

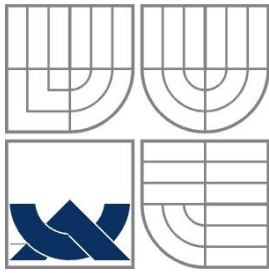
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

Fakulta informačních technologií
Faculty of Information Technology

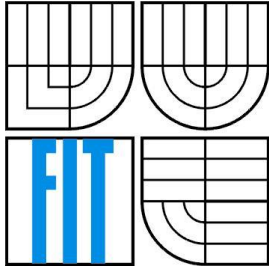
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

Brno, 2016

Zdeněk Bittner



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

UŽIVATELSKÁ AGENDA DOPRAVA PRO
EKONOMICKÝ SYSTÉM STORMWARE POHODA
USER AGENDA TRANSPORT FOR THE ECONOMIC SYSTEM STORMWARE POHODA

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

Zdeněk Bittner

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Filip Orság, Ph.D

BRNO 2016

Abstrakt

Cílem této bakalářské práce je navrhnout a implementovat novou verzi uživatelské agendy Doprava pro ekonomický systém Stormware POHODA, která slouží k objednávání balíkové přepravy u mnoha různých dopravců a tisku adresních štítků na základě dat přímo ze systému Stormware POHODA. K implementaci stěženi části řešení byla využita platforma .NET Framework a programovací jazyk C#.

Abstract

The goal of this bachelor's thesis is to design and implement new version of the user agenda Transport for the economic system Stormware POHODA, which is used to order parcel transportation from many different shippers and to print address labels based on data straight from the system Stormware POHODA. For implementation of the main part of the solution were used .NET Framework and programming language C#.

Klíčová slova

POHODA, Doprava, C#, .NET, Česká pošta, DPD, Geis Parcel, IN TIME, PPL, Uloženska

Keywords

POHODA, Transport, C#, .NET, Czech post, DPD, Geis Parcel, IN TIME, PPL, Uloženska

Citace

Bittner Zdeněk: Uživatelská agenda Doprava pro ekonomický systém Stormware POHODA, bakalářská práce, Brno, FIT VUT v Brně, 2016

Uživatelská agenda Doprava pro ekonomický systém Stormware POHODA

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Filipa Orsága, Ph.D. Další informace mi poskytl konzultant Ing. Tomáš Zoufalý z firmy BHIT CZ s.r.o.. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Zdeněk Bittner
17. května 2016

© Zdeněk Bittner, 2016

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah.....	1
1 Úvod.....	3
1.1 Struktura práce.....	3
2 Ekonomické systémy a logistika.....	4
2.1 Popis problému	4
2.2 Ekonomické systémy	4
2.2.1 Stormware POHODA	4
2.2.2 Money	5
2.2.3 Helios	6
2.3 Dopravci.....	6
2.3.1 Česká pošta	6
2.3.2 DPD	8
2.3.3 Geis Parcel	8
2.3.4 IN TIME	9
2.3.5 PPL.....	10
2.3.6 Uloženska	11
2.4 Shrnutí.....	12
3 Návrh řešení	13
3.1 Původní řešení.....	13
3.2 Požadavky a návrh řešení.....	14
3.2.1 Číselné řady	15
3.2.2 Adresní štítky	15
3.2.3 Odeslání dat	15
3.2.4 Uživatelská agenda Doprava.....	15
3.2.5 Aplikace Doprava	16
3.3 Použité technologie.....	17
3.3.1 Microsoft SQL Server.....	17
3.3.2 .NET Framework	18
3.3.3 Knihovna DevExpress Reporting	18
4 Implementace a testování	19
4.1 Rozšíření Stormware POHODA E1.....	19
4.1.1 Struktura databáze.....	19
4.1.2 Volitelné parametry a uživatelské seznamy.....	21
4.1.3 Tvorba uživatelských agend	21

4.1.4	Úprava formuláře.....	23
4.1.5	Tiskové sestavy.....	23
4.1.6	Externí nástroje.....	23
4.2	Popis tříd aplikace Doprava.....	24
4.2.1	Document.....	24
4.2.2	Package.....	24
4.2.3	DocumentCreator.....	24
4.2.4	Config.....	25
4.2.5	ConfigHelper.....	25
4.2.6	Shipper.....	25
4.3	Podrobný popis třídy Shipper.....	25
4.3.1	Základní metody.....	25
4.3.2	Příklad vytvoření nového dopravce.....	26
4.4	Testování.....	27
4.4.1	Statistiky za měsíc březen 2016.....	27
5	Závěr.....	28

1 Úvod

Elektronické nakupování není pro nikoho z nás žádná novinka, stačí si z pohodlí domova vybrat produkt, po kterém toužíme, zadat adresu a kontaktní informace, zvolit svého oblíbeného dopravce a pak už stačí pouze čekat, než se někdo za nás o vše postará. V ideálním případě máme zásilku do dvou pracovních dnů u sebe. Z pohledu prodejce to ovšem tak jednoduché není, zvláště pokud musí vyřizovat denně desítky, ne-li stovky objednávek. Prodejce musí všechny údaje od zákazníků ručně přepisovat většinou do několika různých systémů, protože každý dopravce má na to své vlastní rozhraní. Při přepisování údajů z jednoho systému do druhého pak navíc může vzniknout chyba z nepozornosti.

Za účelem zrychlení a zjednodušení tohoto procesu byla vytvořena uživatelská agenda Doprava. Snahou bylo vytvořit jednotné rozhraní pro všechny dopravce, díky kterému by bylo možné několika kliknutími myší objednávat zásilky u dopravců na základě dat přímo z ekonomického systému Stormware POHODA.

Ekonomický systém POHODA byl vybrán pro realizaci práce kvůli jednoduchosti tvorby nových uživatelských agend, které jsou plnohodnotnou součástí celého systému a zároveň také protože původní řešení vytvořené firmou BHIT CZ s.r.o. bylo také určeno pro systém POHODA a jakožto zadavatel této práce vyžaduje alespoň částečnou kompatibilitu s původním řešením, aby přechod na novou verzi u existujících zákazníků byl co nejsnazší. Původní řešení bude postupně u všech zákazníků nahrazeno verzí vytvořenou v rámci této bakalářské práce a do budoucna se počítá i s nasazením u desítek nových zákazníků.

1.1 Struktura práce

V druhé kapitole bude pojednáno o nejvýznamnějších ekonomických systémech na českém trhu a jejich možnostech napojení na dopravce. Dále budou popsány základní požadavky pro komunikaci a vytváření adresních štítků dopravců implementovaných v rámci této práce. Třetí kapitola se bude zabývat návrhem nového řešení. Nejprve bude nastíněno původní řešení, z kterého tato práce vychází a dále budou popsány požadavky na nové řešení, návrh uživatelského rozhraní Doprava a budou uvedeny hlavní technologie využití pro realizaci. Čtvrtá kapitola se bude věnovat tvorbě uživatelských agend v rámci ekonomického systému Stormware POHODA, bude proveden rozbor základních tříd využitých pro realizaci uživatelského rozhraní Doprava a také bude popsán průběh testování výsledného řešení. Na závěr budou zhodnoceny přínosy nového rozhraní a možnosti využití uživatelské agendy pro jiné systémy.

2 Ekonomické systémy a logistika

V této kapitole bude nastíněna základní myšlenka uživatelské agendy Doprava, dále budou popsány nejznámější ekonomické systémy na českém trhu a nakonec bude proveden rozbor základních požadavků pro vytvoření doručitelné zásilky dopravci implementovaných v rámci této práce.

2.1 Popis problému

Jak již bylo zmíněno v úvodní kapitole, cílem uživatelské agendy Doprava je především usnadnění a zrychlení procesu vytváření adresních štítků a objednávání svozu zásilek u mnoha různých dopravců pomocí společného rozhraní, aby byla co nejvíce eliminována uživatelská interakce se systémy dopravců a ruční přepisování údajů z jednoho systému do druhého. Vytvoření nové zásilky by mělo probíhat především na základě dokladů z agend ekonomického systému Stormware POHODA, nejlépe bez jakékoliv potřeby upravovat vstupní data. V ideálních případech budou všechny informace nutné pro vytvoření zásilky přenášeny do systému POHODA přímo z elektronického obchodu za pomoci importního můstku. Tím jsou myšleny například informace o zvoleném způsobu dopravy, vybrané služby a případně výdejní pobočky. Ale mohou nastat i situace, kdy takovéto informace nejsou k dispozici a bude tedy potřeba mít možnost vše ještě upravit na dokladu, případně přes uživatelské rozhraní Doprava. Obsluha expedice zásilek bude v průběhu dne vytvářet adresní štítky, kterými bude polepovat jednotlivé zásilky. Data o již vytvořených zásilkách se budou uchovávat v uživatelské agendě Doprava pro evidenci historie odeslaných zásilek. Ke konci dne nebo až uzná obsluha za vhodné, dojde k odeslání informací o zásilkách na servery dopravců a vytištění předávacích protokolů, pokud je dopravce vyžaduje. V případě potřeby bude umožněno znovu vytištění adresního štítku na základě již vytvořeného záznamu v uživatelské agendě. A také bude mít obsluha možnost přímo z uživatelské agendy spustit nástroj na sledování stavu zásilek.

2.2 Ekonomické systémy

Na českém trhu se nacházejí desítky společností, které se zabývají vývojem či distribucí ekonomických systémů. Mnohé z nich jsou opravdu kvalitní a dokáží poskytnout srovnatelný komfort, jako několikanásobně dražší ekonomické systémy zahraničních společností. V další části této kapitoly budou popsány nejčastěji využívané systémy v České republice a jejich možnosti napojení na dopravce.

2.2.1 Stormware POHODA

Stormware s.r.o. je přední česká softwarová společnost působící na trhu aplikací pro kanceláře a domácnosti již od roku 1993. Je významným partnerem společnosti Microsoft s kompetencí Application Development na úrovni Gold a je také držitelem certifikátu systému řízení jakosti podle normy ISO 9001:2009 jak v oblasti návrhu, vývoje, implementace a podpory softwarových produktů, tak v oblasti organizace a provádění kurzů a seminářů. [1]

Jeden z hlavních produktů této firmy je ekonomický a účetní systém Stormware POHODA (dále jen „POHODA“), který je určen pro malé, střední a větší firmy z řad fyzických i právnických osob. Systém obsahuje velké množství agend, které slouží ke komplexnímu a správnému řízení firmy. Z nejčastěji používaných agend jsou to například Adresář, Fakturace, Banka, Mzdy, Majetek, Sklady

a mnoho dalších. POHODA je k dispozici ve třech řadách, které od sebe odlišují použité technologie a různé úrovně funkcí obsažených v programu. V základní řadě systému POHODA je možné využívat všechny jeho součásti, ale díky použité technologii file-server a databáze ve formátu Microsoft Access je však vhodný zejména pro menší počet uživatelů v menších podnicích nebo pro fyzické osoby. Velký kvalitativní skok je pak vyšší řada POHODA SQL, která využívá technologie klient-server a databáze Microsoft SQL, díky kterým je celý systém výrazně bezpečnější z hlediska neoprávněného přístupu i uchování dat, poskytuje výrazné zvýšení výkonu a také možnost zpracování mnohem většího množství dat naproti základní řadě POHODA. Náročnější uživatelé pak využívají nejvyšší řadu POHODA E1, která je postavena na stejné technologii jako POHODA SQL, ale obsahuje celou řadu rozšiřujících funkcí. Všechny řady systému POHODA je možno zakoupit v různých variantách. Jednotlivé varianty pak určují součásti celého systému ve formě přístupu ke konkrétním agendám a funkcím. [1]

Pro účely uživatelské agendy Doprava můžeme uvažovat pouze řadu POHODA E1, která podporuje definování nových volitelných parametrů a tvorbu uživatelských agend. Ve výsledku to znamená, že si lze přidávat celou řadu nových agend, které budou plnohodnotnou součástí celého systému. Je tedy možné si přizpůsobovat ekonomický systém na míru vlastním požadavkům.

