admin	08.08.2018	21:09:24	
<input type="checkbox"/>	100000015	2018	10	10		FERT 10 MB	8592766000520	5,000	0,000		0,000	KS		00:00:00	
<input type="checkbox"/>	100000015	2018	11	11		FERT 11 MB	8592766000537	5,000	0,000		0,000	KS		00:00:00	
<input type="checkbox"/>	100000015	2018	12	12		FERT 12 MB	8592766000544	10,000	0,000		55,000	KS admin	08.08.2018	21:09:50	
<input type="checkbox"/>	100000015	2018	13	23	000000CH01	FERT 12 MB - šarže	7894561237806	10,000	0,000		11,000	KS admin	08.08.2018	21:10:22	
<input type="checkbox"/>	100000015	2018					8594001021529	0,000	0,000		15,000	KS admin	08.08.2018	22:03:02	

Obrázek 6.7: Hlavní obrazovka řídicí aplikace

Režimy zpracování

Jak již bylo zmíněno, jsou v aplikaci implementovány celkem tři režimy, ve kterých lze pracovat:

- Zobrazení inventury - pouhé prohlížení stavu bez možnosti zásahů.
- Vytvoření - nejprve proběhnou kontroly, zda inventura ve standardu existuje, zda již nebyla pro zpracování terminály aktivována, a stavu zpracování (zda již není kompletně hotová). Pokud kontroly proběhnou v pořádku, dojde k aktivaci inventury kopií dat do nových tabulek (viz 6.1.1).
- Zpracování - provedení obdobných kontrol jako v předchozím bodě. Následně proběhnou kontroly objektů oprávnění na příslušné akce (kap. 5.3) a pokud některé z oprávnění chybí, odeberou se příslušná tlačítka z nástrojové lišty.

Pro příslušné kontroly inventur, oprávnění, založení inventury a získání dat z datového modelu jsou využity metody globální třídy ZMBCINV (kap. 6.1.3).

Zpracování inventury

Při spuštění programu v režimu zpracování se zobrazí přehled všech položek inventury a nálezů (červený řádek na obr. 6.7), ve kterém je zobrazena suma celkového nalezeného množství. Při dvoukliku na konkrétní položku se zobrazí detailní přehled jednotlivých nasnímaných záznamů (obr. 6.8) včetně případné poznámky (lze zadat při snímání na terminálu).

Inv.doklad	Fiskál.rok	Položka	Pořad...	ČPP	Jméno	Kód EAN/UPC	Mater...	Šarže	Množs...	Jednotka	Datum	Čas	Zpráva
100000015	2018	12	3		admin	8592766000544	12		31	KS	08.08.2018	21:09:50	
100000015	2018	12	1		admin	8592766000544	12		21	KS	08.08.2018	21:08:07	
100000015	2018	12	2		admin	8592766000544	12		3	KS	08.08.2018	21:08:07	Oprava množství

Obrázek 6.8: Řídicí program - historie zadávání

V seznamu položek jsou zobrazena zaškrťávací pole pro označení jednotlivých řádků. Jednotlivé akce na nástrojové liště (zaúčtovat, smazat nález atd.) vždy zpracovávají pouze konkrétní označené záznamy.

Co se týče akcí inspirovaných standardem, je vždy využito konkrétních existujících BAPI (kap. 3.1.1). Nejdříve provedu základní kontroly označených položek, například zda nebyla k zaúčtování předána již zúčtovaná položka, následně naplním struktury pro BAPI relevantními položkami a nakonec využiji příslušnou SAP funkci. Pokud cokoliv selže, zobrazí se uživateli příslušný protokol o chybách. Jelikož se informace o provedených změnách zaznamenávají do standardních SAP tabulek, provedu ještě aktualizaci pomocných tabulek ZIKPF a ZISEG. Ukázka takového využití BAPI je na obrázku 6.9.

```

LOOP AT it_data ASSIGNING FIELD-SYMBOL(<i>)
      WHERE status <> 'U' AND " nebylo účtováno
            zeili <> ' ' AND " nejedná se o nález
            checkb = 'X'.

Zero count
IF <i>-cmenge = 0.
  IF <i>-xnull = 'X'.
    OK
  ELSE.
    APPEND VALUE #(
      type      = 'W'
      id        = 'AD'
      number    = '010'
      message_v1 = |Počítání položky { <i>-zeili }|
      message_v2 = 'nebylo převzato.'
      message_v3 = 'Má nulový součet a nebyla'
      message_v4 = 'označena příznakem nuly.'
    ) TO rt_protocol.
    CONTINUE.
  ENDIF.
ENDIF.

APPEND VALUE #(
  item      = <i>-zeili
  material  = <i>-matnr
  batch     = <i>-charg
) TO lt_items.
ENDLOOP.

IF lines( lt_items ) < 0.
  MESSAGE |Žádné položky pro zpracování.|
  TYPE 'S' DISPLAY LIKE 'W'.
  EXIT.
ENDIF.

CALL FUNCTION 'BAPI_MATPHYSINV_POSTDIFF'
  EXPORTING
    physinventory = p_iblnr
    fiscalyear    = p_gjahr
    pstng_date    = sy-datum
  TABLES
    return        = lt_return
    items         = lt_items
  EXCEPTIONS
    OTHERS        = 1.

