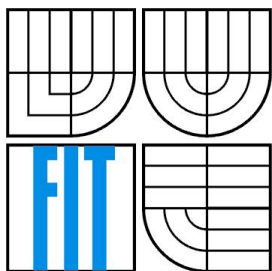


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



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

SOFTWARE PRO SPRÁVU REKLAMNÍCH PLOCH V TISKOVINÁCH

SOFTWARE FOR MANAGEMENT OF ADVERTISEMENTS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Václav Maleček

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Bohuslav Křena, Ph.D.

BRNO 2008

Abstrakt

Tato diplomová práce popisuje vývoj informačního systému pro správu reklamních ploch v tiskovinách včetně úvodu do problematiky, definice požadavků, návrhu, implementaci a diskuzi o plánovaných rozšířeních aplikace. Při návrhu a implementaci aplikace byl kladen důraz na intuitivní a pohodlné ovládání uživateli. Systém je naprogramován v jazyce Java, pro zajištění persistence dat byl využit systém Hibernate a grafické uživatelské rozhraní využívá framework Swing. Vlastní uložení dat je zprostředkováno databází MySQL.

Klíčová slova

Java, Hibernate, Swing, informační systém, Model-view-controller, databáze, MySQL, reklamní noviny, inzertní systém, správa reklamních ploch

Abstract

This master's thesis describes development of information system for management of printed paper advertisements, including the problem description, definition of requirements, design, implementation and discussion of planned extensions. During design and implementation, special attention was paid to intuitive and easy user interface. System is programmed in Java language. Hibernate is used for data persistence. Graphical user interface makes use of the Swing framework. MySQL is used as the database system.

Keywords

Java, Hibernate, Swing, information system, Model-view-controller, database, MySQL, advertising paper, advertising system, management of advertisements

Citace

Maleček Václav: Software pro správu reklamních ploch v tiskovinách. Brno, 2008, diplomová práce, FIT VUT v Brně.

Software pro správu reklamních ploch v tiskovinách

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Bohuslava Křeny, Ph.D. Další informace z oblasti reklamy a procesu přípravy a tvorby inzertních novin mi poskytl ředitel reklamní agentury GOLEM GROUP s.r.o. Ing. Petr Krátký, který je zároveň zadavatelem diplomové práce. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Václav Maleček
19.5.2008

Poděkování

Rád bych poděkoval Ing. Bohuslavu Křenovi, Ph.D., za jeho čas, konzultace a cenné připomínky k diplomové práci. Dále bych chtěl poděkovat Ing. Petru Krátkému za vstřícný přístup a ochotu při upřesnění požadavků. V neposlední řadě děkuji celé rodině za trpělivost a podporu, kterou mi poskytli v průběhu celého studia.

© Václav Maleček, 2008.

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
2 Úvodní studie požadavků.....	4
2.1 Proces přípravy a prodeje reklamní plochy	4
2.2 Inzertní noviny GOLEMPRESS.....	4
2.2.1 Životní cyklus novin	4
2.2.2 Formát novin a reklamní plochy	5
2.2.3 Inzeráty	6
2.2.4 Objednávky	6
3 Specifikace požadavků.....	8
3.1 Role uživatelů a přístupová práva.....	8
3.2 Případy použití.....	9
3.3 Obecné požadavky	20
3.3.1 Autentizace uživatelů.....	20
3.3.2 Hromadný import objednávek do systému	20
3.3.3 Práce s grafickou maketou	20
4 Návrh systému	22
4.1 Návrh architektury	22
4.2 Návrh struktury dat	23
4.2.1 Popis jednotlivých entit	24
5 Implementace a testování.....	27
5.1 Použité technologie.....	27
5.1.1 Programovací jazyk Java	27
5.1.2 Framework Hibernate	28
5.1.3 Databáze MySQL	29
5.2 Popis implementace	29
5.2.1 Struktura aplikace	30
5.2.2 Hlavní okno aplikace	32
5.2.3 Otevření makety.....	33
5.2.4 Práce s maketou	34
5.2.5 Datový model pro výpis entit.....	36
5.2.6 Prohlížení objednávek	36
5.2.7 Import objednávek z CSV souboru.....	38
5.3 Použité knihovny a komponenty.....	39
5.4 Nasazení a testování	39
5.4.1 Testování na reálných datech.....	39
5.5 Instalace a požadavky na prostředí	40
6 Plánovaná rozšíření	41
6.1 Šablony stránek.....	41
6.2 Tiskové výstupy	41
6.2.1 Tisk makety stránky.....	41
6.2.2 Tisk seznamu objednávek z makety	41
6.3 Podrobnější výpisy u importu objednávek.....	42

6.4	Možnost vkládat obrázky na místo ploch	42
6.5	Přesun části aplikační logiky na server.....	42
7	Závěr	44
	Literatura	45
	Seznam příloh	46

1 Úvod

Informační technologie stále více nabývají na významu především při nasazení ve specifických odvětvích podnikání. Informační systém či zakázkový software dnes uživatelům ulehčuje práci, přístup k informacím a pomáhá při analýze business procesů a jejich plánování.

Náplní mé diplomové práce je navrhnout a implementovat software pro správu reklamní plochy v tiskovinách. V této technické zprávě se zaměřuji především na část analýzy a návrhu tohoto informačního systému. Při analýze a návrhu celého řešení problému využívám mojí dlouholeté spolupráce se zadavatelem a znalosti firemních a procesních pochodů při vydávání reklamních novin a prodeji inzertních ploch.

Výsledný software bude ulehčovat práci při procesu prodeje reklamní plochy a jejího organizování v maketě připravovaných vydáních novin. Strategie plánování a rychlá komunikace mezi jednotlivými business rolemi, které se procesu prodeje reklamních ploch v inzertních novinách procesu tvorby těchto novin účastní, je úzkým hrdlem celého procesu. Právě vytvoření a nasazení specifického softwaru by mělo ulehčit a zefektivnit tuto část procesu.

V druhé kapitole uvádím čtenáře do problematiky práce s maketami inzertních novin a prodeje reklamních ploch. Další dvě kapitoly jsou věnovány specifikaci požadavků kladených na systém a návrhu celého informačního systému. V kapitole 5 popisují stěžejní body implementace a nasazení aplikace do testovacího provozu. V další kapitole popisují nové požadavky a možnosti dalšího rozšíření aplikace, které vyplynuly z testovacího provozu u zadavatele. Poslední kapitola shrnuje přínos celého systému a zhodnocuje výsledek celé práce.

2 Úvodní studie požadavků

Po přečtení této kapitoly získá čtenář přehled o procesu tvorby a přípravy reklamních novin. Zejména je popsána část prodeje reklamní plochy v návaznosti na práci s maketou čísel jednotlivých oblastí novin.

2.1 Proces přípravy a prodeje reklamní plochy

Reklamní agentura zabývající se reklamou, webovými službami a propagací firem požaduje zhotovení informačního systému pro správu a přehled obsazenosti reklamní plochy v inzertních tiskovinách. Jednou z hlavních činností reklamní agentury je vydávání inzertních novin GOLEMPRESS. Noviny vycházejí v několika oblastech cca jednou za měsíc. Agentura nabízí a prodává plochu v těchto novinách prostřednictvím obchodních zástupců, kteří mají provizi z množství prodaných služeb. Pro každou oblast může existovat v agentuře jiný šéfredaktor. To je osoba, která organizuje a spravuje uspořádání plošných inzerátů v každém čísle novin. Šéfredaktor komunikuje s obchodními zástupci a sděluje jim podstatné informace o obsazenosti a aktuálně neobsazených plochách na stránkách novin. Šéfredaktor doposud používal papírovou maketu novin a jednotlivé plochy (inzeráty) v nich vylepoval v podobě nadepsaných štítků. Pokud došlo ke změně uspořádání, musel šéfredaktor štítky odtrhnout a přelepit.

Tento systém práce s papírovými maketami novin je značně neefektivní. Nasazením elektronického systému maket novin se slibuje zefektivnění prodeje. Obchodní zástupci budou moci online zjišťovat zaplněnost novin a klientům tak nabízet konkrétní služby. Očekáváme sice v počátcích práce s novou aplikací nárůst práce šéfredaktorů díky obsluze a korektnosti rozmístění inzerátů v maketě, ale ulehčení komunikace šéfredaktorů s obchodními zástupci slibuje návratnost celého projektu.

2.2 Inzertní noviny GOLEMPRESS

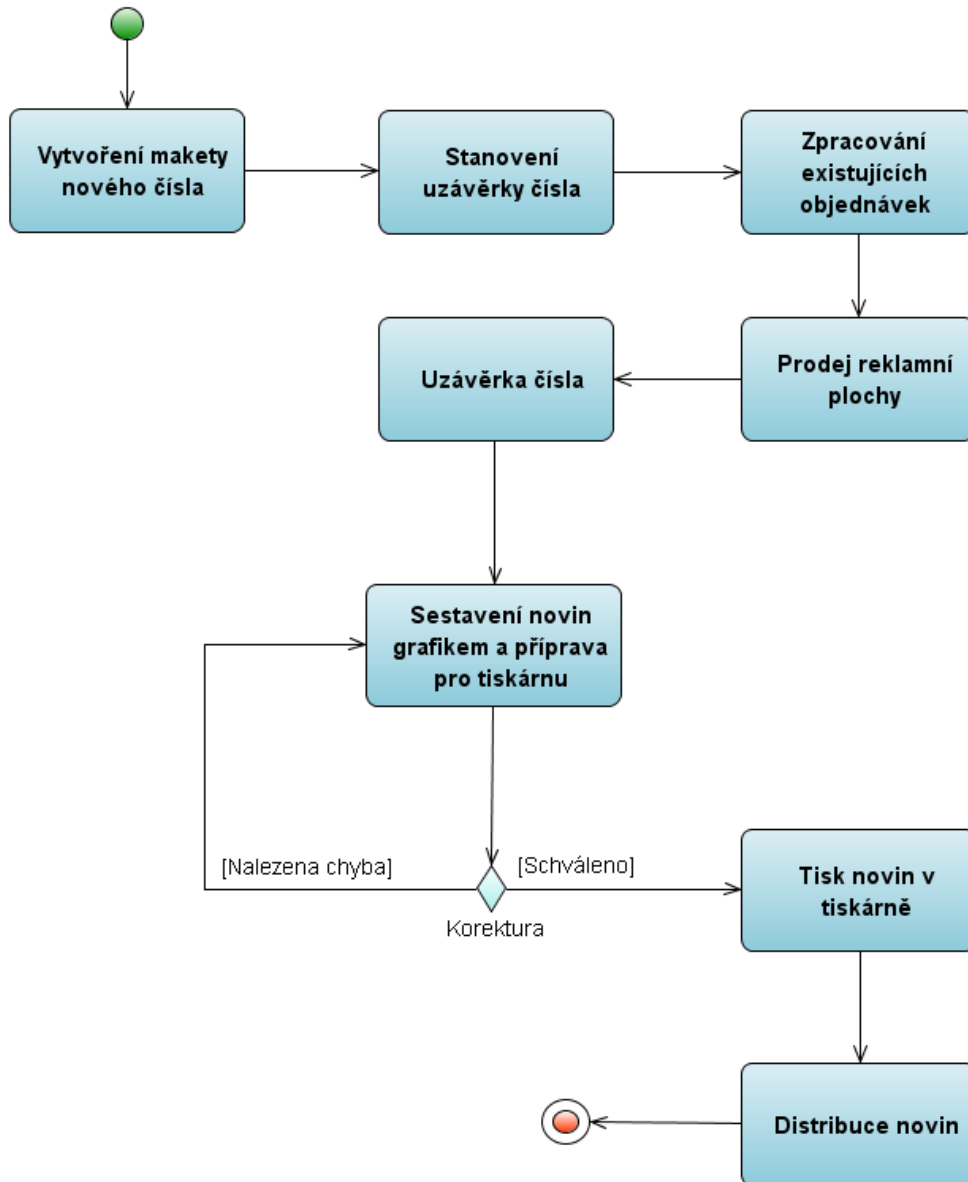
Inzertní noviny GOLEMPRESS vycházejí ve východočeském regionu v 19 okresech. Náklad se pohybuje od 4.000 do 50.000 výtisků pro danou oblast (lokace může zahrnovat více okresů). V reklamní agentuře jsou zaměstnáni lidé, kteří se starají o životní cyklus novin pro daný okres – šéfredaktori. Šéfredaktor zpravidla spravuje 1 až 5 okresů. Noviny vycházejí až na výjimky v pravidelných intervalech (14 dní, měsíc apod.).

2.2.1 Životní cyklus novin

Na počátku šéfredaktor sestaví maketu novin (nyní papírová podoba) s počtem stránek, který závisí na počtu stránek předchozího čísla daného okresu a předpokladu zaplněnosti pro nové číslo. Stanoví se datum uzávěrky daného čísla a ten se distribuuje obchodním zástupcům. Obchodní zástupci prodávají reklamní plochu v tomto čísle v podobě inzerátů. Můžou však prodat reklamní plochu také do několika následujících čísel nebo pro více okresů. Pokud je reálná prodejnost vyšší, než stačí pokrýt nabízená plocha v novinách, šéfredaktor jednoduše přidá potřebný počet stran.

Právě před uzávěrkou čísel je největším problémem zefektivnění práce a rozmístění reklamních ploch v novinách. Spousta plochy může zůstat volná a tak se „na poslední chvíli“ může nabízet se slevou. Obchodní zástupci jsou tedy pořád v kontaktu s danými šéfredaktory. V den uzávěrky čísla se

maketa předá grafikovi, který rozmístí inzeráty podle makety a připraví číslo pro tisk v tiskárně. Po všech korekturách a jiných procesech jdou noviny do tisku a distribuce. Pro znázornění na obrázku uvádím stavový diagram životního cyklu novin a prodeje reklamních ploch (viz Obrázek 2.1 a Obrázek 2.2).

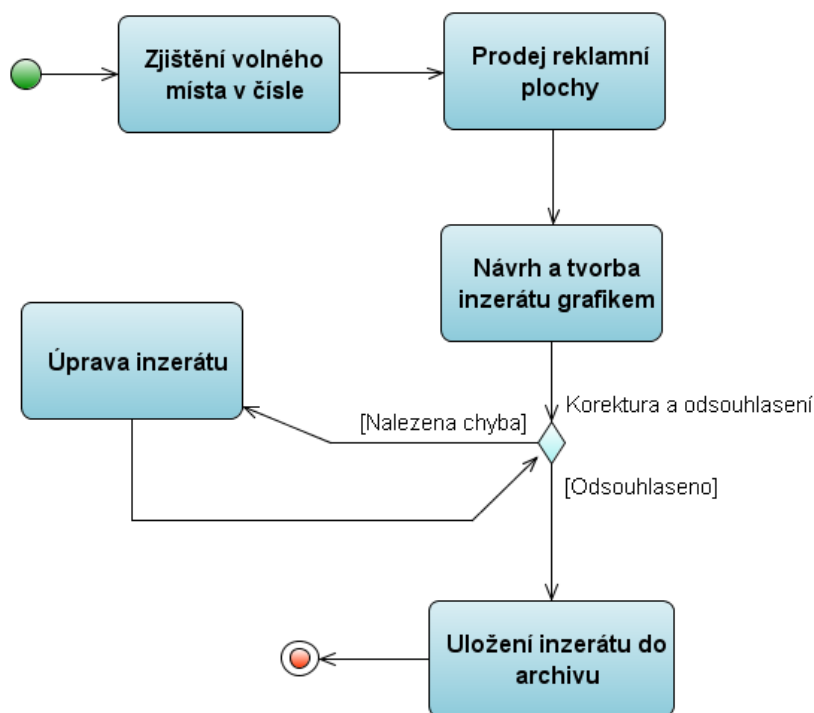


Obrázek 2.1 - Stavový diagram životního cyklu novin

2.2.2 Formát novin a reklamní plochy

Jeden list novin má formát A3, tedy rozměry 297×420 mm. Listy jsou spárovány do dvojlistů. Pokud vychází počet listů liše, je prostřední list samostatný. Reklamní plocha je určena pro umístění inzerátů. Reklamní plocha v novinách nezasahuje až do krajů. Počítá se s 2,5 cm volného místa od krajů stránky, které může být dodatečně zvětšeno nebo zmenšeno o 0,5 cm. Na přední straně je také hlavička novin a na poslední straně je ve spodní části stránky tiráž. Speciální reklamní plocha se

nachází ve formě proužku mezi v páru spojenými listy, v software však není potřeba tuto speciální plochu uvažovat.



Obrázek 2.2 - Stavový diagram prodeje reklamní plochy

Inzertní noviny (čísla) mohou být buď celobarevné, částečně barevné nebo černobílé. U částečně barevných novin je v barvě pouze obálka (tj. nejsvrchnější dvojlist – první, druhá, předposlední a poslední strana).