Nejnovější verze systému POHODA umožňuje vytváření zásilek a export dat pro Českou poštu, ale ovšem ve velice omezené míře. Je nutné nastavit číselnou řadu a zároveň s ní i použité služby a poté na dokladu zvolit, v které řadě se má doklad odeslat. Nelze vybírat služby dynamicky podle potřeby nebo přebírat informace o službách z dokladu. Dále existuje i řešení na vytváření datových souborů pro různé dopravce od společnosti FIRMADAT s.r.o., které ale pouze vytvoří datové soubory. Nelze vybrat žádné volitelné služby nebo vytisknout adresní štítek. Datový soubor je nutné nahrát ručně do systému dopravce a poté vytisknout štítky z webové aplikace, pokud to dopravce umožňuje.

2.2.2 Money

Dalšími populárními systémy jsou produkty od společnosti CÍGLER SOFTWARE, a.s.. Tato společnost působí v oblasti informačních systémů od roku 1990 a řadí se na přední pozice na trhu. Stejně jako společnost Stormware vlastní certifikát systému řízení jakosti podle normy ISO 9001:2008 a také dosáhla nejvyšší úrovně certifikace od společnosti Microsoft a to právě Microsoft Gold Certified Partner. [2]

Pro menší a střední společnosti lze doporučit účetní a ekonomický systém Money S3, pro větší společnosti potom ERP (Enterprise Resource Planning) systémy Money S4 nebo Money S5. Stejně jako POHODA jsou i tyto systémy dodávány v různých variantách, které se liší jednotlivými moduly. Již Money S3 nabízí běžně využívané moduly pro vedení podvojného účetnictví i daňové evidence. Nechybí základní moduly jako adresář, objednávky, fakturace, skladová evidence, mzdy a podobně. Pro náročnější uživatele jsou pak vhodnější produkty Money S4 či Money S5. Tyto systémy už umožňují poměrně značné přizpůsobení na míru uživateli. Money S5 je v podstatě na míru šitá instalace, která by bez problémů měla uspokojit i složité požadavky a komplikované vnitropodnikové procesy. Jakousi pomyslnou střední cestou je systém Money S4, který umožňuje přizpůsobení vnitropodnikovým procesům i poměrně podrobným požadavkům uživatelů. Obecně jsou systémy Money velmi dobře ovladatelné a uživatelsky přívětivé. Vzhledově jsou podobné klasickému rozhraní známému díky produktům z řady Office od společnosti Microsoft. [2]

V současné době jsou k dispozici rozšiřující moduly pro export zásilek pro Českou poštu a PPL, ale stejně jako tomu bylo u POHODY, jedná se pouze o vytvoření datových souborů, které je nutné ručně nahrát do systému dopravců. Adresní štítky je také potřeba vytisknout pomocí webové aplikace.

2.2.3 Helios

Systémy Helios pochází od společnosti ASSECO SOLUTIONS, a.s. Tyto systémy jsou na trhu také od roku 1990 a vyskytují se v opět v různých variantách. Mezi ty nejrozšířenější produkty patří Helios Red, Helios Orange a Helios Green. Opět se vzájemně liší počtem obsažených modulů a svojí flexibilitou. Umožňují vedení podvojného účetnictví i daňové evidence a obsahují všechny klasické moduly od objednávek přes fakturace až po agendy pro controlling a business intelligence. [3]

Helios Orange je jeden z nejrozšířenějších a velice oblíbených systémů v České republice. Funguje na operačních systémech Windows a je opět řešený architekturou klient-server. Hladký běh celého systému zajišťuje technologie SQL serveru. Celý systém je navržen pro práci až několika set uživatelů zároveň, což z něj dělá vhodného kandidáta i pro velké společnosti. Obsahuje moduly potřebné pro výrobní, obchodní i servisní společnosti. Velmi detailně jsou pak zpracovány agendy týkající se ekonomiky, personalistiky, plánování a řízení výroby i logistiky. Svoji oblíbenost si udržuje také díky vynikajícímu poměru cena/výkon a poměrně rozsáhlým možnostem uživatelských úprav. Distribuce systému Helios Orange probíhá z větší části nepřímo přes implementační partnery, kterých jsou v České republice a na Slovensku desítky. Uživatelské úpravy v systému Helios lze rozčlenit do několika kategorií. Ty nejjednodušší úpravy jako nastavení vzhledu a uživatelských seznamů dokáže téměř každý i bez speciálního zaškolení. Složitější úpravy jsou pak opět oblastí pro zaškolený personál a vývojáře se znalostmi SQL. [4]

Pro systém Helios existují rozšiřující moduly na vytváření zásilek a tisk adresních štítků u společností Geis Parcel a PPL. Na rozdíl od předchozích systémů není nutné nahrávat data ručně na servery dopravců. Řešení podporuje i úpravu využitých služeb, ale jejich nastavení musí být provedeno ručně.

2.3 Dopravci

Tato kapitola popisuje datové napojení na jednotlivé dopravce a další prostředky vyžadované k vytvoření doručitelného balíku. Každý dopravce využívá jiné technologie a postupy, ale přesto lze mezi nimi nalézt shodné vlastnosti.

Typické požadavky dopravců pro vytvoření doručitelné zásilky:

- Adresní štítek
- Číslo balíku
- Trasovací informace
- Datový soubor
- Předávací protokol

2.3.1 Česká pošta

Česká pošta vyžaduje opatření zásilky adresním štítkem (obrázek 1) s čárovým kódem typu Code128, ve kterém je obsaženo identifikační číslo zásilky a předání údajů k jednotlivým zásilkám pomocí datového souboru typu „M“. Jedná se o textový soubor s pevnou délkou věty, což znamená, že každé pole souboru má vždy stejnou, předem určenou délku. Struktura tohoto souboru je detailně popsána v technické dokumentaci dostupné na webu České pošty. Výsledný soubor je pak potřeba nahrát do systému České pošty pomocí webového rozhraní `Podání online`. Bohužel tuto operaci nelze automatizovat, ale některá depa podporují zaslání dat emailem nebo je možné soubor předat fyzicky

na USB datovém médiu. Nevýhoda této metody ale spočívá v tom, že data nejsou nijak validována a hrozí větší počet vrácených zásilek. Naopak rozhraní Podání online upozorní uživatele na případné chyby a umožní data dodatečně opravit. [5]

2.3.1.1 Čárový kód a identifikační číslo zásilky

Jak již bylo zmíněno výše, je vyžadován čárový kód s technickým označením Code128. Jedná se o jednorozměrný kód, který umožňuje zakódovat jak číslice, tak i písmena. Celkově se číslo zásilky skládá z 13 znaků.

Složení čísla zásilky:

- Prefix – označení druhu zásilky (2 znaky)
- Číslo zásilky – složeno z čísla podavatele, podacího čísla a kontrolní číslice (9 nebo 10 znaků podle druhu zásilky)
- Postfix – označení typu podavatele nebo obecný postfix „CZ“ (1 nebo 2 znaky podle druhu zásilky)

2.3.1.2 Výpočet kontrolní číslice

Pro výpočet kontrolní číslice se využívá metoda modulo 11. Algoritmus pro výpočet je závislý na délce čísla zásilky.

Pro zásilky s číslem o délce 9 znaků je výpočet následující:

X	X	X	X	X	X	X	X	X
*	*	*	*	*	*	*	*	*
1	8	6	4	2	3	5	9	7
=	=	=	=	=	=	=	=	=
S1 + S2 + S3 + S4 + S5 + S6 + S7 + S8 + S9 = SUM								

Kontrolní číslice **K** se vypočte dle zbytku **A = SUM % 11** takto:

pokud zbytek **A > 1**, pak kontrolní číslice **K = 11 – A**

pokud zbytek **A = 0**, pak kontrolní číslice **K = 5**

pokud zbytek **A = 1**, pak kontrolní číslice **K = 0**

Pro zásilky s číslem o délce 8 znaků je výpočet obdobný, pouze se vynechá první součin **X * 1**. [5]



Obrázek 1: Adresní štítek České pošty

2.3.2 DPD

Společnost DPD vyžaduje stejně jako Česká pošta adresní štítek (obrázek 2) s čárovým kódem typu Code128, ve kterém je ale kromě čísla zásilky zakódováno i poštovní směrovací číslo, číselný kód země a číslo využití služby. Každý zákazník obdrží číselný rozsah pro označování zásilek a po jeho vyčerpání si musí zažádat o nový rozsah. Dále jsou na štítku uvedeny trasovací informace na základě cílového PSČ a země. Z tohoto důvodu je potřeba udržovat aktuální databázi směrovacích informací, které jsou dostupné na webu DPD.

Kromě adresního štítku je také nutné vytvořit datový soubor ve formátu MPSEXPDATA obsahující informace o zásilkách. Jedná se o takzvaný CSV (Comma-separated values) soubor, který se skládá z řádků, ve kterých jsou jednotlivá datová pole oddělena dělicím znakem (typicky středník). Podrobněji je struktura souboru MPSEXPDATA popsána v dokumentaci dostupné na webu DPD. Výsledný soubor se nahraje do systému DPD pomocí FTP (File Transfer Protocol) nebo protokolu XML-RPC (Remote procedure call). [6]

Všechna na poště nerespektovaná poškození musí být písemně nahlášena DPD během 2 dnů po doručení zásilky / Notification on damage which is not recognizable from the outside has to be submitted to DPD within 2 days in writing.

TEST s.r.o František Tester Testovací ulice 123		ADRESNÍ ŠTÍTEK		DPD DPD s.r.o. Dopravní podniky České republiky (DP) s.r.o. Máje 100 CZ-16001 Mladá Tel. +420 223 173 373	
Tel. 777123456 CZ-62100 Brno		Dobře 1 / 1 Váha 2,50 kg		Směrovací Zásilka	
Číslo skupinky DOP01043 Číslo dokladu: 15010342		Doběročná částka: CZK 1234,00 Způsob platby: HOTOVOST VS: 12345678		DPD Logo	

C.O.D.

0000 Track R D-COD Service

CZ-1385

109-CZ-62100 530A

21 04 15 14 44 20 10 80 00 POHODA, BHE CZ V1.1



0062 100 0000 1000 1210 92 03 6

Obrázek 2: Adresní štítek DPD

2.3.3 Geis Parcel

Adresní štítek (obrázek 3) pro zásilky Geis Parcel musí obsahovat čárový kód s číslem zásilky včetně kontrolní číslice, který je ve formátu typu 2 z 5 a také další čárový kód typu Code39, ve kterém se nachází trasovací informace nutné pro správné doručení. Tyto trasovací informace jsou průběžně aktualizovány, a je tedy potřeba udržovat vždy aktuální databázi. Stejně jako u DPD je zákazníkovi přidělen číselný rozsah, z kterého postupně čerpá a po jeho vyčerpání musí zažádat o nový. Rozdíl ale

spočívá v tom, že zákazník obdrží zvlášť číselnou řadu pro vnitrostátní a mezinárodní přepravu, které se dále dělí na dobírkovou a nedobírkovou řadu. Celkem tedy může zákazník obdržet čtyři intervaly.

