



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

NÁVRH RELAČNÍ DATABÁZE PRO OBECNÍ KNIHOVNU

DESIGN OF RELATIONAL DATABASE FOR MUNICIPAL LIBRARY

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jan Vlk

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jiří Kříž, Ph.D.

BRNO 2020

Zadání diplomové práce

Ústav:	Ústav informatiky
Student:	Bc. Jan Vlk
Studijní program:	Systémové inženýrství a informatika
Studijní obor:	Informační management
Vedoucí práce:	Ing. Jiří Kříž, Ph.D.
Akademický rok:	2019/20

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Návrh relační databáze pro obecní knihovnu

Charakteristika problematiky úkolu:

Úvod
Cíle práce, metody a postupy zpracování
Teoretická východiska práce
Analýza současného stavu
Vlastní návrhy řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem práce je vytvoření funkčního návrhu relační databáze pro obecní knihovnu obce Senožaty.

Základní literární prameny:

CONOLLY, Thomas, Carolyn E. BEGG a Richard HOLLOWCZAK. Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází. Brno: Computer Press, 2009. ISBN 978-80-251-2328-7.

KOCH, Miloš a Bernard NEUWIRTH. Datové a funkční modelování. 4. vyd. Brno: Akademické nakladatelství CERM, 2010. ISBN 978-80-214-4125-5.

OPPEL, Andrew J. SQL bez předchozích znalostí: [průvodce pro samouky]. Brno: Computer Press, 2008. ISBN 978-80-251-1707-1.

STEPHENS, Ryan K., Ronald R. PLEW a Arie JONES. Naučte se SQL za 28 dní: [stačí hodina denně]. Brno: Computer Press, 2010. ISBN 978-80-251-2700-1.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2019/20

V Brně dne 29.2.2020

L. S.

doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Tato diplomová práce se zaměřuje na problematiku spojenou s návrhem relační databáze. Je rozdělena do několika částí, ve kterých se zabývá teoretickými východisky, analýzou současného stavu a návrhem vlastního řešení.

Abstract

This diploma thesis focuses on problematics associated with design of relational database. It is divided into several parts where it deals with theoretical basis, analyses of current state and the design of own solution.

Klíčová slova

data, databáze, návrh databáze, relační datový model, SQL

Key words

data, database, design of database, relational data model, SQL

Bibliografická citace

VLK, Jan. *Návrh relační databáze pro obecní knihovnu* [online]. Brno, 2020 [cit. 2020-05-08]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/127645>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Jiří Kříž.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 8. května 2020

.....

podpis autora

Poděkování

Tímto bych chtěl poděkovat panu Ing. Jiřímu Kříži, Ph.D. za vedení mé diplomové práce a za jeho cenné rady i čas. Dále bych chtěl poděkovat panu Bc. Ladislavu Marešovi za poskytnutí informací pro analytickou část práce a v neposlední řadě patří mé díky rodině, přátelům a známým, kteří mi po celou dobu byli oporou.

OBSAH

ÚVOD	13
1 CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ.....	14
1.1 Cíle práce	14
1.2 Metody a postupy zpracování	14
2 TEORETICKÁ VÝCHODISKA PRÁCE.....	16
2.1 Data, informace a znalosti.....	16
2.2 Databázový systém.....	17
2.2.1 Uživatelé	17
2.2.2 Databázová aplikace	17
2.2.3 Systém řízení databáze.....	18
2.2.4 Databáze.....	18
2.3 Architektura systému řízení databáze	18
2.3.1 Jednovrstvá architektura	18
2.3.2 Dvouvrstvá architektura.....	18
2.3.3 Třívrstvá architektura.....	19
2.4 Datové modely	19
2.5 Lineární datový model	19
2.6 Objektový datový model.....	20
2.7 Relační datový model.....	21
2.7.1 Relace.....	21
2.7.2 Relační databáze	21

2.8	Integrita relačního modelu	22
2.8.1	Integritní omezení pro entity (relace)	22
2.8.2	Integritní omezení pro vztahy entit.....	22
2.9	Klíče relace.....	23
2.10	Normalizace	23
2.10.1	První normální forma.....	24
2.10.2	Druhá normální forma	24
2.10.3	Třetí normální forma.....	24
2.11	Životní cyklus vývoje databázového systému.....	24
2.12	Metodologie návrhu databáze	27
2.12.1	Konceptuální úroveň.....	27
2.12.2	Logická úroveň	27
2.12.3	Fyzická úroveň.....	27
2.13	Jazyk SQL	28
2.13.1	Základní příkazy	28
2.13.2	Pohled	29
2.13.3	Uložená procedura	29
2.13.4	Spoušť	29
2.13.5	Kurzor	29
2.13.6	Transakce	30
2.14	Zálohovací pravidlo 3-2-1	30

2.15	Druhy záloh.....	30
2.15.1	Úplná záloha	30
2.15.2	Rozdílová záloha.....	31
2.15.3	Přírůstková záloha.....	31
3	ANALÝZA SOUČASNÉHO STAVU.....	32
3.1	O knihovně.....	32
3.1.1	Organizační struktura.....	33
3.1.2	Umístění a zabezpečení knihovny	34
3.2	Lokální počítačová síť.....	36
3.2.1	Server	37
3.2.2	Datové úložiště NAS	37
3.3	Technické vybavení knihovny	38
3.3.1	Klientská stranice.....	38
3.4	Přístup k datům a aplikacím.....	38
3.5	System zálohování dat.....	39
3.5.1	Stupně zálohy	39
3.6	Přihlášení čtenáře do knihovny	40
3.6.1	Přihlášení čtenáře staršího 15 let	40
3.6.2	Přihlášení čtenáře mladšího 15 let	41
3.7	Evidence čtenářů	43
3.8	Rozdělení a evidence knih	43

3.9	Systém evidence výpůjček	44
3.9.1	Čtenářský průkaz	45
3.9.2	Knižní lístek	46
3.9.3	Výpůjčka knihy	46
3.9.4	Vrácení knihy	47
3.10	Nedostatky současného stavu	48
4	VLASTÍ NÁVRHY ŘEŠENÍ	50
4.1	Konceptuální úroveň návrhu databáze	50
4.1.1	Identifikace základních entit	50
4.1.2	Identifikace vztahů mezi základními entitami	51
4.2	Logická úroveň návrhu databáze	53
4.2.1	Dekompozice vztahů s kardinalitou N:M	53
4.2.2	Návrh tabulek	55
4.2.3	Kontrola tabulek z hlediska integrity a normality	66
4.2.4	Konečná verze ER diagramu	67
4.3	Fyzická úroveň návrhu databáze	68
4.3.1	Implementace tabulky Ctenar včetně vazeb	68
4.3.2	Naplnění tabulky Vypujcka testovacími daty	69
4.3.3	Implementace a ověření funkčnosti pohledu Ctenari	69
4.3.4	Implementace o ověření funkčnosti pohledu Tiskoviny	71
4.3.5	Implementace a ověření funkčnosti procedury Najdi_tiskovinu	72

4.3.6	Implementace a ověření funkčnosti procedury Vloz_ctenare.....	74
4.4	Umístění databáze	76
4.5	Přístup k databázi	76
4.6	Zálohování databáze.....	76
4.7	Fyzické zabezpečení dat databáze.....	77
4.8	Budoucí rozvoj databáze	77
	ZÁVĚR	78
	SEZNAM POUŽITÝCH ZDROJŮ	79
	SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ	80
	SEZNAM OBRÁZKŮ.....	81
	SEZNAM TABULEK	83
	SEZNAM PŘÍLOH.....	85
	PŘÍLOHY	I

ÚVOD

Informace jsou v dnešní době jednou z nejcennějších komodit jednotlivce nebo organizace, ne-li tou vůbec nejcennější. Informace jsou určitým způsobem zpracovaná data. Pokud jsou tedy informace takto cenné a získávají se z dat, je potřeba data vhodným způsobem uchovávat.

Jedním z možných způsobů uchování dat je databáze. Databázi si obecně můžeme představit jako strukturované úložiště dat, kde jsou data podle dané struktury organizovaně uložena. Databáze existují v mnoha variantách provedení. Pokud však o databázi mluvíme dnes, většinou máme na mysli databázi elektronickou. Ta je realizována pomocí informačních technologií a umožňuje snadný přístup k datům a jednoduchou manipulaci s nimi.

1 CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Cíle práce a metody a postupy jejího zpracování jsou uvedeny v následujících dvou kapitolách.

1.1 Cíle práce

Hlavním cílem práce je vytvoření funkčního návrhu relační databáze pro obecní knihovnu obce Senožaty.

Díličními cíli práce jsou:

vypracování teoretické, analytické a návrhové části,

začlenění řešení nedostatků, zjištěných během vypracování analytické části, do návrhové části ať již jako součást návrhu databáze nebo samostatně,

vyřešení fyzického zabezpečení databáze, přístupu k databázi a budoucího rozvoje databáze.

1.2 Metody a postupy zpracování

Informace potřebné pro vypracování teoretické části práce byly získány z dostupných zdrojů uvedených v seznamu použité literatury.

Informace potřebné pro vypracování analytické a návrhové části byly buď čerpány z teoretické části nebo byly získány pomocí dvou výzkumných metod, a to analýzou dokumentů a rozhovory s vybranou osobou.

Analyzované dokumenty byly získány od člena zastupitelstva, přesněji místostarosty a jednalo se například o přihlášku čtenáře staršího 15 let.

Vybranou osobou, se kterou probíhaly rozhovory byl již zmíněný místostarosta znalý provozu knihovny a pověřený správou informačních technologií budovy obecního úřadu.

Během rozhovorů jsem kladl otázky a místostarosta mi na ně poskytoval odpovědi. Otázky kladené místostarostovi se týkaly provozu knihovny a technického vybavení obecního úřadu a knihovny.

2 TEORETICKÁ VÝCHODISKA PRÁCE

Informace zde uvedené slouží jako podpůrný prostředek pro vypracování analytické a návrhové části této práce.

2.1 Data, informace a znalosti

Data

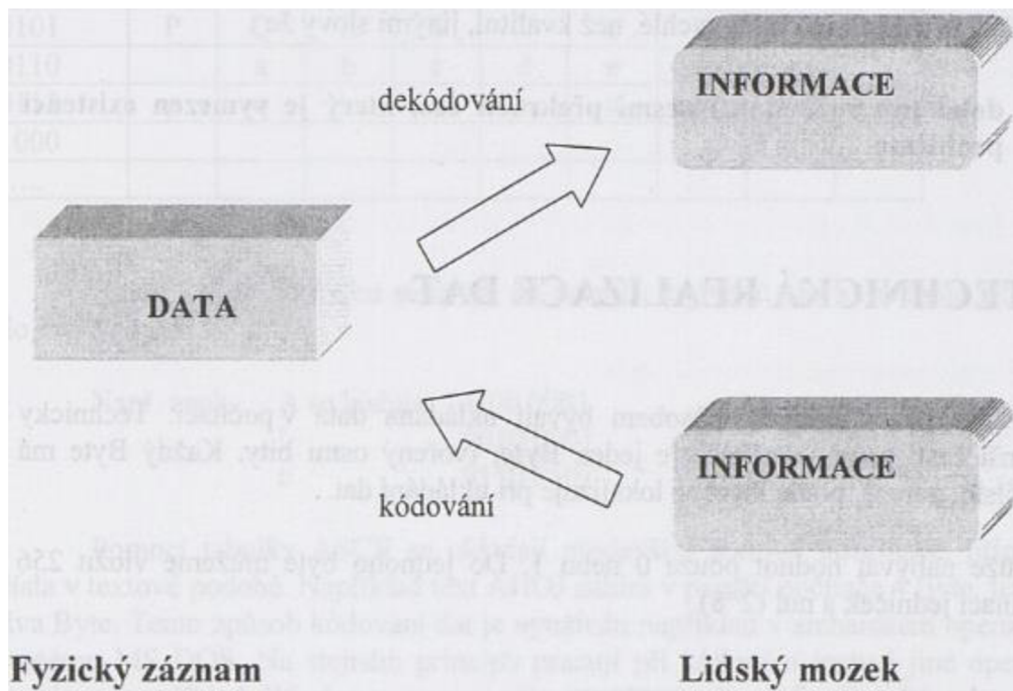
Data si lze představit, jako surová (nezpracovaná) fakta, která mají pro jednotlivce či organizaci určitou důležitost (1, s. 36).

Informace

Informace jsou zpracovaná data takovým způsobem, že jsou srozumitelná a použitelná, pokud je zobrazíme nebo s nimi pracujeme (7, s. 67).

Znalosti

„Znalosti jsou výsledkem porozumění informací, která byla právě sdělena, a její integrace s dřívějšími informacemi.“ (2, s. 5)



Obr. 1: Vztah mezi data a informace (Zdroj:2, s. 20)

2.2 Databázový systém

Databázový systém se skládá ze čtyř komponentů, kterými jsou uživatelé, databázová aplikace, systém řízení databáze a vlastní databáze (5, s. 29).

2.2.1 Uživatelé

Uživatelé udržují přehled o určitých skutečnostech pomocí databázové aplikace. Prostřednictvím formulářů zadávají a načítají data a dotazují se na ně a vytvářejí sestavy (5, s. 29).

2.2.2 Databázová aplikace

Databázová aplikace je počítačový program, který zastává funkci prostředníka mezi uživateli a systémem řízení báze dat (5, s. 29).

2.2.3 Systém řízení databáze

„Systém řízení databáze je počítačový program, který umožňuje vytvářet, zpracovávat a spravovat databáze. Systém řízení databáze přijímá požadavky v jazyce SQL a převádí je na aktivity databáze.“ (5, s. 29)

2.2.4 Databáze

Databázi lze nejobecněji popsat jako kolekci záznamů, které spolu souvisejí a obsahují vlastní popis (5, s. 30).

2.3 Architektura systému řízení databáze

Rozlišujeme tři základní typy architektury systému řízení databáze, kterými jsou jednovrstvá, dvouvrstvá a třívrstvá architektura (6).

2.3.1 Jednovrstvá architektura

V rámci jednovrstvé architektury též nazývané centrální jsou logika provozu, logika zpracování dat, databáze a datové služby umístěny na jednom centrální počítači (6).

2.3.2 Dvouvrstvá architektura

Dvouvrstvou architekturu též označovanou jako klient/server můžeme rozdělit na dvě varianty provedení, kterými jsou architektura soustředěná u klienta a architektura soustředěná na serveru (6).

2.3.3 Třívrstvá architektura

„Třívrstvá architektura rozděluje služby na vrstvu uživatelského rozhraní, která běží na klientovi, vrstvu logiky provozu a logiky zpracování dat, která běží na aplikačním serveru a databázi a datové služby, která běží na databázovém serveru.“ (6)

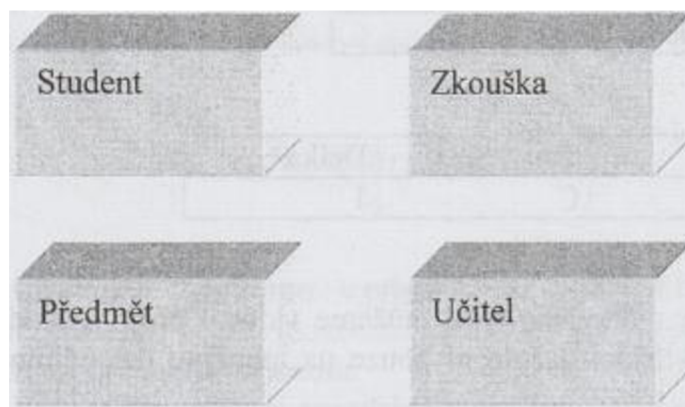
2.4 Datové modely

„Při vytváření informačních systémů zpravidla nevystačíme s jedinou strukturou věty. Pro každý typ datového objektu, který je částí informačního systému, potřebujeme navrhnout samostatnou datovou strukturu věty. Budeme-li budovat jednoduchý IS posluchačů vysoké školy, potřebujeme navrhnout datové struktury přinejmenším pro objekty student, zkouška, předmět a učitel. Kromě toho, v realitě spolu objekty souvisí. Student má složené zkoušky z konkrétních předmětů a zkoušel ho také konkrétní učitel. V informačním systému musíme vytvořit odpovídající obraz reality tak, aby data vložená do systému této realitě plně odpovídala.“ (2, s. 20)

V současné době máme v podstatě tři možné datové modely, které můžeme použít. Těmito modely jsou lineární, objektový a relační (2, s. 20).

2.5 Lineární datový model

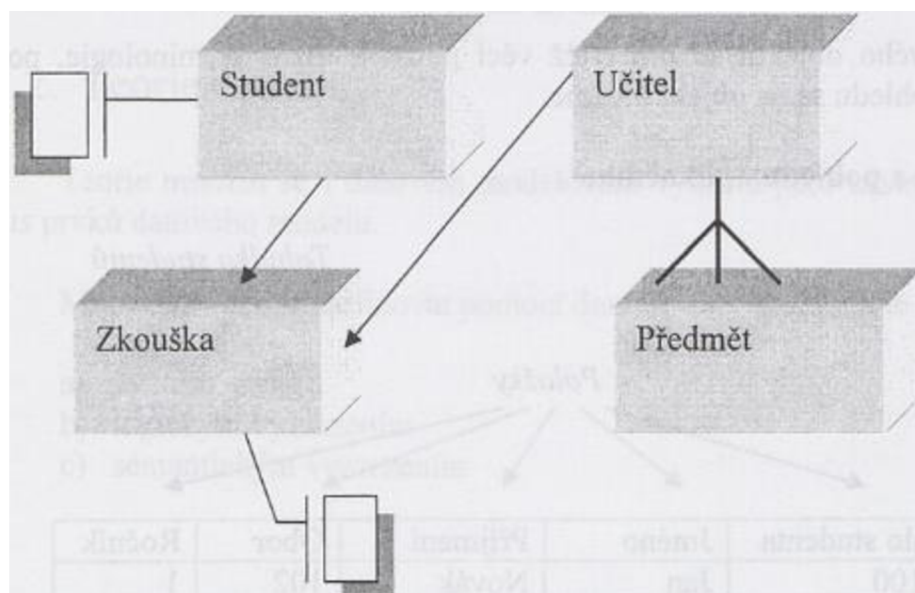
Lineární datový model je jediným modelem, který lze implementovat na libovolné médium. V modelu neexistují žádné vazby mezi jednotlivými objekty. Nelze tedy například přímo stanovit, který student složil zkoušku a z jakého předmětu (2, s. 21).