```

Obrázek 6.9: Ukázka použití BAPI pro zaúčtování rozdílů

Přepočítání

Možnost označit položky pro opětovné přepočítání je jednou z funkcionalit, které nevyužívají BAPI. Princip spočívá v nastavení příznaku `prepocitat` konkrétní položky dokladu v tabulce `ZISEG`. Zároveň se na položce inkrementuje počítadlo `countcycle`. V zobrazeném seznamu položek v řídicí aplikaci se následně zobrazí ve sloupci „status“ příznak, že položka byla poslána k přepočítání (PP).

Suma zjištěného množství pomocí mobilních terminálů je vždy vypočítávána pouze z údajů s aktuálním cyklem přepočítání, které je rovněž poznamenáno v datech odeslaných z terminálů (viz datový model na obr. 6.1). Pro kontrolu a porovnání průběhů jednotlivých cyklů počítání jsou při zobrazení detailu (dvouklik na položku) zobrazeny údaje ze všech cyklů.

Tuto funkcionalitu jsem se rozhodl implementovat i pro případné nálezy, které nebyly uloženy do nového dokladu. V tomto případě se v tabulce `ZISEG` založí fiktivní položka, jejíž následné zpracování probíhá stejným způsobem jako u standardních. Tato položka se nijak neukládá do SAP standardu a nelze ji tedy například přímo zaúčtovat, což zároveň kontroluji před všemi voláními BAPI, viz ukázkou na obr. 6.9. Fiktivní položky slouží pouze jako pomůcka pro možnost odeslat nález zpět do terminálu pro opětovnou kontrolu předtím, než bude případně zanesen do SAP standardu jedním ze způsobů popsaných v kap. 5.1.8.

6.1.5 Rozhraní systému

Pro předávání dat mezi systémem SAP a mobilními terminály slouží skupina funkčních modulů implementovaných na straně SAP zveřejněných jako SOAP webová služba. Tyto moduly slouží pro získání uživatelských účtů, inventurních dokladů pro počítání, odesílání dat počítání do systému a validaci případných nálezů (v online režimu terminálů).

Veškeré funkční moduly jsem zahrnul do skupiny funkcí (kap. 3.1.1) nazvané `Z_M_BC_INVENTORY`, ze které jsem následně v SAP vygeneroval SOAP webovou službu. Stěžejními částmi rozhraní jsou následující funkční moduly:

- `Z_M_BC_GETUSERS` - získání seznamu uživatelů mobilních terminálů a jejich zašifrovaných hesel. Volitelně lze seznam omezit na konkrétní závod.
- `Z_M_MATPHYSINV_GETDATA` - získání dat pro inventurní počítání. Vstupem je identifikace požadované inventury (číslo inventury, fiskální rok) a uživatel, který o data žádá. Zprvu jsou provedeny příslušné kontroly (existence/aktiva-ce inventury, oprávnění uživatele atd.) a následně jsou získána data inventury. Odesílána jsou kompletní data, tedy všechny položky, příslušné měrné jednotky a případná existující data počítání. Odeslání existujících dat počítání jsem se rozhodl přidat pro možnost případného přerušení práce, případně dokončení počítání na jiném zařízení, se zachováním možnosti průběžného přehledu na straně terminálu (viz 5.5). Pro získání dat jsou využívány statické metody pomocné třídy `ZMBCINV` (kap. 6.1.3).
- `Z_M_MATPHYSINV_GET_MAT_INFO` - validace předaných dat, případně získání dodatečných dat z podnikového systému. Jedná se pouze o zprostředkované volání metody `GET_MATERIAL_INFO` třídy `ZMBCINV` (kap. 6.1.3), aby ji bylo možné volat mimo systém SAP.
- `Z_M_MATPHYSINV_POSTCOUNTDATA` - nahrání dat počítání do SAP. Před samotným uložením dat do DB tabulky `ZMINV_COUNTING` proběhne v případě nálezů s chybějícím číslem materiálu (např. pokud se pracovalo v offline režimu bez možnosti volat `*GET_MAT_INFO`) pokus o jeho dohledání, například podle zadaného EAN.

6.2 Aplikace pro mobilní terminály

Jak již bylo řešeno v kapitole 5.5, mobilní terminály slouží převážně jako náhrada za původní systém zapisování zjištěných skutečností tužkou do papírových podkladů.

Pro tuto práci mi byl zapůjčen mobilní terminál s operačním systémem android (viz specifikace v tab. 6.1), nicméně jsem se rozhodl využít takové nástroje, které usnadní případný přesun na jiné platformy. Jedním z těchto nástrojů je framework

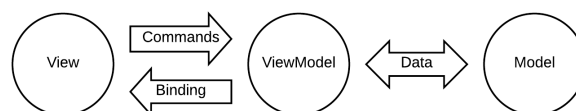
Xamarin od společnosti Microsoft. Základem tohoto prostředí je možnost vytvořit jeden společný základní kód (C# .NET) a následně ho využít na různých platformách jen s minimem potřebných změn [19].