Pro předání dat do systému Geis Parcel je využit datový formát EDI (Electronic Data Interchange). Stejně jako u České pošty se jedná o formát s pevnou délkou věty. Podrobný popis je k dispozici v manuálu pro systémovou integraci. Výsledný soubor se odešle pomocí FTP. [7]

2.3.3.1 Výpočet kontrolní číslice

Na adresním štítku je potřeba na rozdíl od datového souboru uvádět i kontrolní číslici, která se vypočítává na základě čísla zásilky.

Algoritmus bude vysvětlen na následujícím příkladu [7]:

Mějme číslo zásilky **22126690500**.

1. Sečtou se zvlášť číslice na lichých a na sudých pozicích.
 $2+1+6+9+5+0=23$
 $2+2+6+0+0=10$
2. Součet čísel na lichých pozicích se vynásobí třemi a přičte se součet čísel na sudých pozicích a jednička.
 $(23 * 3) + 10 + 1 = 80$
3. Kontrolní číslice je doplněk mezivýsledku z předchozího kroku do následující celé desítky.
80 a do následující desítky (80) = 0



Obrázek 3: Adresní štítek Geis Parcel

2.3.4 IN TIME

Vytváření adresního štítku (obrázek 4) i veškerá komunikace se systémem společnosti IN TIME probíhá pomocí REST API (Representational state transfer). Je to architektura rozhraní navržena pro distribuované prostředí, to znamená, že části aplikace běží na různých strojích a pro vzájemnou komunikaci využívají síť. REST pro svou datovou výměnu používá standardizovaný formát JSON (JavaScript Object Notation) a komunikační protokol HTTP (Hypertext Transfer Protocol) a jeho metody GET a POST. Podrobný popis rozhraní je dostupný v dokumentaci pro vývojáře. Díky tomu,

že se o vytváření adresního štítku stará systém IN TIME, tak není ani potřeba udržovat žádnou číselnou řadu nebo nějaké další trasovací informace. Nevýhoda ale spočívá v tom, že komunikace musí probíhat online a je tedy pro vytvoření zásilky potřeba připojení k internetu. S tím se pojí další nevýhoda a to rychlost této komunikace, která je závislá na rychlosti připojení klienta i současnému stavu vytížení systému společnosti IN TIME. Pro vytvoření zásilky je nutné zaslat data pomocí metody `/package` a následné získání štítků využitím metody `/package/ID/labels.zpl`, kde ID je číslo zásilky získané předchozí metodou. Získané štítky jsou ve formátu ZPL (Zebra Programming Language), což je jazyk pro popis tištěné stránky vytvořený společností Zebra Technologies. [8]



Obrázek 4: Adresní štítek IN TIME

2.3.5 PPL

Adresní štítek (obrázek 5) pro zásilky PPL musí obsahovat čárový kód typu 2 z 5, v kterém je uloženo číslo zásilky. Existuje několik typů číselných řad, které rozlišují různé druhy služeb. Jsou to například řady pro balíky pro fyzické osoby (B2C), pro firemní balíky (B2B) a dopolední balík. Každá z těchto řad se dále dělí na dobírkovou a nedobírkovou řadu. Dobírková řada je rozlišena číslicí 8 nebo 9 na čtvrté pozici číselné řady. Stejně jako u DPD a Geis Parcel řadu přiděluje obchodní zástupce a po její vyčerpání si musí zákazník zažádat o novou. Pro některé služby je nutné udržovat aktuální databázi PSČ, pro které jsou tyto služby dostupné.

Komunikace se serverem PPL probíhá pomocí webové služby, kterou nazývá IEGate. Webová služba je softwarový systém umožňující interakci dvou zařízení pomocí sítě. Definice služby je popsána ve formátu WSDL (Web Services Description Language). WSDL je jazyk pro popis funkcí nabízených webovou službou a také pro popis vstupních a výstupních parametrů těchto funkcí. Komunikace s klientskou aplikací probíhá pomocí protokolu SOAP (Simple Object Access Protocol). Jako aplikační

vrstva pro SOAP se využívá především protokol HTTP. SOAP i WSLD jsou založeny na technologii XML (Extensible Markup Language). [9]

PPL CZ s.r.o. - Odštěpný
závod jižní Morava
Drčkova 3
628 00 Brno - Líšeň



8238600011

CONSIGNEE/PRIJEMCE:	
TEST s.r.o František Tester Testovací ulice 123 62100 Brno Tel: 777123456 Pozn:	
1 / 1	
Hmotnost:	2.5 Kg
	27.4.2016
DOBÍRKA VÝBĚRČÍ	
CONSIGNOR/ODESILATEL:	
Vysoké učení technické v Brně Božetěchova 1 612 66 Brno	

DOBÍRKA
--1234.00-- CZK

Při převzetí zásilky zkontrolujte obal zásilky. Při jeho poškození proveďte zápis do přepravního listu dopravce. Veškeré ostatní reklamace uplatní příjemce zásilky bez zbytečného odkladu, nejpozději do 3 dnů od převzetí zásilky.

Vytisknuto ekonomickým systémem POHODA, zajišťuje www.bhit.cz

Obrázek 5: Adresní štítek PPL

2.3.6 Uloženska

Společnost Uloženska využívá pro vytváření zásilek a tisk adresních štítků (obrázek 6) REST API poskytované službou apiary.io. Apiary nabízí možnost velmi rychle a jednoduše vytvořit vlastní aplikační rozhraní. Výměna dat se službou Apiary je založena na HTTP komunikaci pomocí metod GET a POST. Data jsou zasílána ve formátu JSON. Podrobný popis využitelných metod je dostupný v dokumentaci Uloženska.cz API v3. Stejně jako u dopravce IN TIME, není potřeba udržovat žádné informace o číselné řadě, ale na rozdíl od všech dopravců je potřeba udržovat aktuální databázi výdejních poboček a přepravních služeb. U každé zásilky musí být vybrána transportní služba. K dispozici je například možnost zasílat zásilky pomocí České pošty, DPD, Slovenské pošty, Balíkomatů a dalších. Pro některé z transportních služeb je nutno vybrat výdejní pobočku, například pro partnerské pobočky Uložensky a poštovní novinové stánky, kde si zákazník svoji zásilku vyzvedne. A dále pro všechny transportní služby je potřeba zvolit podací pobočku. Pro vytvoření zásilky je nutné odeslat data pomocí metody `consignments`, která vrací zároveň i adresní štítek ve formátu ZPL nebo base64 kódovaný PDF soubor podle vyžádaného formátu. [10]

Uloženka

Vyzvednutí: Brno, Heršpická 800/6

BRNO3



3971655

ID zásilky: 3971655

Č. objednávky: 15010342

Počet balíků: 1

Podatelna: brno

ID partnera: ---

Dobírka: 1 234,00 CZK

Vrácení: brno3

Odesílatel:

Vysoké učení technické v Brně

Fakulta Informačních technologií

Telefon: +420123456789

Příjemce:

Telefon: +420777123456

E-mail: test@test.cz

TEST s.r.o

František Tester

Pozn.:

Provozovatelem služby Uloženka je společnost: Uloženka s.r.o., IČ: 24299162

Tisk: 27.04.2016, 21:48:01

Obrázek 6: Adresní štítek Uloženky

2.4 Shrnutí

V této kapitole byly popsány nejvýznamnější ekonomické systémy v České republice. Každý z nich má dostupnou jistou formu napojení na dopravce, ať už jako součást systému nebo jako volitelné rozšíření pomocí aplikací vytvořených externími společnostmi, ale ani v jednom z případů se nejedná o jednotné rozhraní pro všechny dopravce, které zajišťuje jak tisk adresního štítku, tak i odeslání dat na servery dopravců. Ve většině případů je nutná manuální interakce se systémy dopravců, což pouze dělá složitější práci obsluhy expedice zásilek.

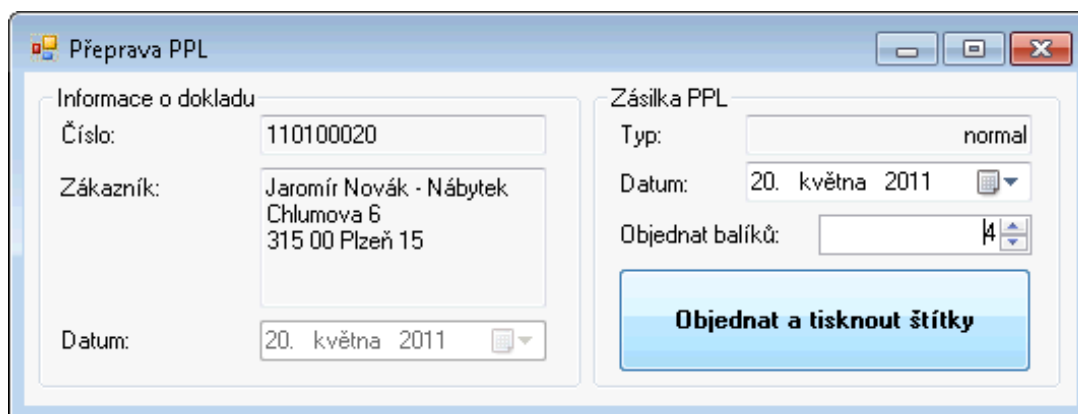
Dále byly popsány základní požadavky vybraných dopravců na vytvoření doručitelné zásilky a popis způsobu komunikace s jejich servery. Každý dopravce používá jiné technologie a postupy, ale ve všech případech je nutné vytisknout adresní štítek a odeslat data na server dopravce.

3 Návrh řešení

Jak již bylo zmíněno v kapitole Ekonomické systémy a logistika, POHODA umožňuje snadné rozšíření o další uživatelské agendy, které jsou plnohodnotnou součástí celého systému. Díky tomu bude vytvořena nová agenda Doprava, která bude přístupná uživatelům přímo v POHODĚ a bude sloužit jako evidence již vytvořených a odeslaných zásilek. Aby bylo možné objednávat zásilky na základě dokladů z agend POHODY, bude nutné vytvořit uživatelské rozhraní Doprava, které se bude spouštět nad jednotlivými doklady jako externí nástroj dostupný v konkrétní agendě. Bude se tedy jednat o aplikaci postavenou mimo POHODU a bude zajišťovat založení nového záznamu v agendě Doprava a tisk adresních štítků. Export dat na servery dopravců bude probíhat také externím nástrojem, do kterého budou vstupovat zásilky vyfiltrované v agendě Doprava.

3.1 Původní řešení

Původní řešení, zobrazené na obrázku 7, bylo navrženo pouze pro jednoho dopravce a to pro společnost PPL. K dispozici bylo pouze velmi jednoduché uživatelské rozhraní, které neumožňovalo úpravu vstupních informací ani výběr použitých služeb a podobně. Bylo možné pouze zadat počet objednaných zásilek, datum svozu a následně vytisknout štítky pro zásilky.