Obr. 2: Lineární datový model (Zdroj:2, s. 20)

2.6 Objektový datový model

„Nejnovějším datovým modelem je objektový datový model. Objektové datové modely jsou vystavěny na základním prvku – objekt (odpovídá přibližně pojmu věta), kde tento objekt má kromě svých atributů i definované metody, které určují chování objektu.“ (2, s. 22)

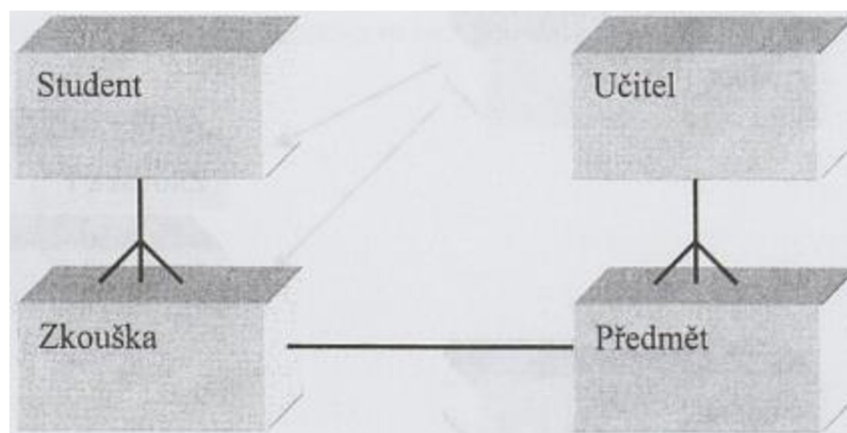


Obr. 3: Objektový datový model (Zdroj:2, s. 22)

2.7 Relační datový model

Relační model je založen na teorii relací. Umožňuje zachytit data nejen o zkoumaných objektech, ale také vztahy mezi těmito objekty, což nám umožňuje se více přiblížit reálnému světu (2, s. 23).

Pět hlavními složkami modelu jsou relace, atribut, datová n-tice, doména a relační databáze (1, s. 63).



Obr. 4: Relační datový model (Zdroj:2, s. 22)

2.7.1 Relace

Relaci si lze představit jako speciální druh tabulky, která slouží pro ukládání dat o sledovaných objektech reálného světa, kterým se říká entity (5, s. 78).

Řádky relace odpovídají jednotlivým **datovým n-ticím** a sloupce **atributů** (1, s. 63).

Množina přípustných hodnot pro jeden či více atributů je **doména** (1, s. 63).

2.7.2 Relační databáze

Relační databáze je kolekcí normalizovaných tabulek (relací) (1, s. 63).

2.8 Integrita relačního modelu

Integritu relačního modelu lze chápat jako stav, kdy data uložená v databázi odpovídají vlastnostem objektů reálného světa (entit) (2, s. 27).

Integritní omezení se dělí na omezení pro entity (relace) a pro vztahy entit (relační vazby) (2, s. 27).

2.8.1 Integritní omezení pro entity (relace)

Doménová integrita

Doménová integrita (integrita hodnot) říká, že každá hodnota každého z atributů relace musí být z množiny (domény) hodnot pro daný atribut přípustných (2, s.27).

Entitní integrita

Podle entitní integrity musí mít každá relace určen primární klíč (2, s. 28).

Referenční integrita

Referenční integrita říká, že cizí a jemu odpovídají primární klíč musí být definovány na stejné doméně (soulad hodnot) (2, s. 29).

2.8.2 Integritní omezení pro vztahy entit

Kardinalita vztahů mezi entitami (relacemi) je omezena na poměry 1:1, 1:N a N:M (2, s. 30).

Vztah 1:1

Dvojice relací je ve vztahu 1:1 pokud jedna n-tice první relace je ve vztahu pouze s jednou n-ticí druhé relace a naopak (7, s. 77).

Vztah 1:N

N-tice první relace může být ve vztahu s mnoha n-ticemi druhé relace, ale n-tice druhé relace může být ve vztahu pouze s jednou n-ticí první relace (7, s. 77).

N:M

N-tice první relace může být ve vztahu s mnoha n-tice druhé relace a naopak (7, s. 78).

2.9 Klíče relace

Kandidátní klíč

Kandidátní klíč jedinečně identifikuje každou n-tici relace. V relaci může být více kandidátních klíčů. Kandidátním klíčem může být jak jeden atribut relace, tak se může jednat o složeninou z více atributů nebo klíčů (5, s. 66).

Primární klíč

Primární klíč je zvolený kandidátní klíč, jenž má dvě vlastnosti, kterými jsou jedinečnost (klíč se v rámci relace neopakuje, je jedinečný) a neredukovatelnost (žádný z dílčích atributů klíče nelze vypustit, aniž by byla porušena jedinečnost) (5, s. 66).

Cizí klíč

Cizí klíč relace odpovídá primárnímu klíči jiné relace. Prostřednictvím cizího a k němu příslušného primárního klíče jsou uskutečněny vazby mezi relacemi (2, s. 28).

2.10 Normalizace

„Normalizace je činnost, při které upravujeme návrhy datových struktur tak, aby splňovaly zvolené normalizační formy – úrovně. Tyto normalizační formy či pravidla vycházejí z požadavku na efektivní ukládání dat a minimalizují redundance při zachování integrity a konsistence dat. Datový model, který porušuje některou z normalizačních forem není navržený optimálně. Při normalizace databáze na vyšší normalizační úrovni musí být normalizovaný na všech předcházejících.“ (2, s. 54).

2.10.1 První normální forma

Všechny atributy relace musí být jednoduché, nikoli složené nebo vícehodnotové (2, s. 55).

Jediná kriticky důležitá norma pro vytvoření vhodných relací. Ostatní normy jsou volitelné. Doporučením je však používat třetí normální formu (1, s. 191).

2.10.2 Druhá normální forma

Relace je v první normální formě a všechny atributy, které nejsou součástí primárního klíče jsou na něm závislé (1, s. 192).

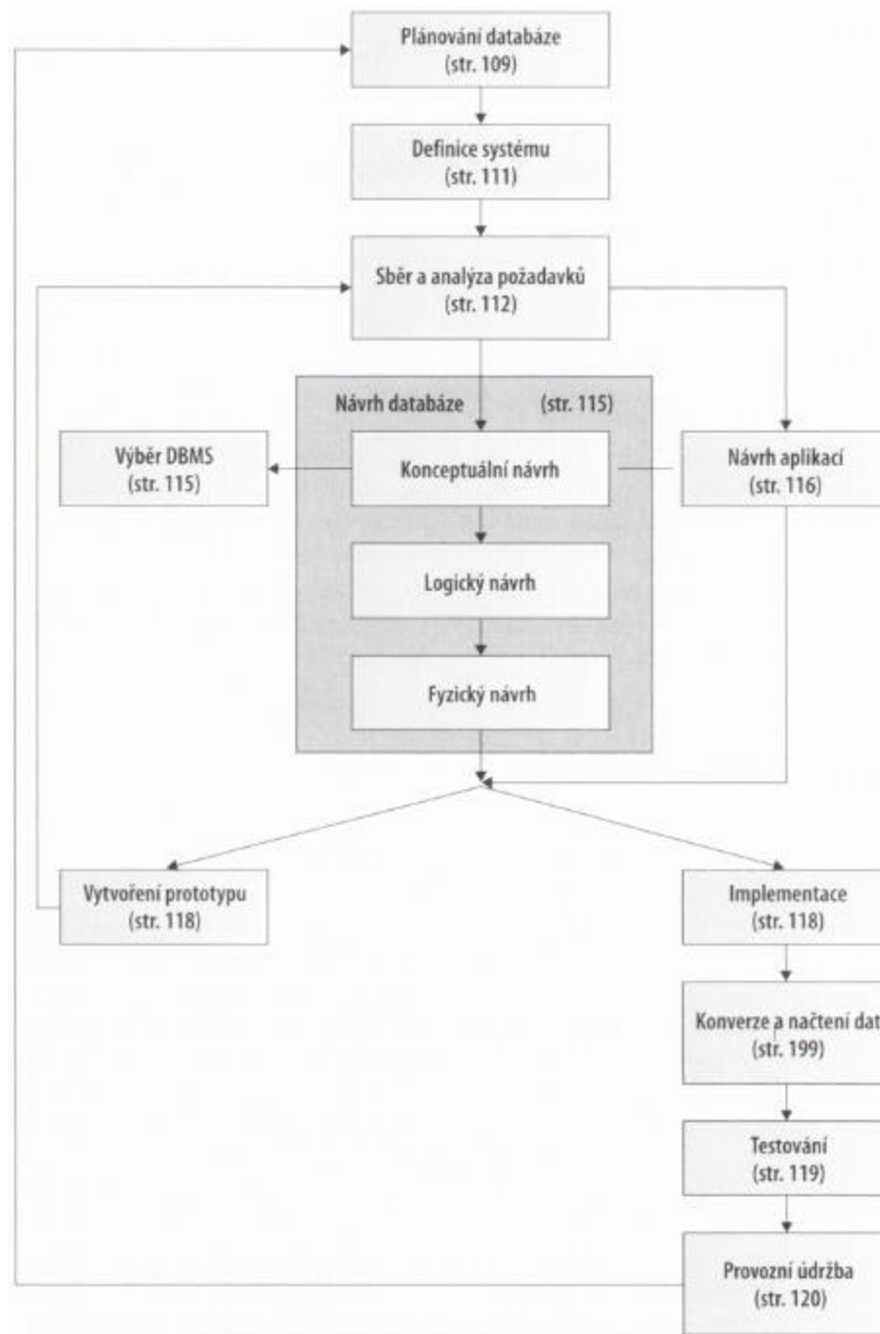
2.10.3 Třetí normální forma

Relace je ve druhé normální formě a všechny atributy, které nejsou součástí primárního klíče jsou vzájemně nezávislé (2, s. 58).

2.11 Životní cyklus vývoje databázového systému

Životním cyklem vývoje databázového systému se rozumí uspořádaný seznam fází, který popisuje odpovídající techniky a nástroje, které je třeba při vývoji databázového systému použít (1, s. 109).

Je důležité si uvědomit, že jednotlivé fáze cyklu nemusí následovat striktně po sobě, ale také mohou někdy zahrnovat opakování předchozích fází na základě zpětné vazby. Například problémy objevené během návrhu databáze mohou vyžadovat dodatečné kroky při sběru a analýze požadavků (1, s. 109).



Obr. 5: Fáze životního cyklu vývoje databázového systému (Zdroj:1, s. 110)

Plánování databáze

„Činnosti managementu, které umožní v mezích možností hladký a efektivní průběh jednotlivých fází životního cyklu vývoje databázového systému.“ (1, s. 109)

Definice systému

Fáze definice systému zahrnuje identifikaci rozsahu a hranic databázového systému včetně jeho hlavních uživatelských pohledů (1, s. 111).

Sběr a analýza požadavků

Tato fáze se zabývá sběrem a analýzou informací o organizaci nebo její části, které bude databáze sloužit (1, s. 112).

Návrh databáze

„Proces vytvoření návrhu, který bude podporovat celkové poslání a dílčí cíle pro požadovaný databázový systém organizace.“ (1, s. 115)

Výběr DBMS

Výběr vhodného systému řízení databáze pro podporu databázového systému. Nejvhodnější částí cyklu pro výběr je doba mezi logickým a fyzickým návrhem databáze. Výběr však lze provést kdykoli, a to i před logickým návrhem (1, s. 115).

Návrh aplikací

„Návrh uživatelského rozhraní a databázových aplikací, které budou používat a zpracovávat databázi.“ (1, s. 116)

Vytvoření prototypu

„Vytvoření fungujícího modelu databázového systému.“ (1, s. 118)

Implementace

Implementací se rozumí fyzická realizace databáze a aplikací (1, s. 118).

Konverze a načtení dat

„Příprava všech dat (pokud je nezbytná) do podoby vhodné pro transfer do nové databáze.“ (1, s. 119)

Testování

„Proces provozu databázového systému se záměrem nalézt chyby.“ (1, s. 119)

Provozní údržby

„Proces monitorování a údržby databázového systému po instalaci.“ (1, s. 120)

2.12 Metodologie návrhu databáze

„Pod pojmem metodologie návrhu databáze si představíme strukturovaný přístup s využitím technik, nástrojů a dokumentace pro usnadnění procesu návrhu.“ (6)

Návrh databáze zahrnuje tři úrovně, kterými jsou konceptuální, logická a fyzická (6).

2.12.1 Konceptuální úroveň

V rámci konceptuální úrovně probíhá identifikace základních entit a vztahů mezi nimi (6).

2.12.2 Logická úroveň

Logická úroveň navazuje na konceptuální a probíhá zde dekompozice vztahů s kardinalitou N:M, návrh tabulek pro jednotlivé entity a jejich kontrola z hlediska integrity a normality (8).

2.12.3 Fyzická úroveň

Fyzická úroveň navazuje na logickou a dochází během ní k implementaci jednotlivých částí databáze (8).

2.13 Jazyk SQL

Jazyk SQL je standartním jazykem pro manipulaci a získávání dat z relačních databází. Pomocí jazyka SQL může upravovat strukturu databáze, měnit nastavení zabezpečení systému, přidávat uživatelská oprávnění k databázím či tabulkám, dotázat se databáze na nějakou informaci nebo aktualizovat obsah databáze (4, s. 38).

2.13.1 Základní příkazy

CREATE

Příkaz CREATE umožňuje vytvořit nový databázový objekt včetně databáze samotné. Druh objektu je uveden za slovem CREATE (3, s. 57).

ALTER

Pomocí příkazu ALTER lze modifikovat již existující databázový objekt. Druh objektu je uveden za slovem ALTER (3, s. 57).

DROP

Příkaz DROP zruší existující databázový objekt. Druh objektu je uveden za slovem DROP (3, s. 57).

INSERT

Příkaz INSERT slouží ke vkládání nových řádků (n-tic) do tabulky (relace) (3, s. 135).

UPDATE

Pomocí příkazu UPDATE můžeme modifikovat řádek (n-ticu) tabulky (relace) (3, s. 135).

DELETE

Příkaz DELETE slouží ke smazání řádku (n-tice) tabulky (relace) (3, s. 135).

SELECT

„Příkaz SELECT slouží pro vyhledávání dat v databázi podle různých kritérií. Vzhledem k tomu, že dotazy mohou být velmi různorodé, lze příkaz SELECT zapisovat mnoha různými variantami.“ (9)

2.13.2 Pohled

„Pohled je dotaz v jazyce SQL, který je trvale uložen v databázi a je mu přiřazen název. Výsledky uloženého dotazu jsou zobrazovány pomocí pohledu a SQL umožňuje přistupovat k těmto výsledkům, jako by se jednalo o skutečnou tabulku v databázi.“ (10, s. 379)

2.13.3 Uložená procedura

Uloženou proceduru lze chápat jako script, který není uložen mimo databázi, ale přímo v ní (11).

„Uložená procedura může například realizovat aplikační logiku, která se stará o příjem zákaznických objednávek nebo o převod peněz z jednoho bankovního účtu na druhý.“ (10, s. 630)

2.13.4 Spoušť

„Princip spouště je poměrně jednoduchý. Ke každé události, která vyvolá změnu v obsahu tabulky, může uživatel připojit akci, kterou databáze v takovém případě provede. Těmito třemi událostmi, jež mohou spustit akci, jsou pokusy o provedení příkazu INSERT, UPDATE nebo DELETE na řádcích tabulky. Akce spouštěná takovou událostí bude dána posloupností příkazů SQL.“ (10, s. 301)

2.13.5 Kurzor

Kurzor není nic jiného než ukazatel na jediný řádek v sadě výsledků (3, s. 173).

„Použití kurzoru lze přirovnat k práci s tradičním jednoduchým souborem. Kurzor je totiž nutné definovat a otevřít dříve, než jej lze použít, umožňuje čtení dat při načítání řádků v programové smyčce a nakonec je vhodné ho uzavřít, když v programu nadále není potřebný.“ (3, s. 173)

2.13.6 Transakce

„Transakce je oddělená sada akcí, které je nutné buď zpracovat jako celek, nebo vůbec. Transakce se někdy označují jako pracovní jednotky, což dále zdůrazňuje jejich nedělitelnou povahu.“ (3, s. 162)

2.14 Zálohovací pravidlo 3-2-1

Zálohovací pravidlo 3-2-1 nám říká, že by měli existovat minimálně tři kopie dat, které budou uloženy minimálně dvou rozdílných médiích, přičemž alespoň jedna z kopií bude umístěna mimo lokální pracoviště (12).

2.15 Druhy záloh

Rozlišujeme tři základní typy zálohy dat, kterými jsou úplná, rozdílová a přírůstková (13, s. 123).

2.15.1 Úplná záloha

Úplná záloha je zálohou všech uložených dat. Je zve všech typů nejnáročnější na kapacitu úložiště a zároveň její provedení trvá nejdéle. Nejčastěji se využívá v kombinaci s jedním ze dvou zbývajících typů zálohy (13, s. 123).