Tabulka 6.1: Specifikace terminálu Zebra MC3200

Operační systém	Android 4.1
Snímač Č.K.	1D laser, otočná hlava
Konektivita	WiFi: 802.11a/b/g/n Bluetooth: v2.1
Rozměry	Délka 60.9 mm, šířka 81,9 mm, hloubka 40 mm
Hmotnost	372 g včetně baterie, řemínku a stylusu
Displej	320 x 320 px, podsvícený
Dotykový panel	tvrzené sklo, odporový snímač
Klávesnice	38 tlačítek
Paměť	1 GB RAM / 4 GB Flash, rozšiřitelné kartou microSDHC o kapacitě až 32 GB

6.2.1 Návrhový vzor MVVM

Při vývoji aplikace jsem se snažil dodržovat strukturu návrhového vzoru MVVM (Model-View-ViewModel). Základní struktura spočívá v oddělení uživatelského rozhraní (View), třídy držící stav aplikace (ViewModel) a modelu, který popisuje data [20].



Obrázek 6.10: Návrhový vzor MVVM

- **Model** popisuje data, například tabulky databáze,

- **View** je uživatelské rozhraní, které může být popsáno v jazyce XAML. Jedná se o stránku aplikace, ovládací prvek apod.
- **ViewModel** drží stav aplikace a spojuje View s Modelem. View čerpá data pro zobrazení z ViewModelu (VM) pomocí tzv. bindingu s podporou propagace změn. To znamená, že pokud se provázaná proměnná ve VM změní, projeví se okamžitě změna v příslušném View.

6.2.2 Struktura aplikace

Aplikaci jsem rozdělil do jednotlivých logických celků s ohledem na návrhový vzor MVVM (kap. 6.2.1) a principy platformy Xamarin, kdy je oddělena na platformě nezávislá část aplikace od té, jejíž implementace závisí na konkrétním cílovém systému. Jako příklad na platformě závislé implementace v této práci mohu uvést přístup k filesystému nebo zařízení pro snímání čárových kódů.

Pro napojení na platformě závislé části a dodatečných funkčních celků, které jsou využívány napříč celým systémem, jsem vytvořil statickou třídu `AppModel`. Její instance je založena a inicializována při startu aplikace a následně je dostupná po celou dobu běhu programu.

6.2.3 Konfigurační soubor

Pro ukládání a možnost úprav konfigurace prostřednictvím libovolného textového editoru jsem se rozhodl využít `*.ini` konfigurační soubor. Slouží pro uchování uživatelského a technického nastavení, mezi které patří:

- přístupové údaje do SAP:
 - uživatelské jméno,
 - heslo,
 - adresa webové služby,
- výchozí rok inventury pro usnadnění zadávání na terminálu,

- nastavení online režimu,
- prodleva pro synchronizaci dat v online režimu,
- výchozí množství, které se při snímání položek předem vyplní.

Přístup k datům uloženým v konfiguračním souboru jsem realizoval pomocí třídy `AppConfig` přístupné z `AppModelu` (kap. 6.2.2).

Některé hodnoty lze nastavovat přímo z aplikace (např. předvyplněné hodnoty, online mode), jiné je nutné nastavovat externě. Hodnotami, které nelze nastavit přímo přes aplikaci, jsou údaje pro přístup do SAP, u nichž nepředpokládám časté změny a měl by je měnit pouze správce systému.

V případě, že by při startu aplikace nebyl soubor nalezen, nebo byl poškozen, vytvoří se automaticky nový s výchozími hodnotami.

6.2.4 Databáze a datový model

Pro ukládání dat v aplikaci jsem se rozhodl použít SQLite databázi, díky čemuž budou data dostupná i při opakovaném vypnutí a zapnutí aplikace v prostředí bez možnosti připojení k síti. Například v offline režimu tedy nedojde ke ztrátě zpracovaného počítání v případě, že se baterie zařízení vybije dřív, než bylo možné odeslat data do SAP.

Veškerá data jsou ukládána do objektů příslušných tříd, dle kterých byla vytvořena struktura databáze díky ORM (Objektově relační mapování). Jako rozhraní pro přístup k takto vytvořeným DB tabulkám jsem následně implementoval generické repositáře, což v budoucnu usnadňuje případný přechod na jinou databázi. Stačilo by pouze nainstalovat jinou DB a upravit repositář, bez nutnosti programových úprav v ostatních částech aplikace přístupujících k datům. Ukázku části implementace generického repositáře lze vidět na obr. 6.11, jeho inicializaci na obr. 6.12 a použití na obr. 6.13.

```

public class BaseRepository<T> : IRepository<T> where T : class, new()
{
    protected SQLiteConnection db;

    public BaseRepository(SQLiteConnection db)
    {
        this.db = db;
    }

    public int Insert(T entity) {...}

    public int Update(T entity) =>
        db.Update(entity);

    public int Delete(T entity) =>
        db.Delete(entity);
}

```

Obrázek 6.11: Ukázka implementace repositáře

```

InvHeaderRepository = new BaseRepository<InventureHeader>(DB);
InvItemRepository = new BaseRepository<InventureItem>(DB);