2.2.3 Inzeráty

Obchodní zástupci prodávají reklamní plochu v podobě inzerátů. Inzerát je grafická reklama zabírající souvislou, pravouhle ohraničenou plochu. Inzerát může mít pouze podobu čtverce nebo obdélníku. Podle umístění inzerátu a typu čísla novin je inzerát buď barevný, nebo černobílý. Barevnost inzerátu se tedy odvíjí od stránky, na které je umístěn. Rozměry inzerátu se odvíjí od objednávky. Ovšem je možné, že finální podobna inzerátu bude mít jiné rozměry (zpravidla větší díky volnému místu na stránce).

2.2.4 Objednávky

Objednávka reklamní plochy může zahrnovat objednání různých reklamních ploch v libovolných vydáních dané oblasti inzertních novin. Pokud klient požaduje objednání stejné plochy ve více číslech, vytvoří se samostatná objednávka pro každý požadavek. Objednávka má jednoznačné číslo, které ji identifikuje. U každé objednávky je uveden atribut *doklad*, který je reprezentován řetězcem čísel, který je v rámci všech objednávek rovněž jednoznačný. Pro systém práce s maketou se nebere v úvahu žádná databáze klientů – zákazníků. Je však nutné zajistit sémantickou korespondenci s interním účetním systémem. U každé objednávky je tedy nutné uvést potřebné informace o klientovi či firmě.

Objednávka nespecifikuje konkrétní číslo vydání. Konkrétní číslo volí šéfredaktor tím, že umístí inzerát korespondující s objednávkou do makety čísla.

3 Specifikace požadavků

V této kapitole popisují jednotlivé požadavky, které jsou na systém kladeny. Stěžejní částí této kapitoly je use-case diagram s popisem případů použití, který mimo jiné částečně ukazuje i přístupová práva jednotlivých uživatelů systému. Další podkapitola upřesňuje některé požadavky, které nevyplývají přímo z případů použití.

3.1 Role uživatelů a přístupová práva

V systému budou pracovat uživatelé čtyř úrovní:

- správce,
- šéfredaktor,
- obchodní zástupce,
- grafik.

Je nutné zajistit omezení přístupových práv v celém systému podle konkrétních rolí uživatelů.

Grafik

Grafický pracovník mimo tvorby jednotlivých inzerátů může mít za úkol i sazbu a přípravu novin pro tisk. Má rovněž na starosti fyzické umístění inzerátů na jednotlivých stránkách právě vydávaného čísla.

Uživatel typu „grafik“ má v systému pouze práva pro prohlížení maket čísel. Dále má právo na prohlížení objednávek za použití výběrového filtru. Nemůže tedy měnit uskupení inzertních ploch v maketách a ani měnit objednávky v libovolných číslech. Jeho role je „pozorovací“.

Obchodní zástupce

Prvořadým úkolem obchodních zástupců je prodávat reklamní plochu a pružně reagovat na změny rozmístění ploch v jednotlivých maketách čísel. Obchodní zástupci pak na základě aktuálního stavu makety prodávají volnou plochu v novinách a zasílají šéfredaktorům objednávky.

Obchodní zástupce může v systému na rozdíl od grafika zadávat objednávky do libovolných oblastí novin. Nemůže však objednávky mazat nebo měnit jejich stav či ostatní atributy.

Šéfredaktor

Šéfredaktor aktivně pracuje s maketou jednotlivých čísel jemu svěřené oblasti. Mění rozmístění inzerátů a přiřazení objednávek k zadaným reklamním plochám v maketě. Výsledkem jeho práce je pak podklad pro grafika a podklad pro fakturaci objednávek, které byly vyřízeny v rámci připravovaného čísla novin.

Šéfredaktor má za úkol primárně spravovat jemu přiřazené oblasti (dále jen „přiřazené oblasti“). U přiřazených oblastí může přidávat, editovat a mazat jednotlivá čísla, měnit a mazat objednávky.

Správce

Správce systému má práva pro správu všech entit a práci s maketami čísel všech oblastí definovaných v systému. Má právo definovat nové oblasti novin, měnit a mazat oblasti.

3.2 Případy použití

Typické akce, které budou moci jednotliví uživatelé v systému provádět, jsou znázorněny v USE CASE diagramu (viz Obrázek 3.1) a přehledně popsány v následujících tabulkách.



Obrázek 3.1 - Diagram případů použití

Jméno případu použití: Prohlížení objednávek
ID: 1
Stručný popis: Všichni aktéři mají možnost prohlížet objednávky zadané v systému a zjišťovat stav a umístění reklamní plochy. Prohlížení objednávek je zprostředkováno jejich seznamem, který lze filtrovat podle nejrůznějších kritérií (datum, stav objednávky, oblast novin).
Primární aktéři: Všichni autentizovaní uživatelé
Předpoklady: Uživatel je přihlášen v systému.
Hlavní tok: <ol style="list-style-type: none"> 1. Příklad použití se spustí, když aktér klikne na volbu „Objednávky“. 2. Systém zobrazí okno s výpisem všech objednávek. 3. Aktér zvolí filtr, podle kterého se zobrazí seznam objednávek. 4. Po kliknutí na tlačítko „Potvrdit filtr“ se vypíše objednávky vyhovující vytvořenému filtru.
Následné podmínky: Jsou zobrazeny objednávky (výpis v tabulce) podle zvolených kritérií.

Jméno případu použití: Prohlížení maket
ID: 2
Stručný popis: Všichni aktéři mají možnost prohlížet makety jednotlivých čísel reklamních novin. Pokud má uživatel dostatečná oprávnění, může jednotlivé inzeráty v maketě rovnou měnit (přesun, změna atributů apod.).
Primární aktéři: Všichni autentizovaní uživatelé
Předpoklady: Uživatel je přihlášen v systému.
Hlavní tok: <ol style="list-style-type: none"> 1. Příklad použití se spustí, když aktér klikne na volbu „Otevřít maketu“. 2. Systém zobrazí okno pro výběr jednotlivých čísel podle oblastí. 3. Po kliknutí na tlačítko „Otevřít“ se zobrazí okno s první stránkou makety čísla. 4. Je zobrazena první stránka makety. Aktér může zobrazit libovolnou stránku z makety volbou z vhodně umístěného seznamu stránek makety.
Následné podmínky: Je zobrazena stránka makety vybraného čísla novin.

Jméno případu použití: Přidání uživatele
ID: 3
Stručný popis: Příklad použití slouží k přidání nového uživatele do systému.
Primární aktéři: Správce
Předpoklady: Příklad použití je spuštěn správcem.

<p>Hlavní tok:</p> <ol style="list-style-type: none"> 1. Příklad použití se spustí, když aktér klikne na volbu „Přidat uživatele“. 2. Systém nabídne dialog s formulářem pro zadání atributů nového uživatele. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není uživatel se stejným loginem. 5A. Systém uloží nového uživatele.
<p>Následné podmínky: Nový uživatel je uložen v databázi.</p>
<p>Alternativní toky:</p> <ol style="list-style-type: none"> 4B. Nejsou vyplněny všechny údaje. Aktér je upozorněn a vyzván k opravě údajů vkládaného uživatele. 5B. V systému již existuje uživatel se zadaným loginem. Aktér je upozorněn a vyzván k opravě loginu vkládaného uživatele.

<p>Jméno případu použití: Změna uživatele</p>
<p>ID: 4</p>
<p>Stručný popis: Příklad použití slouží ke změně atributů již existujícího uživatele.</p>
<p>Primární aktéři: Správce</p>
<p>Předpoklady: Příklad použití je spuštěn správcem. V seznamu uživatelů je vybrán právě jeden uživatel.</p>
<p>Hlavní tok:</p> <ol style="list-style-type: none"> 1. Příklad použití se spustí, když aktér klikne na volbu „Změnit uživatele“. 2. Systém nabídne dialog s formulářem pro změnu atributů uživatele. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není uživatel se stejným loginem. 5A. Systém uloží nové atributy uživatele. Pokud nebylo zadáno heslo editovaného uživatele, zůstává v platnosti původní.
<p>Následné podmínky: Nové atributy uživatele jsou uloženy.</p>
<p>Alternativní toky:</p> <ol style="list-style-type: none"> 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě údajů editovaného uživatele. 5B. V systému již existuje jiný uživatel se zadaným loginem. Aktér je upozorněn a vyzván k opravě loginu vkládaného uživatele.

<p>Jméno případu použití: Smazání uživatele</p>
<p>ID: 5</p>
<p>Stručný popis: Příklad použití slouží ke smazání existujícího uživatele. Je možné mazat více uživatelů zároveň.</p>
<p>Primární aktéři: Správce</p>
<p>Předpoklady:</p>

Případ použití je spuštěn správcem. V seznamu uživatelů je vybrán alespoň jeden uživatel.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Smazat uživatele“. 2. Systém nabídne potvrzovací dialog pro smazání uživatelů. 3. Po kliknutí na tlačítko „Ano“ dojde k smazání vybraných uživatelů.
Následné podmínky: Všichni vybraní uživatelé jsou smazáni z databáze včetně jejich log záznamů.

Jméno případu použití: Přidání oblasti
ID: 6
Stručný popis: Případ použití slouží k přidání nové oblasti inzertních novin.
Primární aktéři: Správce
Předpoklady: Případ použití je spuštěn správcem.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Přidat oblast“. 2. Systém nabídne dialog s formulářem pro zadání atributů nové oblasti. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není oblast se stejnou zkratkou a stejným kódem. 5A. Systém uloží novou oblast.
Následné podmínky: Nová oblast je uložena v databázi.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě atributů oblasti. 5B. V systému již existuje oblast se stejným kódem nebo zkratkou. Aktér je upozorněn a vyzván k opravě kódu či zkratky oblasti.

Jméno případu použití: Změna oblasti
ID: 7
Stručný popis: Případ použití slouží ke změně atributů již existující oblasti novin.
Primární aktéři: Správce
Předpoklady: Případ použití je spuštěn správcem. V seznamu oblastí je vybrána právě jedna oblast.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Změna oblasti“. 2. Systém nabídne dialog s formulářem pro změnu oblasti. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není oblast se stejným kódem nebo stejnou

zkratkou. 5A. Systém uloží nové atributy oblasti.
Následné podmínky: Nové atributy oblasti jsou uloženy.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě atributů oblasti. 5B. V systému již existuje oblast se stejným kódem nebo zkratkou. Aktér je upozorněn a vyzván k opravě kódu či zkratky oblasti.

Jméno případu použití: Smazání oblasti
ID: 8
Stručný popis: Případ použití slouží ke smazání existující oblasti. Je možné mazat více oblastí zároveň.
Primární aktéři: Správce
Předpoklady: Případ použití je spuštěn správcem. V seznamu oblastí je vybrána alespoň jedna oblast.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Smazat oblast“. 2. Systém nabídne potvrzovací dialog pro smazání oblastí. 3. Po kliknutí na tlačítko „Ano“ dojde k smazání vybraných oblastí.
Následné podmínky: Oblasti jsou smazány z databáze včetně všech jejich čísel/stránek/inzerátů.

Jméno případu použití: Přidání čísla novin do oblasti
ID: 9
Stručný popis: Případ použití slouží k přidání nového čísla oblasti.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem, který je přiřazen příslušné oblasti.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Přidat číslo novin do oblasti“. 2. Systém nabídne dialog s formulářem pro zadání atributů nového čísla. 3. Uživatel vyplní rok a pořadové číslo v tomto roce. Dále vyplní datum uzávěrky a zvolí barevnost čísla oblasti. 4. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 5A. Systém uloží nové číslo oblasti.
Následné podmínky: Nové číslo oblasti je uloženo v databázi.
Alternativní toky:

5B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů.

Jméno případu použití: Změna čísla novin oblasti
ID: 10
Stručný popis: Případ použití slouží ke změně atributů již existujícího čísla oblasti novin.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. V seznamu čísel oblastí je vybráno právě jedno číslo.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Změnit číslo novin oblasti“. 2. Systém nabídne dialog s formulářem pro změnu čísla. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém uloží nové atributy čísla.
Následné podmínky: Nové atributy čísla oblasti jsou uloženy.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů.

Jméno případu použití: Smazání čísla novin oblasti
ID: 11
Stručný popis: Případ použití slouží ke smazání existujícího čísla novin oblasti. Je možné mazat více čísel zároveň.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. V seznamu čísel novin oblasti je vybráno alespoň jedno číslo.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Smazat číslo novin“. 2. Systém nabídne potvrzovací dialog pro smazání čísel. 3. Po kliknutí na tlačítko „Ano“ dojde ke smazání vybraných čísel novin oblasti.
Následné podmínky: Čísla jsou smazána z databáze včetně všech jejich stránek/inzerátů.

Jméno případu použití: Přidání stránky
ID: 12
Stručný popis:

Případ použití slouží k přidání nové stránky do čísla novin.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem, který je přiřazen příslušné oblasti. Je vybráno číslo oblasti, do kterého se má stránka vložit.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Přidat stránku“. 2. Systém nabídne dialog s formulářem pro zadání atributů nové stránky. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není stránka s tímto pořadovým číslem pro dané číslo oblasti. 5A. Systém uloží novou stránku.
Následné podmínky: Nová stránka je uložena v databázi.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů. 5B. Pořadové číslo stránky již existuje pro dané číslo oblasti. Aktér je upozorněn a vyzván k opravě pořadového čísla.

Jméno případu použití: Změna stránky
ID: 13
Stručný popis: Případ použití slouží ke změně atributů již existující stránky čísla novin.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. V seznamu stránek čísla novin je vybrána právě jedna stránka.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Změnit stránku“. 2. Systém nabídne dialog s formulářem pro editaci stránky. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není jiná stránka s tímto pořadovým číslem pro dané číslo oblasti. 5A. Systém uloží nové atributy stránky.
Následné podmínky: Nové atributy stránky jsou uloženy.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů. 5B. Pořadové číslo stránky již existuje pro dané číslo oblasti. Aktér je upozorněn a vyzván

k opravě pořadového čísla.

Jméno případu použití: Smazání stránky

ID: 14

Stručný popis:

Případ použití slouží ke smazání existující stránky čísla. Je možné mazat více stránek zároveň.

Primární aktéři:

Správce

Šéfredaktor

Předpoklady:

Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti.

V seznamu stránek čísla je vybrána alespoň jedna stránka.

Hlavní tok:

1. Případ použití se spustí, když aktér klikne na volbu „Smazat stránku“.
2. Systém nabídne potvrzovací dialog pro smazání stránek.
3. Po kliknutí na tlačítko „Ano“ dojde ke smazání vybraných stránek.

Následné podmínky:

Stránky jsou smazány z databáze včetně všech jejich inzerátů.

Jméno případu použití: Zadání objednávky

ID: 15

Stručný popis:

Případ použití slouží k přidání nové objednávky.

Primární aktéři:

Správce

Šéfredaktor

Obchodní zástupce

Předpoklady:

Případ použití je spuštěn správcem, šéfredaktorem nebo obchodním zástupcem.

Hlavní tok:

1. Případ použití se spustí, když aktér klikne na volbu „Přidat objednávku“.
2. Systém nabídne dialog s formulářem pro zadání atributů nové objednávky.
3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů.
- 4A. Systém zkontroluje, zda v systému již není objednávka se stejným číslem nebo dokladem.
- 5A. Systém uloží novou objednávku.

Následné podmínky:

Nová objednávka je uložena v databázi.

Alternativní toky:

- 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů.
- 5B. V systému již existuje objednávka se stejným číslem nebo dokladem. Aktér je upozorněn a vyzván k opravě čísla objednávky nebo dokladu.

Jméno případu použití: Změna objednávky
ID: 16
Stručný popis: Případ použití slouží ke změně atributů již existující objednávky, především ke změně stavu objednávky (čeká, vyřízena, storno).
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti, pro kterou je objednávka definována. V seznamu objednávek je vybrána právě jedna objednávka.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Změnit objednávku“. 2. Systém nabídne dialog s formulářem pro editaci objednávky. 3. Po kliknutí na tlačítko „Uložit“ dojde k validaci údajů. 4A. Systém zkontroluje, zda v systému již není objednávka se stejným číslem nebo dokladem. 5A. Systém uloží nové atributy objednávky.
Následné podmínky: Nové atributy objednávky jsou uloženy.
Alternativní toky: 4B. Nejsou vyplněny všechny potřebné údaje. Aktér je upozorněn a vyzván k opravě zadaných údajů. 5B. V systému již existuje objednávka se stejným číslem nebo dokladem. Aktér je upozorněn a vyzván k opravě čísla objednávky nebo dokladu.

Jméno případu použití: Smazání objednávky
ID: 17
Stručný popis: Případ použití slouží ke smazání existující objednávky ze systému. Na rozdíl od změny stavu objednávky na „storno“ je při smazání objednávky odstraněn záznam z databáze. Je možné smazat více objednávek najednou.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti, pro kterou je objednávka definována. V seznamu objednávek je vybrána alespoň jedna objednávka.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Smazat objednávku“. 2. Systém nabídne potvrzovací dialog pro smazání objednávek. 3. Po kliknutí na tlačítko „Ano“ dojde ke smazání vybraných objednávek.
Následné podmínky: Objednávky jsou smazány z databáze. U inzerátů, které k mazaným objednávkám náležely, je

pouze zrušena vazba k smazané objednávce a v systému zůstávají.