2.15.2 Rozdílová záloha

Nejčastěji se rozdílová záloha využívá v kombinaci se zálohou úplnou. Rozdílová záloha je záznamem všech změn, které proběhly od poslední úplné. Není příliš náročná na kapacitu úložiště a je nezávislá na ostatních rozdílových zálohách, tedy poškození jedné neovlivní ostatní. V případě obnovy dat je potřeba vlastnit příslušnou rozdílovou a příslušnou úplnou zálohu (14).

2.15.3 Přírůstková záloha

Přírůstková záloha se nejčastěji využívá v kombinaci se zálohou úplnou. Na rozdíl od zálohy rozdílové není záznamem změn provedených od poslední zálohy úplné, ale od přecházející přírůstkové. Ze všech tří typů klade nejmenší požadavek na kapacitu úložiště. Jednotlivé rozdílové zálohy tvoří celek a v případě poruchy jedné, dochází k nepoužitelnosti ostatních. V případě obnovy dat je potřeba vlastnit jak původní úplnou zálohu, tak všechny přírůstkové (14).

3 ANALÝZA SOUČASNÉHO STAVU

Nejdříve jsou uvedeny informace o knihovně. Následuje lokální počítačová síť, technickém vybavení knihovny, přístupu k datům a aplikacím a systému zálohování.

Poté je popsáno přihlášení čtenáře do knihovny, evidence čtenářů, rozdělení a evidence knih a systém evidence výpůjček.

Na závěr je uvedeno zhodnocení aktuálního stavu, které slouží jako podklad pro návrhovou část této práce.

3.1 O knihovně

Daná obecní knihovna, která je podkladem této analytické části diplomové práce se nachází v obci Senožaty. Knihovna je otevřena třikrát týdně po dobu čtyř hodiny. Její služby mohou využívat nejen občané Senožaty, ale i jiných obcí a měst.

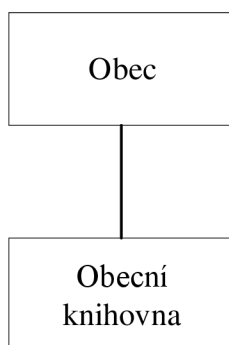
Mezi služby, které knihovna nabízí svým čtenářům (klientům) patří půjčování knih a časopisů, tisk dokumentů a přístup na internet.

Tab. 1: Provoz knihovny (Zdroj: vlastní zpracování)

Den	Otevírací doba
Pondělí	Zavřeno
Úterý	16:00 – 20:00
Středa	Zavřeno
Čtvrtek	16:00 – 20:00
Pátek	Zavřeno
Sobota	8:00 – 12:00
Neděle	Zavřeno

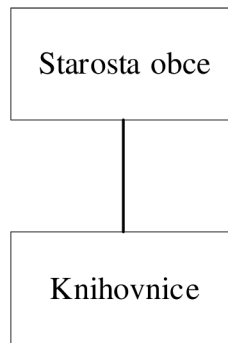
3.1.1 Organizační struktura

Obecní knihovna je součástí obce viz. obrázek č. 1. Z toho vyplývá, že veškerý majetek knihovny, je majetkem obce.



Obr. 6: Zařazení obecní knihovny v rámci obce (Zdroj: vlastní zpracování)

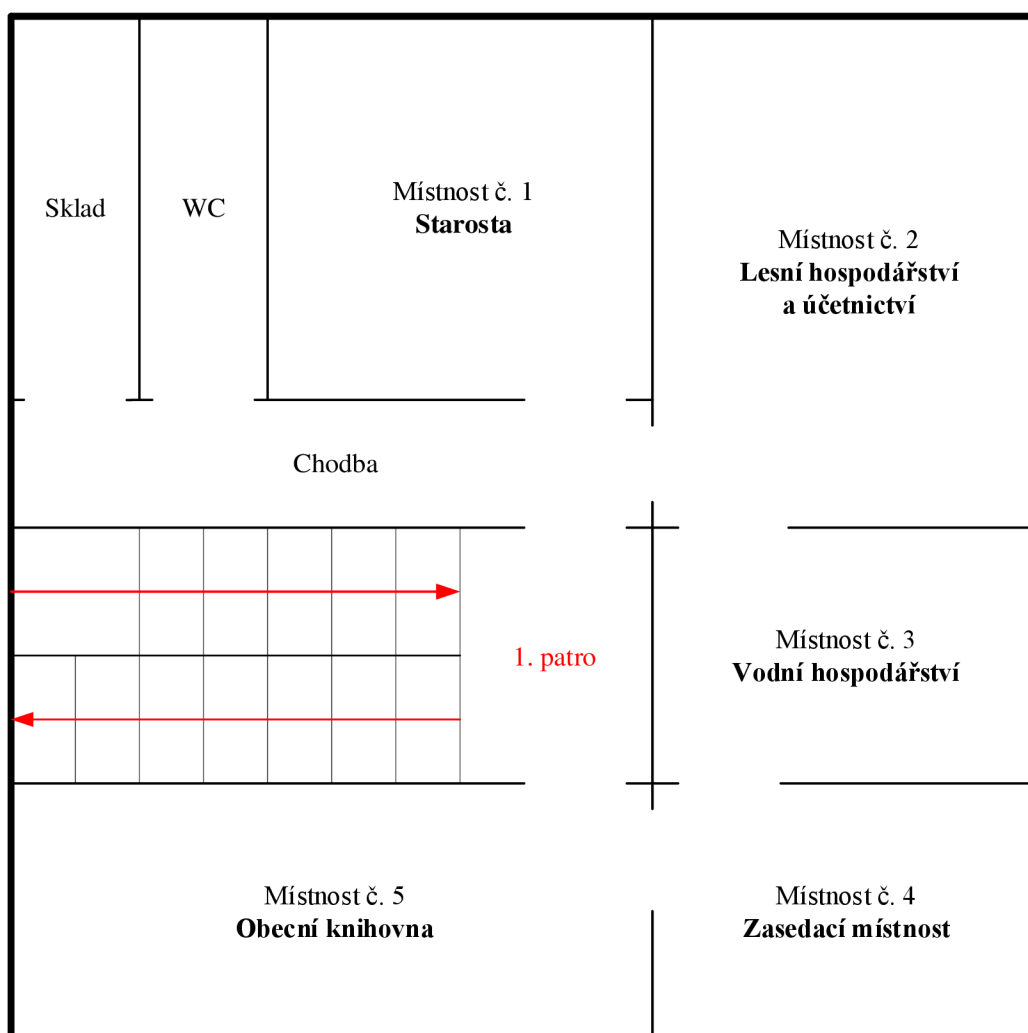
Zaměstnanec obecní knihovny neboli knihovnice, je podřízena starostovi obce viz. obrázek č. 2. Knihovnice pracuje na dohodu o provedení práce (DPP).



Obr. 7: Zařazení knihovnice v rámci obecního úřadu (Zdroj: vlastní zpracování)

3.1.2 Umístění a zabezpečení knihovny

Knihovna je umístěna v 1. patře jednopatrové budovy obecního úřadu viz. obrázek č. 3. V tomto patře sídlí i ostatní složky obce.

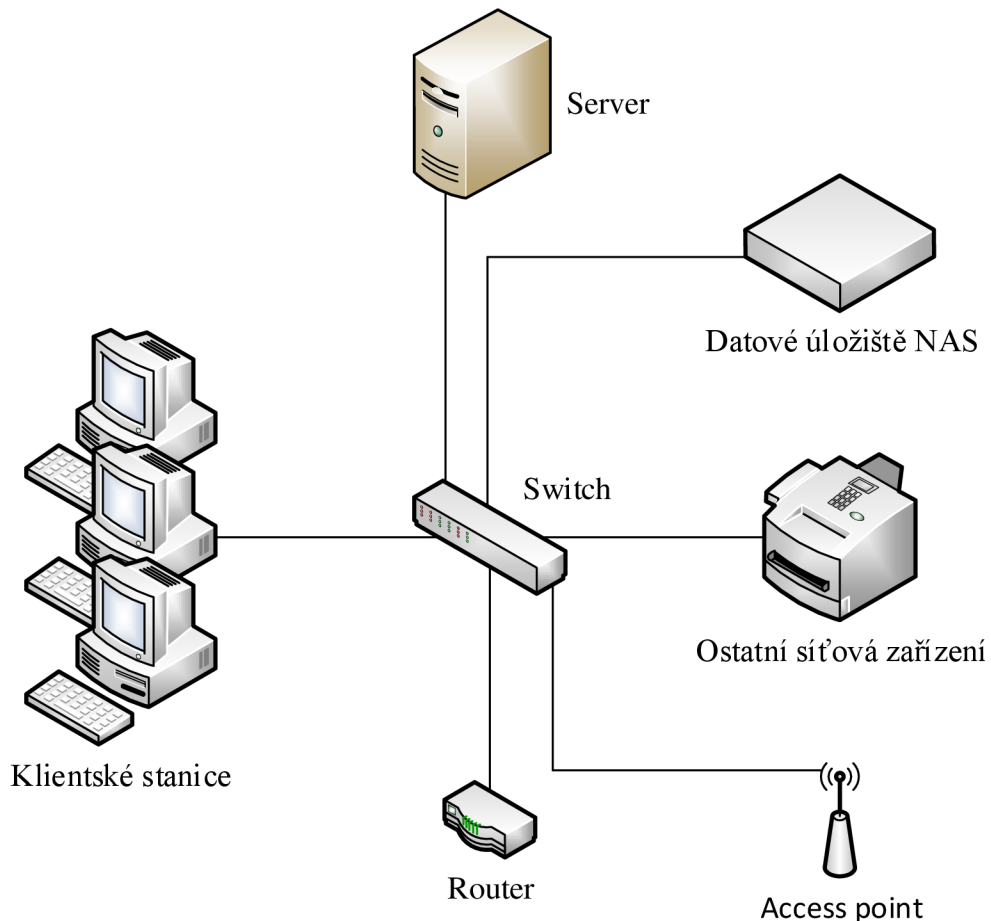


Obr. 8: 1. patro budovy obecního úřadu (Zdroj: vlastní zpracování)

Prostory 1. patra budovy obecního úřadu jsou zabezpečeny pomocí EZS JA-100 od firmy JABLOTRON. Na chodbě a v místnostech č. 2, 4, 5 jsou umístěny sběrníkové detektory pohybu a rozbití skla JA-120PB. Ústředna EZS je umístěna v místnosti č. 4. Modul pro ovládání systému je pak umístěn na chodbě.

3.2 Lokální počítačová síť

Jednotlivé složky obce, kromě obecní knihovny sdílí jednu společnou lokální počítačovou síť. Tuto síť si můžeme rozdělit na dvě části. První z nich je síťová infrastruktura a druhá koncové uzly.



Obr. 9: Lokální počítačová síť (Zdroj: vlastní zpracování)

Síťová infrastruktura

Infrastruktura je topologie hvězda. Její součástí je vrstva aktivní (switch, router, access point) a vrstva pasivní (metalické kabely UTP kategorie 5e), která zajišťuje propojení koncových uzlů s aktivními prvky.

Koncové uzly

Koncovými uzly jsou klientské stanice, datové úložiště NAS, server a ostatní síťová zařízení.

3.2.1 Server

Na serveru je nainstalovaný operační systém Windows Server 2012 R2 Foundation 64bit a aplikace využívané uživateli. Jeho funkce je aplikační a souborová. Pevné disky jsou seskupeny podle RAID 1. Pro případ výpadku přísunu elektrické energie je připojen k záložnímu zdroji elektrické energie APC Back-UPS CS 350I.

Server se nachází v místnosti č. 3.

Konfigurace

- Model: Dell PowerEdge T20
- Procesor: Intel Pentium G3220 (3.0 GHz)
- Operační paměť: DDR3 4 GB (1600 MHz)
- Pevný disk: 2 x 500 GB SATA III

3.2.2 Datové úložiště NAS

Disky datového úložiště NAS jsou seskupeny podle RAID 1. Úložiště podporuje funkci Hot Swap a je připojeno k záložnímu zdroji elektrické energie APC Back-UPS CS 350I.

Datové úložiště se nachází v místnosti č. 3.

Konfigurace

- Model: Synology DiskStation DS214
- Procesor: Marvell Armada XP dvoujádrový (1.066 GHz)
- Operační paměť: 512 MB DDR3
- Pevný disk: 2 x 1 TB SATA II

3.3 Technické vybavení knihovny

Technické vybavení knihovny tvoří jedna klientská stanice a jedna laserová tiskárna Canon i-Sensys MF4320D.

Daná klientská stanice není součástí lokální sítě sdílené ostatními složkami obce, stejně tak i tiskárna.

Klientská stanice je určena jak pro potřeby knihovnice, tak pro přístup na internet čtenáři knihovny. Tiskárna slouží jak pro potřeby knihovnice, tak pro potřeby čtenářů.

Jak stanice, tak i tiskárna jsou umístěny v místnosti č. 4.

3.3.1 Klientská stanice

Na klientské stanici je nainstalovaný operační systém Windows 10 Pro od společnosti Microsoft.

Konfigurace

- Model: Comfor Office Hit I100
- Procesor: Intel Pentium G3260 (3.3 GHz)
- Operační paměť: 4 GB DDR3 (1600 MHz)
- Pevný disk: 1 x 1 TB SATA III

3.4 Přístup k datům a aplikacím

Každý zaměstnanec složek obce, kromě knihovnice, má vytvořený vlastní doménový účet, jehož data jsou uložena na serveru. K tomuto účtu se přihlašuje pomocí přiřazených přihlašovacích údajů.

Prostřednictvím tohoto účtu má každý zaměstnanec přístup k aplikacím, pro které má přidělena oprávnění.

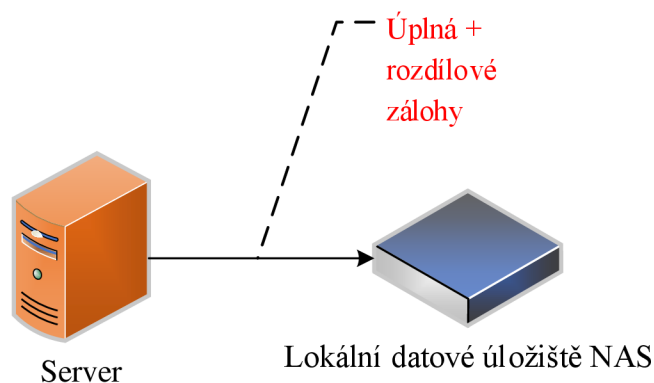
3.5 Systém zálohování dat

Systém zálohování dat disponuje dvěma stupni, kdy zálohování probíhá pouze u serveru, a to v nočních hodinách. Důvodem zálohování pouze serveru je fakt, že veškerá data jsou ukládána pouze na něj.

3.5.1 Stupně zálohy

1. stupeň

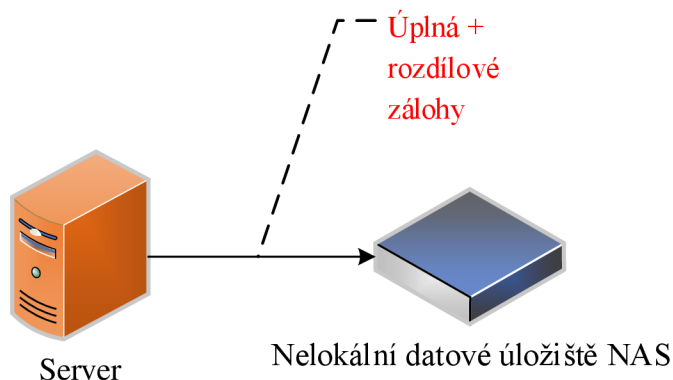
Každý týden v pátek se provádí úplná záloha serveru na lokální datové úložiště NAS, které je součástí lokální sítě. Dále se každé pondělí, úterý, středu a čtvrtky provádí rozdílová záloha serveru, která se též ukládá na lokální datové úložiště NAS.



Obr. 10: 1. stupeň zálohy (Zdroj: vlastní zpracování)

2. stupeň

Každý týden v pátek se provádí úplná záloha serveru na datové úložiště NAS (stejně parametry jako u NAS lokální sítě), které není součástí lokální sítě, ale je součástí sítě základní školy, ve které je i zároveň umístěno. Dále se každé pondělí, úterý, středu a čtvrtky provádí rozdílová záloha serveru, která se též ukládá na datové úložiště NAS základní školy.



Obr. 11: 2. stupeň zálohy (Zdroj: vlastní zpracování)

3.6 Přihlášení čtenáře do knihovny

Každý občan, který se chce využívat služeb obecní knihovny, musí být do knihovny přihlášen, tedy stát se čtenářem knihovny.

3.6.1 Přihlášení čtenáře staršího 15 let

Občan starší 15 let obdrží tištěnou přihlášku (verze pro čtenáře staršího 15 let), knihovní řád a poučení o ochraně osobních údajů od knihovnice. Seznámí se s přihláškou, knihovním řádem a poučením o ochraně osobních údajů a u dané přihlášky vyplní minimálně všechny povinné údaje (může i volitelné). Poté předá vyplněnou přihlášku knihovnici, která zkontroluje, zda je náležitě vyplněna. Pokud je vše v pořádku, občan uhradí zápisné (dle ceníku služeb a sankcí), knihovnice založí přihlášku do kartotéky a předá občanovi jeho čtenářský průkaz (vyplněná úvodní strana).

Mezi povinné údaje patří jméno, příjmení, datum narození, trvalá adresa (ulice a číslo, PSČ, obec//město), kontaktní adresa (ulice a číslo, PSČ, město/obec), datum a podpis čtenáře.