Obrázek 7: Původní uživatelské rozhraní PPL

Tisk adresních štítků probíhal na základě tiskových sestav vytvořených v ekonomickém systému POHODA pomocí programu REPORT Designer, což s sebou přinášelo spoustu nevýhod. Například problém s přístupovými právy pro jednotlivé uživatele systému POHODA a nutnost při datové uzávěře vždy znovu nakonfigurovat tyto sestavy. Další problém spočíval přímo v tisku těchto sestav. Před tiskem bylo nutné v databázi změnit výchozí tiskovou sestavu pro konkrétní agendu a uživatele. Následný tisk pak probíhal simulací stisku klávesové zkratky pro tisk výchozí sestavy, která byla nutná nastavit pro každého uživatele zvlášť. Toto řešení navíc nebylo vždy funkční, protože simulace stisku klávesové zkratky byla vždy provedena nad aktuálním aktivním oknem, takže se stačilo v průběhu procesu vytváření zásilky přepnout do jiné aplikace a tisk přestal fungovat. A jako poslední problém bylo samotné nasazení těchto štítků. Pro všechny štítky, které obsahovaly nějaké obrázky, bylo potřeba jednotlivě nastavit síťovou cestu k umístění každého obrázku, jinak se štítek vytiskl bez obrázků.

Postupem času, když se stával program populárním, tak bylo potřeba rozšířit toto rozhraní pro další dopravce, protože většina elektronických obchodů nabízí různé druhy dopravců, aby si mohl zákazník vybrat dle svých osobních preferencí. Byla tedy umožněna volba dopravce přímo

v uživatelském rozhraní a dále byla přidána možnost nastavit přídatné služby, ale vždy bylo potřeba tyto služby nastavit přímo v rozhraní Doprava. Nebylo možné přebírat tyto informace rovnou z dokladu agendy. Ovládací prvky služeb byly rozmístěny přes sebe na stejném formuláři a při změně dopravce se pouze zneviditelnily nepoužitelné služby, takže se stávaly značně nepřehlednými zvláště při úpravách a dalším přidání nové služby. Těmito úpravami se stával programový kód velice nepřehledný a těžko udržovatelný. Přidání nového dopravce vždy znamenalo ještě další větvení programu pomocí podmínek. Každý dopravce navíc vyžadoval rozdílné techniky k vytvoření zásilky, ale přitom podstata tohoto procesu byla vždy stejná. Docházelo tak ke zbytečnému kopírování stejných částí kódu. Samotné vytvoření záznamu do uživatelské agendy probíhalo pomocí uložených databázových procedur a funkcí, které při každé údržbě databáze byly smazány samotným ekonomickým systémem POHODA, takže se velmi často stávalo, že řešení přestalo zákazníkům fungovat a museli kontaktovat technickou podporu, aby provedla ruční nahrání všech procedur, funkcí a triggerů.

Dalším důvodem pro vytvoření nové verze Dopravy bylo napojení na jiné systémy a umožnění složitějších funkcí jako například hromadné vytváření zásilek na více dokladů zároveň. Toto bylo v původní verzi nemožné, protože funkčnost byla rozmístěna přes několik aplikací a spoustu databázových procedur a funkcí.

3.2 Požadavky a návrh řešení

Hlavním požadavkem pro vytvoření nové verze byla neustále se zvyšující složitost a nepřehlednost programového kódu způsobená nevhodným návrhem. Při přidávání nového dopravce vždy docházelo ke kopírování téměř stejného kódu a větvení běhu programu pomocí podmínek. Cílem nového řešení je tedy toto co nejvíce eliminovat, aby přidání dopravce, probíhalo především zděděním obecné třídy pro dopravce a případné doplnění specifických vlastností. Každý dopravce by měl být zvlášť ve své knihovně, tak aby instalace nového dopravce znamenala pouze nakopírování knihovny dopravce do kořenového adresáře a konfiguraci údajů specifických pro konkrétního uživatele pomocí rozhraní Doprava.

Tisk adresních štítků by již nadále neměl probíhat na základě tiskových sestav z POHODY, ale pomocí sestav vytvořených knihovnou DevExpress Reporting, aby byla zajištěna bezproblémová funkčnost a byla eliminována zdlouhavá instalace a konfigurace sestav pro každého uživatele POHODY, jak již bylo zmíněno v kapitole Původní řešení.

Dalším požadavkem je vytvořit nový systém konfigurace jednotlivých dopravců pomocí společného rozhraní dostupného z aplikace Doprava. Každý dopravce vyžaduje spoustu různých proměnných závislých na konkrétním uživateli a zároveň existují někteří uživatelé, kteří potřebují mít možnost pro stejného dopravce nastavit více různých variant. Například když zákazník vlastní více různých elektronických obchodů, ale všechny data vstupují pouze do jedné účetní jednotky a potřebuje mít na každém štítku jinou provozovnu. Kromě této statické konfigurace je dále potřeba přebírat některá nastavení z dokladů, především služby dopravců, které závisí na volbě zákazníka, který si zásilku objednal, ale zároveň je nutné mít možnost nastavit výchozí hodnoty v případě, že není možné přebírat tyto informace z dokladu nebo je žádoucí zapnout konkrétní službu pro všechny uživatele.

A v neposlední řadě alespoň částečně zpřístupnit vytváření zásilek i pro jiné aplikace jako například program Expedice, který zajišťuje optimalizaci firemního procesu od přijetí objednávky od zákazníka až po samotnou expedici objednávky zákazníkovi. V tomto případě postačí umožnit vytvořit zásilku bez uživatelského rozhraní spuštěním programu s parametrem určujícím, že se nemá spouštět grafické rozhraní.

3.2.1 Číselné řady

Většina dopravců vyžaduje, aby si zákazník ve své vlastní režii spravoval číselné řady, které jsou buď přidělovány jako interval nebo jsou složeny z čísla zákazníka a počítadla jednotlivých zásilek. Pro správu číselných řad bude využita agenda Číselné řady s využitím volitelných parametrů, která je dostupná v systému POHODA v nabídce Nastavení/Seznamy/Číselné řady a databázové tabulce sCRady. Aby bylo možné spravovat řady jednoduše, bude vždy řada rozdělena na prefix, počítadlo zásilek a postfix. Prefix bude uložen ve volitelném parametru a bude obsahovat část, která je v rámci řady neměnná a to i v případě, že dopravce přiděluje pouze číselný interval, protože se typicky přiděluje pouze několik tisíců zásilek a vždy se tedy najde část, která je fixní. Jako počítadlo zásilek se využije standardní pole Číslo v agendě Číselných řad. Toto pole může obsahovat maximálně 5 znaků, takže pro některé případy bude nutné hlídat přetečení řady oproti prefixu, takže bude nutné občas inkrementovat i část prefixu, pokud se bude jednat o číselnou hodnotu. Postfix se bude zatím využívat pouze pro Českou poštu, která vyžaduje uvádět typ podatele v rámci řady, ale v budoucnu může přibýt další dopravce, který bude postfix také využívat. Nedílnou součástí každé řady bude horní limit, který nesmí být překročen, protože pak může dojít k tomu, že se uživatel dostane do intervalu jiného zákazníka.

3.2.2 Adresní štítky

Existují dvě možnosti vytvoření adresního štítku. První z nich je možnost získat štítek online přímo ze systému dopravce ve formátu ZPL nebo PDF, takže není potřeba vytvářet žádné tiskové sestavy a ani udržovat číselnou řadu. Druhý způsob je tisk štítků pomocí vlastních sestav vytvořených pomocí knihovny DevExpress Reporting. Štítky stažené od dopravce budou uloženy v záložce dokumenty v uživatelské agendě Doprava, aby bylo možné případně štítky znovu vytisknout. Pro vlastní štítky bude vytvořeno rozhraní na přetisk již vytvořených zásilek.

3.2.3 Odeslání dat

Pokud se adresní štítky získávají přímo od dopravce, tak se data odesílají rovnou po založení záznamu v agendě Doprava a zpětně se propisují údaje přijaté od dopravce, typicky číslo zásilky či dávky. V druhém případě se data odesílají na konci dne naráz, když už jsou vytvořeny všechny zásilky za daný den. Každý dopravce využívá jiný formát i způsob předání dat, ale uživateli by mělo být umožněno pouze odsouhlasit počty vytvořených zásilek a jedním kliknutím myši odeslat data pro všechny dopravce zároveň pomocí exportní aplikace. V obou případech jsou exportovaná data čerpána pouze z agendy Doprava, protože na dokladech mohly mezitím vzniknout nevyžádané změny.

3.2.4 Uživatelská agenda Doprava

Uživatelská agenda Doprava bude obsahovat všechny potřebné informace o vytvořených balíkových soupiskách. Jeden záznam se bude vždy skládat z hlavičky a položek. Hlavička bude obsahovat vazbu do agendy Adresář a agendy zdrojového dokladu. Uživatel bude moci pomocí tlačítka na formuláři přejít do těchto agend v případě potřeby. Dále budou na hlavičce uvedeny vybrané služby, hodnota dobírky, variabilní symbol a další údaje společné pro celou soupisku. Položky dokladu budou obsahovat informace o konkrétních zásilkách soupisky, jako jsou číslo zásilky, rozměry, váha, poznámka a jiné. Jedna soupiska může obsahovat jednu a více zásilek, které se doručují zároveň. Z agendy Doprava bude také možné pomocí tlačítka na formuláři vyvolat webovou stránku na sledování zásilek, aby byla usnadněna práce obsluhy v případě zákazníkovi poptávky po stavu zásilky.

Z technických důvodů bude vytvořena agenda Trasy, která bude obsahovat trasovací informace nutné pro správné doručení zásilek. Data se budou typicky aktualizovat každý den ráno před prvním spuštěním aplikace Doprava.