```

Obrázek 6.12: Ukázka vytvoření instancí generického repositáře

```

// uložit počítání do lokální DB
var invCounting = new InventureCounting
{
    InvNum          = Item.InvNum,
    InvYear         = Item.InvYear,
    Zeili          = Item.Zeili,
    CountCycle     = Item.CountCycle,
    EAN            = Item.EAN,
    Matnr          = Item.Matnr,
    Charg          = Item.Charg,
    Menge          = quantity,
    Meins          = MJText,
    Usnaz          = AppModel.CurrentUser.Login,
    DateTime       = DateTime.Now,
    Comment        = CommentText,
    IsSynchronized = false
};

AppModel.DB.InvCountingRepository.Insert(invCounting);

```

Obrázek 6.13: Ukázka uložení dat pomocí repositáře

6.2.5 Snímání čárových kódů

Společnost Zebra, od které pochází v této práci využitý mobilní terminál, vydala pro vývojáře knihovnu pro práci se čtečkou čárových kódů. Její implementace umožňuje mimo samotného snímání hodnot čárových kódů i aktivaci a deaktivaci skenování a nastavení přípustných druhů kódů.

Zmíněných možností knihovny jsem využil při vývoji aplikace. Snímání kódů je aktivní pouze v relevantních částech - například pokud se uživatel nachází v nastavení aplikace, není důvod aby bylo snímání zapnuté.

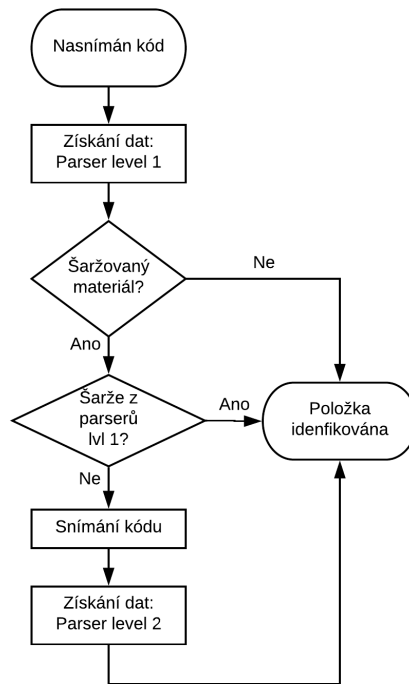
Parsery

Pro eliminaci možnosti snímání nesprávných kódů a případné zpracování kódů (nesoucích více než jednu informaci) jsem se rozhodl implementovat systém tříd, které nazývám BC (BarCode) parsery. Umožňují jednoduše vydefinovat, jaké kódy jsou (s ohledem na stav aplikace) aktuálně přípustné, případně určit přesné části kódu a informaci, kterou nesou.

Aplikaci jsem naprogramoval tak, aby tento systém využívala (algoritmus viz obr. 6.14). Lze tak jednoduše pouhou úpravou parametrů příslušných objektů zcela změnit způsob, jakým snímání položek probíhá. Díky tomu je usnadněna případná implementace do podniků, kde každý využívá jiný systém značení, případně nasazení v podniku, kdy se značení na jednotlivých skladech liší. Při vývoji aplikace jsem například několik testovacích parserů uložil do lokální databáze a následně mezi nimi jednoduše za běhu přepínal v nastavení.

Jako příklad uvedu dva možné scénáře, jak lze tuto funkcionalitu využít:

1. Podnik využívá jeden složený čárový kód (např. CODE-128), kdy na prvních třinácti pozicích je zapsán EAN a následně za ním je šarže.
2. Podnik využívá pro označení položek dva čárové kódy - jedním je EAN, na druhém je (například v kódu CODE-128) uvedena šarže. Při zpracování inventury tedy pracovník sejme nejdříve EAN a poté nasnímá šarži.



Obrázek 6.14: Algoritmus snímání čárových kódů s využitím parserů

V obou dvou případech není nijak nutné zasahovat do logiky skenování kódů, stačí pouze změnit způsob vytváření instance parserů. Rozdíly lze vidět v ukázce na obrázku 6.15.

V případě nasnímání položky, která není obsažena v inventurním dokladu, mohou nastat dva možné scénáře:

1. Offline režim - pokud pracovník potvrdí, že se jedná o nález, zadá případnou šarži, měrnou jednotku a množství. Jediná kontrola, která proběhne, je kontrola měrné jednotky - zda se jedná o reálnou jednotku existující v systému SAP.
2. Online režim - proběhne kontrola, zadaná položka existuje v podnikovém systému (číslo materiálu, EAN). Pokud ano, jsou ze systému načtena dodatečná data - například krátký text materiálu, měrná jednotka nebo zda se jedná o šaržovaný materiál.

```

var BCParser2 = new BCParser
{
    Name = "Code128: EAN13+Charg"
};
BCParserRepository.Insert(BCParser2);
parserId = BCParser2.Id;

var EANGetter1 = new Getter
{
    Level = 1,
    Name = "EAN",
    BCParserId = parserId,
    Position = 0,
    Length = 13,
    CodeName = "Code128"
};
BCGetterRepository.Insert(EANGetter1);
BCParser2.Getters.Add(EANGetter1);

var ChargGetter2 = new Getter
{
    Level = 1,
    Name = "Charg",
    BCParserId = parserId,
    Position = EANGetter1.Length,
    Length = SapChargLength,
    CodeName = "Code128"
};
BCGetterRepository.Insert(ChargGetter2);
BCParser2.Getters.Add(ChargGetter2);

var BCParser4 = new BCParser
{
    Name = "EAN13 -> Charg(Code128)"
};
BCParserRepository.Insert(BCParser4);
parserId = BCParser4.Id;

var EANGetter4 = new Getter
{
    Level = 1,
    Name = "EAN",
    BCParserId = parserId,
    Position = 0,
    Length = SapEanLength,
    CodeName = "Ean13"
};
BCGetterRepository.Insert(EANGetter4);
BCParser4.Getters.Add(EANGetter4);

var ChargGetter4 = new Getter
{
    Level = 2,
    Name = "Charg",
    BCParserId = parserId,
    Position = 0,
    Length = SapChargLength,
    CodeName = "Code128"
};
BCGetterRepository.Insert(ChargGetter4);
BCParser4.Getters.Add(ChargGetter4);

```

Obrázek 6.15: Ukázka parserů čárových kódů

6.2.6 Přístup do SAP

Pro výměnu dat s podnikovým systémem v aplikaci využívám SOAP webové služby. Systém SAP pro tyto účely umožňuje vygenerování WSDL souboru s popisem služby, který jsem následně využil pro automatické vygenerování takzvaných proxy tříd zaštiťujících komunikaci. Jediným zásahem, který jsem ve vygenerovaném kódu provedl, bylo nahrazení napevno vepsané adresy serveru za kód, který získává adresu z konfiguračního souboru (kap. 6.2.3).

Jako prostředníka mezi generovanými třídami, které se v případě změny rozhraní celé přegenerují, jsem vytvořil novou třídu (nazvanou `SAPConnection`). Díky tomuto řešení není nutné v případě úprav rozhraní procházet kód celé aplikace a aktualizovat všechna místa, ve kterém jsou služby využity. Tato třída zároveň obstarává konverzi dat mezi strukturami získanými z rozhraní SAP a datovým modelem aplikace, případné ukládání získaných dat do lokální databáze nebo odeslání dat z databáze do SAP (data počítání). Před každým voláním webové služby jsem samozřejmě implementoval i kontrolu dostupnosti spojení.

Synchronizační vlákno

Synchronizační vlákno slouží pro automatické odesílání dat počítání v pravidelných intervalech do systému SAP. Prodlevu mezi jednotlivými intervaly lze nastavit v konfiguračním souboru.

Vlákno je spuštěno pouze v případě, že terminál pracuje v online režimu. Při jednotlivých iteracích je nejprve provedena kontrola dostupnosti spojení, následována voláním metody pro odeslání dat počítání pomocí třídy `SAPConnection` (kap. 6.2.6), která odešle všechna doposud nesynchronizovaná data počítání.

6.2.7 Uživatelské rozhraní

Při tvorbě uživatelského rozhraní pro mobilní terminály jsem kladl důraz na jednoduchost a přehlednost, jelikož se jedná o aplikaci primárně určenou pro sběr dat.

Celé uživatelské rozhraní se skládá z několika relativně jednoduchých obrazovek, na kterých jsou zobrazovány jen nezbytně nutné údaje.

Obrazovky aplikace:

- **Přihlašovací obrazovka.** Uživatel vyplní jméno a heslo, které je následně zašifrováno algoritmem SHA1 (stejně jako v SAP viz kap. 6.1.2) a porovnáno s uživatelskými údaji ze SAP.
- **Menu.** Základní menu umožňující ukončení aplikace, přechod do **nastavení** či spuštění inventury. V případě inventury je uživatel vyzván k zadání čísla inventury (možno též naskenovat kód s číslem) a fiskálního roku (předvyplněn dle nastavení v konfiguračním souboru). Následně jsou stažena data inventury ze SAP a přepne se stránka na inventuru.
- **Inventura.** Jedná se o základní obrazovku pro inventuru, na které je zobrazen základní přehled o aktuálním průběhu počítání. Na obrazovce je vypsáno číslo inventury, závod a sklad na kterém počítání probíhá. Dále je zobrazen počet zpracovaných položek a položek, ke kterým doposud nebylo zadáno žádné množství. V této části aplikaci je aktivováno skenování čárových kódů s využitím algoritmu na obr. 6.14. Po nasnímání přípustného čárového kódu je uživatel přesměrován na stranu s detailem položky, případně je dotázán, zda se jedná o nález. V případě nálezu proběhne validace položky a stažení dodatečných dat ze SAP.
- **Detail položky.** Na této stránce jsou zobrazeny dodatečné informace k položce inventury a slouží k zadání nalezeného množství. V případě nálezů v offline režimu lze vyplnit i měrnou jednotku pro zadávání, v ostatních případech je tato možnost potlačena a je využita měrná jednotka získána ze systému SAP. Konkrétní záznam počítání lze navíc doplnit o libovolnou textovou poznámku. Pokud již pro položku bylo v minulosti zadáno počítání, je zpřístupněna volba pro zobrazení **historie zadávání**.