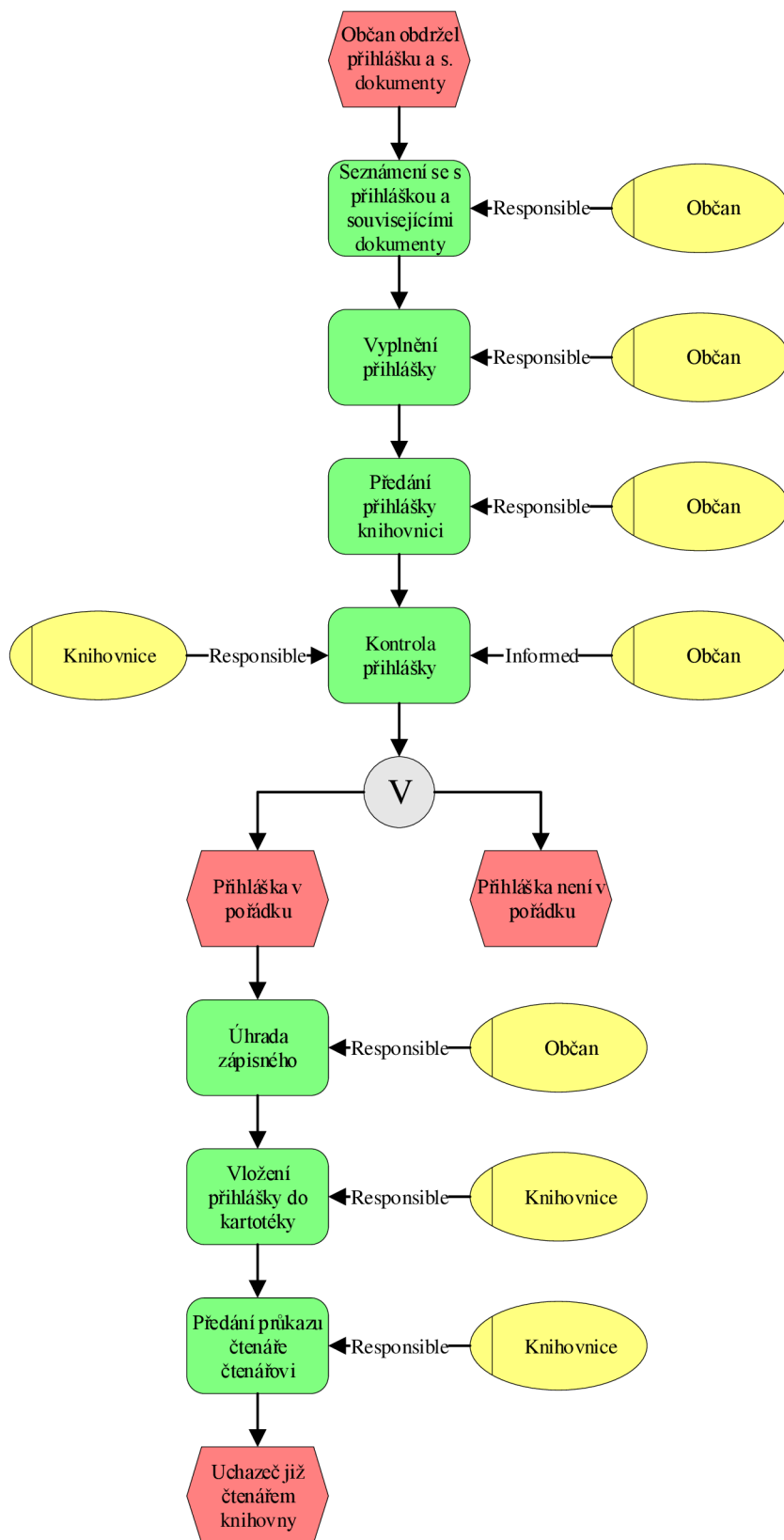
Mezi nepovinné údaje patří tituly, pohlaví, průkaz ZTP, ZTP/P, e-mail(y) a telefon(y).

3.6.2 Přihlášení čtenáře mladšího 15 let

Občan mladší 15 let obdrží tištěnou přihlášku (verze pro čtenáře mladšího 15 let), knihovní řád a poučení o ochraně osobních údajů od knihovnice. Seznámí se s přihláškou, knihovním řádem a poučením o ochraně osobních údajů a u dané přihlášky vyplní minimálně všechny povinné údaje (může i volitelné). Poté předá vyplněnou přihlášku knihovnici, která zkontroluje, zda je náležitě vyplněna. Pokud je vše v pořádku, občan uhradí zápisné (dle ceníku služeb a sankcí), knihovnice založí přihlášku do kartotéky a předá občanovi jeho čtenářský průkaz (vyplněná úvodní strana).

Mezi povinné údaje patří jméno, příjmení, datum narození, trvalá adresa (ulice a číslo, PSČ, obec//město), kontaktní adresa (ulice a číslo, PSČ, město/obec), údaje zákonného zástupce (stejně jako povinné údaje čtenáře staršího 15 let), datum a podpisy čtenáře, zákonného zástupce a knihovníka.

Mezi nepovinné údaje patří tituly (zákonný zástupce), pohlaví (zákonný zástupce), průkaz ZTP, ZTP/P, e-mail(y) a telefon(y).



Obr. 12: Přihlášení čtenáře (bez ohledu na věk) (Zdroj: vlastní zpracování)

3.7 Evidence čtenářů

Evidence čtenářů je provedena pomocí sešitu, ve kterém jsou uvedeny čísla čtenářských průkazů (číslování od jedničky vzestupně) a jejich vlastníci (jméno a příjmení čtenáře). Dále je zde uvedeno, zda čtenář uhradil uživatelský poplatek pro daný rok. Zbytek informací o čtenářích je evidován pomocí přijatých přihlášek.

Problémem tedy je, že informace o čtenářích nejsou evidována na jednom místě (sešit a přihlášky). V případě potřeby nastává zdlouhavé dohledávání jednotlivých informací o čtenářích, kdy je v případě více stejnojmenných čtenářů nutno využít i čtenářského průkazu k jednoznačné identifikaci čtenáře.

3.8 Rozdělení a evidence knih

V základu jsou knihy v knihovně rozděleny do dvou skupin, a to literaturu pro dospělé a na literaturu pro děti. Toto rozdělení je provedeno pomocí umístění knih v rozdílných regálech. Zmíněné dvě skupiny se pak dále dělí na beletrii a naučnou literaturu, kdy jsou jednotlivé regály podle příslušné skupiny označeny.

Každé knize je přiřazena značka (jedna ze čtyř), která značí, do které skupiny kniha patří. Tato značka je uvedena na knižním lístku vloženém uvnitř knihy.

Tab. 2: Značení skupin literatury (Zdroj: vlastní zpracování)

Značka	Popis
B	beletrie
N	naučná pro dospělé
M	mládež beletrie
MN	mládež naučná

Beletrie pro mládež je dále rozdělena pomocí barevných štítků umístěných na knihách, kdy každá z barev odpovídá určení/doporučení knihy pro čtenáře.

Tab. 3: Rozdělení beletrie pro mládež (Zdroj: vlastní zpracování)

Barva	Určení
žlutá	první čtení
červená	pohádky
modrá	příběhy
purpurová	školní četba
hnědá	divčí četba
zelená	dobrodružství

Podle pravidel knihovny by každá kniha měla mít přiřazené přírůstkové číslo (číslování od jedničky vzestupně). Toto číslo by pak bylo uvedeno na vloženém knižním lístku.

Jednotlivé knihy by pak měly být evidovány v určeném sešitě pomocí přírůstkového čísla a popisných informací knihy.

V současné době takový sešit v knihovně neexistuje. Navíc u většiny knih není vyplněno přírůstkové číslo ani na knižním lístku. Pouze je vyplněno u knih, které byly získány darem z jiných knihoven, které též využívali knižní lístky a přírůstková čísla vyplňovali.

3.9 Systém evidence výpůjček

Systém evidence výpůjček je založen na dvou prostředcích. První z nich je čtenářský průkaz, který občan obdrží po přihlášení do knihovny. Druhý je knižní lístek, který je umístěn v jednotlivých knihách.

Čtenářský průkaz a knižní lístek jsou spolu při realizaci výpůjčky propojeny pomocí čísla legitimace (číslo čtenářského průkazu) a značky (skupina do které kniha spadá).

Evidence vypůjčených knih je na základě vyjmutých knižních lístků, které má knihovnice v kartotéce.

3.9.1 Čtenářský průkaz

Úvodní strana čtenářského průkazu obsahuje číslo průkazu, rok, jméno a příjmení a adresu (bydliště). Čtenářský průkaz dále obsahuje datum výpůjčky, počet svazků (značka) a potvrzení o vrácení. Číslo průkazů spolu s jejich vlastníky jsou evidováni v papírové podobě.

Jednotlivá čísla průkazů jsou tvořena tak, že každý nový čtenář obdrží číslo průkazu o 1 vyšší než čtenář předcházející (poslední přihlášený), přičemž první průkaz/čtenář obdržel číslo průkazu 1.

Datum výpůjčky	Počet svazků (značka)	Potvrzení o vrácení

razítko knihovny

ČTENÁŘSKÝ PRŮKAZ

Číslo			
Rok			

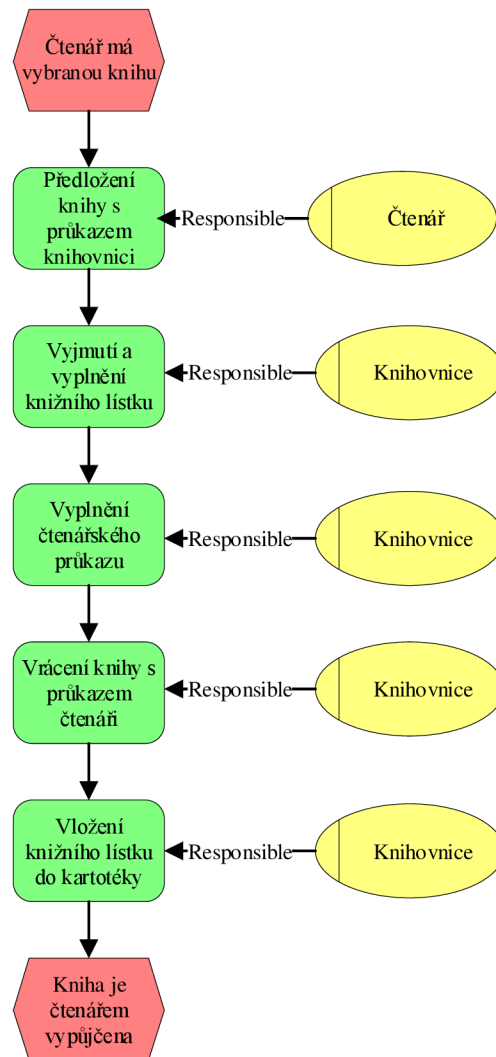
příjmení a jméno

adresa (bydliště)

SEVT - 49 557 1 I/88 TZ 6-79 6434 88

Obr. 13: Čtenářský průkaz (Zdroj: vlastní zpracování)

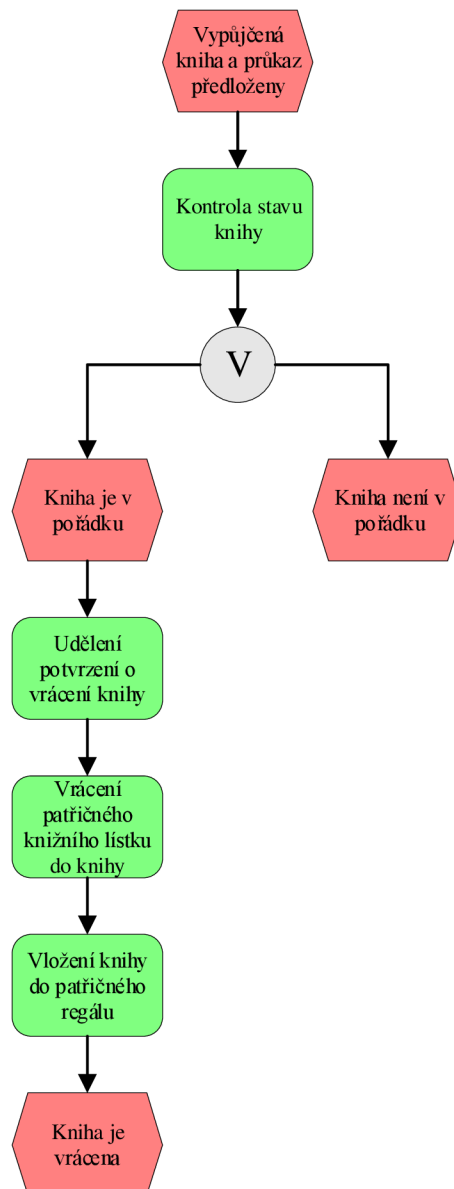
Každá výpůjčka podle pravidel knihovny může trvat maximálně 4 týdny.



Obr. 15: Výpůjčka knihy (Zdroj: vlastní zpracování)

3.9.4 Vrácení knihy

Čtenář předloží vypůjčenou knihu spolu se svým čtenářským průkazem knihovnici. Ta knihu zkontroluje, zda je v pořádku a pokud ano, tak udělí čtenáři do průkazu potvrzení o vrácení. Poté vyjme knižní lístek půjčené knihy z kartotéky a vrátí ho zpátky do knihy. Poté knihu i s vloženým lístkem vrátí do patřičného regálu.



Obr. 16: Vrácení knihy (Zdroj: vlastní zpracování)

3.10 Nedostatky současného stavu

Veškeré záznamy jsou vedeny v papírové podobě a existují pouze ve formě originálu. Nejsou tedy žádným způsobem zálohovány. V případě ztráty či poškození některého z prostředků obsahující záznamy knihovny dojde k jejich trvalé ztrátě.

Dále je zde nedostupnost dat v daném okamžiku na jednom místě. S tím také z velké části souvisí problém s dohledáváním záznamů. To je buď zdlouhavé (např. dohledávání informací o čtenářích stejného jména a příjmení) nebo nemožné (např. dohledání vypůjčených knih čtenářem pomocí záznamů z čtenářského průkazu).

Krom toho je zde naprostá absence evidence knih vlastněných knihovnou.

4 VLASTÍ NÁVRHY ŘEŠENÍ

Nejdříve je proveden návrh (konceptuální, logická a fyzická úroveň) relační databáze naplňující hlavní cíl práce a zároveň odstraňující část zjištěných nedostatků, jimiž jsou nedostupnost dat v daném okamžiku na jednom místě, nevidování knihovou vlastněných knih a zdlouhavé či nemožné dohledávání záznamů.

Poté je vyřešeno umístění databáze a její zálohování, díky čemuž je odstraněn zjištěný nedostatek existence pouze originálních záznamů.

V závěru je vyřešeno fyzické zabezpečení dat databáze, přístup k databázi a budoucí rozvoj databáze.

4.1 Konceptuální úroveň návrhu databáze

V rámci konceptuální úrovně návrhu databáze je nejdříve provedena identifikace základních entit. Následně jsou určeny vztahy (včetně kardinality) mezi těmito entitami. Na závěr je vypracována základní verze ER diagramu.

4.1.1 Identifikace základních entit

Následující slovník dat reprezentovaný tabulkou obsahuje názvy identifikovaných základních entit, jejich popis a počet výskytů.

Tab. 4: Slovník dat (Zdroj: vlastní zpracování)

Název entity	Popis entity	Počet výskytů
Ctenar	Evidence informací o čtenářích (čtenáři, zákonní zástupci, popř. čtenáři/zákonní zástupci) knihovny	Řádově desítky až stovky
Adresa	Evidence kontaktních adres jednotlivých čtenářů	Řádově desítky až stovky
Tiskovina	Evidence tiskovin vlastněných knihovnou	Řádově stovky
Autor	Evidence autorů jednotlivých tiskovin vlastněných knihovnou	Řádově desítky až stovky
Telefon	Evidence telefonních čísel jednotlivých čtenářů	Řádově desítky až stovky
Email	Evidence emailových adres jednotlivých čtenářů	Řádově desítky až stovky
Platba	Evidence plateb jednotlivých čtenářů ve prospěch knihovny	Řádově desítky až stovky

4.1.2 Identifikace vztahů mezi základními entitami

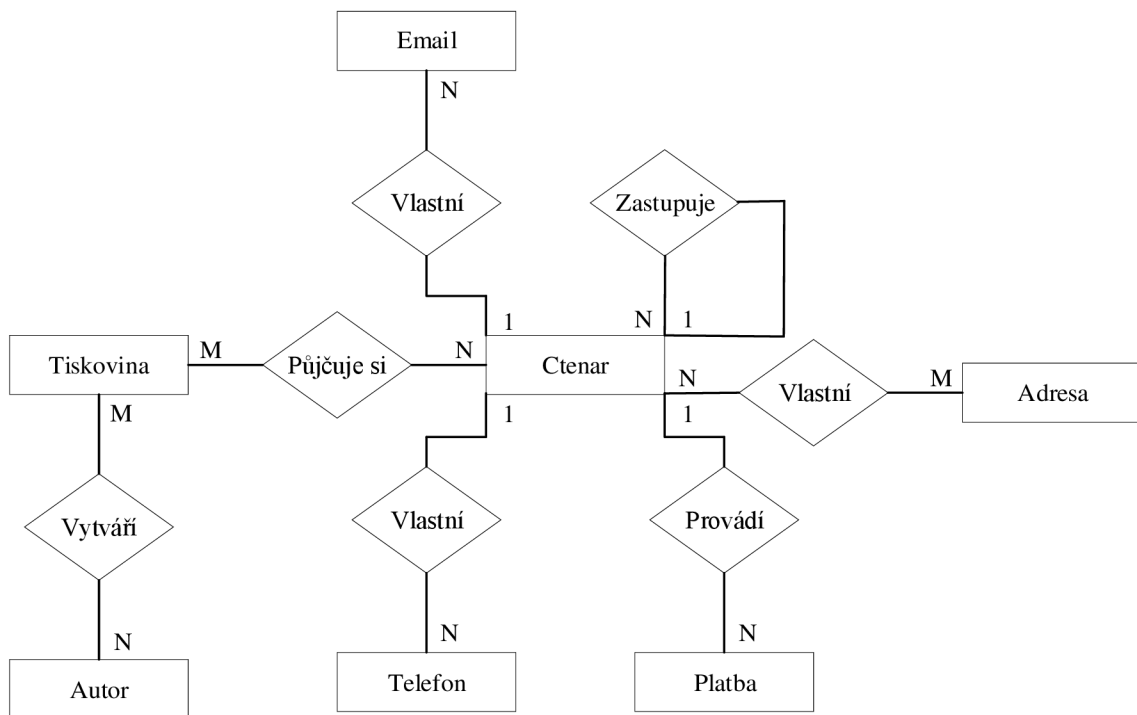
Identifikované vztahy mezi základními entitami eviduje následující tabulky. Pro každý vztah je uvedena jeho kardinalita.