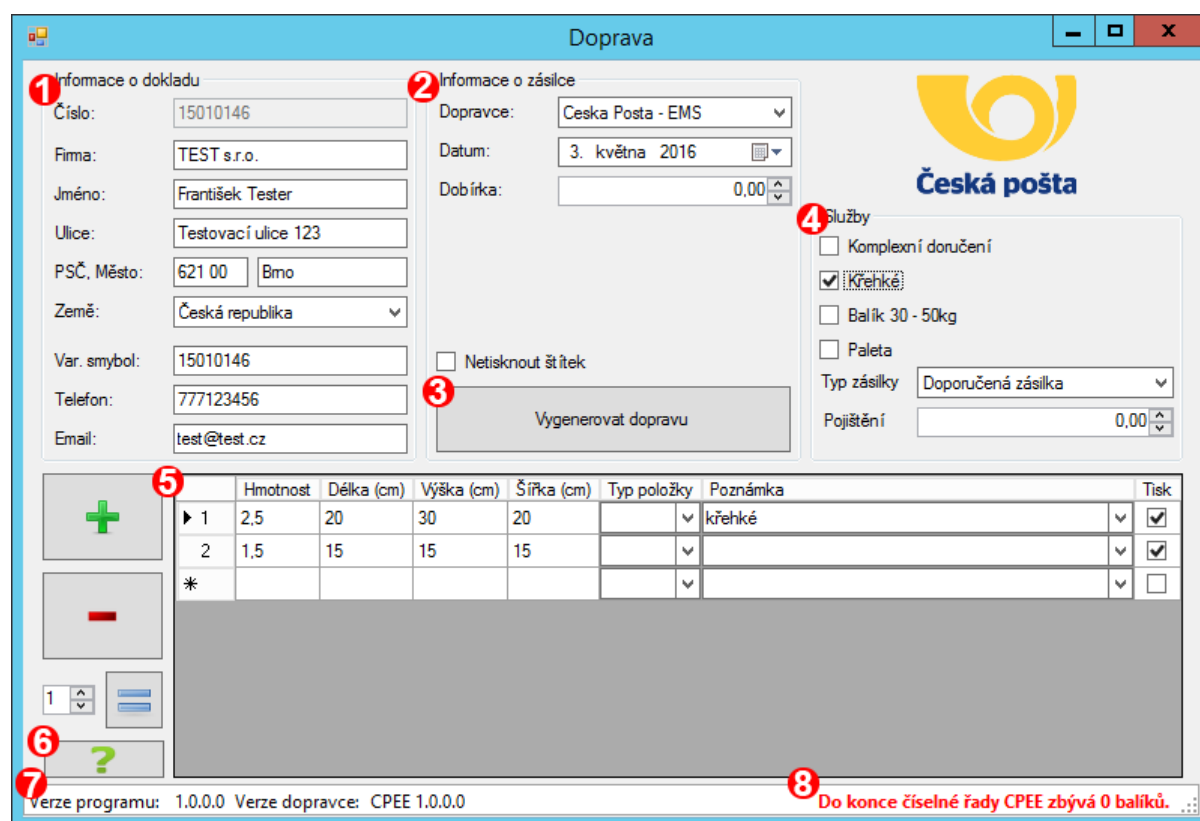
3.2.5 Aplikace Doprava

Při návrhu nové aplikace Doprava, byla snaha dodržet původní rozložení ovládacích prvků a vylepšit uživatelské rozhraní podle nových požadavků.

Rozhraní bylo rozšířeno o možnost úpravy vstupních informací z dokladu. Uživatel bude mít možnost v případě potřeby upravit adresu a kontaktní údaje zákazníka.

Pro každého dopravce bylo vytvořeno vlastní rozhraní pro správu využitých služeb, které se skládá z loga firmy a oblasti pro ovládací prvky služeb. Většina služeb se zapíná pouze pomocí volby ANO/NE, ale existují i takové, které vyžadují vybrat konkrétní hodnotu, například výdejní pobočku. Jelikož každý dopravce podporuje jiné služby a ukládání těchto informací musí probíhat jednotně, tak budou tyto informace uloženy v uživatelské agendě v poli Příznaky dopravce v textové podobě. Jednotlivé služby budou odděleny středníkem. Pro služby typu ANO/NE bude v tomto poli uložen jednoznačný identifikátor služby. V případě služeb, které vyžadují hodnotu, bude syntaxe velmi podobná, za identifikátorem služby bude pouze následovat dvojtečka a zvolená hodnota.

Dále přibyla rozšířená úprava informací o zásilce pomocí tabulky. Každý řádek této tabulky obsahuje informace o jedné zásilce. Uživatel může nově nastavit hmotnosti a rozměry zásilek, přidat ke každé zásilce poznámku a také zvolit, zda se má pro tuto zásilku tisknout adresní štítek. Pro některé dopravce je také možné vybrat typ zásilky.



Obrázek 8: Uživatelské rozhraní Doprava

Popis rozhraní:

1. Úprava kontaktních údajů
2. Výběr dopravce, data svozu a dobírky
3. Spuštění procesu vytvoření zásilky
4. Výběr služeb
5. Úprava informací o zásilkách a tlačítka umožňující rychlé přidání nových zásilek
6. Nabídka pro konfiguraci aplikace
7. Aktuální verze aplikace a knihovny dopravce
8. Varovná hláška o docházející číselné řadě (posledních 100 zásilek)

3.3 Použité technologie

V této kapitole budou popsány základní technologie, které byly využity pro realizaci uživatelské agendy Doprava. Jelikož ekonomický systém Stormware POHODA vyžaduje pro svůj běh operační systém Microsoft Windows, tak i všechny využití technologie musí být na této platformě.

3.3.1 Microsoft SQL Server

Microsoft SQL Server je vysoce výkonný relační databázový systém vytvořený společností Microsoft. Jeho hlavní funkce je ukládání a získávání dat vyžadovaných jinými aplikacemi, které mohou běžet na stejném stroji nebo na jiném zařízení v rámci sítě včetně internetu. Komunikace klientských aplikací s databázovým serverem probíhá pomocí dotazů v jazyce Transact-SQL. Microsoft nabízí mnoho různých edicí SQL Serveru zaměřených na rozdílné uživatele. Od malých jednoduchých aplikací až po rozsáhlé internetové aplikace s mnoha souběžnými uživateli. V rámci ekonomického systému POHODA si uživatelé s nízkým datovým zatížením vystačí s bezplatnou variantou serveru Microsoft SQL Server Express, která je limitována na jeden procesor, 1 GB operační paměti a 10 GB databázových souborů. Pro zabezpečení plynulého provozu při větším datovém zatížení je doporučeno využívat variantu Microsoft SQL Server Standard. [11]

3.3.1.1 Transact-SQL

Transact-SQL je rozšíření SQL (Structured Query Language), což je standardizovaný deklarativní jazyk, který byl původně vytvořen firmou IBM pro dotazování, úpravu a definování relačních databází. T-SQL rozšiřuje standardní SQL o procedurální programování, lokální proměnné a spoustu podpůrných funkcí pro zpracování řetězců, zpracování dat, matematické funkce a jiné. Tyto změny dělají T-SQL turingovsky kompletní. [11]

Rozšiřující vlastnosti:

- Proměnné – příkazy pro definici a nastavování hodnot lokálních proměnných (DECLARE, SET, SELECT)
- Řízení toku – klíčová slova pro řízení toku aplikace jako jsou podmínky a cykly (BEGIN, END, BREAK, CONTINUE, GOTO, RETURN, WAITFOR, WHILE, IF, ELSE)
- Změny dotazů pro mazání a aktualizaci dat – pro obě možnosti přibyla možnost přidat do dotazu FROM klauzuli, která umožňuje využití složených dotazů

- Hromadné vkládání – nový příkaz `BULK INSERT` pro hromadné vkládání dat
- Výjimky – možnost zachycení výjimek pomocí příkazů `TRY`, `CATCH`

3.3.2 .NET Framework

.NET Framework je rozsáhlá softwarová platforma od firmy Microsoft, která běží převážně na operačním systému Microsoft Windows. Tato platforma je určena pro vývoj mnoha různých druhů aplikací. Umožňuje vyvíjet nejen klasické aplikace pro Windows, ale i webové aplikace a služby, aplikace pro mobilní zařízení a další. Framework se skládá z běhového prostředí CLR (Common Language Runtime) a rozsáhlou knihovnou tříd FCL (Framework Class Library). CLR je virtuální stroj, který se stará o vykonávání kódu pomocí just-in-time kompilace, která překládá kompilovaný kód do strojového jazyku za běhu programu. Zároveň poskytuje služby jako je například typová bezpečnost, správa paměti, zachycování výjimek a správu vláken. Všechny programy bez rozdílu programovacího jazyka jsou vykonávány pomocí CLR. FCL poskytuje velké množství tříd pro tvorbu uživatelského rozhraní, přístupu k datům, konektivitu k databázi, matematické algoritmy, síťovou komunikaci a mnoho dalších. [12]

3.3.2.1 Jazyk C#

C# je objektově orientovaný programovací jazyk vyvinutý firmou Microsoft zároveň s platformou .NET Framework. C# byl vytvořen na základě jazyků C++ a Java. Cílem bylo vytvořit moderní, jednoduchý, víceúčelový jazyk. Důležitá je hlavně robustnost, spolehlivost a snadný vývoj aplikací. Podporuje principy softwarového inženýrství jako je silné typování, kontrola hranice polí, detekce použití neinicializovaných proměnných, parametrizované typy, reflexi a podobně. [12]

3.3.2.2 Windows Forms

Windows Forms je knihovna tříd pro snadnou tvorbu grafických uživatelských rozhraní, která je součástí .NET Frameworku. Aplikace vytvořené pomocí WinForms jsou založené na událostech. Na rozdíl od konzolových programů tráví většinu času čekáním na interakci od uživatele jako například kliknutí na tlačítko, vyplnění formuláře a podobně. Všechny vizuální prvky WinForms jsou zděděny od třídy Control, která poskytuje minimální funkcionalitu uživatelského rozhraní jako je například pozice, velikost, barva, písmo ale i nejčastěji používané události jako je kliknutí, pohyb myši, stisk klávesy a podobně. [12]

3.3.3 Knihovna DevExpress Reporting

DevExpress je vývojářská společnost zabývající se především vývojem komponent pro grafické uživatelské rozhraní. V rámci této práce je využita knihovna Reporting, která slouží pro snadné vytváření tiskových sestav a jejich následný tisk. Tato knihovna umožňuje vytváření široké škály tiskových sestav všech možných druhů od tabulkových sestav přes šítkové sestavy až po master-detail sestavy. K dispozici je řada prvků, které se dají na sestavy vložit. Jsou to například čárové kódy, grafy, měřítko, kontingenční tabulky, geometrické tvary, obrázky a podobně. Sestavy je možné tvořit na základě databázového dotazu či vstupních parametrů. Nad daty je možné provádět různé operace pomocí vestavěných funkcí. Pomocí této knihovny jsou vytvořeny adresní štítky dopravců. Tvorba sestav je plně integrována do vývojového prostředí Microsoft Visual Studio. [13]

4 Implementace a testování

4.1 Rozšíření Stormware POHODA E1

4.1.1 Struktura databáze

Aby bylo možné vytvářet balíky na základě dat přímo z POHODY, musíme znát alespoň základní tabulky, odkud se data budou čerpat. V tabulce 1 je výpis nejdůležitějších tabulek potřebných pro získání všech vstupních informací pro vytvoření balíku. Tabulky obsahující informace o dokladech se skládají vždy ze dvou částí, například pro faktury jsou to tabulky FA a FA_{pol}. Tabulka FA obsahuje informace z hlavičky dokladu jako je dodací a fakturační adresa, kontaktní informace, celková částka, měna, variabilní symbol a podobně. Ve FA_{pol} jsou pak informace o položkách dokladu. V ostatních dokladových agendách (objednávky, výdejky, příjemky a podobně) je to téměř stejné, z toho důvodu budeme uvažovat v rámci práce pouze faktury a také protože obsahují nejvíce informací.

Název tabulky	Údaje, které tabulka obsahuje
AD	Adresář
FA	Faktury (přijaté, vydané, zálohové faktury...)
FA _{pol}	Položky faktur
sCIN	Činnosti
sCRady	Číselné řady
sExtTools	Externí nástroje
sFormUh	Seznam forem úhrady
SKz	Zásoby
sSTR	Seznam středisek
sVP	Volitelné parametry
sVP _{pol}	Položky volitelných parametrů
sVPUL	Uživatelské seznamy
sVPUL _{pol}	Položky uživatelských seznamů
sZAK	Seznam zakázek
sZeme	Seznam zemí

Tabulka 1: Zdrojové tabulky

V tabulkách 2 a 3 jsou zobrazeny nejdůležitější sloupce z faktur respektive položek faktur. Pole s názvem ID obsahuje vždy kladná celá čísla začínající od jedničky, které jsou jednoznačný identifikátor každého řádku. Datová pole začínající prefixem Re_l nebo Re_f jsou odkazy do tabulek, ve kterých lze teprve zjistit konkrétní hodnoty. Například pole Re_fAD odkazuje do agendy Adresář, ve které se uchovávají informace o zákazníkovi, na kterého byl doklad vystaven. Ostatní pole nesou přímou hodnotu a jejich název je vždy volen podle hodnoty, kterou obsahují.