Jméno případu použití: Import objednávek

ID: 18

Stručný popis:

Případ použití slouží k importu objednávek z CSV (Comma-separated values) souboru.

Primární aktéři:

Správce

Šéfredaktor

Předpoklady:

Případ použití je spuštěn správcem nebo šéfredaktorem. V tomto případě použití má možnost libovolný šéfredaktor pomocí importu vložit objednávky do libovolné oblasti novin.

Hlavní tok:

1. Případ použití se spustí, když aktér klikne na volbu „Import objednávek“.
2. Systém nabídne dialog průběhu importu s možností vybrání importovaného souboru z disku.
3. Po kliknutí na tlačítko „Start“ dojde k validaci importního souboru.
- 4A. Soubor je korektní a dojde k importu jednotlivých záznamů - objednávek. Průběh importu bude znázorněn na liště s průběhem.
5. Systém vypíše sumář pro provedený import, ve kterém bude uvedeno:
 - kolik bylo záznamů v souboru celkem a
 - kolik záznamů bylo vloženo do databáze.

Následné podmínky:

Importované objednávky jsou uloženy v systému a mají nastaven stav na „čeká“.

Alternativní toky:

- 4B. Formát souboru není korektní, aktér je informován o nekorektnosti importního souboru.

Jméno případu použití: Přidání inzerátu

ID: 19

Stručný popis:

Případ použití slouží k umístění inzerátu do stránky makety.

Primární aktéři:

Správce

Šéfredaktor

Předpoklady:

Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. Je otevřena právě jedna maketa stránky.

Hlavní tok:

1. Případ použití se spustí, když aktér klikne na volbu „Přidat inzerát“.
2. Uživatel táhnutím myši ve směru shora dolů a zleva doprava vytvoří obdélníkovou plochu v maketě stránky.
3. Po puštění tlačítka myši dojde k trvalému umístění inzerátu v maketě.
4. Systém spustí případ použití „Změna inzerátu“.

Následné podmínky:

Inzerát je umístěn v maketě stránky.

Jméno případu použití: Změna inzerátu
ID: 20
Stručný popis: Případ použití slouží ke změně umístění, rozměrů inzerátu a k přiřazení objednávky k inzerátu. Pokud je k inzerátu přiřazena objednávka, je stav této objednávky nastaven na „vyřízena“.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. Je otevřena právě jedna maketa stránky a v ní označen jeden inzerát.
Hlavní tok: 1. Případ použití se spustí buď kliknutím na volbu „Změnit inzerát“ za současně označeného právě jednoho inzerátu nebo poklepaním na vybraný inzerát. Případ použití je rovněž spuštěn bezprostředně po případu užití „Přidání inzerátu“. 2. Systém zobrazí dialogové okno pro změnu atributů inzerátu. 3. Po kliknutí na tlačítko „Uložit“ systém provede validaci údajů. 4A. Systém uloží nové atributy inzerátu.
Následné podmínky: Inzerát je uložen v databázi s novými atributy. V maketě stránky se promítnou změněné atributy (rozměry, pozice ve stránce).
Alternativní toky: 4B. Zadané údaje nejsou validní nebo některé potřebné údaje chybí. Aktér je upozorněn a vyzván k opravě zadaných údajů.

Jméno případu použití: Smazání inzerátu
ID: 21
Stručný popis: Případ použití slouží ke smazání vybraného inzerátu v maketě.
Primární aktéři: Správce Šéfredaktor
Předpoklady: Případ použití je spuštěn správcem nebo šéfredaktorem příslušné oblasti. Je otevřena právě jedna maketa stránky a vybrán právě jeden inzerát.
Hlavní tok: 1. Případ použití se spustí, když aktér klikne na volbu „Smazat inzerát“. 2. Systém nabídne potvrzovací dialog pro smazání inzerátu. 3. Po kliknutí na tlačítko „Ano“ dojde ke smazání vybraného inzerátu.
Následné podmínky: Inzerát je odstraněn z databáze. Pokud byla k inzerátu přiřazena objednávka, je pouze zrušena vazba k mazanému inzerátu. Stav této objednávky je změněn na „čeká“.

3.3 Obecné požadavky

V této části uvádím další požadavky na systém, které upřesňují nebo doplňují případy použití.

3.3.1 Autentizace uživatelů

Uživatelé všech čtyř typů budou identifikováni jednoznačným přihlašovacím jménem (dále jen „login“), které je zvoleno při přidání uživatele do systému. Uživatel se do systému přihlašuje pomocí dvojice login a heslo. Heslo musí být neprázdným řetězcem o minimální délce 5 znaků. Každý uživatel v systému si bude moci změnit své heslo.

3.3.2 Hromadný import objednávek do systému

Z účetního softwaru Pohoda 2007 je proveden hromadný export vybraných objednávek. Export z účetního systému je ve formě prostého strukturovaného textu – CSV, kde každý řádek reprezentuje jednu objednávku. Počet i pořadí sloupců bude vždy stejné a pevně dané: Doklad, Datum, Číslo, Cena, Firma, Jméno, Ulice, Obec, PSČ, IČ, Telefon, Fax, Středisko, Činnost, Číslo (kód) oblasti, Poznámka.

Pole „Poznámka“ obsahuje informace ve formátu:

p/SxV\$POPIS\$X,

kde:

- S = šířka objednaného inzerátu,
- V = výška objednaného inzerátu,
- POPIS = popisná, upřesňující informace o poloze na stránce a barevnosti,
- X = zkratka čísla – nebere se v potaz, oblast je určena polem „Číslo oblasti“.

Systém by měl mít možnost vybrat soubor pro import a v průběhu importu objednávek zobrazovat jeho průběh.

3.3.3 Práce s grafickou maketou

Uživatel bude pracovat vždy s jedním číslem oblasti. Pokud bude chtít pracovat s jiným číslem oblasti, musí aktuální otevřenou maketu zavřít. Zobrazena bude vždy jedna stránka makety. Mezi stránkami se bude možné pohodlně přepínat (např. roletkou s výběrem všech dostupných stránek čísla novin).

Práce s inzeráty musí být intuitivní a snadno ovladatelná. Změnu rozměrů inzerátu a jeho přesun bude možné provést graficky – uchopením a tažením myši a také bude možné rozměry a umístění změnit ve formuláři pro editaci inzerátu. Další pomůckou pro práci s maketou bude možnost definovat tzv. vodící linky, ke kterým se budou jednotlivé inzeráty svými hranami přichycovat.

Maketa bude poskytovat specifické, práci usnadňující možnosti:

- označení a odznačení inzerátu ve stránce,
- označení a odznačení vodící linky,
- přesun označené linky nebo inzerátu za pomoci uchopení a tažení myši,
- přidání inzerátu pomocí tažení myši,
- zobrazení editačního formuláře inzerátu poklepáním na grafický objekt inzerátu,
- zobrazení editačního formuláře vodící linky poklepáním na vodící linku.

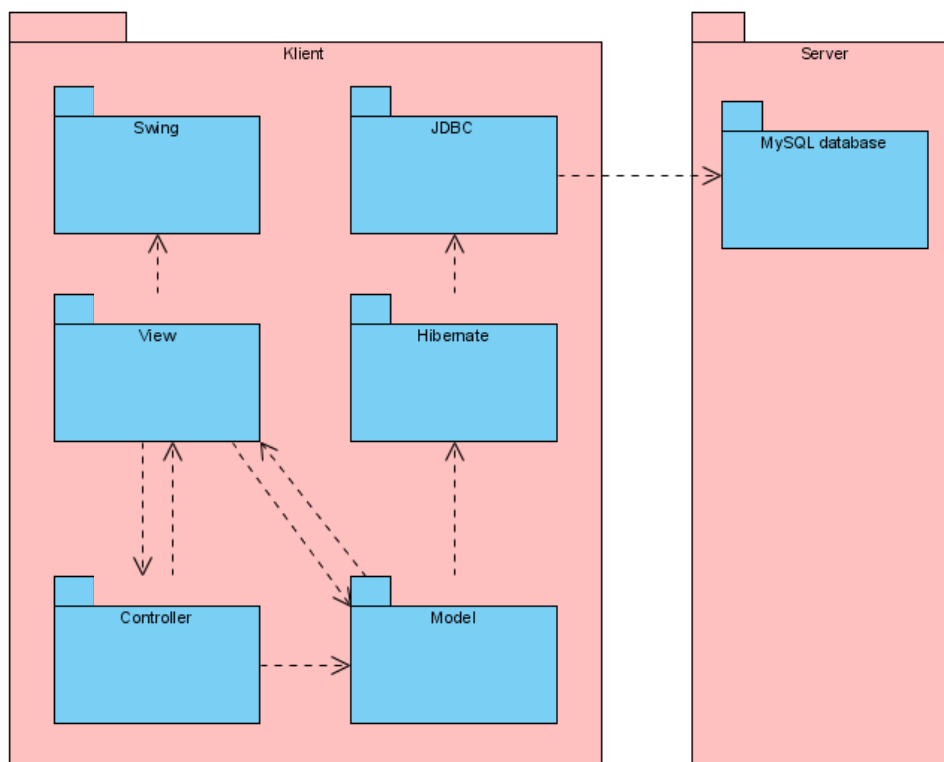
Inzerátům bude možno přiřadit barvu, kterou se pak vykreslí plocha inzerátu v maketě. Barvy budou předdefinovány napevno (alespoň 5 barev).

4 Návrh systému

V této kapitole popisují návrh informačního systému na základě požadavků, které byly popsány v předchozí kapitole. Při návrhu systému také využívám znalostí o procesech ze své několikaleté spolupráce s reklamní agenturou zadavatele a také využívám poznatků získaných při konzultacích a rozpravách se zaměstnanci a budoucími uživateli aplikace.

4.1 Návrh architektury

Návrh architektury systému jednoznačně udává směr celého dalšího vývoje. Úkolem správného návrhu architektury je navrhnout základní kostru a další aspekty aplikace tak, aby splňovala všechny funkční i nefunkční požadavky a aby byl vývoj této aplikace přímočarý a ubíral se nejjednodušší cestou (viz [1]).



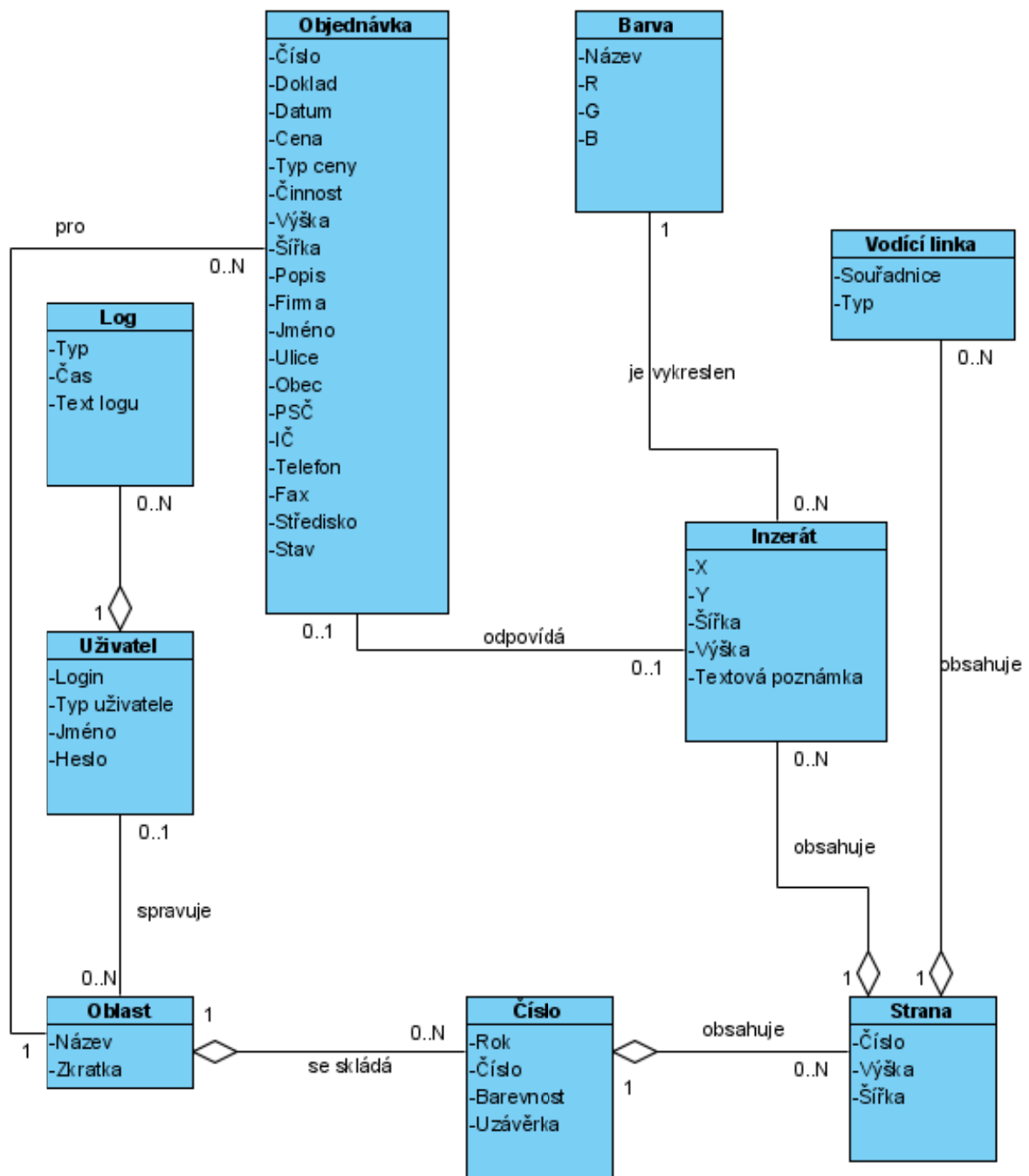
Obrázek 4.1 - Jednotlivé vrstvy architektury aplikace

V dnešní době jsou velmi rozšířenou architekturou systémy na bázi webových systémů s klientem ve formě internetového prohlížeče. Tyto systémy poskytují komplexní řešení architektury klient-server. Avšak právě prezenční vrstva mého systému je stěžejním místem pro splnění účelu aplikace. Je zapotřebí zajistit snadné a intuitivní ovládání makety novin, přesun reklamních ploch a změnu rozměrů pomocí tažením myši. Proto jsem se rozhodl pro tvorbu vlastního klienta v programovacím jazyce Java s použitím grafického frameworku Swing a celý systém postavit na architektuře s tzv. tlustým klientem (viz [2]) a s užitím návrhového vzoru MVC (Model-view-controller).

Navržená architektura je znázorněna v diagramu (viz Obrázek 4.1). Šipky znázorňují závislost jednotlivých vrstev. Jednotlivé vrstvy MVC jsou v diagramu reprezentovány komponentami Model, View a Controller.

4.2 Návrh struktury dat

Pro perzistenci dat jsem se rozhodl použít framework Hibernate. Mapování jednotlivých entit na databázi je realizováno prostřednictvím anotací v kódu. Strukturu datového modelu v aplikaci popisuje diagram tříd (viz Obrázek 4.2). Návrh databáze pak koresponduje s tímto diagramem.



Obrázek 4.2 - Diagram tříd

4.2.1 Popis jednotlivých entit

V návaznosti na diagram tříd popisují jednotlivé entity, které jsou v něm uvedeny.

Log

Log je záznam, prostřednictvím něhož je zaznamenána zmínka o provedené akci konkrétního uživatele. Jednotlivé atributy jsou:

- *Čas* – udává datum a čas zaznamenané akce,
- *Typ* – určuje kategorii logu. V budoucnu bude možné v aplikaci procházet jednotlivé záznamy a filtrovat je také podle typu. Možné hodnoty atributu *typ* jsou:
 - LOGIN – záznam o přihlášení uživatele do systému,
 - OBJEDNAVKY – záznam o přidání, smazání nebo editaci objednávky,
 - MAKETY – otevření či zásah do maket některého z čísel oblasti novin,
 - SPRAVA – ostatní operace správy systému (například přidání, mazání a editace uživatelů, oblastí apod.).
- *Text logu* – upřesňuje provedenou operaci (například „Smazání uživatele novakm.“)

Uživatel

Do systému mají přístup pouze autentizovaní uživatelé. Tabulka databáze pak tedy bude obsahovat všechny uživatele systému. Entita uživatel obsahuje atributy:

- *Login* – (uživatelské jméno) jednoznačný identifikátor, který lze však v rámci všech uživatelů měnit. Uživatelské jméno se může skládat pouze z písmen ASCII tabulky znaků.
- *Heslo* – textový atribut obsahující MD5 hash hesla uživatele,
- *Typ uživatele* – zajišťuje příslušnost k jedné ze skupin uživatelů.

Oblast

Oblast koresponduje s územními celky (bývalé okresy), ve kterých pak vycházejí jednotlivá čísla novin. Atributy jsou:

- *Zkratka* – označuje danou oblast a slouží k textovému doplnění čísla oblasti (například pro zkratku UO je číslo 1 z roku 2008 identifikováno štítkem „UO 1/2008“),
- *Název* – udává textovou informaci o dané oblasti (například „Ústí nad orlicí“).