Tab. 5: Vztahy mezi základními entitami (Zdroj: vlastní zpracování)

Název entity	Vztah	Kardinalita vztahu	Název entity
Ctenar	Vlastní	N:M	Adresa
Ctenar	Vlastní	1:N	Telefon
Ctenar	Vlastní	1:N	Email
Ctenar	Půjčuje si	N:M	Tiskovina
Ctenar	Provádí	1:N	Platba
Ctenar	Zastupuje	1:N	Ctenar
Autor	Vytváří	N:M	Tiskovina

4.1.3 Základní verze ER diagramu

Základní verze ER diagramu vychází z identifikace entit, vztahů mezi nimi a kardinality těchto vztahů.



Obr. 17: Základní verze ER diagramu (Zdroj: vlastní zpracování)

4.2 Logická úroveň návrhu databáze

V rámci logické úrovně návrhu databáze je nejdříve provedena dekompozice (určení průnikových entit) identifikovaných vztahů mezi základními entitami s kardinalitou N:M.

Následně jsou pro základní a průnikové entity navrženy tabulky. Kromě těchto tabulek jsou také navrženy tabulky sloužící jako číselníky vybraných atributů tabulek základních a průnikových entit.

Na závěr jsou všechny navržené tabulky zkontrolovány z hlediska integrity a normality a je vypracována konečná verze ER diagramu.

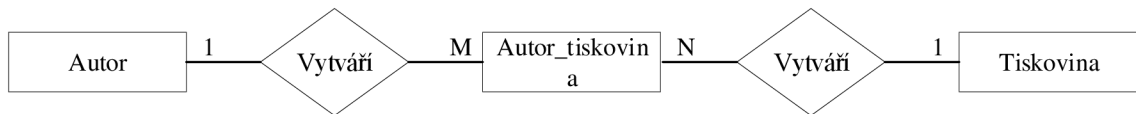
4.2.1 Dekompozice vztahů s kardinalitou N:M

Výsledkem dekompozice vztahů mezi základními entitami s kardinalitou N:M je identifikace nových tzv. průnikových entit.

Dekompozice vztahu Autor/Tiskovina



Obr. 18: Původní vztah Autor/Tiskovina (Zdroj: vlastní zpracování)

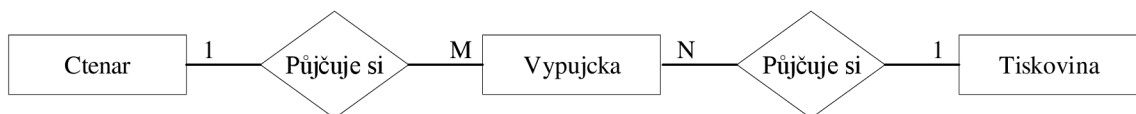


Obr. 19: Vztah Autor/Tiskovina po dekompozici (Zdroj: vlastní zpracování)

Dekompozice vztahu Ctenar/Tiskovina



Obr. 20: Původní vztah Ctenar/Tiskovina (Zdroj: vlastní zpracování)

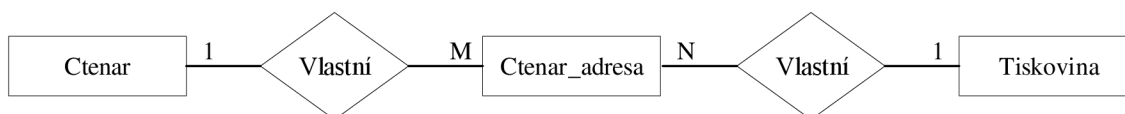


Obr. 21: Vztah Ctenar/Tiskovina po dekompozici (Zdroj: vlastní zpracování)

Dekompozice vztahu Ctenar/Adresa



Obr. 22: Původní vztah Ctenar/Adresa (Zdroj: vlastní zpracování)



Obr. 23: Vztah Ctenar/Adresa po dekompozici (Zdroj: vlastní zpracování)

Dekompozicí nově vzniklými průnikovými entitami jsou Autor_tiskovina, Vypujeka a Ctenar_adresa.

4.2.2 Návrh tabulek

Atribut/atributy tabulek odpovídající primárnímu klíči jsou označeny zkratkou PK a zelenou barvou, zatímco cizímu FK a barvou žlutou. Atributy, které jsou FK a zároveň součástí PK, jsou označeny zkratkou FK/PK.

Tabulka Ctenar

Tabulka *Ctenar* slouží pro evidenci čtenářů (čtenáři, zákonní zástupci, popř. čtenáři/zákonní zástupci) knihovny. Hodnota primárního klíče je generována automaticky. Pomocí cizích klíčů je tabulka propojena s tabulkami *Ctenar* (přiřazení zákonného zástupce čtenáři) a *Prukaz_ozp* (přiřazení průkazu OZP čtenáři).

Tab. 6: Tabulka Ctenar (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
ctenar_id (PK)	INTEGER		ANO	Jedinečný identifikátor čtenáře
jmeno	VARCHAR	15	ANO	Jméno čtenáře
prijmeni	VARCHAR	25	ANO	Příjmení čtenáře
datum_narozeni	DATE		ANO	Datum narození čtenáře
pohlavi	BIT		NE	Pohlaví čtenář (0 je muž, 1 je žena)
titul_pred	VARCHAR	25	NE	Titul/tituly čtenáře před jménem
titul_za	VARCHAR	20	NE	Titul/tituly čtenáře za jménem
zakonny_zastupce (FK)	INTEGER		NE	Propojení s tabulkou Ctenar
prukaz_ozp (FK)	INTEGER		NE	Propojení s tabulkou Prukaz_ozp

Tabulka Adresa

Tabulka *Adresa* slouží pro evidenci adres čtenářů. Hodnota primárního klíče je generována automaticky. Adresy jsou přiřazeny čtenářům pomocí tabulky *Ctenar_adresa*.

Tab. 7: Tabulka Adresa (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
adresa_id (PK)	INTEGER		ANO	Identifikační číslo adresy
psc	VARCHAR	5	ANO	Poštovní směrovací číslo
mesto	VARCHAR	25	ANO	Město
ulice	VARCHAR	25	ANO	Ulice, pokud neexistuje, stejná hodnota jako atribut mesto
cislo	VARCHAR	5	ANO	Číslo popisné

Tabulka Tiskovina

Tabulka *Tiskovina* slouží pro evidenci tiskovin (knihy a časopisy) vlastněných knihovnou. Hodnota primárního klíče je generována automaticky. Pomocí cizích klíčů je tabulka propojena s tabulkami *Tiskovina_jazyk* (přiřazení jazyka tiskovině), *Tiskovina_skdop* (přiřazení literární skupiny a doporučení tiskovině) a *Vydavatel* (přiřazení vydavatele tiskovině).

Tab. 8: Tabulka Tiskovina (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
tiskovina_id (PK)	INTEGER		ANO	Jedinečný identifikátor tiskoviny
nazev	VARCHAR	200	ANO	Název tiskoviny
druh	INTEGER		ANO	Druh tiskoviny (0 je časopis, 1 je kniha)
isbn_issn	VARCHAR	17	ANO	ISBN (kniha) nebo ISSN (časopis) podle druhu tiskoviny
rok_vydani	VARCHAR	4	ANO	Rok vydání tiskoviny
jazyk (FK)	INTEGER		ANO	Propojení s tabulkou Tiskovina_jazyk
rocnik	VARCHAR	4	NE	Ročník, pouze u časopisu
cislo	VARCHAR	3	NE	Číslo, pouze u časopisu
skdop (FK)	INTEGER		ANO	Propojení s tabulkou Tiskovina_skdop
vydavatel (FK)	INTEGER		ANO	Propojení s tabulkou Vydavatel
cena	VARCHAR	4	ANO	Požizovací cena tiskoviny bez slev v Kč

Tabulka Autor

Tabulka *Autor* slouží pro evidenci autorů tiskovin vlastněných knihovnou. Hodnota primárního klíče je generována automaticky.

Tab. 9: Tabulka Autor (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
autor_id (PK)	INTEGER		ANO	Jedinečný identifikátor autora
jmeno	VARCHAR	15	ANO	Jméno autora
prijmeni	VARCHAR	25	ANO	Příjmení autora
rok_narozeni	VARCHAR	4	ANO	Datum narození autora

Tabulka Telefon

Tabulka *Telefon* slouží pro evidenci telefonních čísel čtenářů. Hodnota primárního klíče je generována automaticky. Pomocí cizího klíče je tabulka propojena s tabulkou *Ctenar* (přiřazení čísla čtenáři).

Tab. 10: Tabulka Telefon (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
telefon_id (PK)	INTEGER		ANO	Jedinečný identifikátor telefonního čísla
cislo	VARCHAR	12	ANO	Telefonní číslo
ctenar (FK)	INTEGER		ANO	Propojení s tabulkou Ctenar

Tabulka Email

Tabulka *Email* slouží pro evidenci emailových adres čtenářů. Hodnota primárního klíče je generována automaticky. Pomocí cizího klíče je tabulka propojena s tabulkou *Ctenar* (přiřazení emailové adresy čtenáři).

Tab. 11: Tabulka Email (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
email_id (PK)	INTEGER		ANO	Jedinečný identifikátor emailové adresy
email	VARCHAR	50	ANO	Emailová adresa
ctenar (FK)	INTEGER		ANO	Propojení s tabulkou Ctenar

Tabulka Platba

Tabulka *Platba* slouží pro evidenci plateb čtenářů ve prospěch knihovny. Hodnota primárního klíče je generována automaticky. Pomocí cizích klíčů je tabulka propojena s tabulkami *Platba_predmet* (přiřazení předmětu platby platbě) a *Ctenar* (přiřazení platby čtenáři).

Tab. 12: Tabulka Platba (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
platba_id (PK)	INTEGER		ANO	Jedinečný identifikátor platby
predmet (FK)	INTEGER		ANO	Propojení s tabulkou Platba_predmet
hodnota	VARCHAR	4	ANO	Hodnota platby
popis	TEXT		NE	Podrobnosti platby, vztahující se k atributu predmet_platby
datum_vzniku	DATE		ANO	Datum vzniku požadavku na platbu
datum_uhrady	DATE		NE	Datum provedení platby
ctenar (FK)	INTEGER		ANO	Propojení s tabulkou Ctenar

Tabulka Vypujcka

Tabulka *Vypujcka* slouží pro evidenci výpůjček tiskovin knihovny čtenáři. Hodnota primárního klíče je generována automaticky. Pomocí cizích klíčů je tabulka propojena s tabulkami *Ctenar* (přiřazení výpůjčky čtenáři) a *Tiskovina* (přiřazení tiskoviny k výpůjčce).

Tab. 13: Tabulka Vypujcka (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
vypujcka_id (PK)	INTEGER		ANO	Identifikační číslo výpůjčky
ctenar (FK)	INTEGER		ANO	Propojení s tabulkou Ctenar
tiskovina (FK)	INTEGER		ANO	Propojení s tabulkou Tiskovina
datum_vypujcky	DATE		ANO	Datum vzniku výpůjčky
vraceno_dne	DATE		NE	Datum vrácení vypůjčené tiskoviny

Tabulka Autor_tiskovina

Tabulka *Autor_tiskovina* slouží pro přiřazení autorů tiskovinám. Primární klíč je složeninou z cizích klíčů odkazujících na tabulky *Autor*, *Tiskovina*.

Tab. 14: Tabulka Autor_tiskovina (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
autor (FK/PK)	INTEGER		ANO	Propojení s tabulkou Autor
tiskovina (FK/PK)	INTEGER		ANO	Propojení s tabulkou Tiskovina

Tabulka Ctenar_adresa

Tabulka *Ctenar_adresa* slouží pro přiřazení adres čtenářům. Primární klíč je složeninou z cizích klíčů odkazujících na tabulky *Ctenar*, *Adresa* a *Adresa_druh* (určení druhu adresy).

Důvodem přiřazení adresy čtenáři tímto způsobem je fakt, že někteří čtenáři vlastní stejnou adresu a tímto je zabráněno výskytu duplikátních záznamů v tabulce *Adresa*.

Tab. 15: Tabulka Ctenar_adresa (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
ctenar_id (FK/PK)	INTEGER		ANO	Propojení s tabulkou Ctenar
adresa_id (FK/PK)	INTEGER		ANO	Propojení s tabulkou Adresa
druh (FK/PK)	INTEGER		ANO	Propojení s tabulkou Adresa_druh

Tabulka Platba_predmet

Tabulka *Platba_predmet* je číselník a slouží pro evidenci předmětů plateb přiřaditelných platbám čtenářů ve prospěch knihovny. Hodnota primárního klíče je generována automaticky.

Tab. 16: Tabulka Platba_predmet (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
predmet_id (PK)	INTEGER		ANO	Jedinečný identifikátor předmětu platby
nazev	VARCHAR	20	ANO	Předmět platby, například zápisné

Tabulka Prukaz_ozp

Tabulka *Prukaz_ozp* je číselník a slouží pro evidenci průkazů OZP, přiřaditelných čtenářům. Hodnota primárního klíče je generována automaticky.

Tab. 17: Tabulka Prukaz_ozp (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
prukaz_id (PK)	INTEGER		ANO	Jedinečný identifikátor průkazu
nazev	VARCHAR	5	ANO	Název průkazu

Tabulka Tiskovina_druh

Tabulka *Tiskovina_druh* je číselník a slouží pro evidenci druhů tiskovin přiřaditelných tiskovinám. Hodnota primárního klíče je generována automaticky.

Tab. 18: Tabulka Tiskovina_druh (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
druh_id (PK)	INTEGER		ANO	Jedinečný identifikátor druhu tiskoviny
nazev	VARCHAR	7	ANO	Název druhu tiskoviny

Tabulka Tiskovina_jazyk

Tabulka *Tiskovina_jazyk* je číselník a slouží pro evidenci jazyků tiskovin přiřaditelných tiskovinám. Hodnota primárního klíče je generována automaticky.

Tab. 19: Tabulka Tiskovina_jazyk (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
jazyk_id (PK)	INTEGER		ANO	Jedinečný identifikátor druhu tiskoviny
nazev	VARCHAR	20	ANO	Název druhu tiskoviny

Tabulka Adresa_druh

Tabulka *Tiskovina_druh* je číselník a slouží pro evidenci druhů adres přiřaditelných spojením adres a čtenářů. Hodnota primárního klíče je generována automaticky.

Tab. 20: Tabulka Adresa_druh (Zdroj: vlastní zpracování)

Název atributu	Datový typ	Délka	NOT NULL	Popis
druh_id (PK)	INTEGER		ANO	Jedinečný identifikátor druhu adresy
nazev	VARCHAR	17	ANO	Název druhu adresy

Tabulka Vydavatel

Tabulka *Vydavatel* je číselník a slouží pro evidenci vydavatelů tiskovin přiřaditelných tiskovinám. Hodnota primárního klíče je generována automaticky.