- **Historie zadávání.** Slouží pro zobrazení v minulosti zadaných hodnot počítání pro konkrétní položku (viz předchozí bod). V tomto přehledu je uveden například přesný datum a čas vstupu dat, zadané množství, případně textová poznámka. V této části aplikace je zároveň umožněno provést opravu předchozích dat formou anulování příslušného záznamu, případně opravou zadaného množství. Pro možnost přehledu, jak proces inventury reálně probíhal (viz kap. 6.1.4), neprovádím editaci dat, nýbrž založení opravného záznamu s odpovídající poznámkou (viz záznam na obr. 6.8 s poznámkou „oprava množství“). Pokud například pracovník upravuje množství z „23 KS“ na „25 KS“, je vložen nový záznam s hodnotou „2 KS“ a poznámkou „oprava počítání“. Z pohledu uživatele došlo k pouhé úpravě hodnoty, tyto doplňkové záznamy jsou v přehledu aplikace skryty, aby nepůsobily zmatky.
- **Přehled zbývajících položek.** Na této stránce je vypsán seznam všech položek inventurního dokladu, ke kterým prozatím neexistuje žádný záznam se zadaným množstvím. Poslouží například v situaci, kdy již pracovník prošel celý sklad, nicméně stále v účetnictví existují materiály, které nenalezl. Může se rozhodnout, že celý sklad znovu projde a bude hledat pouze konkrétní chybějící položky. V případě neúspěchu při hledání je na této stránce umožněno označit položku jako nenalezenou - technicky to znamená, že se do dat uloží záznam počítání s nulovým množstvím. Položka se nadále v seznamu zobrazovat nebude a do SAP se přenesou informace, že je tato položka vyřízená a skutečně nebyla nalezena.
- **Nastavení.** Slouží pro úpravu uživatelského nastavení aplikace - například navrhovaný (předvyplněný) fiskální rok, přepínání mezi online a offline režimem, výběr příslušného BC Parseru (kap. 6.2.5).
- **Univerzální obrazovka** pro zadávání hodnot, která je využívána ve většině částí uživatelského rozhraní. Zobrazí stránku s libovolným popisem, volitelně zobrazeným polem pro zadávání hodnot, případně je možné povolit či zakázat

čtení čárových kódů. Příklad použití této stránky lze vidět v kódu na obr. 6.16.

```
var InputPageVM = new InputPageViewModel(Navigation)
{
    Question = "Zadejte kód šarže",
    Btn1Txt = "OK",
    Btn2Txt = "Cancel",
    IsInputVisible = true,
    IsScannerEnabled = true
};

InputPageVM.OnBtn1Clicked += OnItemChargConfirm;
InputPageVM.OnBtn2Clicked += OnInputAbort;

var inputPage = new InputPage(InputPageVM);
await Navigation.PushAsync(inputPage);
}

private void OnItemChargConfirm(object sender, string data)
{
```

Obrázek 6.16: Ukázka použití generické zadávací obrazovky

7 Testování

Bohužel jsem před dokončením této práce neměl příležitost vyvinuté řešení implementovat a otestovat v reálném podniku.

Systém v této práci popsany jsem vyvíjel nad „čistou“ instalací podnikového systému bez reálných dat. Pro účely průběžného testování jsem v něm vytvořil nový sklad a založil několik materiálů (šaržovaných i bez šarží), které jsem následně na sklad z hlediska účetnictví umístil.

Pro založené materiály jsem vytvořil fiktivní kódy EAN a následně pro testy snímání čárových kódů vytvořil několik různých variant systému čárových kódů, kterými by položky mohly být v reálném provozu značeny (viz kap. 6.2.5).

Systém jsem testoval průběžně během vývoje obou částí aplikace (SAP, terminály) za použití výše zmíněných testovacích dat a opakovaného zpracování inventury na testovacím skladu. Za zmínku jistě stojí možnost systému SAP testovat dílčí funkční moduly, případně metody funkcí, pomocí vlastních nástrojů. Je v nich umožněno vyplnit příslušné vstupní parametry, spustit funkci a sledovat výstupní hodnoty (bez psaní kódu, pomocí grafického rozhraní), případně si hodnoty parametrů uložit pro opakované spouštění.

7.1 Testy procesu neproškolenými subjekty

V kapitole 6.2.7 bylo zmíněno, že jsem se aplikaci pro mobilní terminály snažil navrhnout pokud možno co nejvíce uživatelsky přívětivě a intuitivně. Pro ověření, zda se mi toho podařilo dosáhnout, jsem se rozhodl založit v systému inventuru a pře-

dat neproškoleným testovacím subjektům (čtyři osoby, individuálně) terminál zároveň s předměty označenými čárovými kódy.

Subjektům jsem kromě přihlašovacích údajů předal pouze informaci o číslu inventury (pro stažení dat ze SAP na terminálu) a požadavek, aby vyzkoušeli nasnímat kódy se zadáním (libovolného) nalezeného množství a ukončení práce s alespoň jednou nenalezenou položkou (kterou jsem jim nepředal). Jediné co jsem jim z hlediska ovládání terminálu ukázal, bylo tlačítko, které aktivuje čtecí zařízení čárových kódů.