Název sloupce	Význam hodnoty sloupce
ID	Jednoznačný identifikátor
RefCin	Odkaz do seznamu činností
RefStr	Odkaz do seznamu středisek
CisloZAK	Číslo zakázky
RefZeme	Odkaz do seznamu zemí
RelForUh	Odkaz do seznamu forem úhrady
Cislo	Číslo faktury
PDoklad	Číslo objednávky
VarSym	Variabilní symbol
Datum	Datum vystavení
KcCelkem	Celková částka v Kč
RefCM	Odkaz do seznamu měn
CmCelkem	Celková částka v cizí měně
RefAD	Odkaz do adresáře
Firma	Firma
Jmeno	Jméno
Ulice	Ulice
PSC	PSC
Obec	Obec
Email	Email
Tel	Telefon
Pozn	Poznámka

Tabulka 2: Vybrané sloupce tabulky FA

Tabulka položek dokladu je důležitá především kvůli své vazbě na zásoby, zvláště pro výpočet hmotnosti balíku, která bude vypočítána na základě sloupce *Hmotnost* z tabulky *SKz* vynásobeným počtem kusů a koeficientem měrné jednotky.

Název sloupce	Význam hodnoty sloupce
ID	Jednoznačný identifikátor
RefAg	Odkaz na hlavičku dokladu
RefSKz	Odkaz do tabulky zásob
Kod	Kód zásoby
Mnozstvi	Množství
MJ	Měrná jednotka
MJKoef	Koeficient měrné jednotky

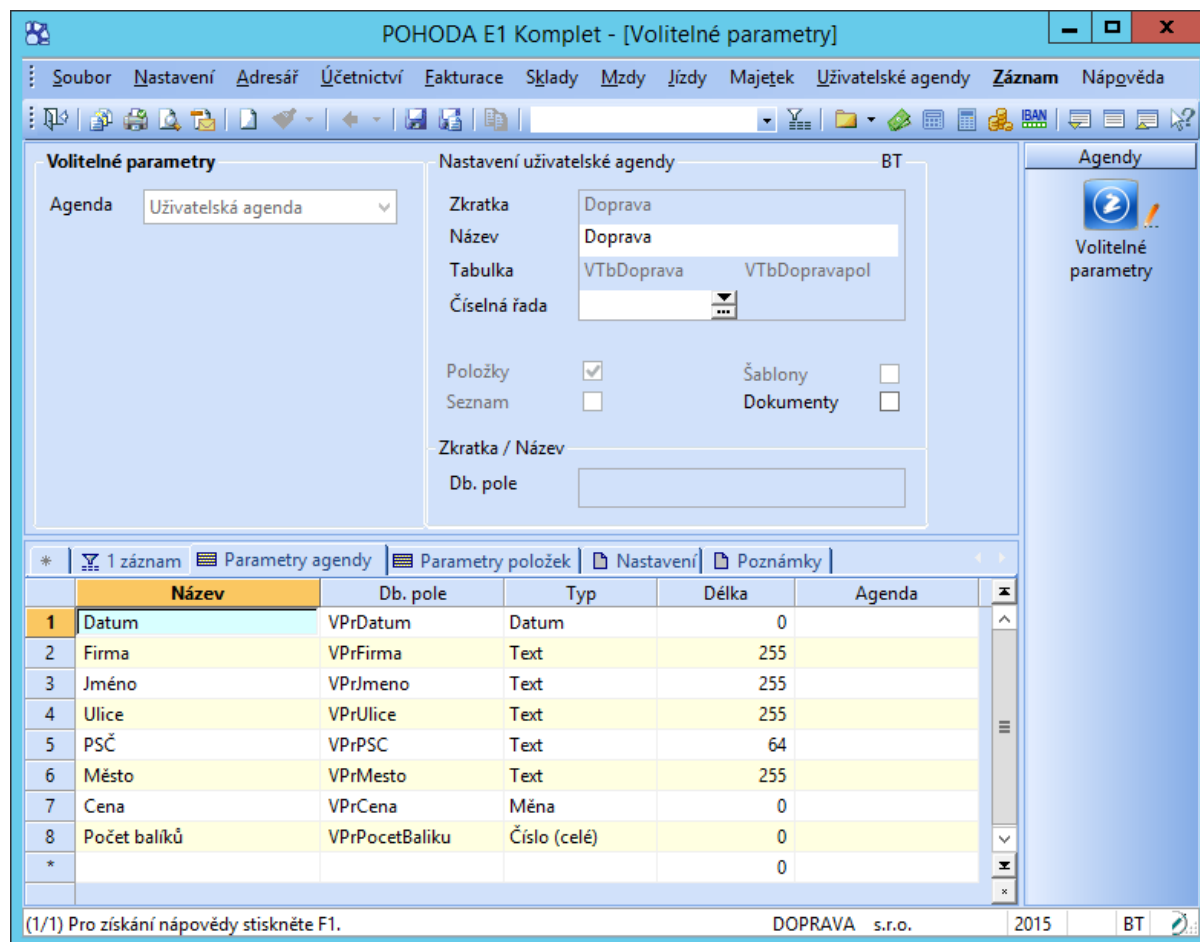
Tabulka 3: Vybrané sloupce tabulky FApol

4.1.2 Volitelné parametry a uživatelské seznamy

Všechny základní i uživatelské agendy lze rozšířit o další volitelné parametry. Celkově je možné přidat 64 parametrů do základních agend a 128 parametrů do uživatelských agend, tento počet se ovšem může v budoucnu změnit. Parametry lze přidávat do hlavičky dokladu i položek. Nově přidávané parametry začínají vždy prefixem `VPr` nebo `RefVPr` podle toho, jestli se jedná o hodnotový nebo seznamový typ. K dispozici je 7 hodnotových typů (text, dlouhý text, měna, ano/ne, číslo, celé číslo a datum) a dva druhy seznamového parametru, které se od sebe liší pouze hodnotou, která je ve sloupci uložena. První z nich je konstantový seznam, pro který platí, že hodnota uložená v tomto sloupci je vždy přímo hodnota konstanty, ale v případě druhého typu seznamu se jedná o odkaz do tabulky `sVPULPOL`, kde jsou uloženy konkrétní hodnoty. Volitelné parametry je možné aktivovat z nabídky Nastavení/Globální nastavení, kde je nutné vybrat možnost Ostatní a následně zatrhnout volbu Povolit použití volitelných parametrů. Po uložení tohoto nastavení je možné začít využívat agendu Volitelné parametry a Uživatelské seznamy. Nastavení volitelných parametrů může provádět pouze administrátor.

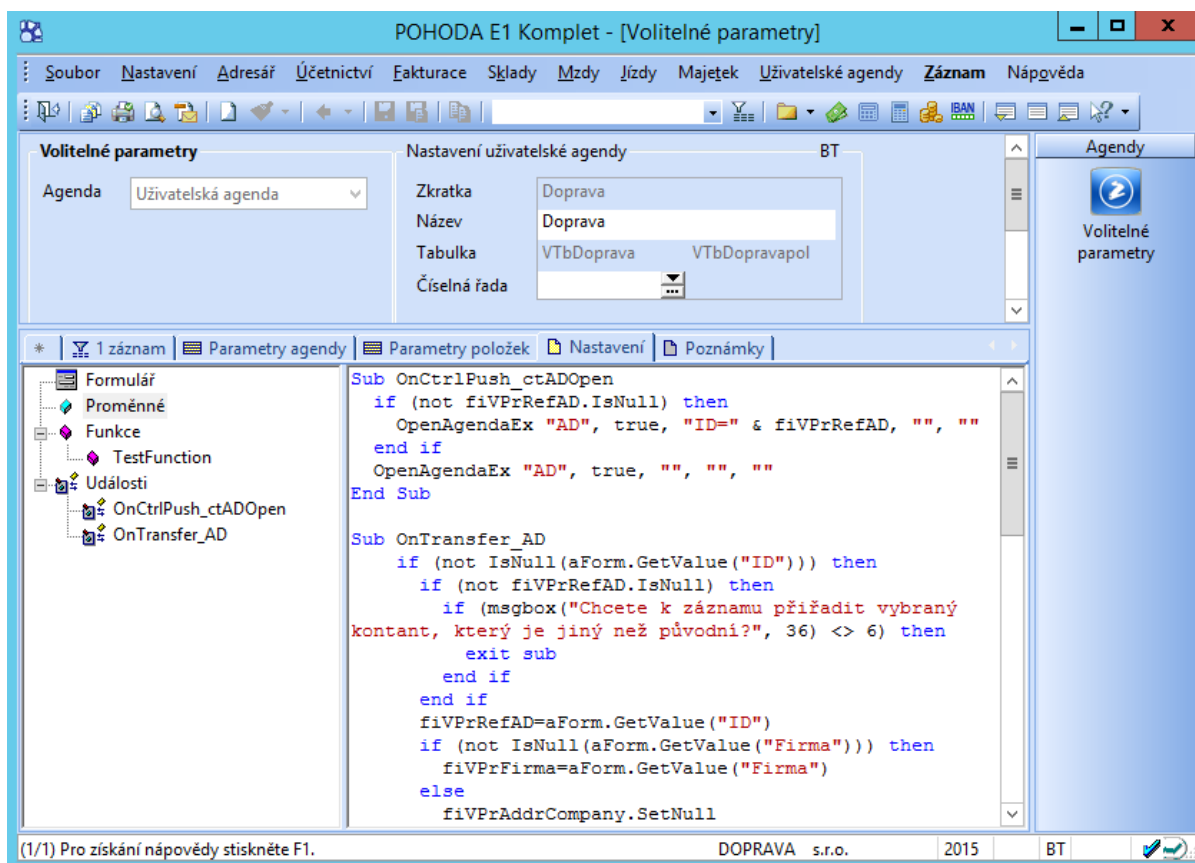
4.1.3 Tvorba uživatelských agend

POHODA E1 umožňuje uživatelská rozšíření programu, která spočívají v možnosti doplnění datového modelu a následné využití VBA (Visual Basic for Applications) pro definování požadované uživatelské funkčnosti.



Obrázek 9: Definice uživatelské agendy

Na obrázku 9 je naznačena definice nové uživatelské agendy. Pokud vybereme z výklopného seznamu Agenda hodnotu Uživatelská agenda, dojde k zpřístupnění druhé části formuláře, ve které dochází k nastavení nové uživatelské agendy. Pro nastavení je nutné vyplnit Zkratku, Název a Tabulku. Název nové tabulky začíná vždy prefixem VTb. V záložce nastavení, zobrazené na obrázku 10, je pak možné doplnit programový kód v jazyce VBScript (Visual Basic Scripting Edition), kterým lze reagovat na události v agendě, upravovat a kontrolovat zadané hodnoty parametrů a přizpůsobit si tak uživatelskou agendu vlastním potřebám. Záložka nastavení obsahuje dvě části. V levé části můžeme vidět jednotlivé Funkce, Události a Proměnné, které pro přizpůsobení agendy používáme, tato část se nazývá Navigační panel. Pravá část záložky Nastavení slouží k zápisu samotného kódu a nazýváme ji Editačním oknem.