Tab. 21: Tabulka Vydavatel (Zdroj: vlastní zpracování)

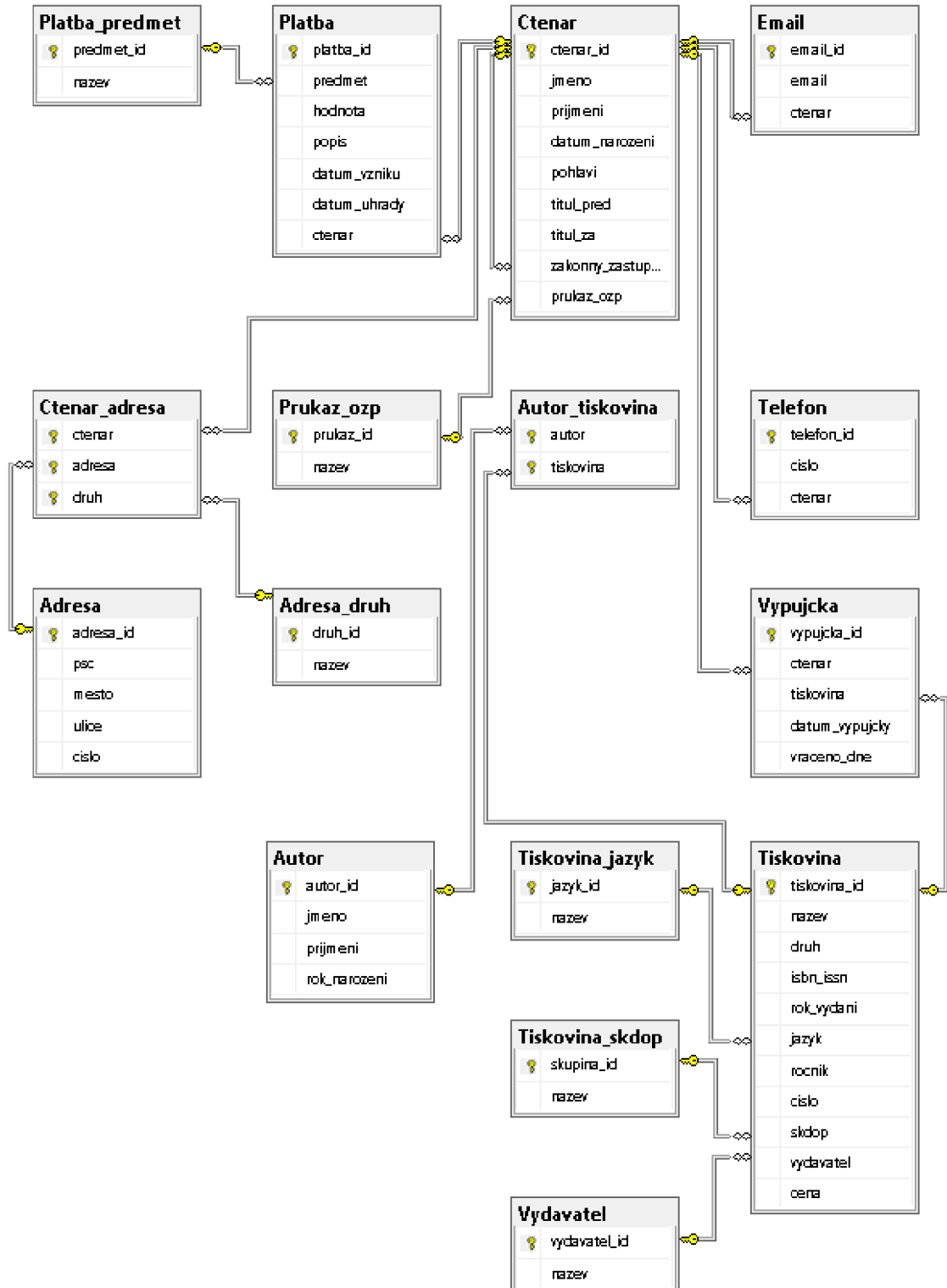
Název atributu	Datový typ	Délka	NOT NULL	Popis
vydavatel_id (PK)	INTEGER		ANO	Jedinečný identifikátor vydavatele
nazev	VARCHAR	100	ANO	Název vydavatele

4.2.3 Kontrola tabulek z hlediska integrity a normality

Všechny navržené tabulky splňují požadavky jak z pohledu integrity (integritní omezení entit), tak z pohledu normality (třetí normální forma).

4.2.4 Konečná verze ER diagramu

Konečná verze ER diagramu databáze zachycuje jednotlivé tabulky včetně struktury a vztahů mezi nimi.



Obr. 24: Konečná verze ER diagramu (Zdroj: vlastní zpracování)

4.3 Fyzická úroveň návrhu databáze

V rámci fyzické úrovně návrhu databáze je provedena implementace jednotlivých částí databáze v následujícím pořadí:

- implementace tabulek a vazem mezi nimi (příloha č. 1),
- naplnění tabulek testovacími daty (příloha č. 2),
- implementace několika pohledů a procedur (příloha č. 3).

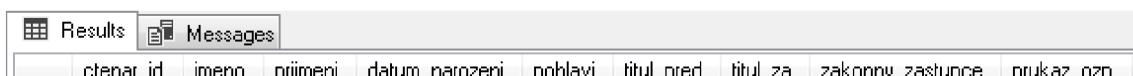
Implementace je provedena pomocí nástroje Microsoft SQL Server Management Studio 18. Vybrané příklady implementace jsou umístěny zde.

4.3.1 Implementace tabulky Ctenar včetně vazeb

Vazby tabulky *Ctenar* s jinými tabulkami jsou realizovány pomocí atributů *zakonny_zastupce* (vazba s tabulkou *Ctenar*) a *prukaz_ozp* (vazba s tabulkou *Prukaz_ozp*), které jsou cizími klíči.

```
create table Ctenar (  
ctenar_id integer not null identity (1,1) primary key,  
jmeno varchar (15) not null,  
prijmeni varchar (25) not null,  
datum_narozeni date not null,  
pohlavi bit,  
titul_pred varchar (25),  
titul_za varchar (20),  
zakonny_zastupce integer references Ctenar (ctenar_id),  
prukaz_ozp integer references Prukaz_ozp (prukaz_id)  
);
```

Náhled na vytvořenou tabulku je možné provést pomocí příkazu: `select*from Ctenar`.



ctenar_id	jmeno	prijmeni	datum_narozeni	pohlavi	titul_pred	titul_za	zakonny_zastupce	prukaz_ozp
-----------	-------	----------	----------------	---------	------------	----------	------------------	------------

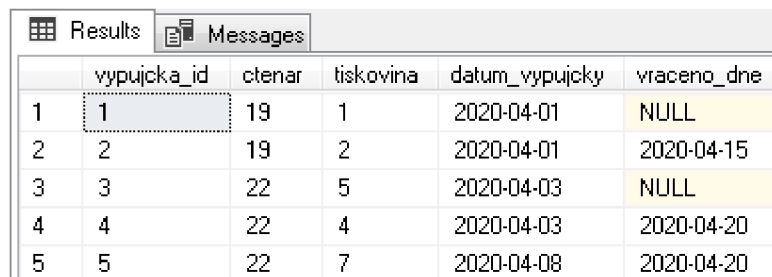
Obr. 25: Tabulka Ctenar (Zdroj: vlastní zpracování)

4.3.2 Naplnění tabulky Vypujcka testovacími daty

Tabulka *Vypujcka* je naplněna testovacími daty, na nichž je ověřena funkčnost pohledu *Vypujcky*.

```
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (19,1,'2020-4-1',null);
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (19,2,'2020-4-1','2020-4-15');
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,5,'2020-4-3',null);
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,4,'2020-4-3','2020-4-20');
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,7,'2020-4-8','2020-4-20');
```

Náhled na naplněnou tabulku je možné provést pomocí příkazu: `select*from Vypujcka`.



	vypujcka_id	ctenar	tiskovina	datum_vypujcky	vraceno_dne
1	1	19	1	2020-04-01	NULL
2	2	19	2	2020-04-01	2020-04-15
3	3	22	5	2020-04-03	NULL
4	4	22	4	2020-04-03	2020-04-20
5	5	22	7	2020-04-08	2020-04-20

Obr. 26: Tabulka *Vypujcka* naplněná testovacími daty (Zdroj: vlastní zpracování)

4.3.3 Implementace a ověření funkčnosti pohledu *Ctenari*

Pohled *Ctenari* zobrazuje všechny dostupné informace (id čtenáře, jméno vč. titulu, datum narození, pohlaví, průkaz OZP, adresa, email, telefonní číslo, id zákonného zástupce) evidované o jednotlivých čtenářích v databázi.

```
create view Ctenari as
select c.ctenar_id as 'Id čtenáře',
ltrim (isnull (c.titul_pred, "") + ' ' + c.jmeno + ' ' + c.prijmeni + ', ' + isnull (c.titul_za, ""))
as 'Jméno vč. titulu',
c.datum_narozeni as 'Datum narození',
replace (replace (isnull (cast (c.pohlavi as varchar), ""), 0, 'muž'), 1, 'žena') as 'Pohlaví',
isnull (h.nazev, "") as 'Průkaz OZP',
```

```

string_agg (isnull ((d.nazev + ':' + a.psc + ',' + a.mesto + ',' + a.ulice + ',' + a.cislo), ''),
',')
within group (order by d.nazev asc) as 'Adresa (druh: PSČ, město, ulice, číslo popisné)',
string_agg (isnull (e.email, ''), ',') as 'Email',
string_agg (isnull ('+' + t.cislo, ''), ',') as 'Telefoní číslo',
isnull (cast (c.zakonny_zastupce as varchar), '') as 'ID zákonného zástupce'
from Ctenar c
left join Prukaz_ozp h on c.prukaz_ozp = h.prukaz_id
left join Ctenar_adresa m on c.ctenar_id = m.ctenar
left join Adresa a on m.adresa = a.adresa_id
left join Adresa_druh d on m.druh = d.druh_id
left join Telefon t on t.ctenar = c.ctenar_id
left join Email e on e.ctenar = c.ctenar_id
group by c.ctenar_id, c.titul_pred, c.datum_narozeni, c.jmeno, c.prijmeni, c.titul_z,
c.pohlavi, c.zakonny_zastupce, h.nazev

```

Ověření funkčnosti pohledu je možné provést pomocí příkazu: `select*from Ctenari`.

	Id členáře	Jméno vč. titulu	Datum narození	Pohlaví	Průkaz OZP	Adresa (druh: PSČ, město, ulice, číslo popisné)
1	19	Bc. Jan Wolf,	1994-11-25	muž		Trvalá/Kontaktní: 39456, Senožaty, Senožaty, 14...
2	21	Jakub Vokal,	1993-10-18	muž		Trvalá/Kontaktní: 39456, Senožaty, Senožaty, 120
3	22	Bc. Dominika Krásná,	1996-11-20	žena		Kontaktní: 39456, Otavožaty, Otavožaty, 12, Trva...
4	23	Petr Šolc,	2000-01-11			Kontaktní: 39601, Humpolec, Jiráskova, 30, Trval...
5	24	Tereza Zelená,	1990-04-11	žena		
6	25	Martin Zelený,	2008-01-11	muž		
7	39	Martin Dvořák,	1994-09-21	muž		
8	20	Bc. Jan Vokal,	1994-11-19	muž	TP	Trvalá/Kontaktní: 39456, Senožaty, Senožaty, 120

Obr. 27: Pohled Ctenari, část 1 (Zdroj: vlastní zpracování)

Email	Telefoní číslo	ID zákonného zástupce
wolf.jan@seznam.cz, wolfik@seznam.cz	+420721596679, +420721596679	
domca.krasna@seznam.cz, domca.krasna@seznam.cz	+420720521654, +420720521654	
.	.	
		24

Obr. 28: Pohled Ctenari, část 2 (Zdroj: vlastní zpracování)

4.3.4 Implementace o ověření funkčnosti pohledu Tiskoviny

Pohled *Tiskoviny* zobrazuje všechny dostupné informace (id tiskoviny, název, autor, vydavatel, datum vytvoření, jazyk, identifikátor, ročník, číslo, skupina, doporučení, pořizovací cena) evidované o jednotlivých tiskovinách v databázi.

```
create view Tiskoviny as
select t.tiskovina_id as 'Id tiskoviny',
t.nazev as 'Název',
string_agg (isnull ((a.jmeno + ' ' + a.prijmeni + ' ' + a.rok_narozeni), ''), ', ')
within group (order by a.jmeno, a.prijmeni asc) as 'Autor',
v.nazev as 'Vydavatel',
t.rok_vydani as 'Datum vytvoření',
j.nazev as 'Jazyk',
(replace (replace (t.druh, 0, 'ISSN:'), 1, 'ISBN:') + ' ' + t.isbn_issn ) as 'Identifikátor',
isnull (t.rocnik, '') as 'Ročník',
isnull (t.cislo, '') as 'Číslo',
d.nazev as 'Skupina, doporučení',
t.cena as 'Pořizovací cena'
from Tiskovina t
left join Tiskovina_jazyk j on t.jazyk = j.jazyk_id
left join Tiskovina_skdop d on t.skdop = d.skupina_id
left join Vydavatel v on t.vydavatel = v.vydavatel_id
left join Autor_tiskovina m on m.tiskovina = t.tiskovina_id
left join Autor a on m.autor = a.autor_id
group by t.tiskovina_id, t.nazev, t.rok_vydani, d.nazev, t.rocnik, t.cislo, t.druh, t.cena,
v.nazev, j.nazev, t.isbn_issn
```

Ověření funkčnosti pohledu je možné provést pomocí příkazu: `select*from Tiskoviny`.

	Id tiskoviny	Název	Autor	Vydavatel
1	1	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM
2	2	Infrastruktura komunikačních systémů I, Univerzá...	Viktor Ondrák 1958, Vilém Jordán 1955	Brno, Akademické nakladatelství CERM
3	3	Číslicová technika	Marcela Antošová 1965, Václav Davídek 1942	České budějovice, Kopp
4	4	Můj boj/Tvůj boj	Ronda Rousey 1987	Bratislava, Noxi
5	5	Finanční tihy	Oldřich Rejnuš 1954	Praha, Grada
6	6	Automata, časopis pro automatizační techniku		Děčín, odborný časopis pro automatizační techniku
7	7	Automata, časopis pro automatizační techniku		Děčín, odborný časopis pro automatizační techniku
8	8	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM
9	9	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM

Obr. 29: Pohled Tiskoviny, část 1 (Zdroj: vlastní zpracování)

Datum vytvoření	Jazyk	Identifikátor	Ročník	Číslo	Skupina, doporučení	Požizovací cena
2010	Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200
2015	Čeština	ISBN: 978-80-214-5115-5			Naučná pro dospělé	400
2009	Čeština	ISBN: 978-80-7232-394-4			Naučná pro dospělé	400
2016	Čeština	ISBN: 978-80-8111-394-9			Beletrie	350
2014	Čeština	ISBN: 978-80-247-3671-6			Naučná pro dospělé	300
2018	Čeština	ISSN: 1210-9592	2018	1	Naučná pro dospělé	100
2018	Čeština	ISSN: 1210-9592	2018	2	Naučná pro dospělé	100
2010	Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200
2010	Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200

Obr. 30: Pohledu Tiskoviny, část 2 (Zdroj: vlastní zpracování)

4.3.5 Implementace a ověření funkčnosti procedury Najdi_tiskovinu

Procedura *Najdi_tiskovinu* po zadání vstupních parametrů (*nazev*, *identifikator*, *jmeno*, *prijmeni*, *rocnik*, *cislo*) vyhledá požadovanou tiskovinu, upozorní uživatele na její nalezení a poskytne náhled (využití pohledu *Tiskoviny*) na data o ní uložená.

Všechny vstupní parametry jsou nepovinné a *jmeno* a *prijmeni* zastupují jméno a příjmení autora tiskoviny. V případě, že daná tiskovina má více autorů, je vždy zadáno jméno a příjmení pouze jednoho z nich.

Pokud se daná tiskovina v databázi nenachází, je na tento fakt uživatel upozorněn.

```

create procedure Najdi_tiskovinu
@nazev varchar (200) = null,
@identifikator varchar (17) = null,
@jmeno varchar (15) = null,
@prijmeni varchar (25) = null,
@rocnik varchar (4) = null,
@cislo varchar (3) = null
as
begin
declare @ptiskovina_id integer
select @ptiskovina_id = t.tiskovina_id
from Tiskovina t
left join Autor_tiskovina m on t.tiskovina_id = m.tiskovina
left join Autor a on a.autor_id = m.autor
where @nazev = t.nazev or @identifikator = t.isbn_issn or @jmeno = a.jmeno or
@prijmeni = a.prijmeni or @rocnik = t.rocnik or @cislo = t.cislo
if @ptiskovina_id is null
begin
print 'Tiskovina nenalezena.'
end

```

```

else
begin
print 'Tiskovina nalezena.'
select * from Tiskoviny t
where @nazev = t.Název or @identifikator = t.Identifikátor or t.Autor like '%' +
@jmeno + '%' or t.Autor like '%' + @prijmeni + '%' or @rocnik = t.Ročník or @cislo =
t.Číslo
end
end
go

```

Možným příkazem ověření funkčnosti procedury pro případ, kdy se tiskovina nachází v databázi je: `execute Najdi_tiskovinu 'Datové a funkční modelování'`.

Id tiskoviny	Název	Autor	Vydavatel	Datum vytvoření
1	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM	2010
2	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM	2010
3	Datové a funkční modelování	Bernard Neuwirth 1978, Miloš Koch 1961	Brno, Akademické nakladatelství CERM	2010

Obr. 31: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 1 (Zdroj: vlastní zpracování)

Jazyk	Identifikátor	Ročník	Číslo	Skupina, doporučení	Pořizovací cena
Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200
Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200
Čeština	ISBN: 978-80-214-4125-5			Naučná pro dospělé	200

Obr. 32: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 2 (Zdroj: vlastní zpracování)

```

Tiskovina nalezena.

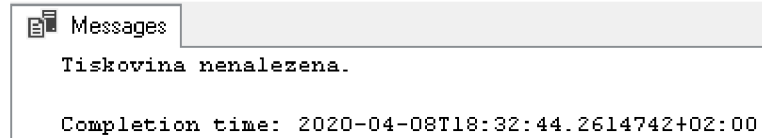
(3 rows affected)

Completion time: 2020-04-09T19:24:22.2662581+02:00

```

Obr. 33: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 3 (Zdroj: vlastní zpracování)

Možným příkazem ověření funkčnosti procedury pro případ, kdy se tiskovina nenachází v databázi je: `execute Najdi_tiskovinu 'Datamining'`.



```
Messages
Tiskovina nenalezena.

Completion time: 2020-04-08T18:32:44.2614742+02:00
```

Obr. 34: Procedura `Najdi_tiskovinu` (tiskovina se nenachází v databázi) (Zdroj: vlastní zpracování)

4.3.6 Implementace a ověření funkčnosti procedury `Vloz_ctenare`

Procedura `Vloz_ctenare` po zadání vstupních parametrů (*jmeno*, *prijmeni*, *datum_narozeni*, *pohlavi*, *titul_pred*, *titul_za*, *prukaz_orp*, *zakonny_zastupce*) vytvoří nového čtenáře v databázi nebo v případě jeho existence na tento fakt uživatele upozorní a poskytne náhled (využití pohledu *Ctenari*) na data o něm uložená.