Do průběhu testování inventury jsem nijak nezasahoval, pouze pozoroval celkový průběh. Subjekty se v ovládání aplikace dokázaly bez pomoci zorientovat a bez nápovědy zvládly i označování nenalezených položek.

Tento test potvrdil, že je ovládání aplikace lehce srozumitelné bez nutnosti zásadních investic do zaškolení uživatelů terminálů. Zároveň díky experimentování testovacími subjekty byly odhaleny nedostatky, které jsem následně v aplikaci dle možností eliminoval:

- Skenování různých druhů kódů - v prvotní verzi jsem sice již měl implementované BC parsery, ovšem bez omezení na druh kódu (skener snímal vše, pouze se vybíraly podřetězce). Stalo se tak například, že terminál byl sice přepnut do režimu skenování složeného kódu Code128, avšak testovací subjekt omylem naskenoval čárový kód typu EAN ze svačiny, kterou měl položenou vedle testovací položky, případně jiný typ kódu, který byl umístěn na testovacím objektu. Z toho důvodu jsem následně aplikaci rozšířil o možnost nastavení druhu kódu, který má být snímán, aby se takovéto chyby alespoň částečně eliminovaly.
- Historie zadávání - uživatelé rychle pochopili práci s opravou počítání. Nicméně zprvu aplikace ukazovala i jednotlivé „opravné“ záznamy, které se pro subjekty jevily matoucí. Jeden z nich se například opakovaně snažil odstranit již jednou smazaný záznam, čímž se z celkového hlediska následně dostal do záporného stavu. Proto jsem se rozhodl, že v přehledu historie zadávání na terminálu

bude zobrazováno sečtené množství všech oprav pro konkrétní zadání, namísto detailního rozepsání, které je zobrazeno v přehledu v SAP aplikaci.

8 Závěr

Cílem práce bylo prozkoumat současné možnosti inventarizace a dále vývoj nástrojů pro proces založený na použití zařízení pro čtení čárových kódů s cílem usnadnění a urychlení procesu.

V práci byl vysvětlen proces inventarizace, jeho částí a účelu, ke kterému slouží. Dále byla provedena rešerše existujících řešení od různých dodavatelů, avšak se nepodařilo získat detailní informace, jak danou problematiku po technické stránce řeší a porovnat finanční nároky, jelikož tyto údaje nejsou veřejně dostupné.

Podařilo se vyvinout systém, který rozšiřuje standardní funkcionalitu podnikového systému SAP, využívající mobilní terminály snímající čárové kódy. Nový systém plně spolupracuje se stávajícím systémem a je možné libovolně využívat a kombinovat funkce původního a nového řešení. Nová funkcionalita byla implementována formou nadstavby nad stávajícím řešením bez jakýchkoliv zásahů do originálních kódů.

Čtecí zařízení dokáží pracovat v režimu, kdy mají soustavně dostupné připojení do podnikového systému a mohou průběžně získávat či odesílat data. Zároveň umožňují práci v tzv. offline módu, kdy spojení není dostupné a data jsou odeslána až když je to možné. Dále byla implementována funkcionalita umožňující označení položek pro opakované provedení fyzického počítání, možnost detailního sledování průběhu inventury včetně prohlížení logů. Vyvinuté nástroje rovněž umožňují vyřazování se s případnými nálezy, tedy položkami které nefigurují v inventurním soupisu, avšak se ve skladě fyzicky nacházejí.

V práci byly rovněž z technického hlediska vysvětleny základní principy kontrol oprávnění v podnikovém systému, které následně byly v praktické části využity. Rovněž byl vyvinut systém vedení a správy uživatelských účtů pro mobilní terminály.

Vyvinutý systém byl podroben testování neproškolenými uživateli, což ověřilo, že je ovládání dostatečně intuitivní a lehce srozumitelné. Před dokončením práce se systémem nepodařilo nasadit a otestovat v reálném provozu, testování tedy probíhalo pouze na virtuálních datech.

Při vývoji aplikace pro terminály byly využity technologie usnadňující přenositelnost na další mobilní platformy. V budoucnosti by tak mohlo být zajímavé upravit aplikaci pro využití s běžně dostupnými mobilními telefony disponujícími kamerou. Díky tomu, že byl při návrhu aplikace kladen důraz na univerzálnost a snadnou rozšiřitelnost, stačilo by nahradit pouze část aplikace, která načítá data ze snímače kódů za systém rozpoznávání obrazu.