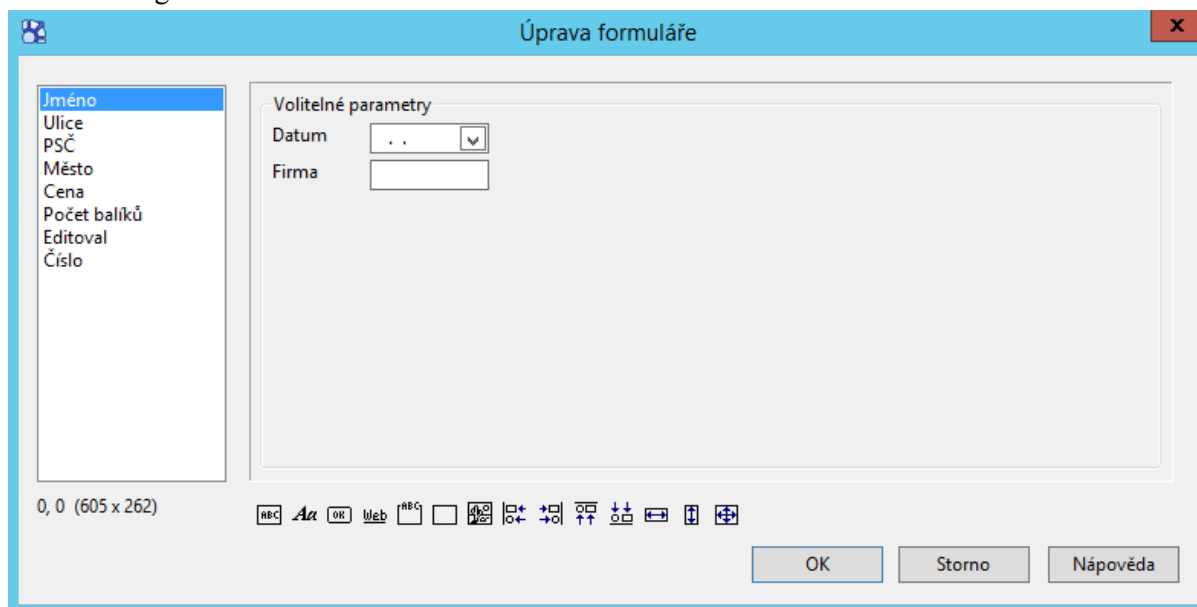


Obrázek 10: Programový kód agendy

Při práci s agendou jsou vyvolávány události, ve kterých lze řídit chování editace dat v agendě. Jednotlivé parametry jsou zpřístupněny jako objekty s metodami přes identifikátor databázového pole s prefixem *fi* pro parametry hlavičky a *gi* pro parametry položek. Například *fiVPrFirma* zpřístupní pole Firma. Na další ovládací prvky formuláře se lze připojit zadáním názvu ve Vlastnostech pole, k těm se pak přistupuje přes prefix *ct*.

4.1.4 Úprava formuláře

Kliknutím pravého tlačítka myši ve formuláři agendy lze vyvolat dialogové okno Úprava formuláře, zobrazené na obrázku 11. V levé části okna lze vidět parametry agendy, které byly dříve nastaveny v agendě Volitelné parametry. Tyto parametry lze dvojklikem vložit na formulář. Jakmile dojde k nastavení těchto parametrů, po stisku tlačítka OK bude formulář agendy dostupný pro další práci ve zvolené agendě.



Obrázek 11: Úprava formuláře

Velikost, šířku a zarovnání parametrů je možné uskutečnit pomocí nástrojové lišty. Umístění jednotlivých popisků je možné měnit. Zároveň lze z nástrojové lišty vložit další ovládací prvky, jako jsou tlačítka, popisky, odkazy, rámečky skupiny, obrázky a podobně.

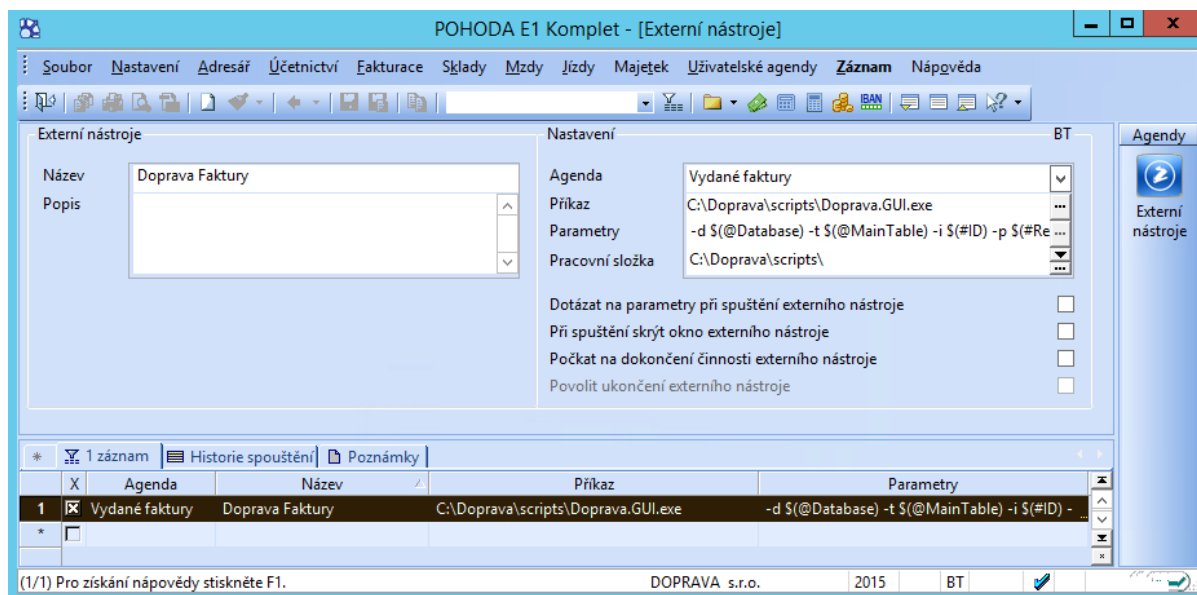
4.1.5 Tiskové sestavy

Pro vytvořenou Uživatelskou agendu je možné vytvořit vlastní tiskové sestavy s využitím volitelných parametrů. Sestavu je potřeba nadefinovat v programu REPORT Designer, který je součástí POHODY. Do sestavy mohou vstupovat data na základě databázového dotazu, takže je možné na sestavu přidat data z více různých tabulek. Je možné vytisknout aktuální záznam pomocí tiskové sestavy Doklad nebo seznam dokladů pomocí tiskové sestavy Seznam dokladů.

4.1.6 Externí nástroje

Další způsob, jak rozšířit funkčnost POHODY je možnost nadefinovat externí nástroje, které jsou posléze dostupné v nabídce Záznam/Externí nástroje pro konkrétní agendu, které byl nástroj přiřazen. Na obrázku 12 je ukázka externího nástroje, který spouští uživatelské rozhraní Doprava z agendy

Faktury. Externímu nástroji lze předávat různé parametry z agendy, na kterou byl nastaven. V příkladu níže lze vidět parametry databáze, tabulka agendy a ID aktuálního záznamu



Obrázek 12: Agenda Externí nástroje

4.2 Popis tříd aplikace Doprava

Bylo vytvořeno několik tříd, které abstrahují funkčnost, umožňují snazší práci s daty a zajišťují správný běh aplikace. Budou popsány jen ty nejdůležitější z pohledu funkčnosti, protože řešení obsahuje velké množství pomocných tříd a je zbytečné v této práci popsat všechny do detailu. Při implementaci výsledné aplikace byla snaha co nejvíce využít potenciál objektově orientovaného paradigma, především reflexe, dědičnosti a polymorfismu.

4.2.1 Document

Třída `Document` abstrahuje doklady z ekonomického systému POHODA. Obsahuje vlastnosti, jako jsou číslo dokladu, adresa zákazníka, kontaktní informace, hodnota dobírky a její měna a také seznam vytvořených zásilek. Neobsahuje žádnou funkčnost, pouze uchovává data a nabízí události, při změně jednotlivých vlastností, které jsou využity dalšími třídami.

4.2.2 Package

Stejně jako třída `Document` neobsahuje žádnou funkčnost, pouze abstrahuje a uchovává informace o vytvořených zásilkách, jako jsou například rozměry, váha, poznámka, typ zásilky a podobně.

4.2.3 DocumentCreator

Funkcí této třídy je vytvořit novou instanci třídy `Document` a naplnit ji daty z ekonomického systému POHODA. Tato funkčnost byla záměrně oddělena, kvůli možnosti napojení na další systém.

4.2.4 Config

Tato třída abstrahuje základní konfiguraci společnou pro všechny dopravce. Každý z dopravců poté tuto třídu dědí a doplňuje svoje specifická nastavení. Třída `Config` obsahuje vlastnost `SubConfig`, ve které se nachází potomci této třídy pro jednotlivé dopravce a také tito potomci mohou obsahovat další dílčí instance své třídy ve vlastnosti `SubConfig`. Díky tomu je zajištěno, že má každý dopravce svoje vlastní nastavení, ale je umožněno přebírat informace z rodičovské konfigurace a zároveň i rozdělit konfiguraci na více podřazených konfigurací rozdělených podle číselné řady. Výsledná struktura má tedy stromový charakter. Při vytváření nové configurační třídy pro dopravce stačí pouze zdědit třídu `Config` a doplnit potřebné vlastnosti a o vše ostatní se postará reflexe, takže je vytvoření konfigurace pro nového dopravce velice jednoduchá a rychlá záležitost.

4.2.5 ConfigHelper

`ConfigHelper` je statická třída, která zajišťuje ukládání a načítání konfigurace pomocí serializace a deserializace třídy `Config` do souboru XML. Formát XML byl zvolen nejen kvůli jednoduchosti uložení dat, ale také aby byla uložená konfigurace čitelná pro zkušeného uživatele i mimo uživatelské rozhraní aplikace Doprava. V budoucnu může být přidáno ještě šifrování pro větší bezpečnost uložených dat, ale ztratí se tím možnost úpravy konfigurace mimo aplikaci Doprava.

4.2.6 Shipper

Tato třída je základní pilíř celého řešení, každý dopravce musí být potomek této třídy. Nabízí metody pro vytvoření nového záznamu v uživatelské agendě Doprava i metody pro odesílání dat na servery dopravců a stahování trasovacích informací a výdejních poboček. Podrobněji bude tato třída popsána v kapitole Podrobný popis třídy Shipper.

4.3 Podrobný popis třídy Shipper

Jak již bylo řečeno výše, tato třída je základní stavební kámen pro všechny dopravce. Každý dopravce je potomkem této třídy a má možnost přetížít její metody, aby bylo možné zajistit specifické chování daného dopravce.

4.3.1 Základní metody

Zde bude uvedeno několik základních metod, která třída `Shipper` nabízí.