Číslo

Číslem se rozumí jednotlivé konkrétní vydání novin v příslušné oblasti. Atributy čísla jsou:

- *Rok* – celočíselný atribut, udávající informaci o roku, ve kterém číslo novin vyšlo,
- *Číslo* – atribut, který udává pořadové číslo novin v konkrétním roce,
- *Barevnost* – udává informaci o barevnosti konkrétního čísla. Může nabývat výčtových hodnot:
 - CB – číslo je černobílé,
 - BARVA – číslo je celobarevné,
 - BARVA_CAST – číslo je částečně barevné (barevná může být například jen svrchní obálka novin).

Strana

Každé číslo může obsahovat libovolné množství stránek. Atributy každé stránky jsou:

- *Číslo stránky* – pořadové číslo stránky v daném vydání (čísle) oblasti novin,
- *Výška* – výška stránky v cm,
- *Šířka* – šířka stránky v cm.

Vodící linka

Vodící linka je pomůckou při práci s grafickou maketou čísla novin. K vodící lince se přichytávají inzeráty při tvorbě, přesunu nebo změně rozměrů. Jednotlivými atributy jsou:

- *Typ* – udává typ vodící linky. Může nabývat hodnot:
 - *VODOROVNA* pro vodorovnou a
 - *SVISLA* pro svislou vodící linku.
- *Souřadnice* – udává pozici vodící linky vzhledem ke stránce. Pokud jde o vodorovnou vodící linku, je souřadnicí Y souřadnice stránky. V opačném případě se jedná o souřadnici ve vodorovném směru X.

Inzerát

Inzerát reprezentuje reklamní plochu v maketě čísla. Inzerát je umístěn na konkrétní stránce. Atributy inzerátu jsou:

- *X* – vodorovná souřadnice vzhledem ke stránce,
- *Y* – svislá souřadnice vzhledem ke stránce,
- *Šířka* – šířka inzerátu v cm,
- *Výška* – výška inzerátu v cm,
- *Textová poznámka* – umožňuje šéfredaktorovi vepsat doplňkové informace o dané reklamní ploše.

Barva

Inzerát je v maketě vykreslen určitou barvou. Barvy jsou v systému předdefinovány z doby instalace a vytvoření databáze. Uživatel pak vybírá barvu, coby atribut inzerátu, pomocí roletky. Atributy barvy jsou:

- *Název* – text v roletce výběru barvy u editace atributů inzerátu,
- *R* – červená složka barvy,
- *G* – zelená složka barvy,
- *B* – modrá složka barvy.

Objednávka

Objednávka reprezentuje v systému požadavek na jedno konkrétní umístění inzerátu v maketě pro konkrétní oblast. Avšak blíže nespecifikuje číslo oblasti, ve kterém se má objednaná inzertní plocha nacházet. Objednávka obsahuje atributy:

- *Číslo* – textová identifikace, která musí být uchována pro spojení s dalším účetním softwarem, který se v reklamní agentuře používá,
- *Doklad* – celočíselný atribut, který koresponduje s variabilním symbolem pro platbu – číslem faktury,
- *Datum* – určuje datum vytvoření objednávky,
- *Cena* – hodnota objednávky v Kč na faktuře,
- *Typ ceny* – udává typ platby faktury. Podle typu lze pak objednávky třídit. Může nabývat hodnot:

- *PLATBA* – platba faktury,
- *RECIPROCE* – klient objednávky platí formou reciproce.
- *Činnost* – několikaznakový kód identifikující typ zakázky v externím účetním systému,
- *Výška a šířka* – udávají objednané rozměry plochy v cm,
- *Popis* – upřesňuje a udává bližší požadavky na umístění. Tyto informace jsou důležité pro šéfredaktora, který sestavuje maketu novin.
- *Firma, Jméno, Ulice, Obec, PSC, IČ, Telefon a Fax* – jsou popisnými atributy, které dávají informaci o objednateli z faktury.
- *Středisko* – je další popisnou informací pro vnitřní potřeby reklamní agentury.
- *Stav* – určuje, v jakém stavu se objednávky nachází. Může nabývat hodnot:
 - *CEKA* – objednávka čeká na vyřízení,
 - *VYRIZENA* – objednávka byla vyřízena (umístěna v maketě),
 - *STORNO* – objednávka byla stornována.

5 Implementace a testování

Po analýze požadavků, jejich specifikaci a navazujícím návrhu byl celý systém implementován. V této kapitole popisují použité technologie, uvádím stěžejní části implementace a pomocí obrazovek aplikace umožňují čtenáři získat představu o podobě výsledné aplikace. Dále se zmiňují o výsledcích a prvotních zkušenostech uživatelů s aplikací.

5.1 Použité technologie

V této podkapitole uvádím a stručně charakterizuji jednotlivé technologie, které jsem použil při vývoji software pro správu reklamních ploch v tiskovinách.

5.1.1 Programovací jazyk Java

Snad nejdůležitějším krokem celé implementace je správné zvolení implementačních prostředků, zejména programovacího jazyka.

Zrození jazyka

Podle [4 a 5] lze prvopočátky Javy datovat do roku 1990, kdy byl na podnět viceprezidenta firmy Sun Microsystems Billa Joe sestaven tým s názvem Green Team, který měl za úkol pracovat na vývoji systému pro domácí spotřebiče. Vše pod vedením Jamese Goslinga.

Po neúspěšných pokusech s programovacími jazyky C/C++ a Pascal nakonec James Gosling navrhl nový jazyk, který pojmenoval „Oak“ (což znamená dub). Gosling pojmenoval tento jazyk podle stromu, který rostl pod jeho kanceláří.

Jelikož původní záměr byl vyvinout systém pro domácí spotřebiče, což znamenalo různé architektury systémů s různými výkony, byl nový jazyk navržen jako jazyk interpretovaný, tj. nezávislý na architektuře.

Jazyk Oak později našel uplatnění na poli Internetu jako nová technologie přinášející možnost využití appletů. Podpora spouštění appletů byla postupně přidávána do většiny internetových prohlížečů přímo, nebo prostřednictvím zásuvných modulů.

V roce 1995 bylo zjištěno, že jazyk s názvem „Oak“ již existuje, proto byl nový jazyk týmu Green Team přejmenován na „Java“. Oficiálně byla Java představena 23. května 1995 na konferenci SunWorld. Technologie pak byla licencována společnostmi IBM, Microsoft, Symantec, Silicon Graphics, Inprise (Borland), Oracle, Toshiba a Novell. Roku 1996 byla vydána první verze JDK (Java Development Kit) 1.0.

Charakteristika jazyka

Podle [4, 5 a 6] je Java programovací jazyk:

- *Jednoduchý*, jedním z cílů Green Teamu bylo vytvořit jazyk, který se lze snadno naučit. Toho bylo dosaženo odstraněním různých konstrukcí jazyka C++.
- *Objektově orientovaný*. Java plně respektuje principy objektově orientovaného návrhu a programování. Veškerý kód musí být vždy zapouzdřen do tříd.
- *Robustní*, určený pro programy, které vyžadují vysokou spolehlivost.

- *Nezávislý na architektuře.* Programovací jazyk Java je interpretovaný. Výsledné programy v Javě jsou distribuovány v tzv. „bytecode“ a při běhu interpretovány JVM (Java virtual machine). To zajišťuje programům snadnou *přenositelnost*.
- Při vývoji aplikací v Javě je práce programátora ulehčena o správu paměti. O správu paměti, alokování, přidělování a uvolňování se stará *Garbage collector*. Tato řízená správa však vyžaduje značnou režii. Podle [5] je dnes tento problém vyřešen použitím samostatného vlákna, ve kterém Garbage collector běží s nízkou prioritou.
- *Bezpečný,* provádějící typovou kontrolu, ošetření výjimek a kolekcí již v době kompilace programu.

Java obsahuje ve svém API (Java Programming Interface) velké množství tříd, které umožňují navrhovat a tvořit programy na vysoké úrovni. Spousta dalších tříd je na internetu volně ke stažení jako Open source (software s otevřenými zdrojovými kódy). Programovou dokumentaci lze snadno generovat z kódu za použití JavaDocu (viz [8]).

Programovací jazyk Java je v IT komunitě rozšířen a velmi oblíben pro jeho výše popsané vlastnosti a dostupnost¹. Využívá se především k vývoji Enterprise aplikací². Na Internetu je k dispozici spousta webových stránek, fór a diskuzí, které se zabývají vývojem aplikací v tomto jazyce.

5.1.2 Framework Hibernate

Hibernate je jedním z frameworků pro jazyk Java, který umožňuje mapování mezi objektovým a relačním světem. Podle [7] při relačním přístupu k databázi je zhruba 70 procent času stráveno psaním kódu zajišťující persistenci objektů. Na domovských stránkách Hibernate (<http://www.hibernate.org>) se uvádí dokonce až 90 procent.

Objektově relační mapování (dále jen ORM) je technikou mapování dat, která jsou reprezentována třídami, na relační struktury reprezentované tabulkami v databázi. Pro získávání dat z databáze je k dispozici vlastní dotazovací jazyk HQL (Hibernate Query Language), jež je syntakticky podobný jazyku SQL.

Hibernate umožňuje mapování objektů zhruba do 20 různých relačních databází. Jde o relační databáze Oracle 9/10g, DB2, DB2 AS/400, DB2 OS390, MySQL, PostgreSQL, Sybase, Sybase Anywhere, SAP DB, HypersonicSQL, Microsoft SQL Server, Informix, FrontBase, Ingres, Progress, Mckoi SQL, Pointbase, Interbase a FireBird.

Hibernate lze použít v různých aplikačních architekturách, jako jsou desktopové či webové aplikace založené na J2EE.

Přístupy k návrhu aplikace a mapování

Při návrhu aplikace v prostředí Hibernate existují čtyři přístupy:

- *Top-down* – z naimplementovaných Java Beans vytvoříme XML mapovací dokument. Poté vygenerujeme schéma relační databáze.

¹ Pro Javu je zdarma dostupných mnoho vývojových prostředí (IDE), aplikačních serverů a knihoven. Mezi nejznámější vývojová prostředí patří NetBeans, Eclipse, Borland JBuilder a Oracle JDeveloper. Z aplikačních serverů můžeme jmenovat Apache Tomcat, JBoss, IBM WebSphere, Apache Geronimo, GlassFish a další.

² Enterprise aplikace jsou myšleny jako webové služby, webové aplikace, Enterprise Java Beans a technologie pro správu a bezpečnost.

- *Bottom-up* – jde o opačný přístup. Ze schématu relační databáze je vygenerován mapovací dokument. Z tohoto dokumentu je dále vygenerován skeleton Java kódu, který programátor doplní o business logiku.
- *Middle-out* – začneme vytvořením mapovacího dokumentu. Z něho poté vygenerujeme schéma relační databáze a skeleton Java kódu.
- *Meet-in-the-middle* – jedná se o nejtěžší přístup. Ručně nebo pomocí nástrojů vytvoříme mapovací dokument z již vytvořených Java Beans a relační databáze.

Mapování můžeme provést dvěma způsoby:

- mapovacím xml souborem
- nebo anotacemi v kódu.

Příklady mapování pomocí xml souboru a příklad mapování pomocí anotací v kódu jsou uvedeny v příloze. Při vývoji mé aplikace jsem využil návrhový top-down model s mapováním pomocí anotací v kódu.

5.1.3 Databáze MySQL

MySQL je databázový systém, který pochází z dílny švédské firmy MySQL AB z roku 1995. Jeho hlavními autory jsou (podle [3]) Michael „Monty“ Widenius a David Axmark. MySQL je multiplatformní databáze, komunikace s ní probíhá prostřednictvím jazyka SQL. Existují verze pro Windows, Linux a jiné operační systémy (viz [3]).

Podle [9] bylo MySQL zpočátku optimalizováno především pro rychlost za cenu některých zjednodušení³. Ovšem omezení, která tato zjednodušení vytvářela, začalo postupně stále více chybět webovým a jiným vývojářům používajícím MySQL.

Systém MySQL využívá hned několik typů úložišť dat - tabulek, které si uživatel může při vytváření databáze v prostředí MySQL zvolit. Jednotlivé typy se liší možnostmi a způsobem ukládání dat do souborů. Mezi podporované typy tabulek patří:

- MyISAM – nejpoužívanější, bez podpory transakcí,
- InnoDB – s podporou transakcí,
- BerkeleyDB,
- MEMORY – používá pouze paměť pro práci s daty,
- NDB Cluster – úložiště pro clusterované databáze (od verze 5.0),
- ARCHIVE – komprimované tabulky, není zde podpora indexů,
- CSV – ukládání dat v textových souborech.

Aktuální verze MySQL má číslo 5.1 a byla vydána v prosinci 2005. Nynější verze MySQL jsou již vyvíjeny firmou Sun Microsystems, která MySQL od švédské firmy odkoupila v prosinci roku 2008.

5.2 Popis implementace

V této podkapitole uvádím stěžejní části implementace software a popisuji strukturu jednotlivých balíčků a tříd.

³ Nižší verze MySQL nepodporovaly cizí klíče a transakce (podporovány od verze 3.23 v tabulkách InnoDB). Poddotazy byly možné až od verze 4.1. Od verze MySQL 5.0 se uživatelé dočkali podpory pohledů, triggerů a uložených procedur.

5.2.1 Struktura aplikace

Celý program je rozdělen podle návrhového vzoru MVC do následujících balíčků:

- *MaketaNovin* – hlavní balík aplikace.
- *MaketaNovin.APP* – obsahuje třídy starající se o logiku aplikace (vrstva *Controller*).
- *MaketaNovin.Entity* – obsahuje třídy entit s mapováním na relační databázi (vrstva *Model*).
- *MaketaNovin.GUI* – obsahuje dialogová okna a formuláře pro zobrazování dat a interakci s uživatelem (vrstva *View*).
- *MaketaNovin.GUI.GO* – obsahuje třídu pro zobrazování a práci s grafickou maketou.
- *MaketaNovin.GUI.Resource* – obsahuje grafické ikony pro tlačítka a hlavičky oken.
- *MaketaNovin.Util* – obsahuje pomocné třídy, které jsou využívány na jiných místech programu.
- *MaketaNovin.Util.Calendar* – obsahuje třídy pro grafický kalendář.

V následujících podkapitolách je uveden souhrn všech tříd rozdělených podle výše zmiňovaných balíčků.

Balík MaketaNovin

Main.java	Vstupní bod programu. Nastavuje globální vzhled oken a předává řízení třídě <i>SessionManager</i> .
Nastaveni.java	Obsahuje globální nastavení celé aplikace a různé konstanty.

Tabulka 5.1 - Třídy balíku MaketaNovin

Balík MaketaNovin.APP

Balík *MaketaNovin.APP* sdružuje třídy, které reprezentují vrstvu *Controller* návrhového vzoru MVC.

MaketaManager.java	Třída poskytující operace pro práci s grafickými objekty ve stránce (inzerát, vodící linka) a pro práci s otevřenou maketou čísla novin.
ManazerCisel.java	Třída poskytuje operace pro práci s čísly oblastí novin.
ManazerObjednavek.java	Třída poskytuje operace pro práci s objednávkami. Obsahuje také privátní třídu pro zpracování CSV souboru ve vláknu pro import objednávek.
ManazerStranek.java	Třída poskytuje operace pro práci s jednotlivými stránkami v číslech novin.
ManazerUzivatelu.java	Třída poskytuje operace pro správu uživatelů v systému.
SessionManager.java	Třída, jejíž instance se při běhu aplikace stará o aktuální relaci a přihlášeného uživatele. Nastavuje také práva k jednotlivým částem programu podle typu uživatele.
myValidator.java	Třída obsahující metody pro validaci údajů ve formulářích.

Tabulka 5.2 - Třídy balíku MaketaNovin.APP

Balík MaketaNovin.Entity

Balík *MaketaNovin.Entity* sdružuje třídy, které reprezentují vrstvu *Model* MVC návrhového vzoru MVC. Obsahuje třídy entit podle diagramu tříd, pomocné třídy a rozhraní pro ukládání, mazání a výběr jednotlivých entit. Třídy entit jsou popsány v kapitole 4.2.1, proto uvádím pouze jejich

seznam: *Barva.java*, *Cislo.java*, *Inzerat.java*, *Log.java*, *Objednavka.java*, *Oblast.java*, *Stranka.java*, *Uzivatel.java*, *VodiciLinka.java*.

Protože v některých dialogových oknech a formulářích je zapotřebí tzv. nulového objektu, kdy je nutné mít možnost definovat hodnotu „nerozhoduje“ (například u filtru při výpisu objednávek), obsahuje balík *MaketaNovin.Entity* ještě tři pomocné třídy. Jsou jimi: *NullObjednavka.java*, *NullOblast.java*, *NullUzivatel.java*.

Třída *DBManazer.java* odvozená od *DBManazerBean.java* zprostředkovává pro všechny entitní třídy operace pro persistenci (uložení, smazání, výběr na základě různých parametrů).