Literatura

1. FOŘTOVÁ, Lucie. Inventarizace majetku a závazků v podniku [online]. 2013 [cit. 2018-08-01]. Dostupné z: <https://theses.cz/id/g2bqn9>.
2. SCHIFFER, Vladimír. *Inventarizace majetku a závazků v praxi podnikatelů*. Vyd. 1. Praha: BOVA POLYGON, 2005. ISBN 80-7273-117-3.
3. PIHIR, I.; PIHIR, V.; VIDAČIĆ, S. Improvement of warehouse operations through implementation of mobile barcode systems aimed at advancing sales process. In: *Proceedings of the ITI 2011, 33rd International Conference on Information Technology Interfaces*. 2011, s. 433–438. ISSN 1330-1012.
4. 60 LET S ČÁROVÝM KÓDEM: POCHOPTĚ JEHO ANATOMIÍ. *National Geographic* [online]. 2012 [cit. 2018-07-31]. Dostupné z: <https://www.national-geographic.cz/clanky/60-let-s-carovym-kodem-pochopte-jeho-anatomii.html>.
5. ZVELEBILOVÁ, Soňa. UCC/EAN-128: pomocník, nebo strašák? *AUTOMA* [online]. Roč. 2009, č. 07 [cit. 2018-08-19]. Dostupné z: http://automa.cz/cz/casopis-clanky/ucc/ean-128-pomocnik-nebo-strasak-2009_07_39322_4754/.
6. DOČEKAL, Daniel. Budoucnost a současnost „čárových“ kódů pro mobily. *Lupa.cz* [online]. 2008 [cit. 2018-08-05]. Dostupné z: <https://www.lupa.cz/clanky/budoucnost-a-soucasnost-carovych-kodu-pro-mobily/>.
7. AKBARI, A.; MIRSHAHI, S.; HASHEMIPOUR, M. Comparison of RFID system and barcode reader for manufacturing processes. In: *2015 IEEE 28th Ca-*

- nadian Conference on Electrical and Computer Engineering (CCECE)*. 2015, s. 502–506. ISSN 0840-7789. Dostupné z DOI: 10.1109/CCECE.2015.7129326.
8. TERESHCHENKO, I.; SHTANGEY, S.; TERESHCHENKO, A. The application SAP® ERP principles for the development and implementation of corporate integrated information system for SME. In: *2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S T)*. 2016, s. 168–170. Dostupné z DOI: 10.1109/INFCOMMST.2016.7905370.
 9. SAP Corporate Fact Sheet [online] [cit. 2018-08-20]. Dostupné z: <http://www.sap.com/corporate-en/factsheet>.
 10. What are SAP Modules: A Complete List of SAP ERP Modules [online]. 2018 [cit. 2018-08-20]. Dostupné z: <http://blog.koenig-solutions.com/2018/02/12/what-are-sap-modules-a-complete-list-of-sap-erp-modules/>.
 11. What is SAP ABAP? [online] [cit. 2018-08-05]. Dostupné z: <https://www.stechies.com/abap/>.
 12. SAP ABAP - Function Modules [online] [cit. 2018-08-15]. Dostupné z: https://www.tutorialspoint.com/sap_abap/sap_abap_function_modules.htm.
 13. ROUSE, Margaret. SAP BAPI (Business Application Programming Interface) [online] [cit. 2018-08-15]. Dostupné z: <https://searchsap.techtarget.com/definition/BAPI>.
 14. What are Authorization Objects [online] [cit. 2018-08-16]. Dostupné z: <https://wiki.scn.sap.com/wiki/display/ABAP/What+are+Authorization+Objects>.
 15. AUTHORITY-CHECK [online] [cit. 2018-08-16]. Dostupné z: https://help.sap.com/doc/abapdocu_752_index_htm/7.52/en-US/abapauthority-check.htm.

16. Využijte našeho know-how pro evidenci skladových zásob [online] [cit. 2018-08-02]. Dostupné z: <https://www.eprin.cz/clanky-evidence-skladovych-zasob-v-sap-titul.html>.
17. Řešení pro řízené sklady [online] [cit. 2018-08-03]. Dostupné z: <https://www.sabris.com/cz/s3511/Reseni-a-sluzby/Oblast-SAP/c1995-Reseni-pro-rizene-sklady>.
18. Produktový list- mStock [online] [cit. 2018-08-04]. Dostupné z: http://www.sapmobility.cz/wp-content/uploads/2014/03/mStock_PL-mail.pdf.
19. Aplikace vytvořené pomocí Xamarinu se chovají a vypadají nativně, protože nativní jsou [online] [cit. 2018-08-25]. Dostupné z: <https://visualstudio.microsoft.com/cs/xamarin/>.
20. mvvm: model-view-viewmodel [online] [cit. 2018-08-25]. Dostupné z: <https://www.dotnetportal.cz/clanek/4994/MVVM-Model-View-ViewModel>.

A Obsah přiloženého CD

K práci je přiloženo CD se zdrojovými kódy vyvinutého řešení a elektronickou podobou zprávy. Jeho obsah:

- zdrojove_kody
 - SAP - ukázky zdrojových kódů částí vyvinutých na straně SAP (v programovacím jazyce ABAP). Ne všechny části řešení bylo možné exportovat v jednoduše čitelné podobě. Lze zde nalézt většinu zdrojových kódů, avšak nelze prohlížet datové struktury, podobu nových DB tabulek, objekty oprávnění, obrazovky (dynpra) atd. Ve složce „packet“ je vyexportováno kompletní řešení v binární podobě, které je případně možné nahrát do ERP systému SAP (vyvinuto bylo na verzi SAP S/4 Hana 1611).
 - Terminaly - kompletní projekt (VisualStudio) vyvinutého řešení se všemi kódy.
- elektronická verze této práce