- `LoadConfig` – slouží k načtení konfigurace a informací z ekonomického systému POHODA
- `OnRowWarning` – ověření dostupnosti číselné řady
- `OnBeforeGenerate` – umožňuje upravit vstupní data těsně před vytvořením nové zásilky, je využita pro zajištění specifického chování některých dopravců
- `Generate` – vytvoření nového záznamu v uživatelské agendě Doprava
- `OnAfterGenerate` – slouží především pro dopravce, kteří využívají online API, po vytvoření záznamu v agendě Doprava dojde k odeslání dat na server dopravce a propsání získaných informací
- `GetVariables` – převod nastavených služeb do textové reprezentace popsané v kapitole Návrh řešení

- `GetNewPackageNumber` – získání nového čísla balíku a dělí se na několik kroků
- `GetNewPackageNumberStep1` – první krok získání nového čísla balíku, z agendy číselných řad vezme aktuální číslo, pro jistotu se zkontroluje, jestli již neexistuje a inkrementuje počítadlo
- `GetNewPackageNumberStep2` – slouží ke složení prefixu, počítadla zásilek a postfixu, případně pro doplnění kontrolní číslice
- `OnDelete` – slouží ke smazání záznamu v agendě Doprava, pokud je nutné vytvořit zásilku znovu. U dopravců využívající online API dojde ke smazání zásilky z jejich systému
- `DownloadRouting` – spouští se typicky každý den ráno pro stažení aktuálních trasovacích informací a výdejních poboček nebo jiných informací nutných pro správnou funkci služeb dopravce
- `Export` – tato metoda slouží pro odeslání dat na servery dopravců

4.3.2 Příklad vytvoření nového dopravce

Pokud není dopravcem vyžadována specifická funkčnost, vytvoření nového dopravce probíhá pouze zděděním třídy `Shipper` a doplnění vlastností pro nastavení služeb. Dále je potřeba vytvořit potomka třídy `Config` pro uchování konfigurace, vytvoření grafického rozhraní pro tyto služby a nakonec doplnit funkci na export dat k dopravci. Níže je zjednodušený příklad tvorby nového dopravce.

```
public class PPL : Shipper
{
    [ShipperSetting(SettingType.BOOL, "VPrPPEveningDel")]
    public bool EveningDelivery { get; set; }

    public override LogMessage[] Export()
    {
        //zde je potřeba doplnit kód pro export dat na server dopravce
    }

    protected override string GetVariables()
    {
        if (EveningDelivery) return "EVENING"; else return "";
    }
}

[ConfigName("PPL Parcel")]
public class PPLConfig : Config
{
    [Setting("Večerní doručení")]
    public bool EveningDelivery { get; set; }

    [Setting("Číslo zákazníka")]
    public string CustomerID { get; set; }

    [Setting("Číslo depa")]
    public string DepoID { get; set; }

    public PPLConfig()
    {
        RowPrefix = "PPL";
        CustomerID = "PPLTEST";
    }
}
```

4.4 Testování

Každý z dopravců musí být před nasazením do reálného provozu důkladně otestován, aby se předešlo problémům s doručením, především u zásilek, které jsou na dobírku, protože by bylo velice složité později vymáhat peníze po zákazníkovi, který již obdržel svoji zásilku a mohlo by dojít ke škodám v rámci tisíců až statisíců.

Pro dopravce, kteří využívají online API pro vytváření zásilek a tisk štítků (IN TIME, Uloženka) bylo nutné pouze vytvořit několik testovacích zásilek a ověřit jestli se data v jejich systému shodují s daty v uživatelské agendě Doprava. Pro ostatní dopravce bylo potřeba provést sérii integračních testů za pomoci obchodních zástupců jednotlivých dopravců. Pro každého z těchto dopravců bylo nutné vytvořit různé typy zásilek s různými kombinacemi služeb a odeslat vytvořené datové soubory včetně štítků obchodnímu zástupci, který následně vrátil připomínky. Ve většině případů to byly pouze drobné připomínky pro zlepšení čitelnosti čárových kódů na adresních štítcích, případně drobné změny v datech jako třeba formát telefonního čísla nebo nepovolené kombinace služeb.

Jakmile byly data i adresní štítky schváleny, přišlo na řadu otestovat funkčnost aplikace na reálných zásilkách. Testování postupně probíhalo u několika zákazníků firmy BHIT CZ s.r.o.. Z počátku se každý den odeslalo pouze několik zásilek, které neměly moc vysokou hodnotu, kdyby náhodou došlo k nějaké chybě. Až bylo zjištěno, že byly zásilky správně doručeny, tak byl spuštěn plný provoz. Obsluha expedice zásilek začala každý den všechny zásilky zasílat pomocí aplikace Doprava.

4.4.1 Statistiky za měsíc březen 2016

V tabulce 4 je uveden přehled zásilek vytvořených pomocí řešení Doprava za měsíc 2016. Statistiky byly získány od několika různých firem, ale v rámci jednoho dopravce jsou data od jedné firmy. Počty objednaných zásilek samozřejmě závisí na oblíbenosti dopravce zákazníky elektronického obchodu a na počtu objednávek. Pro dopravce IN TIME se bohužel statistiky získat nepodařilo.

Dopravce	Počet zásilek	Průměrný počet za den	Celková hodnota dobírek
Česká pošta	3797	165	4 401 071 Kč
DPD	1579	69	800 513 Kč
Geis Parcel	3114	135	4 227 817 Kč
PPL	6043	263	5 842 911 Kč
Uloženka	364	16	167 671 Kč

Tabulka 4: Statistiky za měsíc březen 2016

Ze získaných statistik se tedy celkem za měsíc březen 2016 pomocí nového řešení vytvořilo a doručilo 14 897 zásilek bez jakýchkoliv větších problémů. Pokud nastal nějaký problém, tak byl způsoben chybnými daty zadaných již na objednávce z elektronického obchodu, převážně chybný formát telefonního čísla nebo nesmyslné údaje v poštovním směrovacím čísle a adrese. Tyto chyby ale byly vždy eliminovány při vytváření zásilky, takže doručení proběhlo v pořádku.

5 Závěr

Cílem této práce bylo navrhnout a implementovat novou verzi uživatelské agendy Doprava, která slouží jako jednotné rozhraní pro objednávání zásilek u různých dopravců. Hlavním úkolem bylo především zajistit, aby bylo co nejjednodušší rozšířit toto řešení o další dopravce a vytvořit jednotný systém konfigurace dopravců a načítání dat z ekonomického systému Stormware POHODA. Dále bylo za úkol vytvořit nový systém tisku adresních štítků založeným na knihovně DevExpress Reporting.

Stěženi část řešení tvoří aplikace Doprava, která se spouští jako externí nástroj nad doklady z agend POHODY. Pro implementaci byl vybrán jazyk C# a prostředí .NET, především kvůli jednoduchosti vytváření aplikací pro systém Microsoft Windows, jeho rozsáhlým knihovnám a především kvůli reflexi, pomocí které byl vytvořen jednotný systém konfigurace a načítání služeb z POHODY, založený na attributech tříd a na attributech jejich samotných vlastností. Ve velké míře byla také využita dědičnost, aby bylo možné přistupovat ke všem dopravcům za pomoci stejných metod a nebylo nutné vše implementovat pro každého dopravce znovu.

Dále byla vytvořena uživatelská agenda Doprava, která slouží jako evidence vytvořených zásilek pro statistické účely a také pro snadnou kontrolu stavu odeslaných zásilek.

Z testování v reálném provozu u několika různých firem nebyly nalezeny žádné zásadní nedostatky. Každá firma má sice rozdílné požadavky, ale navržený systém se podařilo vždy nakonfigurovat tak, aby vyhovoval všem požadavkům. Zásilky odeslané během testování byly bez problémů doručeny. Aplikace zvládla i provoz náročných uživatelů, kteří vytvářeli až 300 zásilek během jediného dne, což pomohlo k výraznému zrychlení procesu expedice zásilek k zákazníkům.

Vytvořenou aplikaci lze napojit na další systémy, což v původní verzi bylo sice taky možné, ale velmi omezeně, protože se všechny služby museli zapínat ručně v uživatelském rozhraní a bylo tedy možné vytvořit pouze základní zásilku. V nové verzi probíhá načítání těchto údajů zcela automaticky, takže stačí jen předat správné parametry zdrojové tabulky a dokladu a za pár vteřin je zásilka vytvořena i bez interakce uživatele, pokud jsou všechna data dostupná na dokladu v systému POHODA. Kromě toho lze ale také využít tříd dostupných v knihovně aplikace Doprava, díky kterým je možné integrovat vytváření zásilek přímo do jiných systémů.

Aplikaci je možné rozšířit o jakéhokoliv dopravce a do budoucna má jistě velký potenciál k dalšímu rozvoji.

Literatura

- [1] POHODA - ekonomický a informační systém [online]. © 2014 [cit. 2016-05-14]. Dostupné z: <http://www.stormware.cz/>
- [2] Účetní program Money S3, ERP systém a informační systémy S4 & S5 - CÍGLER SOFTWARE [online]. © 2016 [cit. 2016-05-14]. Dostupné z: <http://www.money.cz/>
- [3] HELIOS - podnikový informační systém, ekonomický a účetní software, systém pro veřejnou správu [online]. © 2016 [cit. 2016-05-14]. Dostupné z: <http://www.helios.eu/>
- [4] Helios Orange: nejrozšířenější ERP systém na českém trhu. In: ERP - podnikové informační systémy. - ERP forum [online]. 2009-12-21 [cit. 2016-05-14]. Dostupné z: <http://www.erpforum.cz/erp-systemy/helios-orange.html>
- [5] Česká pošta, s.p.. Technická dokumentace - Hromadné podání zásilek smluvním podavatelem [online]. 2015-11-18 [cit. 2016-05-14]. Dostupné z: https://www.ceskaposta.cz/documents/10180/282457/apost_Smluvni-podavatel_uziv_dok_v1-11.pdf
- [6] DPD CZ. Specifikace datového formátu MPSEXPDATA pro přenos kompletních dat o zásilkách [online]. 2015-04-14 [cit. 2016-05-14]. Dostupné z: https://www.dpd.com/cz/domu/vas_pruvodce_prepravou/aplikace_a_nastroje/vlastni_software
- [7] Geis Parcel CZ s.r.o.. Manuál pro systémovou integraci [online]. 2016-05-02 [cit. 2016-05-14]. Dostupné z: http://www.geisparcel.cz/support/files/Geis_Integrace_Manual.pdf
- [8] IN TIME - Dokumentace pro vývojáře [online]. [cit. 2016-05-14]. Dostupné z: <https://bridge.intime.cz/doc/index.html>
- [9] IEGate Web Service [online]. [cit. 2016-05-14]. Dostupné z: <https://www.ppl.cz/IEGate/>
- [10] Uložka.cz API v3 [online]. [cit. 2016-05-14]. Dostupné z: <http://docs.ulozkav3.apiary.io>
- [11] WALTERS, R. E. Mistrovství v Microsoft SQL Server 2008: [kompletní průvodce databázového experta]. Brno: Computer Press, 2009. ISBN 978-80-251-2329-4.
- [12] ROBINSON, Simon. C#: programujeme profesionálně. Brno: Computer Press, 2003. ISBN 80-251-0085-5.
- [13] .NET UI Controls for Developers of Mobile, Desktop, Web & Reporting Applications | www.DevExpress.com [online]. © 1998-2016 [cit. 2016-05-16]. Dostupné z: <https://www.devexpress.com/>