Balík *MaketaNovin.GUI* a *MaketaNovin.GUI.GO*

Balík *MaketaNovin.GUI* sdružuje třídy, které reprezentují vrstvu *View* návrhového vzoru MVC. Popis tříd je uveden v tabulce (viz Tabulka 5.3).

AboutDialog.java	Třída reprezentující dialog „O programu“.
AnoNeDialog.java	Potvrzující dialog s volbami <i>Ano</i> a <i>Ne</i> .
CisloDialog.java	Třída reprezentující formulář pro přidání a editaci čísla oblasti novin.
ImportCSVDialog.java	Okno pro zahájení importu objednávek z CSV souboru.
InfoDialog.java	Informační dialog pro zobrazování chybových a jiných informačních zpráv.
InzeratDialog.java	Třída reprezentující formulář pro editaci inzerátu.
LoginDialog.java	Dialog pro přihlášení do aplikace.
mainWindow.java	Hlavní okno aplikace. Okno obsahuje hlavní menu a plochu pro umístění podoken.
MaketaFrame.java	Okno pro práci s grafickou maketou. Na tomto okně je umístěna plocha pro vykreslení stránky makety (třída <i>MyJPanel</i>).
myDialog.java	Rozhraní pro vlastní dialogová okna.
myTableModel.java	Rozšíření <i>AbstractTableModel</i> , použito při plnění tabulek s výpisem entit.
ObjednavkaDialog.java	Formulář pro zadání nové nebo editaci stávající objednávky.
ObjednavkyFrame.java	Okno s výpisem objednávek. Okno obsahuje také filtr umožňující nastavení různých parametrů výpisu.
OblastDialog.java	Formulář pro editaci oblasti novin.
OblastiFrame.java	Okno s výpisem oblastí novin.
OtevritMaketuDialog.java	Dialog pro otevření makety čísla.
PasswdDialog.java	Dialogové okno pro změnu hesla aktuálně přihlášeného uživatele.
StrankaDialog.java	Formulář pro přidání a editaci jednotlivých stránek v čísle oblasti.
StrankyDialog.java	Okno s výpisem stránek vybraného čísla.
UzivatelDialog.java	Dialog pro přidání nebo editaci uživatele.
UzivateleFrame.java	Okno s výpisem uživatelů.
VodiciLinkaDialog.java	Dialog pro přidání nebo editaci vodící linky ve stránce makety čísla.
WindowToolkit.java	Třída poskytující operace, využité v ostatních formulářích a dialogích.

Tabulka 5.3 - Třídy balíku *MaketaNovin.GUI*

V balíku *MaketaNovin.GUI.GO* je umístěna třída *MyJPanel.java*, která poskytuje grafickou plochu pro vykreslení jednotlivých stránek z otevřené makety čísla novin. Třída *MyJPanel* obsahuje také obsluhu událostí myši.

Balík MaketaNovin.GUI.Resource

Balík *MaketaNovin.GUI.Resource* obsahuje grafické objekty (ikony tlačítek, logo aplikace). Ikony pro tlačítka byly staženy z volně dostupného zdroje z URL <http://www.iconspedia.com>.

Balík MaketaNovin.Util

V balíku *MaketaNovin.Util* jsou umístěny třídy, které rozšiřují funkce vrstvy *Controller* a *View*.

DateUtil.java	Třída pro konverze mezi různými formáty data. Umožňuje převádět datum z řetězců do <i>Date</i> a naopak.
NumberUtil.java	Třída poskytující operace pro konverze čísel reprezentujících souřadnice, rozměry nebo finanční částky na řetězce a naopak.
SimpleMD5.java	Třída poskytující operaci pro výpočet MD5 hashe řetězců.

Tabulka 5.4 - Třídy balíku MaketaNovin.Util

Balík MaketaNovin.Util.Calendar

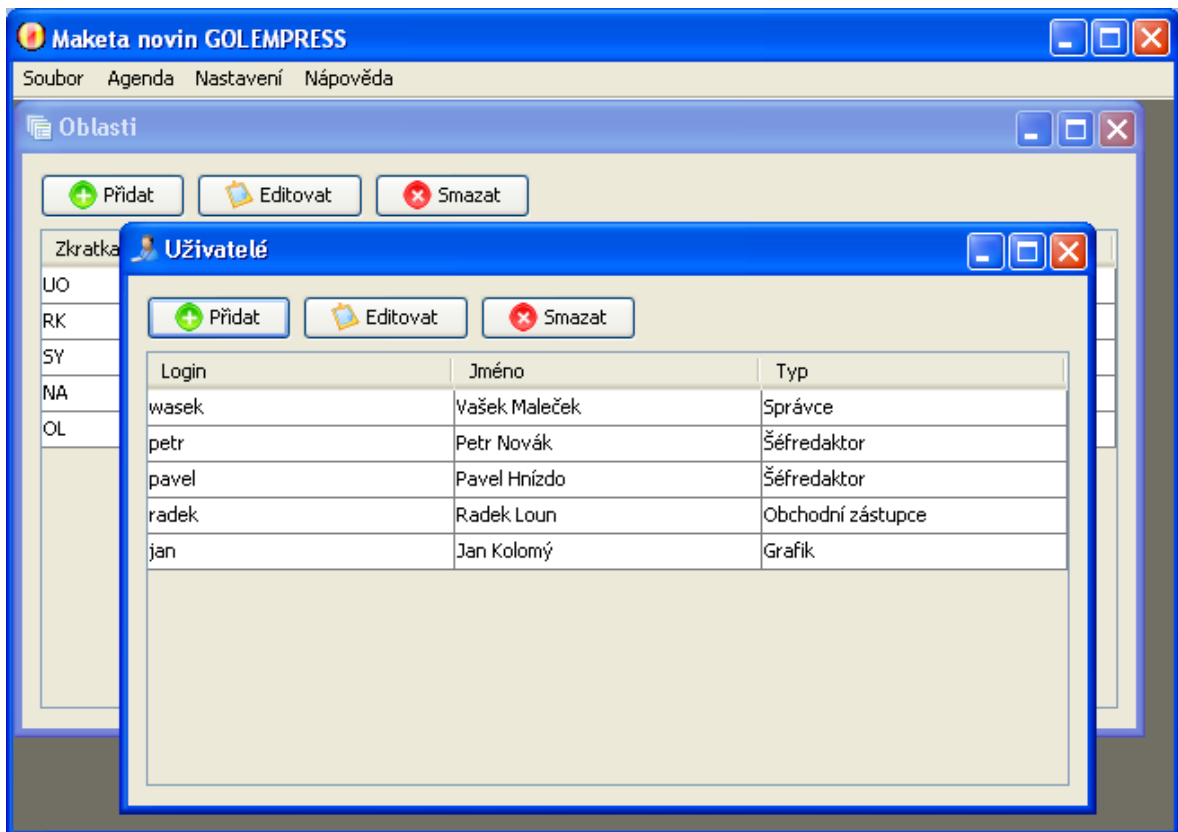
Balík *MaketaNovin.Util.Calendar* obsahuje komponentu pro zobrazení dialogu s výběrem data. Zdrojové kódy byly použity z volně dostupného webového zdroje na adrese <http://www.koders.com/zeitgeist/java>.

5.2.2 Hlavní okno aplikace

Po úspěšném přihlášení uživatele do systému je zobrazeno hlavní okno aplikace s vertikálním menu v horní části okna (viz Obrázek 5.1). Jednotlivé části aplikace jsou zprostředkovány modálními dialogovými okny nebo podokny třídy *JInternalFrame*. Aplikace tedy působí kompaktně a jednotně.

Hlavní menu aplikace obsahuje následující položky:

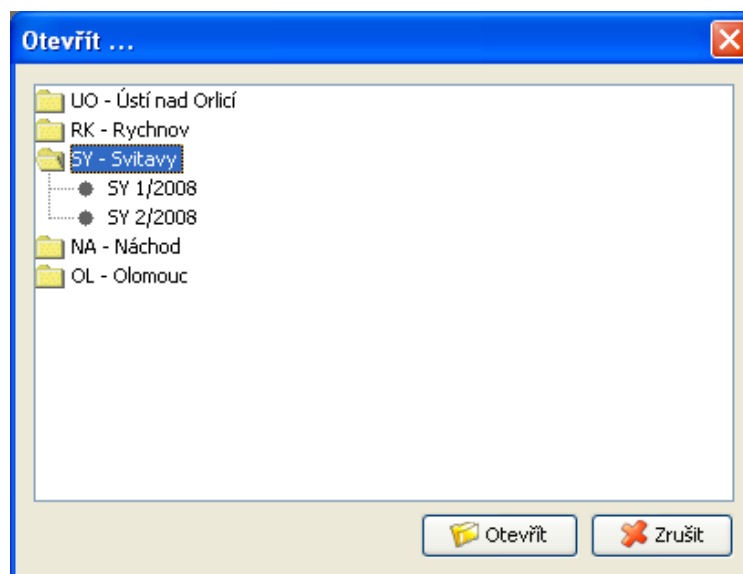
- Soubor
 - Otevřít maketu – Zobrazí dialog pro otevření konkrétní makety čísla.
 - Konec – Ukončí aplikaci.
- Agenda
 - Uživatelé – Otevře podokno pro správu uživatelů v systému.
 - Oblasti – Otevře podokno pro správu oblastí novin.
 - Čísla oblastí – Otevře podokno pro správu jednotlivých čísel podle oblastí novin.
 - Objednávky – Otevře podokno pro správu objednávek.
- Nastavení
 - Změna hesla – Umožní uživateli změnit heslo.
- Nápověda
 - O programu – Zobrazí dialogové okno s informacemi o určení a verzi programu.
 - Uživatelský manuál – Otevře uživatelský manuál ve formátu PDF.



Obrázek 5.1 - Hlavní okno aplikace

5.2.3 Otevření makety

Uživatel přes volbu z menu vyvolá dialogové okno s výpisem všech dostupných čísel. Jednotlivá čísla jsou sdružena do oblastí a zobrazena pomocí dvouúrovňové stromové struktury. K zobrazení stromu byla použita komponenta *JTree*, která je naplněna modelem typu *DefaultTreeModel* obsahující onu stromovou strukturu. Pro představu uvádím ukázkou dialogového okna pro otevření makety čísla (viz Obrázek 5.2).



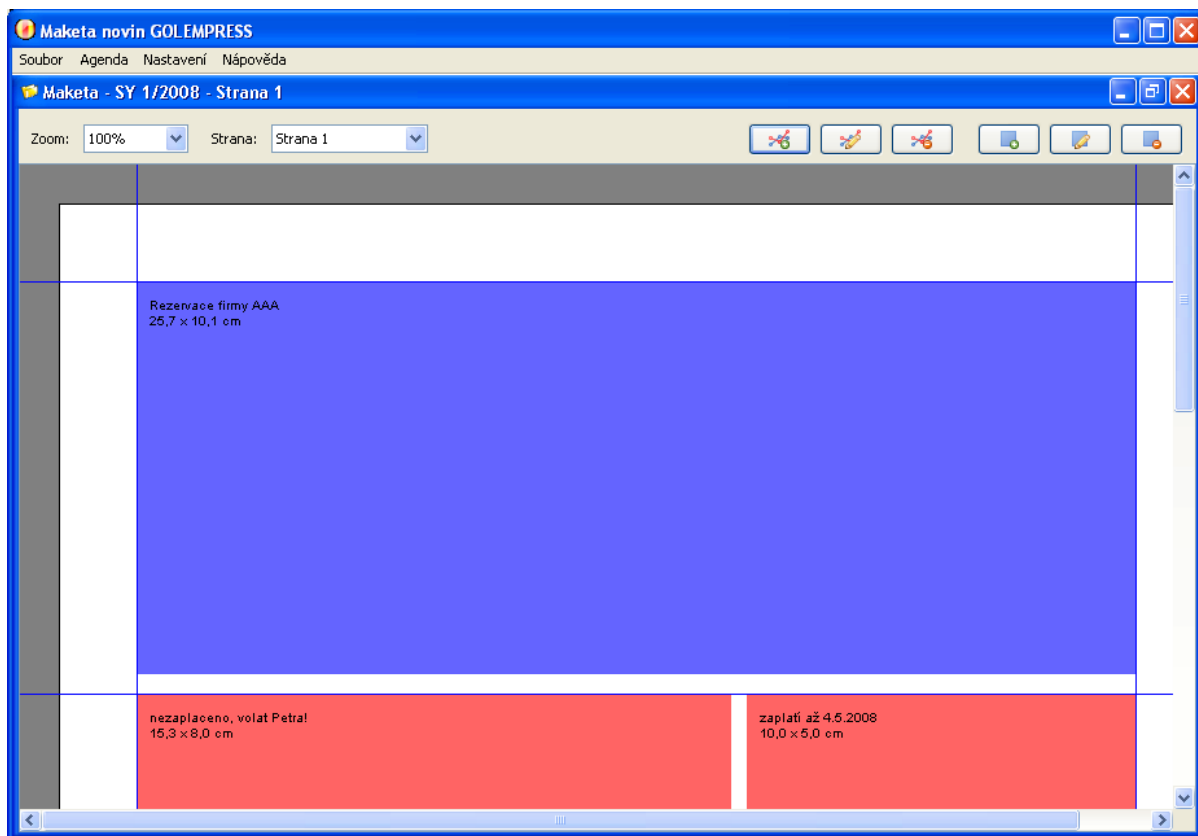
Obrázek 5.2 - Okno pro otevření makety čísla

5.2.4 Práce s maketou

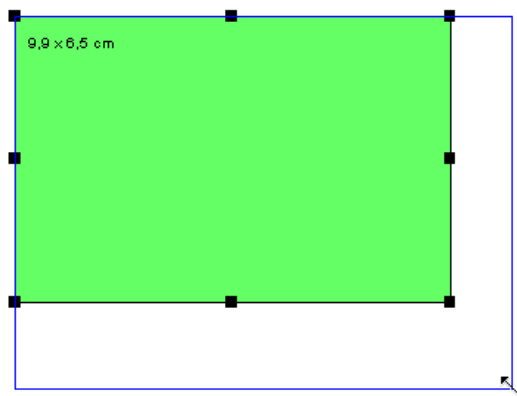
Při otevření makety je zobrazeno podokno s vybranou maketou. Lze se přepínat mezi jednotlivými stránkami makety čísla a volit přiblížení/oddálení. Vlastní stránka makety je pak zobrazována pomocí vlastní grafické třídy *MyJPanel* odvozené od *JPanel*. V třídě *MyJPanel* je implementována veškerá logika pro grafickou práci s inzeráty a vodíci linkami ve stránce. Třída *MyJPanel* umožňuje:

- Vytváření inzerátu pomocí tažení myši s přichytáváním k vodícím linkám.
- Přesun inzerátu tažením myši s přichytáváním k vodícím linkám.
- Změnu rozměrů vybraného inzerátu s přichytáváním k vodícím linkám.
- Označení inzerátu (zobrazení kontury a úchopových symbolů).
- Změnu kurzoru myši podle pozice a právě prováděné akce (například změnu kurzoru při přejetí nad vybraným inzerátem nebo úchopem).
- Označení vodící linky.
- Přesun označené vodící linky.

Možnost smazání vybrané vodící linky nebo inzerátu je zprostředkováno pomocí funkčních tlačítek v horní části podokna pro zobrazení makety. Na obrázcích (viz Obrázek 5.3 a Obrázek 5.4) je ukázka podokna s maketou stránky a ukázka změny rozměrů inzerátu pomocí myši.