Povinnými vstupními parametry jsou *jmeno*, *prijmeni* a *datum_narozeni*.

Ověření, zda již čtenář v databázi existuje, probíhá na základě porovnávání zadaných parametrů (*jmeno*, *prijmeni*, *datum_narozeni*) s informacemi o jednotlivých čtenářích uloženými v databázi.

```
create procedure Vloz_ctenare
@jmeno varchar (15),
@prijmeni varchar (25),
@datum_narozeni date,
@pohlavi integer = null,
@titul_pred varchar (25) = null,
@titul_za varchar (20) = null,
@prukaz_ozp_id integer = null,
@zakonny_zastupce_id integer = null
as
begin
declare @pctenar_id integer
select @pctenar_id = c.ctenar_id
from Ctenar c
where @jmeno = c.jmeno and @prijmeni = c.prijmeni and @datum_narozeni =
c.datum_narozeni
if @pctenar_id is null
begin
```

```

insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_z,
zakonny_zastupce, prukaz_ozp)
values (ltrim (@jmeno), ltrim (@prijmeni), ltrim(@datum_narozeni), @pohlavi,
ltrim(@titul_pred), ltrim(@titul_z),@zakonny_zastupce_id, @prukaz_ozp_id)
print 'Čtenář vložen do databáze.'
end
else
begin
print 'Čtenář je již v databázi.'
select * from Ctenari c
where @pctenar_id = c.[Id čtenáře]
end
end
go

```

Možným příkazem ověření funkčnosti procedury pro případ, kdy se čtenář nachází v databázi je: `execute Vloz_ctenare @jmeno = 'Jan', @prijmeni = 'Wolf', @datum_narozeni = '1994-11-25'.`

Id čtenáře	Jméno vč. titulu	Datum narození	Pohlaví	Průkaz OZP	Adresa (druh: PSČ, město, ulice, číslo popisné)
1	Bc. Jan Wolf,	1994-11-25	muž		Trvalá/Kontaktní: 39456, Senožaty, Senožaty, 14...

Obr. 35: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 1 (Zdroj: vlastní zpracování)

Email	Telefonní číslo	ID zákonného zástupce
wolf.jan@seznam.cz, wolfik@seznam.cz	+420721596679, +420721596679	

Obr. 36: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 2 (Zdroj: vlastní zpracování)

```

Čtenář je již v databázi.

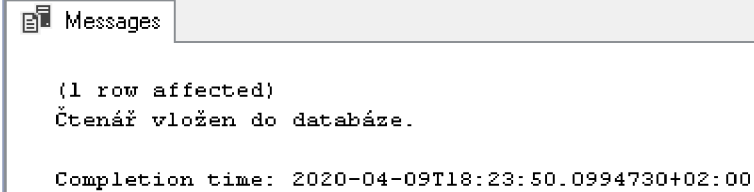
(1 row affected)

Completion time: 2020-04-09T17:26:45.7807133+02:00

```

Obr. 37: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 3 (Zdroj: vlastní zpracování)

Možným příkazem ověření funkčnosti procedury pro případ, kdy se čtenář nenachází v databázi je: `execute Vloz_ctenare @jmeno = 'Martin', @prijmeni = 'Kyselka', @datum_narozeni = '1990-8-13'`.



```
Messages  
  
(1 row affected)  
Čtenář vložen do databáze.  
  
Completion time: 2020-04-09T18:23:50.0994730+02:00
```

Obr. 38: Procedura Vloz_ctenare (čtenář se nenachází v databázi) (Zdroj: vlastní zpracování)

4.4 Umístění databáze

Umístěním databáze je server, který je součástí lokální sítě sdílené jednotlivými složkami obce kromě knihovny. Technické parametry včetně operačního systému toho serveru jsou uvedena v kapitole 3.2.1.

4.5 Přístup k databázi

Klientská stanice knihovny bude začleněna do lokální sítě sdílené jednotlivými složkami obce. Knihovnici bude vytvořen doménový účet, skrze něj získá právo přístupu k databázi.

Technické parametry klientské stanice knihovny včetně operačního systému (podpora doménových účtů) jsou uvedeny v kapitole 3.3.1.

4.6 Zálohování databáze

Umístění databáze řeší problém spojený se zálohováním dat uložených v databázi. Na server je aplikován dvojestupňový systém zálohování, díky němuž data databáze existují ve třech kopiích a je dodrženo zálohovací pravidlo 3-2-1.

System zálohování je popsán v kapitole 3.5. Technické parametry typu datového úložiště NAS, užitého v obou stupních systému zálohování, jsou uvedeny v kapitole 3.3.2.

4.7 Fyzické zabezpečení dat databáze

Jednotlivá zařízení, na kterých jsou data databáze uložena, jsou umístěna v prostorách zabezpečených ať už proti živelným pohromám nebo nekalým živlům.

Umístění a zabezpečení prostor je popsáno v kapitole 3.1.2.

4.8 Budoucí rozvoj databáze

První možností budoucího rozvoje je doprogramování dalších nástrojů (pohledy, procedury atd.) pro práci s databází dle potřeby (např. potřeba zjištění jaké literární skupiny si dané pohlaví nejvíce půjčuje knihy) bez vytvoření aplikace napojené na databázi. Uživatel tak bude přímo databázi obsluhovat pomocí zmíněných nástrojů, což může být pro něj uživatelsky nepřívětivé.

Druhou možností je, ať již vlastní silou nebo externím subjektem, vytvoření uživatelsky přívětivé aplikace napojené na databázi. Pomocí uživatelského rozhraní aplikace pak bude moci uživatel s databází (daty) manipulovat, aniž by přímo on sám pracoval s pohledy, procedurami atd.

ZÁVĚR

Hlavním cílem práce bylo vytvoření funkčního návrhu relační databáze pro obecní knihovnu obce Senožaty.

Díličními cíli práce byly:

vypracování teoretické, analytické a návrhové části,

začlenění řešení nedostatků, zjištěných během vypracování analytické části, do návrhové části ať již jako součást návrhu databáze nebo samostatně,

vyřešení fyzického zabezpečení databáze, přístupu k databázi a budoucího rozvoj databáze.

Všech daných cílů, ať hlavního nebo dílčích, bylo úspěšně dosaženo. Práce zahrnuje tři stěžejní části, kterými jsou teoretická, analytická a návrhová. Řešení nedostatků (nedostupnost dat v daném okamžiku na jednom místě, neevidování knihovou vlastněných knih, zdlouhavé či nemožné dohledávání záznamů a existence pouze originálních záznamů), zjištěných během vypracování analytické části, jsou začleněna do části návrhové, a to ať jako součást návrhu nebo samostatně. Kromě zmíněného je také vyřešeno fyzické zabezpečení databáze, přístup k databázi a budoucí rozvoj databáze.

SEZNAM POUŽITÝCH ZDROJŮ

- (1) CONOLLY, Thomas, Carolyn E. BEGG a Richard HOLOWCZAK. Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází. Brno: Computer Press, 2009. ISBN 978-80-251-2328-7.
- (2) KOCH, Miloš a Bernard NEUWIRTH. Datové a funkční modelování. 4. vyd. Brno: Akademické nakladatelství CERM, 2010. ISBN 978-80-214-4125-5.
- (3) OPPEL, Andrew J. SQL bez předchozích znalostí: [průvodce pro samouky]. Brno: Computer Press, 2008. ISBN 978-80-251-1707-1.
- (4) STEPHENS, Ryan K., Ronald R. PLEW a Arie JONES. Naučte se SQL za 28 dní: [stačí hodina denně]. Brno: Computer Press, 2010. ISBN 978-80-251-2700-1.
- (5) KROENKE, David a David J. AUER. Databáze. Brno: Computer Press, 2015. ISBN 978-80-251-4352-0.
- (6) KŘÍŽ, Jiří. Databázové systémy - DBS: DS a architektury DBMS: Metodologie návrhu DB. Brno, 2016, 1 a 2. přednáška.
- (7) HERNANDEZ, Michael J. Návrh databází. Praha: Grada, 2006. Profesionál. ISBN isbn80-247-0900-7.
- (8) KŘÍŽ, Jiří. Databázové systémy - DBS: Metodologie návrhu DB: Logické schéma a fyzické schéma. 3. přednáška. Brno, 2016.
- (9) KŘÍŽ, Jiří a Jan LUHAN. Databázové systémy: Dotazovací jazyk SQL. 5. přednáška. Brno, 2016.
- (10) GROFF, James R. a Paul N. WEINBERG. SQL: kompletní průvodce. Brno: CP Books, 2005. Programování (CP Books). ISBN isbn80-251-0369-2.
- (11) KŘÍŽ, Jiří. Databázové systémy - DBS: Uložené procedury (parametry, proměnné, řízení roku programu, vnořené procedury). 7. přednáška. Brno, 2016.
- (12) OSHMAN, Shaun. Three rules of data backup for business. BizWest. 2016, 35(10), 20. ISSN 23345721.
- (13) CONRAD, Eric, Seth MISENAR, Joshua FELDMAN a Kevin RIGGINS. Eleventh hour CISSP: study guide. Second edition. Amsterdam: Elsevier, 2014. ISBN 01-241-7142-7.
- (14) Diferenciální - rozdílová záloha. Acronis - softwarové nástroje pro zálohování a ochranu dat [online]. Ostrava: Zebra systems [cit. 2018-03-15]. Dostupné z: <https://www.acronis.cz/kb/diferencialni-zaloha/>

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

DBMS	Database Management System
EZS	Elektronický Zabezpečovací Systém
FK	Foreign Key
NAS	Network Attached Storage
PK	Primary Key
RAID	Redundant Array of Independent Disks
SQL	Structured Query Language
UPS	Uninterruptible Power Supply

SEZNAM OBRÁZKŮ

Obr. 1: Vztah mezi data a informace	17
Obr. 2: Lineární datový model.....	20
Obr. 3: Objektový datový model	20
Obr. 4: Relační datový model	21
Obr. 5: Fáze životního cyklu vývoje databázového systému.....	25
Obr. 6: Zařazení obecní knihovny v rámci obce.....	33
Obr. 7: Zařazení knihovnice v rámci obecního úřadu	34
Obr. 8: 1. patro budovy obecního úřadu	35
Obr. 9: Lokální počítačová síť	36
Obr. 10: 1. stupeň zálohy	39
Obr. 11: 2. stupeň zálohy	40
Obr. 12: Přihlášení čtenáře (bez ohledu na věk).....	42
Obr. 13: Čtenářský průkaz	45
Obr. 14: Knižní lístek.....	46
Obr. 15: Výpůjčka knihy	47
Obr. 16: Vrácení knihy	48
Obr. 17: Základní verze ER diagramu	53
Obr. 18: Původní vztah Autor/Tiskovina.....	54
Obr. 19: Vztah Autor/Tiskovina po dekompozici	54
Obr. 20: Původní vztah Ctenar/Tiskovina	54

Obr. 21: Vztah Ctenar/Tiskovina po dekompozici	54
Obr. 22: Původní vztah Ctenar/Adresa	55
Obr. 23: Vztah Ctenar/Adresa po dekompozici.....	55
Obr. 24: Konečná verze ER diagramu	67
Obr. 25: Tabulka Ctenar	68
Obr. 26: Tabulka Vypujcka naplněná testovacími daty.....	69
Obr. 27: Pohled Ctenari, část 1	70
Obr. 28: Pohled Ctenari, část 2	70
Obr. 29: Pohled Tiskoviny, část 1.....	71
Obr. 30: Pohledu Tiskoviny, část 2.....	72
Obr. 31: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 1.....	73
Obr. 32: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 2.....	73
Obr. 33: Procedura Najdi_tiskovinu (tiskovina se nachází v databázi), část 3.....	73
Obr. 34: Procedura Najdi_tiskovinu (tiskovina se nenachází v databázi).....	74
Obr. 35: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 1	75
Obr. 36: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 2	75
Obr. 37: Procedura Vloz_ctenare (čtenář se nachází v databázi), část 3	75
Obr. 38: Procedura Vloz_ctenare (čtenář se nenachází v databázi)	76

SEZNAM TABULEK

Tab. 1: Provoz knihovny.....	33
Tab. 2: Značení skupin literatury	43
Tab. 3: Rozdělení beletrie pro mládež	44
Tab. 4: Slovník dat.....	51
Tab. 5: Vztahy mezi základními entitami	52
Tab. 6: Tabulka Ctenar	56
Tab. 7: Tabulka Adresa.....	57
Tab. 8: Tabulka Tiskovina	58
Tab. 9: Tabulka Autor.....	59
Tab. 10: Tabulka Telefon.....	59
Tab. 11: Tabulka Email	60
Tab. 12: Tabulka Platba	61
Tab. 13: Tabulka Vypujcka	62
Tab. 14: Tabulka Autor_tiskovina	62
Tab. 15: Tabulka Ctenar_adresa	63
Tab. 16: Tabulka Platba_predmet.....	64
Tab. 17: Tabulka Prukaz_ozp	64
Tab. 18: Tabulka Tiskovina_druh.....	65
Tab. 19: Tabulka Tiskovina_jazyk	65
Tab. 20: Tabulka Adresa_druh	66

Tab. 21: Tabulka Vydavatel.....	66
---------------------------------	----

SEZNAM PŘÍLOH

Příloha č. 1: Implementace tabulek a vazeb mezi nimi

Příloha č. 2: Naplnění tabulek testovacími daty

Příloha č. 3: Implementace několika pohledů a procedur

PŘÍLOHY

Příloha č. 1: Implementace tabulek a vazeb mezi nimi (Zdroj: vlastní zpracování)

```
-- implementace tabulky Pukaz_ozp
create table Pukaz_ozp (
prukaz_id integer not null identity (1,1) primary key,
nazev varchar (5) not null
);

-- implementace tabulky Ctenar
create table Ctenar (
ctenar_id integer not null identity (1,1) primary key,
jmeno varchar (15) not null,
prijmeni varchar (25) not null,
datum_narozeni date not null,
pohlavi bit,
titul_pred varchar (25),
titul_za varchar (20),
zakonny_zastupce integer references Ctenar (ctenar_id),
prukaz_ozp integer references Pukaz_ozp (prukaz_id)
);

-- implementace tabulky Adresa
create table Adresa (
adresa_id integer not null identity (1,1) primary key,
psc varchar(5) not null,
mesto varchar (25) not null,
ulice varchar (25) not null,
cislo varchar (5) not null
);

-- implementace tabulky Tiskovina_jazyk
create table Tiskovina_jazyk (
jazyk_id integer not null identity (1,1) primary key,
nazev varchar (20)
);

-- implementace tabulky Tiskovina_skdop
create table Tiskovina_skdop (
skupina_id integer not null identity (1,1) primary key,
nazev varchar (33) not null
);

-- implementace tabulky Vydavatel
create table Vydavatel (
vydavatel_id integer not null identity (1,1) primary key,
nazev varchar (100) not null
);
```

```

-- implementace tabulky Tiskovina
create table Tiskovina (
tiskovina_id integer not null identity (1,1) primary key,
nazev varchar (200) not null,
druh bit not null,
isbn_issn varchar (17) not null,
rok_vydani varchar (4) not null,
jazyk integer not null references Tiskovina_jazyk (jazyk_id),
rocnik varchar (4),
cislo varchar (3),
skdop integer not null references Tiskovina_skdop (skupina_id),
vydavatel integer not null references Vydavatel (vydavatel_id),
cena varchar (4) not null
);

-- implementace tabulky Autor
create table Autor (
autor_id integer not null identity (1,1) primary key,
jmeno varchar (15) not null,
prijmeni varchar (25) not null,
rok_narozeni varchar (4) not null,
);

-- implementace tabulky Telefon
create table Telefon (
telefon_id integer not null identity (1,1) primary key,
cislo varchar (12) not null,
ctenar integer not null references ctenar (ctenar_id)
);

-- implementace tabulky Email
create table Email (
email_id integer not null identity (1,1) primary key,
email varchar (50) not null,
ctenar integer not null references ctenar (ctenar_id)
);

-- implementace tabulky Platba_predmet
create table Platba_predmet (
predmet_id integer not null identity (1,1) primary key,
nazev varchar (20) not null,
);

-- implementace tabulky Platba
create table Platba (
platba_id integer not null identity (1,1) primary key,
predmet integer not null references Platba_predmet (predmet_id),
hodnota varchar (4) not null,
popis text,
datum_vzniku date not null,
datum_uhrady date,
);

```

```

ctenar integer not null references Ctenar (ctenar_id),
);

-- implementace tabulky Vypujcka
create table Vypujcka (
vypujcka_id integer not null identity (1,1) primary key,
ctenar integer not null references Ctenar (ctenar_id),
tiskovina integer not null references Tiskovina (tiskovina_id),
datum_vypujcky date not null,
vraceno_dne date
);

-- implementace tabulky Autor_tiskovina
create table Autor_tiskovina (
autor integer not null references Autor (autor_id),
tiskovina integer not null references Tiskovina (tiskovina_id),
primary key (autor, tiskovina)
);

-- implementace tabulky Adresa_druh
create table Adresa_druh (
druh_id integer not null identity (1,1) primary key,
nazev varchar (17) not null
);

-- implementace tabulky Ctenar_adresa
create table Ctenar_adresa (
ctenar integer not null references Ctenar (ctenar_id),
adresa integer not null references Adresa (adresa_id),
druh integer not null references Adresa_druh (druh_id),
primary key (ctenar, adresa, druh)
);

```

Příloha č. 2: Naplnění tabulek testovacími daty

```

-- naplnění tabulky Vydavatel
insert into Vydavatel (nazev)
values ('Brno, Akademické nakladatelství CERM');
insert into Vydavatel (nazev)
values ('Praha, Grada');
insert into Vydavatel (nazev)
values ('České budějovice, Kopp');
insert into Vydavatel (nazev)
values ('Děčín, odborný časopis pro automatizační techniku');
insert into Vydavatel (nazev)
values ('Bratislava, Noxi');

-- naplnění tabulky Tiskovina_jazyk
insert into Tiskovina_jazyk (nazev)
values ('Čeština');