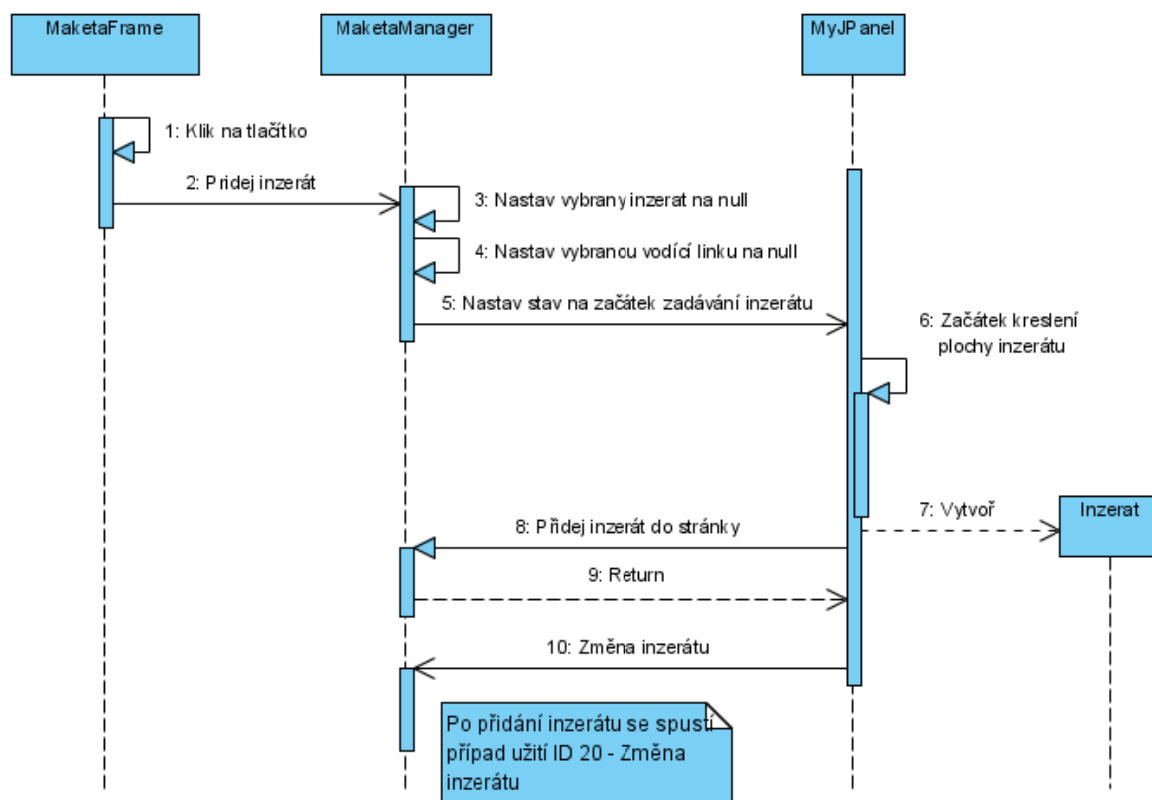


Obrázek 5.3 - Podokno pro práci s maketou



Obrázek 5.4 - Změna rozměrů inzerátu

Pro lepší pochopení sledů jednotlivých kroků při zadávání nové reklamní plochy (inzerátu) jsem vytvořil sekvenční diagram (viz Obrázek 5.5), který znázorňuje průběh a časování komunikace mezi jednotlivými třídami.



Obrázek 5.5 - Sekvenční diagram přidání inzerátu

Akce je spuštěna, pokud uživatel klikne na tlačítko „Přidat inzerát“ v okně pro práci s maketou (*MaketaFrame*). Událost způsobená kliknutím na tlačítko spustí metodu třídy *MaketaManager* pro přidání nového inzerátu. V této metodě se nastaví aktuální vybraný inzerát a vodič linka na *null* (není vybrán/a žádný/á). Poté se grafické třídě *MyJPanel*, která se stará o vykreslování grafických objektů a interakci s maketou, předá parametr, pro počátek zadávání inzerátu. Po stisknutí tlačítka myši dojde

k zahájení vyznačování plochy inzerátu. Uživatel tahem myši shora dolů a zleva doprava vyznačí plochu inzerátu. Po uvolnění tlačítka myši dojde k vytvoření objektu inzerátu s rozměry podle vybrané plochy. Následně je objekt inzerátu pomocí třídy *MaketaFrame* přidán do aktuální stránky a vložen do databáze.

5.2.5 Datový model pro výpis entit

Framework Swing má zabudovanou podporu pro MVC architekturu. Grafická komponenta *JTable* je plněna tzv. table modelem, což je objekt obsahující zobrazovaná data. Uživatel má možnost komponentu *JTable* naplnit dvěma způsoby:

- vytvořit instanci *DefaultTableModel*
- nebo vytvořit vlastní třídu, která dědí z třídy *AbstractTableModel*.

První možnost je jednoduchá a je vhodná tam, kde potřebujeme data pouze zobrazit v tabulce. Příklad na naplnění komponenty *JTable* v Java pseudokódu:

```
String [][] data = new String [colsCnt][rowsCnt];
for (...) {
    //naplnění objektu data
}
String [] colNames = new String [colsCnt] {"A", "B"};
DefaultTableModel tblModel = new DefaultTableModel(data, colNames);
```

Problém však nastane, pokud chceme pracovat s objektem, který koresponduje s vybraným řádkem v tabulce *JTable*. Potřebujeme mít vztah řádek tabulky – příslušný objekt. Pro tyto účely jsem vytvořil vlastní třídu *myTableModel* odvozenou od *AbstractTableModel*. Při vytváření tohoto modelu se zároveň k příslušnému řádku přiřadí daný objekt. Příklad při vytváření table modelu pro výpis uživatelů:

```
myTableModel model = new myTableModel(vectorOfColumnNames, 0);
for (...) {
    model.addRowWithObject(new String[]
        {uzivatel.getLogin(), uzivatel.getJmeno(), uzivatel});
}
```

5.2.6 Prohlížení objednávek

Prohlížení objednávek je realizováno třídou *ObjednavkyFrame*, která je odvozena od třídy *JInternalFrame*. Seznam objednávek je zobrazen v komponentě *JTable*, která je umístěna pod filtrem zobrazení (viz Obrázek 5.6).

Datum	Doklad	Číslo	Cena	TyCeny	Zákazník	Oblast	Stav
08.04.2008	1050027349-01	285103394	5940,00	Platba	KRC SRO & CO, k. s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	1050027349-02	285103395	5940,00	Platba	KRC SRO & CO, k. s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	1050027349-03	285103396	5940,00	Platba	KRC SRO & CO, k. s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	2250298030	285103399	9900,00	Platba	Milan Trnečka	SY - Svitavy	Čeká
08.04.2008	225098029-01	285103400	7260,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-02	285103401	7260,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-03	285103402	7260,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-04	285103403	7260,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-05	285103404	7500,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-06	285103405	7500,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-07	285103406	7500,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	225098029-08	285103407	7500,00	Platba	CEREBRUS-MOTO a.s.	UO - Ústí nad Orlicí	Čeká
08.04.2008	2051420622-01	285103434	1320,00	Platba	Alpo Czech s. r. o.	RK - Rychnov	Čeká
08.04.2008	2051420622-02	285103435	1320,00	Platba	Alpo Czech s. r. o.	RK - Rychnov	Čeká
08.04.2008	2051420622-03	285103436	1320,00	Platba	Alpo Czech s. r. o.	RK - Rychnov	Čeká
08.04.2008	1452040557	285103467	963,00	Platba	Aleš Facek	SU - Šumperk	Čeká
09.04.2008	1050027348-01	285103382	1980,00	Platba	AUTOLAKY - Josef Slán...	UO - Ústí nad Orlicí	Čeká
09.04.2008	1050027348-02	285103383	1980,00	Platba	AUTOLAKY - Josef Slán...	UO - Ústí nad Orlicí	Čeká

Obrázek 5.6 - Podokno s výpisem objednávek

Díky filtru si může uživatel zvolit podmínky, podle kterých se vypíše seznam objednávek. Může si zvolit období, ohraničené shora, zdola nebo shora i zdola zároveň. Dále má možnost si zobrazit objednávky pouze určitého stavu nebo objednávky, které jsou určeny pro určitou oblast.

Po kliknutí na tlačítko „Potvrdit filtr“ se zavolá metoda *aktualizuj()* třídy *ObjednavkyFrame*, která aktualizuje seznam objednávek. V této metodě se naplní komponenta *JTable* novým modelem získaným pomocí třídy *ManazerObjednavek*:

```
Seznam.setModel(ManazerObjednavek.getTableModel());
```

Metoda *getTableModel()* třídy *ManazerObjednavek* zajistí vytvoření modelu pro tabulku na základě zvolených hodnot filtru. Podle filtru je vytvořena mapa argumentů pro třídu *DBManazer*, která pomocí metody *List<Objednavka> getSeznamObjednavek(Map<String, Object> argumenty)* zajistí vrácení seznamu objednávek, vyhovujícímu zvolenému filtru. Pro dynamický počet argumentů je použito asociativní pole *Map<String, Object>*. Příklad použití:

```
ObjednavkyFrame of = ObjednavkyFrame.getInstance();
Map<String, Object> argumenty = new HashMap<String, Object>();
if (of.getDatumOd() != null) {
    argumenty.put("o.datum>=", of.getDatumOd());
}
```

S vytvořenými argumenty je zavolána metoda *getSeznamObjednavek* třídy *DBManazer*:

```
List<Objednavka> list = new
ArrayList<Objednavka>(DBManazer.getSeznamObjednavek(argumenty));
```


V těle metody *getSeznamObjednavek* je tvořen HQL (Hibernate Query Language) dotaz pro výběr objednávek s dynamickým počtem argumentů. V prvním kroku je vytvořen holý (bezargumentový) HQL dotaz:

```
String hql = "select o from Objednavka as o";
```

Dále se pro každý klíč v asociativním poli upraví řetězec *hql* (například pro pole datum vznikne „*select o from Objednavka as o where o.datum<=:arg0*“).

```
Set<String> keys = argumenty.keySet();
int argNum = 0;
for (String key : keys) {
    if (argNum == 0) {
        hql = hql + " where ";
    } else {
        hql = hql + " and ";
    }
    hql = hql + key + ":arg"+argNum;
    argNum++;
}
query = em.createQuery(hql);
```

Nyní již zbývá dosadit objekty za argumenty v celém dotazu. V cyklu *for* pro každý klíč asociativního pole dosadíme korespondující objekt a získáme výsledný seznam objektů:

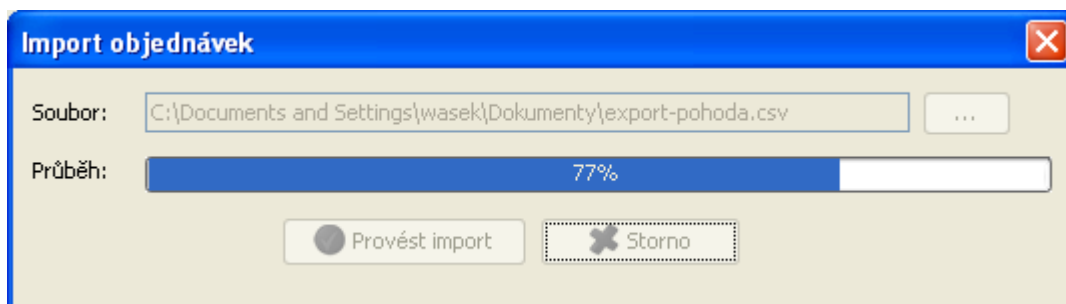
```
argNum = 0;
for (String key : keys) {
    query.setParameter("arg"+argNum, argumenty.get(key));
    argNum++;
}
list = query.getResultList();
```

5.2.7 Import objednávek z CSV souboru

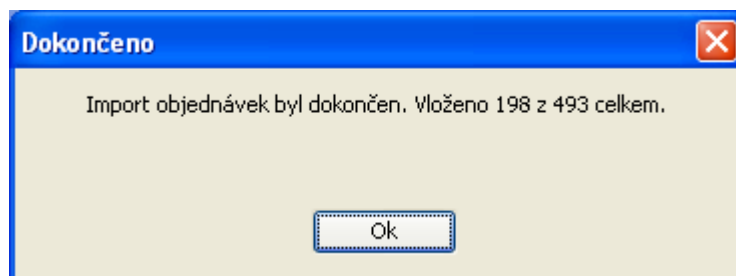
Zahájení importu souboru je realizováno modálním dialogem - instance třídy *ImportCSVDialog* (viz Obrázek 5.7). Po vybrání importního souboru a po kliknutí na tlačítko „Provést import“ se zneaktivní ovládací prvky na tomto dialogu a zavolá se metoda *importCSV_Proces* třídy *ManazerObjednave,ve* které se jako parametry předají název souboru a reference na instanci importního dialogu. V této metodě se otevře soubor pro čtení a pomocí instance *CSVReader* knihovny *OpenCSV* se načte veškerý obsah souboru do dvourozměrného vektoru řetězců. Při načítání je kontrolován importní soubor, zda všechny řádky obsahují přesně 16 polí.

Pokud byl soubor správně načten do paměti, zpracují se všechny řádky v samostatně běžícím vlákně, aby bylo možné uživatele informovat o průběhu zpracování. Běh vlákna zajišťuje třída *ImportCSVThread* v metodě *run*. V metodě *run* je každý záznam zpracován a vytvořena objednávka, která je uložena do databáze pomocí třídy *DBManazer*. Při jakékoliv chybě (např. při převodu řetězce na číslo či datum) je vyvolána výjimka a řádek je přeskočen.

Na konci importu je zobrazeno okno s informacemi o počtu vložených a nevložených řádků (viz Obrázek 5.8).



Obrázek 5.7 - Dialog pro import objednávek



Obrázek 5.8 - Sumář po ukončení importu objednávek

5.3 Použité knihovny a komponenty

Při implementaci aplikace jsem použil grafické ikony tlačítek z volně dostupné webové knihovny <http://www.iconspedia.com>.

Pro práci s CSV soubory jsem využil knihovnu *OpenCSV*, která je volně dostupná na adrese <http://opencsv.sourceforge.net>.

Pro výpočet MD5 hashe hesel využívám knihovny *SimpleMD5*, která je volně dostupná na adrese http://www.anyexample.com/programming/java/java_simple_class_to_compute_md5_hash.xml.

Poslední komponentou, kterou jsem použil z externího zdroje, bylo kalendářové okno pro výběr data. Třídy komponenty jsou volně dostupné na adrese <http://www.koders.com/zeitgeist/java>.

5.4 Nasazení a testování

Aplikace byla při dokončení implementace otestována. Byly otestovány všechny části aplikace a možné případy užití jak s validními, tak i nevalidními daty, kdy systém upozorňoval uživatele na chybu v zadaných údajích. Následně se aplikace nasadila u klienta a poskytla pro testování na reálných datech.

5.4.1 Testování na reálných datech

Zadavatel měl k dispozici první verzi aplikace, kterou testoval na reálných datech. Pro testování byla zvolena méně vytížená oblast reklamních novin, kterou měl na starosti právě jeden šéfredaktor. Šéfredaktor využíval pouze zadávání objednávek pomocí importu CSV souboru ze systému Pohoda 2007.

Dále bylo vytvořeno CD pro distribuci obchodním zástupcům, kteří aplikaci využívali vzdáleně ze svých domovů a z počítačů ve firmě. Celkem se testování aplikace zúčastnilo 5 obchodních

zástupců, kterým byly správcem (zadavatelem) vytvořeny uživatelské účty v aplikaci. Všichni aktéři absolvovali krátké školení, na kterém jim byla představena aplikace, především pak zadávání objednávek a princip práce s maketou novin. Rovněž byli uživatelé zaškoleni pro instalaci JRE (Java Runtime Environment), jehož instalace se také na CD nacházela, a informování o omezeních aplikace a požadavcích pro běh systému.

Při společné schůzce po týdnu testování aplikace „v terénu“ hodnotili uživatelé práci s aplikací za vskutku intuitivní a shledali práci se softwarem při procesu přípravy reklamních novin za ulehčení a zrychlení celého procesu.

5.5 Instalace a požadavky na prostředí

Na přiloženém nosiči CD-ROM jsou v adresáři *src* uložené zdrojové kódy, spustitelná aplikace a soubor *build.xml*. Pro překlad, spuštění a provoz je zapotřebí následující vybavení (jsou uvedeny programy a jejich verze):

- Java 2 Standard Edition Runtime Environment (JRE) verze 1.6.0_04 (na klientovi),
- MySQL verze 5.1 (na serveru),
- Apache Ant verze 1.7.0 (pro překlad zdrojových kódů).

Nastavení přístupu k databázi provedte v souboru *persistence.xml* v adresáři */src/META-INF/*. Poté přeložte⁴ zdrojové kódy pomocí programu *ant* z adresáře *src* se souborem *build.xml*.

Spustitelný program je v adresáři *test*. Přiložený spustitelný program je nakonfigurován pro lokální databázi:

```
Server: localhost
Databáze: dip
Uživatel: test
Heslo: test
```

Program spusťte z adresáře *test* příkazem „`java -jar MaketaNovin.jar`“. Databázi naplníte implicitními daty pomocí přiloženého skriptu *default_db.sql*.

⁴ Pozor, pro překlad je nutné mít zdrojové soubory programu z nosiče CD-ROM zkopírovány na disk počítače kvůli potřebě zápisu.

6 Plánovaná rozšíření

Po nasazení a testování aplikace zadavatelem byly uvolněny další finanční prostředky a schválen další vývoj tohoto nástroje pro organizaci a správu reklamní plochy v novinách. V průběhu testování vznikly požadavky na nové funkce a rozšíření.

6.1 Šablony stránek

Aplikace již nyní usnadňuje práci šéfredaktorům a šetří čas všem zainteresovaným business rolím. Je však potřeba zjednodušit práci při přidávání nových čísel do oblastí a stránek v těchto číslech. Požadavkem je vytvořit podsystém tzv. šablon stránek. Všechny stránky v systému (až na výjimky) mají stejné rozměry. Dalším bodem jsou vodící linky, které mohou vymezovat například netisknutelné okraje a předdefinované sloupce pro sazbu inzerátů. Šablony stránek by tedy obsahovaly už také předdefinované vodící linky.

Proces přidávání stránek by pak byl možný dvěma způsoby:

- pomocí stávajícího dialogového formuláře (uživatel zadává rozměry stránky přímo a v nové stránce nejsou žádné předdefinované vodící linky),
- pomocí šablon, kdy si uživatel zvolí, podle které šablony se má stránka vytvořit. Zároveň se do nově přidané stránky zkopírují vodící linky ze šablony.

Šablony stránek budou přístupné v rámci celého systému všem uživatelům. Tvorba šablony bude možná prostřednictvím stránky z makety. Pokud tedy uživatel bude mít otevřenou maketu nějakého čísla, tak z aktuální otevřené stránky bude moci vytvořit šablonu. Šablona stránky bude identifikována pomocí zvoleného názvu (například „První stránka“ nebo „tříslopcová šablona – vnitřek novin“).

6.2 Tiskové výstupy

Ze systému je zapotřebí mít dva druhy tiskových výstupů:

- grafické rozložení inzerátů v maketě
- a seznam objednávek, které byly přiřazeny k inzerátům ve vybrané maketě čísla novin.

6.2.1 Tisk makety stránky

Pro účely korektur rozmístění inzertní plochy v maketě je zapotřebí mít možnost tisku jednotlivých stránek makety čísla. Tisk by měl být zprostředkován přes klasické tlačítko se symbolem tisku. Výsledný tiskový výstup bude vždy přizpůsoben formátu A4. Na každém listu tiskového výstupu bude uvedeno číslo makety (například „UO 1/2008“), datum uzávěrky, datum a čas tisku, jméno uživatele, který tisk provedl.

6.2.2 Tisk seznamu objednávek z makety

Pro potřeby fakturace je nutné mít přehled o zpracovaných objednávkách daného čísla oblasti. Proto u aktuálně otevřené makety čísla bude možnost vytisknout tabulku realizovaných objednávek. Výstupem bude tabulka (tisk na šířku A4) se všemi atributy jednotlivých objednávek. Tato tabulka pak bude sloužit jako podklad pro účetní, která vystavuje faktury.

6.3 Podrobnější výpisy u importu objednávek

Uživatelé jsou jenom lidé. Proto dělají chyby. Jedním z nejčastějších chybných vstupů od uživatelů budou právě importy objednávek z účetního systému Pohoda. Stávající systém umožňuje detekovat chyby a sumarizovat vložené řádky a řádky, obsahující chybu. Po tomto rozšíření by měl software umožňovat zobrazení podrobného textového výpisu, který by poskytoval bližší informace o důvodu nevložení jednotlivých řádků souboru.

6.4 Možnost vkládat obrázky na místo ploch

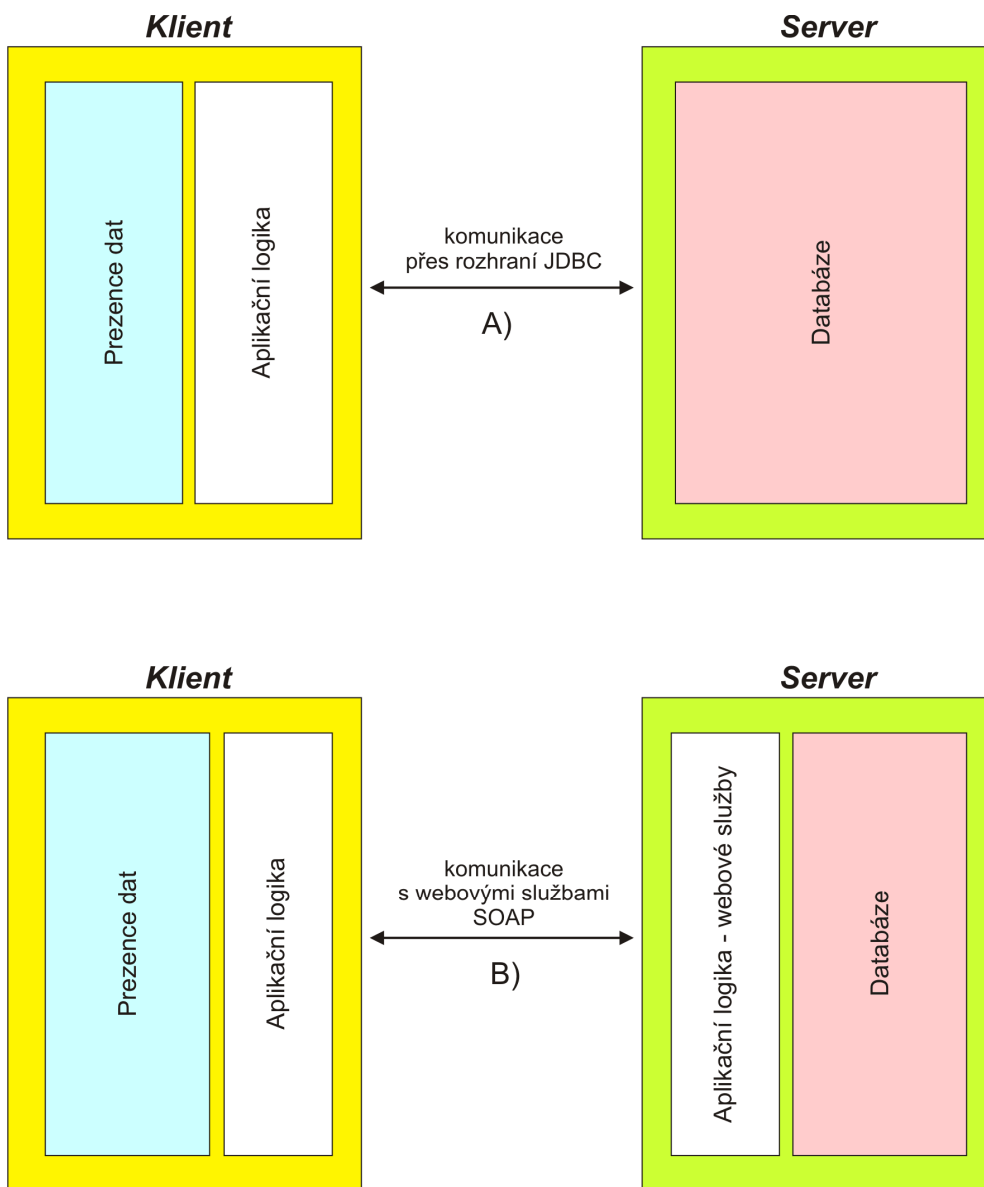
Pro snadnější orientaci v maketě by mělo být možné vkládat k inzertním plochám v maketě náhled příslušného inzerátu, který se vloží k ploše při editaci atributů inzerátu. Náhled inzerátu bude ve formátu JPG. Náhledy by mělo být možné vypnout a zobrazovat tak konceptuální verzi makety (tedy pouze ohraničené barevné plochy).

6.5 Přesun části aplikační logiky na server

Software je nyní navržen na bázi architektury s tlustým klientem se vzdálenou databází. Prvořadým problémem je bezpečnost. Autorizace k jednotlivým akcím je ověřována na straně klientské aplikace, což je slabým místem celé bezpečnosti.

Přesun části aplikační logiky na server (viz Obrázek 6.1) se uvažuje až od druhé verze aplikace, kdy bude zajištěn aplikační server pro J2EE, který bude prostřednictvím webových služeb zprostředkovávat operace pro práci s entitami. Distribuce aktuálního software díky malému (řádově desítky) počtu uživatelů nebude činit problém.

Jelikož jsem s touto změnou počítal již od návrhu první verze software, nebude potřeba kompletního nového návrhu, nýbrž změny na straně klientské aplikace se dotknou několika málo zdrojových souborů.



Obrázek 6.1 - Struktura aplikace A) před a B) po přesunu části aplikační logiky na server

7 Závěr

Úkolem diplomové práce bylo vytvořit software pro práci s reklamními plochami v tiskovinách, zdokumentovat jeho vývoj, zejména část analýzu a návrhu, vytvořit uživatelský manuál a zhodnotit přínos aplikace a popsat možná další rozšíření softwaru.

Vývoj software hodnotím jako velmi vydařený. Všechny body zadání byly splněny, zadavatel je se softwarem spokojen, splnil jeho očekávání. Je přínosem pro celou reklamní agenturu a zaměstnance, kteří se účastní procesu přípravy a vydávání inzertních novin. Tento systém lze použít univerzálně i pro tiskoviny jiného formátu a určení, ve kterých je nutné spravovat reklamní plochy.

Vývoj systému tedy bude nyní pokračovat. Bude provedena analýza nových požadavků a vytvořen návrh pro jejich realizaci v již stávajícím systému.

Literatura

- [1] Microsoft s.r.o. (Visual Studio Team System): *Návrh architektury* [online], 2006, [cit. 2008-03-23], URL: <<http://www.teamsystem.cz/Architect.aspx>>
- [2] Pichlík R.: *Třívrstvá architektura v kostce I.* [online], 2004-11-11, [cit. 2008-04-12], URL: <http://www.sweb.cz/pichlik/archive/2004_11_07_archive.html>
- [3] Wikipedia.org: *MySQL* [online], [cit. 2007-12-30], URL: <<http://en.wikipedia.org/wiki/MySQL>>
- [4] Wikipedia.cz: *Java* [online], [cit. 2008-04-03], URL: <<http://cs.wikipedia.org/wiki/Java>>
- [5] Wikipedia.org: *Java (programming language)* [online], [cit. 2008-04-03], URL: <[http://en.wikipedia.org/wiki/Java_\(programming_language\)](http://en.wikipedia.org/wiki/Java_(programming_language))>
- [6] Novotný L.: *Historie a vývoj jazyka Java* [online], 2003, [cit. 2008-05-12], URL: <<http://www.fi.muni.cz/usr/jkucera/pv109/2003p/xnovotn8.htm>>
- [7] Hamtil V.: *Framework Hibernate* [online], [cit. 2008-04-09], URL: <<http://nb.vse.cz/~zelenyj/it380/eseje/xhamv01/xhamv01.htm>>
- [8] Wikipedia.cz: *JavaDoc* [online], [cit. 2008-05-14], URL: <<http://cs.wikipedia.org/wiki/Javadoc>>
- [9] Wikipedia.org: *MySQL* [online], [cit. 2007-12-30], URL: <<http://en.wikipedia.org/wiki/MySQL>>

Seznam příloh

- Příloha A) Seznam obrázků
- Příloha B) Příklady mapování z objektových dat na relační databázi
- Příloha C) Popis obsahu přiloženého nosiče CD-ROM
- Příloha D) Uživatelský manuál k programu MaketaNovin
- Příloha E) Nosič CD-ROM

Příloha A

Seznam obrázků

Obrázek 2.1 - Stavový diagram životního cyklu novin	5
Obrázek 2.2 - Stavový diagram prodeje reklamní plochy	6
Obrázek 3.1 - Diagram případů použití	9
Obrázek 4.1 - Jednotlivé vrstvy architektury aplikace	22
Obrázek 4.2 - Diagram tříd	23
Obrázek 5.1 - Hlavní okno aplikace	33
Obrázek 5.2 - Okno pro otevření makety čísla	33
Obrázek 5.3 - Podokno pro práci s maketou	34
Obrázek 5.4 - Změna rozměrů inzerátu	35
Obrázek 5.5 - Sekvenční diagram přidání inzerátu	35
Obrázek 5.6 - Podokno s výpisem objednávek	37
Obrázek 5.7 - Dialog pro import objednávek	39
Obrázek 5.8 - Sumář po ukončení importu objednávek	39
Obrázek 6.1 - Struktura aplikace A) před a B) po přesunu části aplikační logiky na server	43

Příloha B

Příklady mapování z objektových dat na relační databázi

Oba následující příklady pro objektově relační mapování jsou převzaty z webových stránek na adrese http://www.laliluna.de/blog/2007/01/28/hibernate_annotations.html.

Příklad mapování pomocí XML souboru:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd" >

<hibernate-mapping package="de.laliluna.example1">
  <class name="Computer" table="tcomputer">
    <id name="id" >
      <generator class="sequence">
        <param name="sequence" >tcomputer_id_seq</param>
      </generator>
    </id>
    <property name="name" type="string"></property>
  </class>
</hibernate-mapping>
```

Příklad mapování pomocí anotací v kódu:

```
package de.laliluna.example1;

import java.io.Serializable;
import javax.persistence.*;

@Entity
@Table(name="computer")
@SequenceGenerator(name="computer_seq", sequenceName="computer_id_seq")
public class Computer implements Serializable{
    private static final long serialVersionUID = 3322530785985822682L;
    private Integer id;
    private String name;

    public Computer() {}

    @Id
    @GeneratedValue(strategy=GenerationType.SEQUENCE,
                    generator="computer_seq")
    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }
}
```