```

```

insert into Tiskovina_jazyk (nazev)
values ('Angličtina');
insert into Tiskovina_jazyk (nazev)
values ('Slovenština');
insert into Tiskovina_jazyk (nazev)
values ('Němčina');

-- naplnění tabulky Tiskovina_skdop
insert into Tiskovina_skdop (nazev)
values ('Beletrie');
insert into Tiskovina_skdop (nazev)
values ('Naučná pro dospělé');
insert into Tiskovina_skdop (nazev)
values ('Mládež naučná');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, první čtení');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, pohádky');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, příběhy');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, školní četba');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, dívčí četba');
insert into Tiskovina_skdop (nazev)
values ('Mládež beletrie, dobrodružství');

--naplnění tabulky Pukaz_ozp
insert into Pukaz_ozp (nazev)
values ('TP');
insert into Pukaz_ozp (nazev)
values ('ZTP');
insert into Pukaz_ozp (nazev)
values ('ZTP/P');

-- naplnění tabulky Platba_predmet
insert into Platba_predmet (nazev)
values ('Zápisné');
insert into Platba_predmet (nazev)
values ('Uživatelská poplatek');
insert into Platba_predmet (nazev)
values ('Pokuta');
insert into Platba_predmet (nazev)
values ('Přístup na internet');

-- naplnění tabulky Adresa_druh
insert into Adresa_druh (nazev)
values ('Trvalá');
insert into Adresa_druh (nazev)
values ('Kontaktní');
insert into Adresa_druh (nazev)

```

```

values ('Trvalá/Kontaktní');

-- naplnění tabulky Tiskovina
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Datové a funkční modelování',1,'978-80-214-4125-
5','2010',1,2,1,'200',null,null);
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Infrastruktura komunikačních systémů I, Univerzální kabelážní systémy',1,'978-
80-214-5115-5','2015',1,2,1,'400',null,null);
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Číslicová technika',1,'978-80-7232-394-4','2009',1,2,3,'400',null,null);
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Můj boj/Tvůj boj',1,'978-80-8111-394-9','2016',1,1,5,'350',null,null);
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Finanční trhy',1,'978-80-247-3671-6','2014',1,2,2,'300',null,null);
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Automata, časopis pro automatizační techniku',0,'1210-
9592','2018',1,2,4,'100','2018','1');
insert into Tiskovina (nazev, druh, isbn_issn, rok_vydani, jazyk, skdop, vydavatel, cena,
rocnik, cislo)
values ('Automata, časopis pro automatizační techniku',0,'1210-
9592','2018',1,2,4,'100','2018','2');

-- naplnění tabulky Autor
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Miloš','Koch','1961');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Bernard','Neuwirth','1978');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Ronda','Rousey','1987');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Oldřich','Rejnuš','1954');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Vilém','Jordán','1955');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Viktor','Ondrák','1958');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Marcela','Antošová','1965');
insert into Autor (jmeno, prijmeni, rok_narozeni)
values ('Vratislav','Davídek','1942');

-- naplnění tabulky Autor_tiskovina
insert into Autor_tiskovina (autor, tiskovina)
values (1,1);
insert into Autor_tiskovina (autor, tiskovina)

```

```

values (2,1);
insert into Autor_tiskovina (autor, tiskovina)
values (5,2);
insert into Autor_tiskovina (autor, tiskovina)
values (6,2);
insert into Autor_tiskovina (autor, tiskovina)
values (7,3);
insert into Autor_tiskovina (autor, tiskovina)
values (8,3);
insert into Autor_tiskovina (autor, tiskovina)
values (3,4);
insert into Autor_tiskovina (autor, tiskovina)
values (4,5);
insert into Autor_tiskovina (autor, tiskovina)
values (1,8);
insert into Autor_tiskovina (autor, tiskovina)
values (2,8);
insert into Autor_tiskovina (autor, tiskovina)
values (1,9);
insert into Autor_tiskovina (autor, tiskovina)
values (2,9);

```

-- naplnění tabulky Adresa

```

insert into Adresa (psc, mesto, ulice, cislo)
values ('39456','Senožaty','Senožaty','145');
insert into Adresa (psc, mesto, ulice, cislo)
values ('39456','Senožaty','Senožaty','120');
insert into Adresa (psc, mesto, ulice, cislo)
values ('39601','Humpolec','Jiráskova','30');
insert into Adresa (psc, mesto, ulice, cislo)
values ('39456','Otavožaty','Otavožaty','12');

```

-- naplnění tabulky Ctenar

```

insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Jan','Wolf','1994-11-25',0,'Bc.',null,null,null);
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Jan','Vokal','1994-11-19',0,'Bc.',null,null,1);
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Jakub','Vokal','1993-10-18',0,null,null,null,null);
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Dominika','Krásná','1996-11-20',1,'Bc.',null,null,null);
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Petr','Šolc','2000-1-11',null,null,null,null,null);
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Matin','Zelený','2008-1-11',0,null,null,24,null);

```

```
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values ('Tereza','Zelený','1990-4-11',1,null,null, null, null);
```

```
-- naplnění tabulky Ctenar_adresa
```

```
insert into Ctenar_adresa (ctenar, adresa, druh)
values (19,1,3);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (20,2,3);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (21,2,3);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (22,3,1);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (22,4,2);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (23,3,2);
insert into Ctenar_adresa (ctenar, adresa, druh)
values (23,4,1);
```

```
-- naplnění tabulky Email
```

```
insert into Email (email,ctenar)
values ('wolf.jan@seznam.cz',19);
insert into Email (email,ctenar)
values ('wolfik@seznam.cz',19);
insert into Email (email,ctenar)
values ('domca.krasna@seznam.cz',22);
```

```
-- naplnění tabulky Telefon
```

```
insert into Telefon (cislo, ctenar)
values ('420721596679',19);
insert into Telefon (cislo,ctenar)
values ('420720521654',22);
```

```
-- naplnění tabulky Platba
```

```
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-3-1','2020-3-1',19);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-3-5','2020-3-5',20);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-3-7','2020-3-7',21);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-2-1','2020-2-1',22);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-3-12','2020-3-12',23);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (2,'150','u. poplatek za rok 2020','2020-3-1','2020-3-1',19);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (2,'150','u. poplatek za rok 2020','2020-2-1','2020-3-1',22);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (2,'150','u. poplatek za rok 2020','2020-3-12','2020-3-1',23);
```



```
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (4,'50','1 hodnina','2020-3-1','2020-3-13',23);
insert into Platba (predmet, hodnota, popis, datum_vzniku, datum_uhrady, ctenar)
values (1,'100',null,'2020-3-1','2020-3-1',25);
```

-- naplnění tabulka Vypujcka

```
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (19,1,'2020-4-1',null);
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (19,2,'2020-4-1','2020-4-15');
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,5,'2020-4-3',null);
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,4,'2020-4-3','2020-4-20');
insert into Vypujcka (ctenar, tiskovina, datum_vypujcky, vraceno_dne)
values (22,7,'2020-4-8','2020-4-20');
```

Příloha č. 3: Implementace několika pohledů a procedur

-- implementace pohledu Ctenari

```
create view Ctenari as
select c.ctenar_id as 'Id čtenáře',
ltrim (isnull (c.titul_pred, '') + ' ' + c.jmeno + ' ' + c.prijmeni + ', ' + isnull (c.titul_za, ''))
as 'Jméno vč. titulu',
c.datum_narozeni as 'Datum narození',
replace (replace (isnull (cast (c.pohlavi as varchar), ''), 0, 'muž'), 1, 'žena') as 'Pohlaví',
isnull (h.nazev, '') as 'Průkaz OZP',
string_agg (isnull ((d.nazev + ': ' + a.psc + ', ' + a.mesto + ', ' + a.ulice + ', ' + a.cislo), ''),
', ')
within group (order by d.nazev asc) as 'Adresa (druh: PSČ, město, ulice, číslo popisné)',
string_agg (isnull (e.email, ''), ', ') as 'Email',
string_agg (isnull ('+' + t.cislo, ''), ', ') as 'Telefoní číslo',
isnull (cast (c.zakonny_zastupce as varchar), '') as 'ID zákonného zástupce'
from Ctenar c
left join Prukaz_ozp h on c.prukaz_ozp = h.prukaz_id
left join Ctenar_adresa m on c.ctenar_id = m.ctenar
left join Adresa a on m.adresa = a.adresa_id
left join Adresa_druh d on m.druh = d.druh_id
left join Telefon t on t.ctenar = c.ctenar_id
left join Email e on e.ctenar = c.ctenar_id
group by c.ctenar_id, c.titul_pred, c.datum_narozeni, c.jmeno, c.prijmeni, c.titul_za,
c.pohlavi, c.zakonny_zastupce, h.nazev
```

-- implementace pohledu Tiskoviny

```
create view Tiskoviny as
select t.tiskovina_id as 'Id tiskoviny',
t.nazev as 'Název',
string_agg (isnull ((a.jmeno + ' ' + a.prijmeni + ' ' + a.rok_narozeni), ''), ', ')
within group (order by a.jmeno, a.prijmeni asc) as 'Autor',
v.nazev as 'Vydavatel',
```

```

t.rok_vydani as 'Datum vytvoření',
j.nazev as 'Jazyk',
(replace (replace (t.druh, 0, 'ISSN:'), 1, 'ISBN:') + ' ' + t.isbn_issn ) as 'Identifikátor',
isnull (t.rocnik, '') as 'Ročník',
isnull (t.cislo, '') as 'Číslo',
d.nazev as 'Skupina, doporučení',
t.cena as 'Pořizovací cena'
from Tiskovina t
left join Tiskovina_jazyk j on t.jazyk = j.jazyk_id
left join Tiskovina_skdop d on t.skdop = d.skupina_id
left join Vydavatel v on t.vydavatel = v.vydavatel_id
left join Autor_tiskovina m on m.tiskovina = t.tiskovina_id
left join Autor a on m.autor = a.autor_id
group by t.tiskovina_id, t.nazev, t.rok_vydani, d.nazev, t.rocnik, t.cislo, t.druh, t.cena,
v.nazev, j.nazev, t.isbn_issn

```

-- implementace pohledu Vypujcky

```

create view Vypujcky as
select v.vypujcka_id as 'Id výpůjčky',
(cast (v.tiskovina as varchar) + ', ' + t.nazev) as 'Tiskovina (id, název)',
ltrim (cast (v.ctenar as varchar) + ', ' + isnull (c.titul_pred, '') + ' ' + c.jmeno + ' ' + c.
prijmeni + ', ' + isnull (c.titul_za, '')) as 'Čtenář (id, jméno vč. titulu)',
datum_vypujcky as 'Začátek výpůjčky',
dateadd (day, 28, datum_vypujcky) as 'Konec výpůjčky',
isnull (cast (vraceno_dne as varchar), '') as 'Vraceno dne'
from Vypujcka v
left join Tiskovina t on v.tiskovina = t.tiskovina_id
left join Ctenar c on v.ctenar = c.ctenar_id

```

-- implementace pohledu Platby

```

create view Platby as
select p.platba_id as 'Id platby',
n.nazev as 'Předmět platby',
(p.hodnota + ' Kč') as 'Částka',
isnull (p.popis, '') as 'Popis platby',
p.datum_vzniku as 'Datum požadavku na platbu',
p.datum_uhrady as 'Datum provedení platby',
ltrim (cast (p.ctenar as varchar) + ', ' + isnull (c.titul_pred, '') + ' ' + c.jmeno + ' ' + c.
prijmeni + ', ' + isnull (c.titul_za, '')) as 'Čtenář (id, jméno vč. titulu)'
from Platba p
left join Platba_predmet n on p.predmet = n.predmet_id
left join Ctenar c on p.ctenar = c.ctenar_id

```

-- implementace procedury Najdi_tiskovinu

```

create procedure Najdi_tiskovinu
@nazev varchar (200) = null,
@identifikator varchar (17) = null,
@jmeno varchar (15) = null,
@prijmeni varchar (25) = null,
@rocnik varchar (4) = null,
@cislo varchar (3) = null

```

```

as
begin
declare @ptiskovina_id integer
select @ptiskovina_id = t.tiskovina_id
from Tiskovina t
left join Autor_tiskovina m on t.tiskovina_id = m.tiskovina
left join Autor a on a.autor_id = m.autor
where @nazev = t.nazev or @identifikator = t.isbn_issn or @jmeno = a.jmeno or
@prijmeni = a.prijmeni or @rocnik = t.rocnik or @cislo = t.cislo
if @ptiskovina_id is null
begin
print 'Tiskovina nenalezena.'
end
else
begin
print 'Tiskovina nalezena.'
select * from Tiskoviny t
where @nazev = t.Název or @identifikator = t.Identifikátor or t.Autor like '%' +
@jmeno + '%' or t.Autor like '%' + @prijmeni + '%' or @rocnik = t.Ročník or @cislo =
t.Číslo
end
end
go

```

```

-- implementace procedury Vloz_ctenare
create procedure Vloz_ctenare
@jmeno varchar (15),
@prijmeni varchar (25),
@datum_narozeni date,
@pohlavi integer = null,
@titul_pred varchar (25) = null,
@titul_za varchar (20) = null,
@prukaz_ozp_id integer = null,
@zakonny_zastupce_id integer = null
as
begin
declare @pctenar_id integer
select @pctenar_id = c.ctenar_id
from Ctenar c
where @jmeno = c.jmeno and @prijmeni = c.prijmeni and @datum_narozeni =
c.datum_narozeni
if @pctenar_id is null
begin
insert into Ctenar (jmeno, prijmeni, datum_narozeni, pohlavi, titul_pred, titul_za,
zakonny_zastupce, prukaz_ozp)
values (ltrim (@jmeno), ltrim (@prijmeni), ltrim (@datum_narozeni), @pohlavi,
ltrim (@titul_pred), ltrim (@titul_za), @zakonny_zastupce_id, @prukaz_ozp_id)
print 'Čtenář vložen do databáze.'
end
else

```

```

begin
print 'Čtenář je již v databázi.'
select * from Ctenari c
where @pctenar_id = c.[Id čtenáře]
end
end
go

-- implementace procedury Vloz_email
create procedure Vloz_email
@email varchar (50),
@ctenar_id integer
as
begin
declare @pemail_id integer
select @pemail_id = e.email_id from Email e
where @email = e.email
if @pemail_id is null
begin
insert into Email (email, ctenar)
values (@email, @ctenar_id)
print 'Email vložen do databáze.'
end
else
begin
print 'Email je již v databázi.'
select e.email_id as 'Id emailu', e.email as 'Email', [Id čtenáře], [Jméno vč. titulu]
from Ctenari c
left join Email e on c.[Id čtenáře] = e.ctenar
where e.email_id = @pemail_id
end
end
go

-- implementace procedury Vloz_telefon
create procedure Vloz_telefon
@cislo varchar (50),
@ctenar_id integer
as
begin
declare @ptelefon_id integer
select @ptelefon_id = t.telefon_id from Telefon t
where @cislo = t.cislo
if @ptelefon_id is null
begin
insert into Telefon (cislo, ctenar)
values (@cislo, @ctenar_id)
print 'Telefon vložen do databáze.'
end
else
begin

```

```

print 'Telefon je již v databázi.'
select t.telefon_id as 'Id telefonu', ('+' + t.cislo) as 'Telefonní číslo', [Id čtenáře], [Jméno
vč. titulu]
from Ctenari c
left join Telefon t on c.[Id čtenáře] = t.ctenar
where t.telefon_id = @ptelefon_id
end
end
go

-- implementace procedury Vloz_adresu
create procedure Vloz_adresu
@psc varchar (5),
@mesto varchar (25),
@ulice varchar (25),
@cislo varchar (25)
as
begin
declare @padresa_id integer
select @padresa_id = a.adresa_id from Adresa a
where @psc = a.psc and @mesto = a.mesto and @ulice = a.ulice and @cislo = a.cislo
if @padresa_id is null
begin
insert into Adresa (psc, mesto, ulice, cislo)
values (@psc, @mesto, @ulice, @cislo)
print 'Adresa vložena do databáze.'
end
else
begin
print 'Adresa je již v databázi.'
select a.adresa_id as 'Id adresy', a.psc as 'PSČ', a.mesto as 'Město', a.ulice as 'Ulice',
a.cislo as 'Číslo popisné'
from Adresa a
where a.adresa_Id = @padresa_id
end
end
go

-- implementace procedury Vloz_ctenar_adresa
create procedure Vloz_ctenar_adresa
@ctenar_id integer,
@adresa_id integer,
@druh_id integer
as
begin
declare @pctenar_id integer
select @pctenar_id = c.ctenar from Ctenar_adresa c
where @ctenar_id = c.ctenar and @adresa_id = c.adresa and @druh_id = c.druh
if @pctenar_id is null
begin
insert into Ctenar_adresa (ctenar, adresa, druh)

```

```

values (@ctenar_id, @adresa_id, @druh_id)
print 'Spojení čtenář, adresa, druh bylo vloženo do databáze.'
end
else
begin
print 'Adresa tohoto druhu je již pro daného čtenáře zadána. Zadejte pro daného čtenáře
jinou adresu jiného druhu.'
select c.ctenar as 'Id čtenáře',
c.adresa as 'Id adresy',
replace (replace (replace (c.druh, 1, 'Trvalá'), 2, 'Kontaktní'), 3, 'Trvalá/Kontaktní') as
'Druh adresy'
from Ctenar_adresa c
where @ctenar_id = c.ctenar and @adresa_id = c.adresa and @druh_id = c.druh
end
end
go

```