```
public String getName() {  
    return name;  
}  
  
public void setName(String name) {  
    this.name = name;  
}  
  
public String toString() {  
    return "Computer: " + getId() + " Name: " + getName();  
}  
}
```

Příloha C

Popis obsahu přiloženého nosiče CD-ROM

Nosič CD-ROM, který je přílohou E této zprávy, obsahuje následující adresářovou strukturu:

- **zprava** – obsahuje tuto technickou zprávu (*DIP.pdf*) ve formátu PDF,
- **test** – obsahuje spustitelnou aplikaci pro testovací účely,
- **src** – obsahuje zdrojové kódy programu a potřebné soubory pro překlad,
- **database** – obsahuje skript *default_db.sql* pro vytvoření databáze.

Příloha D

Uživatelský manuál k programu MaketaNovin

1. Úvod

Aplikace *MaketaNovin* slouží pro správu reklamních ploch v novinách GOLEMPRESS. Do systému mají přístup pouze autentizovaní uživatelé, které své uživatelské jméno a heslo získají u správce systému Ing. Petra Krátkého.

Aplikace MaketaNovin umožňuje:

- správu uživatelů,
- správu oblastí,
- správu čísel v jednotlivých oblastech,
- správu objednávek,
- změnu hesla uživatele,
- práci s grafickou maketou.

2. Požadavky a spuštění programu

Pro spuštění programu je zapotřebí mít na svém počítači nainstalováno prostředí JRE (Java SE Runtime Environment 6), jehož instalaci lze stáhnout na webových stránkách Sun Microsystems na adrese <http://java.sun.com/javase/downloads>.

Program spustíte z adresáře s programem poklepnutím na soubor *run.bat*.

3. Typy uživatelů

V systému se vyskytují čtyři úrovně uživatelů:

- Správce,
- Šéfredaktor,
- Obchodní zástupce,
- Grafik.

Grafik

Uživatel typu „grafik“ má pouze práva pro prohlížení maket čísel. Dále má právo na prohlížení objednávek za použití výběrového filtru. Nemůže tedy měnit uskupení inzertních ploch v maketách a ani měnit objednávky v libovolných číslech. Jeho role je „pozorovací“.

Obchodní zástupce

Obchodní zástupce může v systému na rozdíl od grafika zadávat objednávky do libovolných oblastí novin. Nemůže však objednávky mazat nebo měnit jejich stav či ostatní atributy.

Šéfredaktor

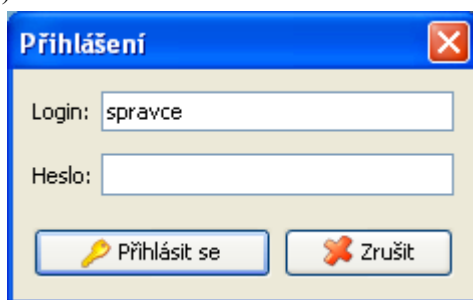
Šéfredaktor má za úkol primárně spravovat jemu přiřazené oblasti. U přiřazených oblastí může přidávat, editovat a mazat jednotlivá čísla, měnit a mazat objednávky.

Správce

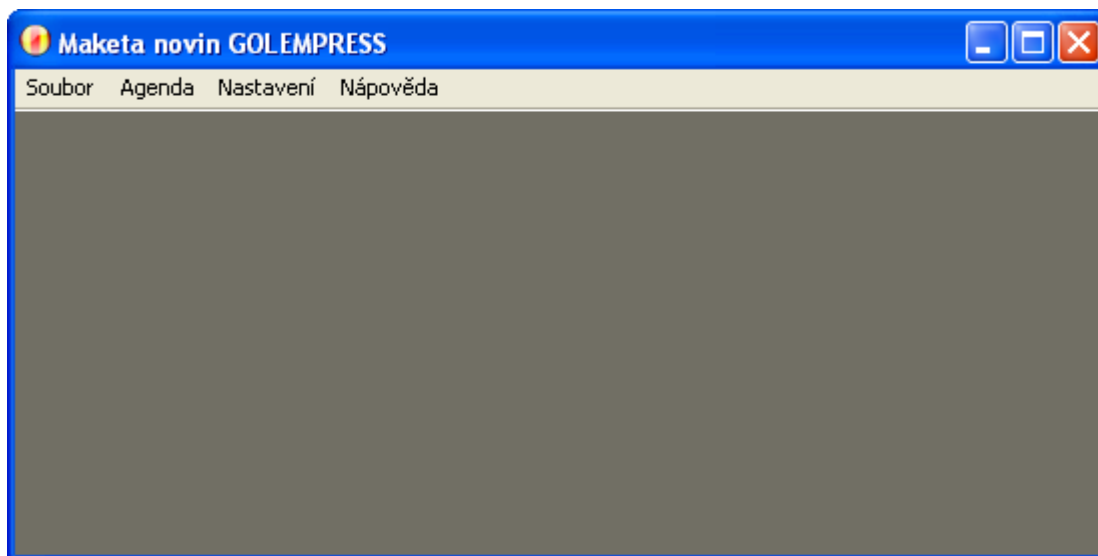
Správce systému má práva pro správu všech entit a práci s maketami čísel všech oblastí definovaných v systému. Má právo definovat nové oblasti novin, měnit a mazat oblasti.

4. Přihlášení a hlavní okno aplikace

Po spuštění aplikace je uživatel vybídnut k přihlášení pomocí kombinace uživatelského jména (loginu) a hesla (viz obrázek 4.1). Pokud jsou zadány správné přihlašovací údaje, je zobrazeno hlavní okno aplikace (viz obrázek 4.2).



Obrázek 4.1 – Přihlašovací dialog



Obrázek 4.2 – Hlavní okno aplikace

Hlavní okno obsahuje v horní části hlavní nabídku. Ostatní místo obsahuje pracovní plochu pro podřízená okna ostatních částí aplikace.

5. Správa uživatelů

Do správy uživatelů má přístup pouze správce systému přes volbu z menu *Agenda/Uživatelé*. Zobrazí se okno se správou uživatelů, které obsahuje tlačítka pro *přidání*, *editaci* a *smazání* uživatele a tabulku se seznamem uživatelů.

Při přidávání nového uživatele nebo při editaci stávajícího je nutné zadat jednoznačné přihlašovací jméno (tzv. login).

Pokud chcete změnit uživatele, nejprve v seznamu vyberte jednoho kliknutím myši, poté klikněte na tlačítko *Editovat* a ve zobrazeném formuláři změňte požadované údaje. Pokud při editaci nevložíte heslo, zůstane v platnosti původní.

Pro smazání je možné vybrat více uživatelů a poté kliknout na tlačítko *Smazat*. Po kliknutí na „Ano“ potvrzovacího dialogu dojde k odstranění vybraných uživatelů ze systému.

6. Správa oblastí novin

Inzertní noviny GOLEMPRESS vycházejí v různých oblastech. Do správy oblastí má přístup pouze správce systému přes volbu z menu *Agenda/Oblasti*. Zobrazí se okno se správou oblastí, které obsahuje tlačítko pro *přidání*, *editaci* a *smazání* oblasti a tabulku se seznamem již definovaných oblastí novin.

Při přidávání či editaci oblasti ve formuláři (viz obrázek 6.1) vyplňujete *zkratku* (např. UO, SY apod.), *kód oblasti* (koresponduje se zakázkou v systému Pohoda 2007), *název* a v roletce vyberte šéfredaktora, který je zodpovědný za danou oblast a který bude mít možnost spravovat inzeráty v maketě.



Obrázek 6.1 – Formulář pro přidání oblasti

7. Správa čísel novin

Jednotlivá vydání – čísla novin jsou sdružená v oblastech. Do správy čísel oblastí má přístup pouze správce systému nebo šéfredaktor přes volbu z menu *Agenda/Číslo oblasti*. Zobrazí se okno se správou oblastí, které obsahuje tlačítko pro *přidání*, *editaci* a *smazání* čísla, tabulku se seznamem již definovaných čísel novin, tlačítko pro zobrazení seznamu stránek v čísle a především roletku s výběrem oblastí, podle které se vypisují čísla novin.

Pokud je uživatel typu *šéfredaktor* a není šéfredaktorem vybrané oblasti, nemůže čísla z oblasti mazat, editovat a ani do oblasti přidávat. Při přidávání nebo změně čísla novin (viz obrázek 7.1) uživatel volí *rok*, ve kterém číslo vychází, a *pořadové číslo* v tomto roce. Následně se zadává datum uzávěrky a v roletce na konci formuláře se volí barevnost čísla.

Obrázek 7.1 – Formulář pro editaci oblasti

Po vybrání právě jednoho čísla v seznamu čísel novin a kliknutí na tlačítko *Stránky* se zobrazí okno pro správu stránek daného čísla novin (viz obrázek 7.2). Stránky je možné *přidávat*, *editovat* a *mazat* obdobně jako bylo výše popsáno u správy uživatelů, oblastí apod.

Číslo stránky	Počet inzerátů	Rozměr stránky
Strana 1	1	29,7 x 42,0 cm
Strana 2	0	29,7 x 42,0 cm
Strana 3	0	29,7 x 42,0 cm
Strana 4	0	29,7 x 42,0 cm

Obrázek 7.2 – Okno pro správu stránek v čísle novin

8. Správa objednávek

Do správy objednávek mají přístup všichni uživatelé a mají možnost si za pomoci filtru zobrazit seznam všech objednávek v systému. Okno pro správu objednávek (viz obrázek 8.1) zobrazíte volbou z menu *Agenda/Objednávky*.

Uživatelé si mohou filtrovat výpis objednávek podle *data*, *stavu objednávky* a *oblasti*, do které je objednávka určena. Přidávání, editace a mazání objednávek probíhá obdobně, jako tomu je u správy oblastí. Pokud přihlášený uživatel není správce nebo šéfredaktor, nemůže provádět import objednávek ze souboru a nemůže také mazat a editovat objednávky. Obchodní zástupci mohou objednávky pouze přidávat.

Import objednávek z CSV souboru

Do systému je možné importovat objednávky hromadně pomocí CSV souboru. Počet i pořadí sloupců bude vždy stejný a pevně daný: *Doklad, Datum, Číslo, Cena, Firma, Jméno, Ulice, Obec, PSČ, IČ, Telefon, Fax, Sředitisko, Činnost, Číslo (kód) oblasti, Poznámka*.

Pole „Poznámka“ obsahuje informace ve formátu:

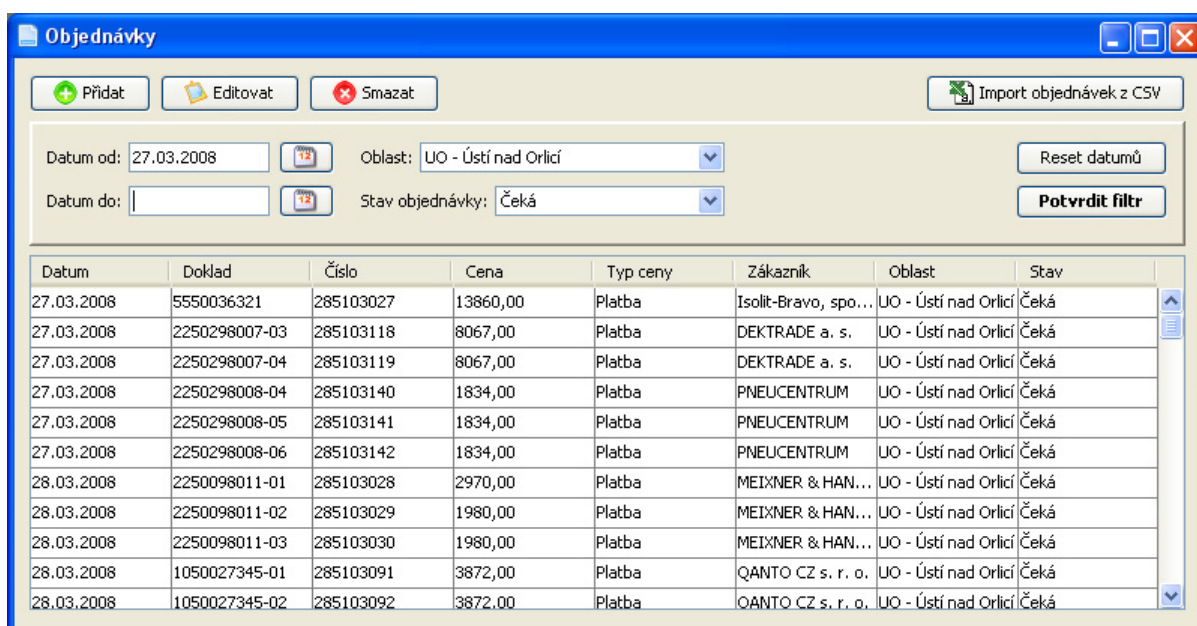
p/SxV\$POPIS\$X,

kde:

- S = šířka objednaného inzerátu,
- V = výška objednaného inzerátu,
- POPIS = popisná, upřesňující informace o poloze na stránce a barevnosti,
- X = zkratka čísla – nebere se v potaz, oblast je určena polem „Číslo oblasti“.

Při importu se kontroluje, zda již objednávka v systému nebyla zadána (podle pole *doklad* a *číslo objednávky*). Stav importované objednávky je nastaven na „čeká“.

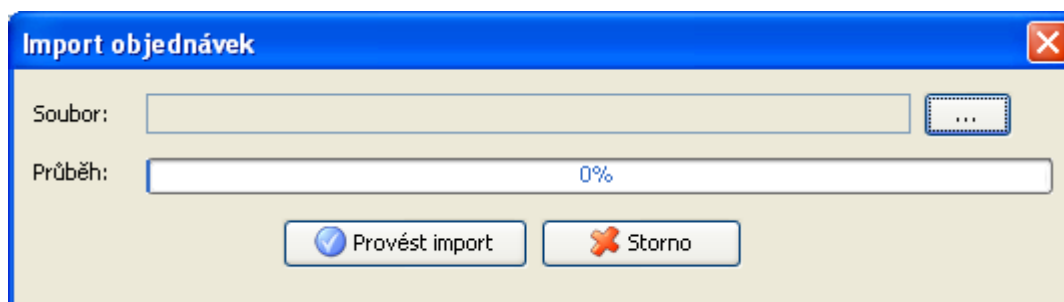
Import zahájíte kliknutím na tlačítko *Import objednávek z CSV*. V dialogovém okně (viz obrázek 8.2) vyberte importní soubor a klikněte na tlačítko *Provést import*. Průběh importu je znázorněn v tomto dialogu. Po ukončení importu je uživatel informován o počtu vložených/nevložených objednávek.



The screenshot shows a window titled 'Objednávky' with a table of orders and a filter section. The filter section includes buttons for 'Přidat', 'Editovat', 'Smazat', and 'Import objednávek z CSV'. It also has input fields for 'Datum od' (27.03.2008), 'Datum do', 'Oblast' (UO - Ústí nad Orlicí), and 'Stav objednávky' (Čeká). A 'Reset datumů' button is also present. The table below has the following data:

Datum	Doklad	Číslo	Cena	Typ ceny	Zákazník	Oblast	Stav
27.03.2008	5550036321	285103027	13860,00	Platba	Isolit-Bravo, spo...	UO - Ústí nad Orlicí	Čeká
27.03.2008	2250298007-03	285103118	8067,00	Platba	DEKTRADE a. s.	UO - Ústí nad Orlicí	Čeká
27.03.2008	2250298007-04	285103119	8067,00	Platba	DEKTRADE a. s.	UO - Ústí nad Orlicí	Čeká
27.03.2008	2250298008-04	285103140	1834,00	Platba	PNEUCENTRUM	UO - Ústí nad Orlicí	Čeká
27.03.2008	2250298008-05	285103141	1834,00	Platba	PNEUCENTRUM	UO - Ústí nad Orlicí	Čeká
27.03.2008	2250298008-06	285103142	1834,00	Platba	PNEUCENTRUM	UO - Ústí nad Orlicí	Čeká
28.03.2008	2250098011-01	285103028	2970,00	Platba	MEIXNER & HAN...	UO - Ústí nad Orlicí	Čeká
28.03.2008	2250098011-02	285103029	1980,00	Platba	MEIXNER & HAN...	UO - Ústí nad Orlicí	Čeká
28.03.2008	2250098011-03	285103030	1980,00	Platba	MEIXNER & HAN...	UO - Ústí nad Orlicí	Čeká
28.03.2008	1050027345-01	285103091	3872,00	Platba	QANTO CZ s. r. o.	UO - Ústí nad Orlicí	Čeká
28.03.2008	1050027345-02	285103092	3872,00	Platba	QANTO CZ s. r. o.	UO - Ústí nad Orlicí	Čeká

Obrázek 8.1 – Okno pro správu objednávek



Obrázek 8.2 – Dialogové okno pro import objednávek

9. Změna hesla

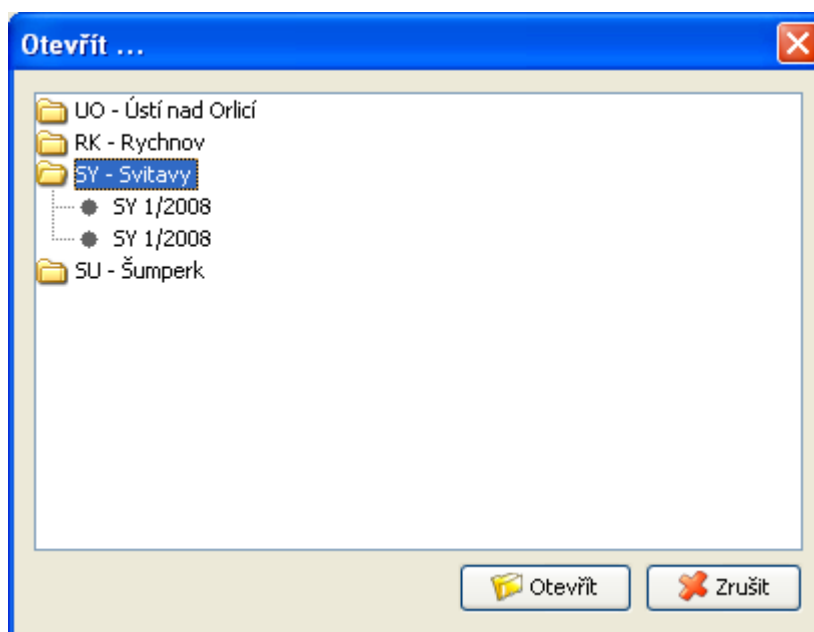
Přihlášený uživatel si může změnit své přihlašovací heslo. V menu klikněte na volbu *Nastavení/Změna hesla*. Zobrazí se formulář pro změnu hesla. V první řadě zadejte své původní heslo. Poté zvolte heslo nové a pro kontrolu ho zadejte i do pole *Kontrola hesla*.

Pokud původní heslo bylo zadáno korektně a kontrola hesla souhlasí, bylo heslo úspěšně změněno.

10. Práce s maketou novin

Maketu čísla otevřete přes volbu z menu *Soubor/Otevřít maketu*. Pokud je přihlášený uživatel typu grafik, obchodní zástupce nebo šéfredaktor, který nespravuje oblast otevírané makety, je maketa otevřena pouze pro čtení (tj. nelze v maketě přidávat či měnit inzeráty).

V dialogu pro otevření makety (viz obrázek 10.1) jsou jednotlivá čísla sdruženy do složek podle oblastí. Kliknutím na složku (oblast) se rozvine seznam čísel novin v oblasti.



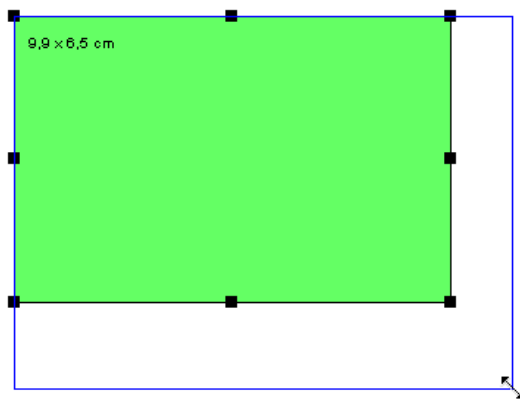
Obrázek 10.1 – Dialog pro otevření makety

Maketa čísla je otevřena v samostatném okně a je zobrazena první stránka čísla. Mezi stránkami je možné se přepínat roletkovým výběrem v horní části okna. Je také možné volit přiblížení (realizováno rovněž pomocí roletky).

Pokud má uživatel právo zápisu do otevřené makety, má možnost přidávat, přesouvat a měnit inzeráty. Dále má možnost přidávat, měnit a mazat vodící linky, ke kterým se pak při přesunu přichycují jednotlivé inzeráty.

Inzerát lze označit kliknutím myši na jeho plochu. Pro změnu inzerátu použijte „double-click“ na plochu inzerátu, nebo tlačítko v horní části okna. Změnu rozměrů či přesun je možné provádět i intuitivně pomocí myši (uchopením a tažením – viz obrázek 10.2). U každého inzerátu je možné definovat jeho rozměry, pozici vůči stránce a barvu, kterou je inzerát vykreslen. Dále je možné vložit poznámku, která se bude zobrazovat u plochy inzerátu v maketě stránky.

K inzerátu je možné přiřadit objednávku. Pokud přiřadíte objednávku k inzerátu, je stav objednávky změněn na „vyřízeno“. Při smazání inzerátu, u kterého byla přiřazena objednávka, se stav objednávky nastaví opět na „čeká“.



Obrázek 10.2 – Změna rozměrů inzerátu tažením myši